Numerical solution of optimal control problems with state constraints by sequential quadratic programming in function space

NUMERICAL SOLUTION OF OPTIMAL CONTROL PROBLEMS WITH STATE CONSTRAINTS BY SEQUENTIAL QUADRATIC PROGRAMMING IN FUNCTION SPACE

K.C.P. Machielsen

NUMERICAL SOLUTION OF OPTIMAL CONTROL PROBLEMS WITH STATE CONSTRAINTS BY SEQUENTIAL QUADRATIC PROGRAMMING IN FUNCTION SPACE

PROEFSCHRIFT

TER VERKRIJGING VAN DE GRAAD VAN DOCTOR AAN DE TECHNISCHE UNIVERSITEIT EINDHOVEN, OP GEZAG VAN DE RECTOR MAGNIFICUS, PROF. DR. F.N. HOOGE, VOOR EEN COMMISSIE AANGEWEZEN DOOR HET COLLEGE VAN DEKANEN IN HET OPENBAAR TE VERDEDIGEN OP
DINSDAG 31 MAART 1987 TE 16.00 UUR

DOOR

KEES CASPERT PETER MACHIELSEN

GEBOREN TE VLAARDINGEN

Druk: Dissertatiedrukkerij Wibro, Helmond.

Dit proefschrift is goedgekeurd
door de promotoren

Prof. Dr. Ir. M.L.J. Hautus

en

Prof. Dr. G.W. Veltkamp

Copromotor Dr. Ir. J.L. de Jong

Aan Angela

Aan mijn ouders

De onderzoekingen in dit proefschrift beschreven zijn verricht in het CAM centrum van het Centrum voor Fabricage Technieken van de Nederlandse Philips Bedrijven B.V. te Eindhoven, in samenwerking met de faculteit der Wiskunde en Informatica van de Technische Universiteit Eindhoven. De directie van het CAM centrum ben ik zeer erkentelijk voor de mij geboden gelegenheid dit werk uit te voeren en in deze vorm te publiceren.

Summary.

The purpose of this thesis is to present a numerical method for the solution of state constrained optimal control problems.

In the first instance, optimization problems are introduced and considered in an abstract setting. The major advantage of this abstract treatment is that one can consider optimality conditions without going into the details of problem specifications. A number of results on optimality conditions for the optimization problems are reviewed.

Because state constrained optimal control problems can be identified as special cases of the abstract optimization problems, the theory reviewed for abstract optimization problems can be applied directly. When the optimality conditions for the abstract problems are expressed in terms of the optimal control problems, the well known minimum principle for state constrained optimal control problems follows.

The method, which is proposed for the numerical solution of the optimal control problems, is presented first in terms of the abstract optimization problems. Essentially the method is analogous to a sequential quadratic programming method for the numerical solution of finite-dimensional nonlinear programming problems. Hence, the method is an iterative descent method where the direction of search is determined by the solution of a subproblem with quadratic objective function and linear constraints. In each iteration of the method a step size is determined using an exact penalty (merit) function. The application of the abstract method to state constrained optimal control problems is complicated by the fact that the subproblems, which are optimal control problems with quadratic objective function and linear constraints (including linear state constraints), cannot be solved easily when the structure of the solution is not known. A modification of the subproblems is therefore necessary. As a result of this modification the method will, in general, not converge to a solution of the problem, but to a point close to a solution. Therefore a second stage, which makes use of the structure of the solution determined in the first stage, is necessary to determine the solution more accurately.

The numerical implementation of the method essentially comes down to the numerical solution of a linear multipoint boundary value problem. Several methods may be used for the numerical solution of this problem, but the collocation method which was chosen, has several important advantages over other methods. Effective use can be made of the special structure of the set of linear equations to be solved, using large scale optimization techniques.

Numerical results of the program for some practical problems are given. Two of these problems are well known in literature and allow therefore a comparison with results obtained by others.

Finally the relations between the method proposed and some other methods is given.

Contents page

Appendices :

1. Introduction.

1.1. State constrained optimal control problems.

Optimal control problems arise in practice when there is a demand to control a system from one state to another in some optimal sense, i.e. the control must be such that some (objective) criterion is minimized (or maximized).

In this thesis we are interested in those optimal control problems which are completely deterministic. This means that the dynamic behaviour of the system to be controlled is determined completely by a set of differential equations and that stochastic influences on the state of the system, which are present in practical systems, may be neglected.

It is assumed that the dynamic behaviour of the system to be controlled can be described by a set of ordinary differential equations of the form :

$$\dot{x}(t) = f(x(t),u(t),t) \qquad 0 \leqslant t \leqslant T, \tag{1.1.1}$$

where x is an n-vector function on $[0,T]$ called the state variable and u is an m-vector function on $[0,T]$ called the control variable. The function f is an n-valued vector function, on $\mathbb{R}^n \times \mathbb{R}^m \times [0,T]$. It is assumed that f is twice continuously differentiable with respect to its arguments.

On the one hand one may note that the dynamic behaviour of a large number of systems, which arise in practice, can be described by a set of differential equations of the form (1.1.1). On the other hand systems with delays are excluded from this formulation.

The system is to be controlled starting from an initial state x_0 at $t = 0$, i.e.

$$x(0) = x_0, \tag{1.1.2}$$

over an interval $[0,T]$. The number T is used to denote the final time. We shall assume that T is finite, which means that we are interested in so-called finite time horizon optimal control problems.

The object criterion is specified by means of a functional which assigns a real value to each triple (x,u,T) of the following form :

$$\int_0^T f_0(x(t),u(t),t)\,dt + g_0(x(T),T). \tag{1.1.3}$$

About the functions f_0 and g_0 it is only assumed that they are twice continuously differentiable with respect to their arguments. We note that the rather general formulation of (1.1.3) includes the formulation of minimum time and minimum energy problems (cf. Falb et al. (1966)).

For most optimal control problems which arise in practice, the control u and the state x must satisfy certain conditions, in addition to the differential equations. It is assumed that these conditions, which enter into the formulation of the optimal control problem as constraints, may take any of the following forms :

* *Terminal point constraints*, i.e. the final state $x(T)$ must satisfy a vector equality of the form :

$$E(x(T),T) = 0. \tag{1.1.4}$$

* *Control constraints*, i.e. the control u must satisfy :

$$S_0(u(t),t) \leqslant 0 \qquad \text{for all} \quad 0 \leqslant t \leqslant T. \tag{1.1.5}$$

* *Mixed control state constraints*, i.e. the control u and the state x must satisfy :

$$S_1(x(t),u(t),t) \leqslant 0 \qquad \text{for all} \quad 0 \leqslant t \leqslant T. \tag{1.1.6}$$

* *State constraints*, i.e. the state x must satisfy :

$$S_2(x(t),t) \leqslant 0 \qquad \text{for all} \quad 0 \leqslant t \leqslant T. \tag{1.1.7}$$

For the numerical method to be presented in this thesis, the distinction between control and mixed control state constraints is not important. The distinction between mixed control state constraints and state constraints however, is essential. The major difficulty involved with state constraints is that these constraints represent implicit constraints on the control, as the state function is completely determined by the control via the differential equations.

The optimal control problems formally stated above are obviously of a very general type and cover a large number of problems considered by the available optimal control theory. The first practical applications of optimal control theory were in the field of aero-space engineering, which involved mainly problems of flight path optimization of airplanes and space vehicles. (See e.g. Falb et al. (1966, 1969), Bryson et al. (1975).) As examples of these types of problems one may consider the problems solved in Sections 8.1 and 8.2. We note that the reentry manoever of an Apollo capsule was first posed as an optimal control problem as early as 1963 by Bryson et al. (1963b). Later optimal control theory found application in many other areas of applied science, such as econometrics (see e.g. van Loon (1982), Geerts (1985)).

Recently, there is a growing interest in optimal control theory arising from the field of robotics (see e.g. Bobrow et al. (1985), Bryson et al. (1985), Gomez (1985), Machielsen (1983), Newman et al. (1986), Shin et al. (1985)). For the practical application of the method presented in this thesis, this area of robotics is of special importance. Therefore we will briefly outline an important problem from this field in the next section.

1.2. An example of state constrained optimal control problems in robotics.

In general, a (rigid body) model of a robotic arm mechanism, which consists of k links (and joints) may be described by means of a nonlinearly coupled set of k-differential equations of the form (see e.g. Paul (1981), Machielsen (1983)) :

$$J(q)\ddot{q} + D(\dot{q},q) = F \tag{1.2.1}$$

where q is the vector of joint positions, $\dot{q}$ is the vector of joint velocities and $\ddot{q}$ is the vector of joint accelerations. $J(q)$ is the $k \times k$ inertia matrix which, in general, will be invertible. The vector $D(\dot{q},q)$ represents gravity, coriolis and centripetal forces. F is the vector of joint torques.

It is supposed that the arm mechanism is to be controlled from one point to another point along a path that is specified as a parameterized curve. The curve is assumed to be given by a set of k functions $\Upsilon_i : [0,1] \rightarrow \mathbb{R}$ of a single parameter s, so that the joint positions $q_i(t)$ must satisfy :

$$q_i(t) = \Upsilon_i(s(t)) \qquad 0 \leq t \leq T \quad 1 \leq i \leq k, \tag{1.2.2}$$

where $s:[0,T] \rightarrow [0,1]$. The value of the function $s(t)$ at a time point t is interpreted as the relative position on the path. Thus, at the initial point we have $s(0)=0$ and at the final point we have $s(T)=1$.

Equation (1.2.2) reveals that for each fixed (sufficiently smooth) function $s:[0,T] \rightarrow [0,1]$, the motion of the robot along the path is completely determined. Differentiation of equation (1.2.2) with respect to the variable t yields the joint velocities and accelerations. †

$$\dot{q}(t) = \Upsilon'(s(t))\dot{s}(t) \qquad 0 \leq t \leq T, \tag{1.2.3}$$

$$\ddot{q}(t) = \Upsilon'(s(t))\ddot{s}(t) + \Upsilon''(s(t))\dot{s}(t)^2 \qquad 0 \leq t \leq T. \tag{1.2.4}$$

The joint torques required to control the robot along the path for a certain function $s:[0,T] \rightarrow [0,1]$, follow from the combination of the equations of motion of the robot (1.2.1) and equations (1.2.2) - (1.2.4), which relate the path motion to the joint positions, velocities and accelerations.

$$F(t) = J(\Upsilon(s(t)))(\Upsilon'(s(t))\ddot{s}(t) + \Upsilon''(s(t))\dot{s}(t)^2) + D(\Upsilon'(s(t))\dot{s}(t), \Upsilon(s(t))) \qquad 0 \leq t \leq T. \tag{1.2.5}$$

For most robotic systems, the motion of the robot is restricted by constraints on the joint velocities and torques. These constraints are of the following type :

$$|\dot{q}_i(t)| \leq V_{max,i} \qquad 0 \leq t \leq T \quad i = 1,\ldots,k, \tag{1.2.6}$$

$$|F_i(t)| \leq F_{max,i} \qquad 0 \leq t \leq T \quad i = 1,\ldots,k. \tag{1.2.7}$$

The optimal control problem can be formulated completely in terms of the function s, i.e. in terms of the relative motion along the path. The joint positions, velocities, accelerations and torques can be eliminated using relations (1.2.2) - (1.2.5). The constraints (1.2.6) - (1.2.7) become :

$$|\Upsilon_i'(s(t))\dot{s}(t)| \leq V_{max,i} \qquad 0 \leq t \leq T \qquad 1 \leq i \leq k, \tag{1.2.8}$$

$$|J(\Upsilon(s(t)))(\Upsilon'(s(t))\ddot{s}(t) + \Upsilon''(s(t))\dot{s}(t)^2) + D(\Upsilon'(s(t))\dot{s}(t), \Upsilon(s(t)))| \leq F_{max} \qquad 0 \leq t \leq T. \tag{1.2.9}$$

The optimal control problem comes down to the selection of a function s, which minimizes some object criterion, is twice differentiable and satisfies the constraints (1.2.8) - (1.2.9), $s(0)=0$ and $s(T)=1$.

The choice of a suitable object criterion depends on the specific robot application. For instance, this criterion may be the final time T which yields minimum time control. This criterion, however, may have the disadvantage in many practical applications that the solution of the optimal control problem is 'not smooth enough', because the second derivative of the function s is likely to be of the bang-bang type. Relation (1.2.5) reveals that discontinuities of $\ddot{s}$ yield discontinuous joint torques which is an undesirable phenomenon in many applications from the mechanics point of view (see e.g. Koster (1973)).

† For equations (1.2.3) - (1.2.5) a vector notation is used.

An alternative to minimum time control is to select a smooth function s that satisfies the constraints, via the minimization of

$$\frac{1}{2}\int_0^T \ddot{s}(t)^2\,dt\,, \tag{1.2.10}$$

for a fixed final time T. It can be shown, that with this objective function the solution of the optimal control problem has a continuous second derivative (provided T is larger than the minimum time) and hence, the joint torques will also be continuous. A drawback of this approach may be that the final time must be specified in advance, which, in general is not known a priori.

A second alternative, which combines more or less the advantages of both objective functions, is to use :

$$T + \frac{1}{2}c\int_0^T \ddot{s}(t)^2\,dt\,. \tag{1.2.11}$$

as an objective function and to 'control' the properties of the solution of the optimal control problem via a suitable (a priori) choice of the parameter c.

A more formal statement of the problem outlined above shows that the optimal control problem is indeed of the type discussed in the previous section and that the solution of this problem is complicated in particular by the presence of the (state) constraints (1.2.8) - (1.2.9).

1.3. Optimality conditions for state constrained optimal control problems.

In this section we shall introduce optimality conditions for state constrained optimal control problems in a formal manner. This is done in view of the central role that optimality conditions play in any solution method for these problems.

It can be shown that the optimal control problems introduced in Section 1.1 are special cases of the following abstract optimization problem :

$$\underset{x\in X}{\textit{minimize}}\ \tilde{f}(x). \tag{1.3.1}$$

$$\textit{subject to}: \tilde{g}(x) \in B\,, \tag{1.3.2}$$

$$\tilde{h}(x) = 0, \tag{1.3.3}$$

where $\tilde{f}: X \to \mathbb{R}$; $\tilde{g}: X \to Y$; $\tilde{h}: X \to Z$ are mappings from one Banach space (X) to another $(\mathbb{R}, Y, Z)$ and $B \subset Y$ is a cone with nonempty interior. The functional $\tilde{f}$ denotes the objective criterion which is to be minimized over the set of feasible points, i.e. the set of points which satisfy the inequality constraints $\tilde{g}(x) \in B$ and the equality constraints $\tilde{h}(x) = 0$.

The problem (1.3.1) - (1.3.3) is a generalization of the well known finite-dimensional mathematical programming problem (i.e. $X = \mathbb{R}^n$, $Y = \mathbb{R}^{m_i}$, $Z = \mathbb{R}^{m_e}$) :

$$\underset{x \in \mathbb{R}^n}{\textit{minimize}} \quad \tilde{f}(x), \tag{1.3.4}$$

$$\textit{subject to} : \tilde{g}(x) \leqslant 0, \tag{1.3.5}$$

$$\tilde{h}(x) = 0, \tag{1.3.6}$$

It is possible to derive optimality conditions for the abstract optimization problem (1.3.1) - (1.3.3), i.e. conditions which must hold for solutions of the problem. Because both the state constrained optimal control problems discussed in Section 1.1 and the finite-dimensional mathematical programming problem are special cases of the abstract problem, optimality conditions for these problems follow directly from the optimality conditions for the abstract problem. As an introduction however, we shall review the optimality conditions for the finite-dimensional mathematical programming problem (1.3.4) - (1.3.6) directly (e.g. cf. Gill et al. (1981); Mangasarian (1969)).

First we recall that, for any minimum of the functional $\tilde{f}$, denoted $\hat{x}$, which is not subject to any constraints, it must hold that :

$$\nabla \tilde{f}(\hat{x}) = 0, \tag{1.3.7}$$

i.e. the gradient of $\tilde{f}$ at $\hat{x}$ must vanish.

For the case that only equality constraints are present the optimality conditions state that when $\hat{x}$ is a solution to the problem, and $\hat{x}$ satisfies some constraint qualification, then there exists a (Lagrange multiplier) vector $\hat{z}$, such that the Lagrangian

$$L(x;\hat{z}) := \tilde{f}(x) - \hat{z}^T \tilde{h}(x), \tag{1.3.8}$$

has a stationary point at $\hat{x}$, i.e.

$$\nabla_x L(\hat{x};\hat{z}) = \nabla \tilde{f}(\hat{x}) - \hat{z}^T \nabla \tilde{h}(\hat{x}) = 0. \tag{1.3.9}$$

Rewriting condition (1.3.9) we obtain :

$$\nabla \tilde{f}(\hat{x}) = \sum_{j=1}^{m_e} \hat{z}_j \nabla \tilde{h}_j(\hat{x}), \tag{1.3.10}$$

which shows that at the point $\hat{x}$, the gradient of the objective functional must be a linear combination of the gradients of the constraints. The numbers $\hat{z}_j$ are called Lagrange multipliers and have the interpretation of marginal costs of constraint perturbations.

When there are, besides equality constraints, also inequality constraints present, the optimality conditions state that when $\hat{x}$ is a solution to the problem, and $\hat{x}$ satisfies some constraint qualification, then there exist vectors $\hat{y}$ and $\hat{z}$, such that the Lagrangian

$$L(x;\hat{y},\hat{z}) := \tilde{f}(x) - \hat{y}^T \tilde{g}(x) - \hat{z}^T \tilde{h}(x), \tag{1.3.11}$$

has a stationary point at $\hat{x}$ and that in addition

$$\hat{y}_j \tilde{g}_j(\hat{x}) = 0 \qquad j = 1,\ldots,m_i, \tag{1.3.12}$$

$$\hat{y}_j \leqslant 0 \qquad j = 1,\ldots,m_i, \tag{1.3.13}$$

Condition (1.3.12) is called the complementary slack condition. This states that all inactive inequality constraints, i.e. constraints for which $\tilde{g}_j(\hat{x}) < 0$, may be neglected, because the corresponding Lagrange multiplier must be zero.

Condition (1.3.13) is directly due to the special nature of the inequality constraints. To see this, a distinction must be made between negative (feasible) and positive (infeasible) perturbations of the constraints. The sign of the multiplier must be nonpositive in order that a feasible perturbation of the constraint does not yield a decrease in cost. Otherwise, the value of the objective function could be reduced by releasing the constraint.

Having introduced optimality conditions for the finite-dimensional mathematical programming problem, we shall now introduce optimality conditions for state constrained optimal control problems in a similar way. The Lagrangian of the state constrained optimal control problem is defined as :

$$L(x,u;\lambda,\eta_1,\xi,\mu) := \int_0^T f_0(x,u,t)\,dt + g_0(x(T),T) - \int_0^T \lambda^T(\dot{x}-f(x,u,t))\,dt$$

$$+ \int_0^T \eta_1^T S_1(x,u,t)\,dt + \int_0^T d\,\xi(t)^T S_2(x,t) + \mu^T E(x(T),T). \qquad (1.3.14)$$

The optimality conditions state that when $(\hat{x},\hat{u})$ is a solution to the state constrained optimal control problem, and $(\hat{x},\hat{u})$ satisfy some constraint qualification, then there exist multipliers $\hat{\lambda}$, $\hat{\eta}_1$, $\hat{\xi}$ and $\hat{\mu}$ such that the Lagrangian has a stationary point at $(\hat{x},\hat{u})$. Using calculus of variations (e.g. cf. Bryson et al. (1963a) or Hestenes (1966)) this yields the following relations on intervals where the time derivative of $\hat{\xi}$ exists :†

$$\dot{\hat{\lambda}}(t) = -H_x[t]^T - S_{1x}[t]^T\hat{\eta}_1(t) - S_{2x}[t]^T\dot{\hat{\xi}}(t) \qquad 0\leqslant t\leqslant T, \qquad (1.3.15)$$

$$H_u[t] + \hat{\eta}_1(t)^T S_{1u}[t] = 0 \qquad 0\leqslant t\leqslant T, \qquad (1.3.16)$$

$$\lambda(T) = g_{0x}[T] + \mu^T E_x[T]. \qquad (1.3.17)$$

where the Hamiltonian is defined as :

$$H(x,u,\lambda,t) := f_0(x,u,t) + \lambda^T f(x,u,t). \qquad (1.3.18)$$

At points t_i where the multiplier function $\hat{\xi}$ has a discontinuity the so-called jump-condition must hold

$$\lambda(t_i+) = \lambda(t_i-) - S_{2x}[t_i]d\,\hat{\xi}(t_i), \qquad (1.3.19)$$

which states that at these points the adjoint variable $\hat{\lambda}$ is also discontinuous.

The complementary slackness condition yields :

$$\hat{\eta}_{1i}(t)S_{1i}[t] = 0 \qquad 0\leqslant t\leqslant T \qquad i=1,\ldots,k_1, \qquad (1.3.20)$$

$$\hat{\xi}_i(t) \text{ *is constant on intervals where* } S_{2i}[t] < 0 \qquad 0\leqslant t\leqslant T \qquad i=1,\ldots,k_2. \qquad (1.3.21)$$

and the sign condition on the multipliers becomes :

$$\hat{\eta}_{1i}(t) \geqslant 0 \qquad 0\leqslant t\leqslant T \qquad i=1,\ldots,k_1, \qquad (1.3.22)$$

$$\hat{\xi}_i(t) \text{ *is nondecreasing on* } [0,T]. \qquad (1.3.23)$$

A more detailed analysis reveals that normally the multiplier function $\hat{\xi}$ is continuously differentiable on the interior of a boundary arc of the corresponding state constraint, i.e. an

† Straight brackets $[t\,]$ are used to replace argument lists involving $\hat{x}(t)$, $\hat{u}(t)$, $\hat{\lambda}(t)$.

interval where the state constraint is satisfied as an equality. The function $\hat{\xi}$ is in most cases discontinuous at junction and contact points, i.e. at points where a boundary arc of the constraint is entered or exited and at points where the constraint boundary is touched.

The combination of relations (1.3.15) - (1.3.19) with the constraints of the problem allow the derivation of a multipoint boundary value problem in the variables x and λ, with boundary conditions at $t=0$, $t=T$ and at the time points t_i where the jump conditions must hold. To obtain this boundary value problem the control u and the multipliers η_1 and ξ must be eliminated. This is usually only possible when the structure of the solution is known, i.e. the sequence in which the various constraints are active and inactive.

Because of the important role that optimality conditions play in any solution procedure of optimal control problems, optimality conditions have experienced quite some interest in the past. We refer to Bryson et al. (1963a, 1975), Falb et al. (1966), Hamilton (1972), Hestenes (1966), Jacobson et al. (1971), Köhler (1980), Kreindler (1982), Maurer (1976, 1977, 1981), Norris (1973), Pontryagin et al. (1962), Russak (1970a, 1970b).

1.4. Available methods for the numerical solution.

Among the methods, available for the numerical solution of optimal control problems, a distinction can be made between *direct* and *indirect* methods. With direct methods the optimal control problem is treated directly as a minimization problem, i.e. the method is started with an initial approximation of the solution, which is improved iteratively by minimizing the objective functional (augmented with a 'penalty' term) along a direction of search. The direction of search is obtained via a linearization of the problem. With indirect methods the optimality conditions, which must hold for a solution of the optimal control problem, are used to derive a multipoint boundary value problem. Solutions of the optimal control problem will also be solutions of this multipoint boundary value problem and hence the numerical solution of the multipoint boundary value problem yields a candidate for the solution of the optimal control problem. These methods are called indirect because the optimality conditions are solved as a set of equations, as a replacement for the minimization of the original problem.

Most direct methods are of the gradient type, i.e. they are function space analogies of the well known gradient method for finite-dimensional nonlinear programming problems (cf. Bryson et al. (1975)). The development of these function space analogies is based on the relationship between optimal control problems and nonlinear programming problems. This relationship is revealed by the fact that they are both special cases of the same abstract optimization problem. With most gradient methods the control $u(t)$ is considered as the variable of the minimization problem and the state $x(t)$ is treated as a quantity dependent on the control $u(t)$ via the differential equations. A well known variant on the ordinary gradient methods is the gradient-restoration method of Miele (cf. Miele (1975, 1980). This is essentially a projected gradient method in function space (cf. Gill et al. (1981)). With this method both the control $u(t)$ and the state $x(t)$ are taken as variables of the minimization problem and the differential equations enter the formulation as (infinite-dimensional) equality constraints. Similar to the finite-dimensional case where gradient methods can be extended to quasi-Newton or Newton-like methods, gradient methods for optimal control problems can be modified to quasi-Newton or Newton-like methods. (cf. Bryson et al. (1975), Edge et al. (1976), Miele et al. (1982)).

With all gradient type methods, state constraints can be treated via a penalty function approach, i.e. a term which is a measure for the violation of the state constraints is added to the objective function. Numerical results however, indicate that this penalty function approach yields a very inefficient and inaccurate method for the solution of state constrained optimal control problems (cf. Well (1983)).

Another way to treat state constraints is via a slack-variable transformation technique, using quadratic slack-variables. This technique transforms the inequality state constrained problem into a problem with mixed control state constraints of the equality type. A drawback of this approach is that the slack-variable transformation becomes singular at points where the constraint is active (cf. Jacobson et al. (1969)). As a result of this, it may be possible that state constraints, which are treated active in an early stage of the solution process, cannot change from active to inactive. Therefore it is not certain whether the method converges to the right set of active points. In addition, the numerical results of Bals (1983) show that this approach may fail to converge at all for some problems.

Another type of direct method follows from the conversion of the (infinite-dimensional) optimal control problem into a (finite-dimensional) nonlinear programming problem. This is done by approximating the time functions using a finite-dimensional base (cf. Kraft (1980, 1984)). The resulting nonlinear programming problem may be solved using any general purpose method for this type of problem. We note that when a sequential quadratic programming method (cf. Gill et al. (1981)) is used, then this direct method has a relatively strong correspondence with the method discussed in this thesis. In view of its significance for the work presented in this thesis, this method is described in more detail in Section 8.1.

A well known indirect method is the method based on the numerical solution of the multipoint boundary value problem using multiple shooting (cf. Bulirsch (1983), Bock (1983), Maurer et al. (1974, 1975, 1976), Oberle (1977, 1983), Well (1983)). For optimal control problems with state constraints, the right hand side of the differential equations of the multipoint boundary value problem will, in general, be discontinuous at junction and contact points.† These discontinuities require special precautions in the boundary value problem solver. The junction and contact points can be characterized by means of so-called switching functions, which are used to locate these points numerically.

Another indirect method, which can only be used for the solution of optimal control problems without state constraints, is based on the numerical solution of the boundary value problem using a collocation method (cf. Dickmans et al. (1975)). The reason that the method cannot be used without modification for the solution of state constrained optimal control problems is that these problems require the solution of a multipoint boundary value problem whereas the specific collocation method discussed by Dickmans et al. is especially suited for the numerical solution of two point boundary value problems. Numerical results indicate that the method is relatively efficient and accurate.

In general, the properties of the direct and indirect methods are somewhat complementary. Direct methods tend to have a relatively large region of convergence and tend to be relatively inaccurate, whereas indirect methods generally have a relatively small region of

† Junction points are points where a constraint changes from active to inactive or vice versa. At contact points the solution touches the constraint boundary.

convergence and tend to be relatively accurate. For state constrained optimal control problems the indirect methods make use of the structure of the solution, i.e. the sequence in which the state constraints are active and inactive on the interval $[0,T]$, for the derivation of the boundary value problem. Direct methods do not require this structure. Because state constraints are treated via a penalty function approach, most direct methods are relatively inefficient. In practice, they are used only for the determination of the structure of the solution. An accurate solution of the state constrained optimal control problem can in most practical cases only be determined via an indirect method, which is started with an approximation to the solution obtained via a direct method.

1.5. Scope of the thesis.

In Chapter 2, optimization problems are introduced and considered in an abstract setting. The major advantage of this abstract treatment is that one is able to consider optimality conditions without going into the details of problem specifications.

The state constrained optimal control problems are stated in Chapter 3. Because these problems can be identified as special cases of the abstract problems considered in Chapter 2, the theory stated in Chapter 2 can be applied to the optimal control problems. This yields the well known minimum principle for state constrained optimal control problems.

In Chapter 4, the method which is proposed for the numerical solution of state constrained optimal control problems is presented first in the abstract terminology of Chapter 2. Essentially, this method is analogous to a sequential quadratic programming method for the numerical solution of a finite-dimensional nonlinear problem. Hence, it is an iterative descent method where the direction of search is determined as the solution of a subproblem with quadratic objective function and linear constraints.

Chapter 5 deals with the solution of the subproblems whose numerical solution is required for the calculation of the direction of search. In addition the active set strategy, which is used to locate the set of active points of the state constraints, is described.

The numerical implementation of the method, which essentially comes down to the numerical solution of a linear multipoint boundary value problem, is discussed in Chapter 6.

The numerical results of the computer program for some practical problems are given in Chapter 7. Two of these problems are well known in literature and therefore allow a comparison with the results obtained by others.

In the final chapter the relation between the method discussed in this thesis and some other methods is established. The chapter is closed with some final comments.

The method used for the solution of one of the subproblems is based on a method for the solution of finite-dimensional quadratic programming problems, which is reviewed in Appendix A. Appendix B deals with a transformation of state constraints to a form which allows a relatively simple solution procedure for the subproblems. Technical results relevant for the active set strategy are summarized in Appendix C. A number of computational details are given in Appendices D and E. Numerical results related to the results contained in Chapter 7 are listed in Appendix F.

2. Nonlinear programming in Banach spaces.

In this chapter, a number of results from the theory of functional analysis concerned with optimization will be reviewed.

In Section 2.1 some optimization problems will be introduced in an abstract formulation and in Sections 2.2 and 2.3 some results on optimality conditions and constraint qualifications in Banach spaces will be reviewed.

2.1. Optimization problems in Banach spaces.

In this chapter, we shall consider optimization problems from an abstract point of view. The major advantage of such an abstract treatment is that one is able to consider the problems without first going into the details of problem specifications. The first optimization problem to be considered is defined as :

Problem (P_0) : *Given a Banach space U, an objective functional $J : U \to \mathbb{R}$ and a constraint set $S_0 \subset U$, find an $\hat{u} \in S_0$, such that*

$$J(\hat{u}) \leqslant J(u) \quad \text{for all } u \in S_0. \tag{2.1.1}$$

A solution $\hat{u}$ of problem P_0 is said to be a *global minimum* of J subject to the constraint $u \in S_0$. In practice it is often difficult to prove that a solution is a global solution to the problem. Instead one therefore considers conditions for a weaker type of solution. This weaker type of solution is defined as :

Definition 2.1 : *In the terminology of problem (P_0) a vector $\tilde{u} \in U$ is said to be a local minimum of J, subject to the constraint $u \in S_0$, if there is an $\epsilon > 0$ such that,*

$$J(\tilde{u}) \leqslant J(u) \quad \text{for all } u \in S_0 \cap S(\tilde{u}, \epsilon), \tag{2.1.2}$$

with :

$$S(\tilde{u}, \epsilon) := \{ u \in U : \|u - \tilde{u}\| < \epsilon \}. \tag{2.1.3}$$

We shall consider two special cases of problem (P_0).

Problem (P_1) : *Given two Banach spaces U and L, two twice continuously Fréchet differentiable mappings $J : U \to \mathbb{R}$ and $S : U \to L$, a convex set $M \subset U$ with nonempty interior and a closed convex cone $K \subset L$ with $0 \in K$, then find an $\hat{u} \in M$, such that $S(\hat{u}) \in K$ and that*

$$J(\hat{u}) \leqslant J(u) \quad \text{for all } u \in M \cap S^{-1}(K). \tag{2.1.4}$$

Comparing problems (P_0) and (P_1), we notice that in problem (P_1) :

* $S_0 = M \cap S^{-1}(K)$, with $S^{-1}(K) := \{u \in U : S(u) \in K\}$. The assumptions on K, M and S are made in order to obtain a suitable linearization of the constraint set S_0.
* J is supposed to be twice Fréchet differentiable.

A further specialization of problem (P_0) is obtained when a distinction is made between equality and inequality constraints.

Problem (EIP): *Given Banach spaces X, Y and Z, twice continuously Fréchet differentiable mappings $\tilde{f} : X \to \mathbb{R}$, $\tilde{g} : X \to Y$ and $\tilde{h} : X \to Z$, a convex set $A \subset X$ having a nonempty interior, and a closed convex cone $B \subset Y$ with $0 \in B$ and having nonempty interior, then find an $\hat{x} \in A$, such that $\tilde{g}(\hat{x}) \in B$ and $\tilde{h}(\hat{x}) = 0$ and that*

$$\tilde{f}(\hat{x}) \leqslant \tilde{f}(x) \quad \text{for all } x \in A \cap \tilde{g}^{-1}(B) \cap N(\tilde{h}). \tag{2.1.5}$$

In problem (EIP), the equality constraints are represented by $\tilde{h}(x) = 0$, whereas the inequality constraints are incorporated in $x \in A$ and $\tilde{g}(x) \in B$ (note that A and B have nonempty interiors).

Throughout this chapter we shall use various basic notions from the theory of functional analysis without giving explicit definitions. For these we generally refer to Luenberger (1969). Because of their central role in the ensuing discussion we explicitly recall the following definitions.

Definition 2.2: *Let X be a normed linear vector space, then the space of all bounded linear functionals on X is called the (topological) dual of X, denoted X^*.*

Definition 2.3: *Given the set K in a normed linear vector space X, then the dual (or conjugate) cone of K is defined as*

$$K^* := \{x^* \in X^* : \langle x^*, x \rangle \geqslant 0 \text{ for all } x \in K\}, \tag{2.1.6}$$

where the notation $\langle x^*, x \rangle$ is employed to represent the result of the linear functional $x^* \in X^*$ acting on $x \in X$.

In a number of occasions we shall also use the notation $x^* x$ instead of $\langle x^*, x \rangle$.
With regard to Definition 2.3 we note that the set K^* is a cone, as an immediate consequence of the linearity of the elements of X^*.

Definition 2.4: *Let S be a bounded linear operator from the normed linear vector space X into the normed linear vector space Y. The adjoint operator $S^* : Y^* \to X^*$ is defined by the equation :*

$$\langle x, S^* y^* \rangle = \langle Sx, y^* \rangle. \tag{2.1.7}$$

The notions of dual cone and adjoint operator play an important role in giving a characterization of the solutions of the optimization problems (P_1) and (EIP). Other concepts which play an important role in the following discussion are conical approximations of the set of feasible points.

Definition 2.5: *Let U be a Banach space, $M \subset U$ and $\bar{u} \in M$. The open cone*

$$A(M, \bar{u}) := \{u \in U : \exists \epsilon_0, r > 0, \forall \epsilon : 0 < \epsilon \leqslant \epsilon_0, \forall v \in U : \|v\| \leqslant r, \bar{u} + \epsilon(u + v) \in M\}, \tag{2.1.8}$$

is called the cone of admissible directions to M at $\bar{u}$.

This cone is referred to differently in literature : cone of feasible directions (Girsanov (1972)); cone of interior directions (Bazaraa et al. (1976)).
In the case that M has no interior, the cone $A(M, \bar{u})$ is empty for every $\bar{u} \in U$.

Definition 2.6 : *Let U be a Banach space, $M \subset U$ and $\bar{u} \in M$, then the set*

$$T(M,\bar{u}) := \{ u \in U : \exists (\epsilon_n)_{n=0}^{\infty},\ \epsilon_n \in \mathbb{R}^+,\ \epsilon_n \to 0,\ \exists (u_n)_{n=0}^{\infty},\ u_n \in M,\ u_n \to \bar{u},$$
$$u = \lim_{n \to \infty} (u_n - \bar{u})/\epsilon_n \}, \qquad (2.1.9)$$

i.e. the set of elements $u \in U$ for which there are sequences $(u_n)_{n=0}^{\infty}$ and $(\epsilon_n)_{n=0}^{\infty}$, with $u_n \to \bar{u}$, $\epsilon_n > 0$ and $\epsilon_n \to 0$, such that

$$u = \lim_{n \to \infty} (u_n - \bar{u})/\epsilon_n ,$$

is called the sequential tangent cone of M at $\bar{u}$.

In literature, the sequential tangent cone as defined in Definition 2.6, is also referred to as tangent cone (e.g. Bazaraa et al. (1976); Norris (1971)) or as local closed cone (Varaiya (1976)).

We note that the cone of admissible directions is always contained in the sequential tangent cone, i.e. $A(M,\bar{u}) \subset T(M,\bar{u})$.

Definition 2.7 : *Let U be a Banach space, $M \subset U$ and $\bar{u} \in M$. The set*

$$C(M,\bar{u}) := \{ \lambda(m - \bar{u}) : \lambda \geq 0,\ m \in M \}, \qquad (2.1.10)$$

is called the conical hull of $M - \{\bar{u}\}$.

This definition is analogous to the definition of the convex hull of a set A, i.e. the smallest convex set which contains the set A. In this context the conical hull of a set A is the smallest cone in which the set A is contained.

In the case that K is a cone with vertex at 0, the conical hull of $K - \{\bar{u}\}$ becomes :

$$C(K,\bar{u}) := \{ m - \lambda\bar{u} : \lambda \geq 0,\ m \in K \}. \qquad (2.1.11)$$

If M is a convex set with nonempty interior, the closure of the cone of admissible directions of M at $\bar{u}$ coincides with the conical hull of $M - \{\bar{u}\}$, i.e. $\overline{A(M,\bar{u})} = C(M,\bar{u})$ (cf. Girsanov (1972)).

Definition 2.8 : *Let U and L be Banach spaces, S a continuously Fréchet differentiable operator $U \to L$ and K a closed convex cone in L with $0 \in K$. At a point $\bar{u} \in U$, the set* †

$$L(S,K,\bar{u}) := \{ u \in U : S'(\bar{u})u \in C(K, S(\bar{u})) \}, \qquad (2.1.12)$$

is called the linearizing cone of $S^{-1}(K)$ at $\bar{u}$.

In Definition 2.8 the notation $S^{-1}(K)$ was used to denote the set

$$S^{-1}(K) := \{ u \in U : S(u) \in K \}. \qquad (2.1.13)$$

In view of the optimality conditions to be stated, the following regularity conditions are defined.

† S' is used to denote the Fréchet derivative of S.

Definition 2.9 : *Let U and L be Banach spaces, S a continuously Fréchet differentiable operator $U \to L$ and K a closed convex cone in L with $0 \in K$. The conditions*

$$L(S,K,\hat{u}) = T(S^{-1}(K),\hat{u}), \tag{2.1.14}$$

$$L(S,K,\hat{u})^* = S'(\hat{u})^* C(K,S(\hat{u}))^*, \tag{2.1.15}$$

$$\text{the set } R(S'(\hat{u})) + C(K,S(\hat{u})) \text{ is not dense in } L, \tag{2.1.16}$$

are respectively called

the Abadie condition,
the Farkas condition,
the Nonsingularity condition,

at $\hat{u}$.

We note that condition (2.1.14) is an abstract version of the Abadie constraint qualification in Kuhn-Tucker theory, which deals with optimality conditions for nonlinear programming problems in finite-dimensional spaces (cf. Bazaraa et al.(1976)). An interpretation of the various conditions is given in the next section in the outline of the proof of Theorem 2.10.

2.2. First order optimality conditions in Banach spaces.

In this section we shall present optimality conditions for solutions of problems (P_1) and (EIP). The results presented are mainly taken from the review article of Kurcyusz (1976).

The conditions involve only the first Fréchet derivatives of the mappings which are used to define the objective function and the constraints of the problem. This is the reason that they are called first order optimality conditions.

The Definitions 2.5 - 2.9 are used for the formulation of the following Lagrange multiplier theorem, which plays a central role in the following discussion.

Theorem 2.10 : *(Kurcyusz (1976), Theorem 3.1) Let $\hat{u}$ be a local solution to problem (P_1).*

(i) *If either condition (2.1.16) or both (2.1.14) and (2.1.15) hold, then there exists a pair $(\hat{\rho},\hat{l}^*) \in \mathbb{R} \times L^*$, such that,*

$$(\hat{\rho},\hat{l}^*) \neq (0,0^*), \tag{2.2.1}$$

$$\hat{\rho} \geq 0, \quad \hat{l}^* \in K^*, \quad \langle \hat{l}^*, S(\hat{u}) \rangle = 0, \tag{2.2.2}$$

$$\hat{\rho} J'(\hat{u}) - S'(\hat{u})^* \hat{l}^* \in A(M,\hat{u})^*. \tag{2.2.3}$$

A pair $(\hat{\rho},\hat{l}^*)$ satisfying (2.2.1) - (2.2.3) is called a pair of nontrivial Lagrange multipliers for problem (P_1).

(ii) *If conditions (2.1.14) and (2.1.15) are satisfied and*

$$A(M,\hat{u}) \cap L(S,K,\hat{u}) \neq \emptyset, \tag{2.2.4}$$

then there exists a vector $\hat{l}^* \in L^*$ such that (2.2.2) and (2.2.3) hold with $\hat{\rho} = 1$. A vector $\hat{l}^*$ satisfying (2.2.2) and (2.2.3) with $\hat{\rho} = 1$ is called a normal Lagrange multiplier for problem (P_1).

Conditions (2.2.1) and (2.2.2) are respectively called the nontriviality and the complementary slackness condition.

Because of the basic nature of this theorem, we shall discuss in a formal way the main lines of the proof.

In the derivation of optimality conditions for the solutions of nonlinear programming problems we are faced with the basic problem of translating the characterization of the optimality of the solution of the problem into an operational set of rules. The way in which this translation is carried out is by making use of conical approximations to the set of feasible points and the set of directions in which the objective function decreases.

A vector $\tilde{u}$ is called a *direction of decrease* of the functional J at the point $\hat{u}$, if there exists a neighborhood $S(\tilde{u},\epsilon_0)$ of the vector $\tilde{u}$ and a number $\alpha=\alpha(J,\hat{u},\tilde{u})$, $\alpha > 0$, such that

$$J(\hat{u}+\epsilon u) \leqslant J(\hat{u}) - \epsilon\alpha \quad \text{for all } \epsilon:0<\epsilon<\epsilon_0, \text{ for all } u \in S(\tilde{u},\epsilon_0). \tag{2.2.5}$$

The set of all directions of decrease at $\hat{u}$, is an open cone $D(J,\hat{u})$ with vertex at zero (cf. Girsanov (1972)). †

Using the definition of the cone of admissible directions to M at $\hat{u}$ and of the sequential tangent cone of $S^{-1}(K)$ at $\hat{u}$, the local optimality property of the solution $\hat{u}$ implies the following condition (cf. Girsanov (1972)) :

$$D(J,\hat{u}) \cap A(M,\hat{u}) \cap T(S^{-1}(K),\hat{u}) = \varnothing, \tag{2.2.6}$$

which states that at a (local) solution point $\hat{u}$ there cannot be a direction of decrease, that is also an admissible direction to the set M at $\hat{u}$ and which is also a tangent direction of the set $S^{-1}(K)$ at $\hat{u}$.

The Abadie condition (2.1.14) is now used to replace (2.2.6) by a more tractable expression :

$$D(J,\hat{u}) \cap A(M,\hat{u}) \cap L(S,K,\hat{u}) = \varnothing. \tag{2.2.7}$$

This completes the conical approximation of the optimization problem, where the sets $D(J,\hat{u})$ and $A(M,\hat{u})$ are open convex cones, and $L(S,K,\hat{u})$ is a (not necessarily open) convex cone.

Condition (2.2.7) is not yet an operational rule. Thereto a further translation is necessary. In particular, the Dubovitskii-Milyutin lemma may be invoked, which is essentially a separating hyperplane theorem. It states that (Girsanov (1972), Lemma 5.11) :

Let $K_1,\ldots,K_n,K_{n+1}$ be convex cones with vertex at zero, where $K_1,\ldots,K_n$ are open. Then

$$\bigcap_{i=1}^{n+1} K_i = \varnothing,$$

if and only if there exist linear functionals $u_i^* \in K_i^*$, not all zero, such that

$$u_1^* + u_2^* \ldots\ldots u_n^* + u_{n+1}^* = 0. \tag{2.2.8}$$

Condition (2.2.3) is a translation of (2.2.8). In this translation, the Farkas condition (2.1.15) is used to establish a characterization of $L(S,K,\hat{u})^*$, which implies the properties (2.2.2) of $\hat{l}^*$.

† We note that strictly speaking, the cone $D(J,\hat{u})$ is only an open cone when the empty set is defined to be an open cone.

We now consider the implication that if (2.1.16) holds then the optimality of $\hat{u}$ implies the existence of nontrivial Lagrange multipliers. The Nonsingularity condition (2.1.16) deals with the convex cone $R(S'(\hat{u}))+C(K,S(\hat{u}))$. Because this set is not dense in L, the origin of L is not an interior point of the set and hence (cf. Luenberger (1969), p.133, Theorem 2) there is a closed hyperplane H containing 0, such that the cone $R(S'(\hat{u}))+C(K,S(\hat{u}))$ lies on one side of H. The element $\hat{l}^* \in L^*$ which defines such an hyperplane, satisfies (2.2.1) - (2.2.3) with $\hat{\rho}=0$.

The second part of Theorem 2.10 is proved by reversing the proof of the implication that (2.1.14) and (2.1.15) together imply the existence of nontrivial multipliers with $\hat{\rho}=0$. It can be shown that under the hypotheses of Theorem 2.10, assuming $\hat{\rho}=0$ yields always $\hat{l}^*=0$, and thus the pair $(\hat{\rho},\hat{l}^*)$ is not a pair of nontrivial Lagrange multipliers. Hence of any pair of nontrivial Lagrange multipliers the number $\hat{\rho}$ cannot be zero.

It is of interest to investigate the role of the constant $\hat{\rho}$, which is called the *regularity constant*. First, consider the case $\hat{\rho}=0$ (pathological case). In this case the *nontriviality condition* (2.2.1) implies $\hat{l}^* \neq 0$, which leaves us with a set of equations (2.2.2) - (2.2.3) involving only the constraints, and not the object functional of the specific problem. If $\hat{\rho}>0$, we may set $\hat{\rho}=1$, because of the homogenity of (2.2.2) - (2.2.3). Clearly in this case equations (2.2.2) and (2.2.3) involve the object functional of the problem. Much research has been devoted to conditions which imply $\hat{\rho}>0$. These conditions, which generally involve only the constraints of the problem, are usually called *constraint qualifications*.

In view of its structure, the set of equations (2.2.1) - (2.2.3) is called a *multiplier rule*. A constraint qualification restricts the multiplier rule as additional conditions are imposed on the problem. These conditions may exclude solutions to problems which admit a nonzero multiplier $\hat{\rho}$. There are also situations in which a constraint qualification may be difficult to validate, whereas the nontriviality condition may be used to establish the case $\hat{\rho}>0$.
Following this reasoning we are led to the definition of two types of multiplier rules, *intrinsic multiplier rules* ($\hat{\rho}\geq 0$) and *restricted multiplier rules* ($\hat{\rho}>0$) (cf. Pourciau (1980), (1983)). In our terminology, part (i) of Theorem 2.10 is an intrinsic multiplier rule, which becomes a restricted one if the conditions stated in part (ii) are added.

Necessary conditions for optimality for solutions to problem (EIP) may be derived from the optimality conditions for problem (P_1), presented in Theorem 2.10. To obtain these conditions for problem (EIP) we first make an intermediate step and consider the constraint operator of problem (P_1) $S : U \to L$, split up as $S = (S_1,S_2)$; $L = L_1 \times L_2$, such that $S_1 : U \to L_1$; $S_2 : U \to L_2$.

The operator S_1 is taken to represent the equality constraints, i.e.

$$S_1(\hat{u}) \in \{0\}.$$

The operator S_2 represents inequality constraints, i.e.

$$S_2(\hat{u}) \in K_2,$$

where K_2 is a closed convex cone having nonempty interior. Taking $K := \{0\} \times K_2$ in Theorem 2.10 leads directly to the following result :

Lemma 2.11 : *Let $\hat{u}$ be a local solution to problem (P_1), and $L = L_1 \times L_2$, $S = (S_1, S_2)$, $K = \{0\} \times K_2$.*

(i) *If int $K_2 \neq \emptyset$ and $R(S_1'(\hat{u}))$ is not a proper dense subspace of L_1, then there exist nontrivial Lagrange multipliers for problem (P_1) at $\hat{u}$.*

(ii) *If*

$$R(S_1'(\hat{u})) = L_1, \tag{2.2.9}$$

$$\{S'_2(\hat{u})u : S'_1(\hat{u})u = 0\} \cap int\, C(K_2, S_2(\hat{u})) \neq \emptyset, \tag{2.2.10}$$

and

$$A(M, \hat{u}) \cap L(S, K, \hat{u}) \neq \emptyset, \tag{2.2.11}$$

then, a normal Lagrange multiplier exist for problem (P_1) at $\hat{u}$.

For a proof see Kurcyusz (1976), Theorem 4.4 and Corollary 4.2.

Using this result we are led to the following multiplier rule for problem (EIP), which has the form of an abstract minimum principle (cf. Neustadt (1969)).

Theorem 2.12 : *Let $\hat{x}$ be a solution to problem (EIP).*

(i) *If*

$$R(h'(\hat{x})) = \text{closed}, \tag{2.2.12}$$

then, there exist a real number $\hat{\rho}$, an $\hat{y}^* \in Y^*$, $\hat{z}^* \in Z^*$, such that :

$$(\hat{\rho}, \hat{y}^*, \hat{z}^*) \neq (0,0,0), \tag{2.2.13}$$

$$\hat{\rho} \geq 0, \tag{2.2.14}$$

$$\langle \hat{y}^*, \tilde{g}(\hat{x}) \rangle = 0, \tag{2.2.15}$$

$$\langle \hat{y}^*, y \rangle \geq 0 \text{ for all } y \in B, \tag{2.2.16}$$

$$[\hat{\rho}\tilde{f}'(\hat{x}) - \hat{y}^*\tilde{g}'(\hat{x}) - \hat{z}^*\tilde{h}'(\hat{x})](x - \hat{x}) \geq 0 \text{ for all } x \in A. \tag{2.2.17}$$

(ii) *The multiplier $\hat{\rho}$ is not zero, when*

$$R(\tilde{h}'(\hat{x})) = Z, \tag{2.2.18}$$

and, in addition, there is some $x \in int\, A$, such that

$$\tilde{h}'(\hat{x})(x - \hat{x}) = 0, \tag{2.2.19}$$

and

$$\tilde{g}(\hat{x}) + \tilde{g}'(\hat{x})(x - \hat{x}) \in int\, B. \tag{2.2.20}$$

Proof : Let $U = X, M = A, L_1 = Z, L_2 = Y, K_2 = B, S_1 = \tilde{h}, S_2 = \tilde{g}$.

Consider first part (i). By definition of problem (EIP), the cone K_2 has nonempty interior. By Lemma 2.11, there exist nontrivial Lagrange multipliers, when $R(S_1'(\hat{u}))$ is not a proper dense subspace of L_1. We shall show that this is the case, whenever this set is closed. Thereto we consider two cases : $R(S_1'(\hat{u})) = L_1$ and $R(S_1'(\hat{u})) \neq L_1$. In the first case the condition is satisfied, because the subspace is not proper. In the second case the condition is satisfied because the subspace cannot be dense in L_1, i.e.

$$R(S_1'(\hat{u})) = \overline{R(S_1'(\hat{u}))} \neq L_1$$

This proves the existence of Lagrange multipliers, or equivalently the conditions (2.2.1) - (2.2.3) of Theorem 2.10. In order to translate these into the conditions (2.2.13) - (2.2.17) we identify $\hat{l}^* = (\hat{z}^*, \hat{y}^*)$. Now consider the relations (2.2.2)

$$\hat{l}^* \in K^* \quad \textit{and} \quad <\hat{l}^*, S(\hat{u})> = 0.$$

In the present situation the dual cone of K is :

$$K^* = \{(y^*, z^*) \in (Y^* \times Z^*) : <z^*, 0> \geq 0, <y^*, y> \geq 0 \ \textit{for all} \ y \in B\},$$

which reduces trivially to :

$$K^* = \{(y^*, z^*) \in (Y^* \times Z^*) : <y^*, y> \geq 0 \ \textit{for all} \ y \in B\}.$$

The relation (2.2.2) thus translates directly into (2.2.15) and (2.2.16). To derive (2.2.17) recall condition (2.2.3) :

$$\hat{\rho} J'(\hat{u}) - S'(\hat{u})^* \hat{l}^* \in A(M, \hat{u})^*.$$

The set $A(M, \hat{u})^*$ is equal with $\overline{A(M, \hat{u})^*}$, if M has nonempty interior (cf. Girsanov (1972), Lemma 5.3). Now (2.2.3) becomes :

$$<\hat{\rho} J'(\hat{u}) - S'(\hat{u})^* \hat{l}^*, u> \geq 0 \ \textit{for all} \ u \in \overline{A(M, \hat{u})},$$

which, by definition of the adjoint operator, is equivalent to :

$$<\hat{\rho} J'(\hat{u}) - \hat{l}^* S'(\hat{u}), u> \geq 0 \ \textit{for all} \ u \in \overline{A(M, \hat{u})}.$$

Identification of the various terms in the terminology of problem (EIP) yields :

$$[\hat{\rho} \tilde{f}'(\hat{x}) - \hat{y}^* \tilde{g}'(\hat{x}) - \hat{z}^* \tilde{h}'(\hat{x})]\tilde{x} \geq 0 \ \textit{for all} \ \tilde{x} \in \overline{A(A, \hat{x})}. \qquad (2.2.21)$$

Here $A(A, \hat{x}))$ is the cone of admissible directions of a convex set with nonempty interior and hence (cf. Girsanov (1972)) :

$$A(A, \hat{x}) = \{\lambda(x - \hat{x}) : x \in \textit{int}\ A, \lambda \geq 0\}.$$

The closure of this set contains the set :

$$\{\lambda(x - \hat{x}) : x \in A, \lambda \geq 0\}.$$

Taking elements $\tilde{x} = x - \hat{x}$ in (2.2.21) yields (2.2.17).

Now consider part (ii). Condition (2.2.18) is a direct translation of condition (2.2.9) of Lemma 2.11. Restating (2.2.10) in terms of problem (EIP), we obtain :

$$\tilde{g}'(\hat{x})(N(\tilde{h}'(\hat{x}))) \cap \textit{int}\ C(B, \tilde{g}(\hat{x})) \neq \emptyset,$$

which is equivalent to (cf. Kurcyusz (1976), eq.(33); Zowe (1978), Theorem 3.2; Zowe (1980)) :

$$\exists x \in X : \tilde{h}'(\hat{x})x = 0 \wedge \tilde{g}(\hat{x}) + \tilde{g}'(\hat{x})x \in \textit{int}\ B. \qquad (2.2.22)$$

Now consider (2.2.11) :

$$A(M, \hat{u}) \cap L(S, K, \hat{u}) \neq \emptyset,$$

which becomes in terms of problem (EIP) :

$$\exists x \in A(A,\hat{x}) : \tilde{h}'(\hat{x})x = 0 \wedge \tilde{g}(\hat{x}) + \tilde{g}'(\hat{x})x \in B. \qquad (2.2.23)$$

Clearly, (2.2.19) - (2.2.20) are a sufficient condition under which both (2.2.22) and (2.2.23) hold. It should be noted that instead of part (ii) of Theorem 2.12 a somewhat stronger theorem could be stated. This would however yield also a more complicated statement.

□

2.3. Second order optimality conditions in Banach space.

In the previous section we considered optimality conditions of first order, i.e. only the first Fréchet derivatives of the mappings involved in the definition of the optimization problem considered, were taken into account. In this section we shall consider optimality conditions of second order, i.e. the second Fréchet derivatives of the mappings will also be used for the derivation of optimality conditions.

The notion of second Fréchet derivatives is somewhat more complicated than that of first Fréchet derivatives. Consider for instance the mapping $J : U \to \mathbb{R}$ of problem (P_1). Its first Fréchet derivative at $u \in U$ is denoted $J'(u)$ and its Fréchet differential, denoted δJ, is

$$\delta J(u ; \delta u) = J'(u)\delta u = \langle J'(u), \delta u \rangle \quad \text{for all } \delta u \in U. \qquad (2.3.1)$$

Equation (2.3.1) reveals that $J'(u)$ can be interpreted as an element of the dual space U^*. Using this interpretation we obtain :

$$J'(\cdot) : U \to U^*. \qquad (2.3.2)$$

It is this interpretation that is used to define the second Fréchet derivative of J, i.e. the second Fréchet derivative of J is the first Fréchet derivative of the mapping $J'(\cdot)$.

The second Fréchet differential of J at u, denoted $\delta^2 J$, becomes :

$$\delta^2 J(u ; \delta u_1, \delta u_2) = J''(u)(\delta u_1)(\delta u_2)$$
$$= \langle J''(u)\delta u_1, \delta u_2 \rangle \quad \text{for all } \delta u_1, \delta u_2 \in U. \qquad (2.3.3)$$

The form (2.3.3) leads to two different interpretations of $J''(u)$, i.e.

$$J''(u)(\cdot) : U \to U^*, \qquad (2.3.4)$$

and

$$J''(u)(\cdot)(\cdot) : U \times U \to \mathbb{R}. \qquad (2.3.5)$$

The interpretation of (2.3.4) is the interpretation of $J''(u)$ as a linear mapping from the space U into its dual, whereas the interpretation (2.3.5) is a bilinear mapping from the productspace $U \times U$ to the space $\mathbb{R}$. Using (2.3.4) concepts like invertibility of $J''(u)$ can be defined, whereas (2.3.5) may be used to define concepts like positive definiteness.

Thusfar we have considered a real valued mapping J, i.e. $J : U \to \mathbb{R}$. The interpretation of the second Fréchet derivative of $S : U \to L$ is even more complicated. For our purposes, however, it suffices to consider only Fréchet derivatives of mappings of the form

$$l^* S(u) = \langle l^*, S(u) \rangle, \qquad (2.3.6)$$

where l^* is a bounded linear functional on the space L, so that

$$l^* S(\cdot) : U \to \mathbb{R}, \tag{2.3.7}$$

is a real valued mapping.

We now return to the subject of optimality conditions for problem (P_1).
The purpose of considering second order optimality conditions, is to augment the set of first order conditions in some way. This leads quite naturally to the investigation of directions which satisfy the first order optimality conditions.
To simplify such an investigation, we use a somewhat more tractable form than (2.2.1) - (2.2.3) for the optimality conditions by assuming :

$$\hat{\rho} > 0, \tag{2.3.8}$$

$$M = U \text{ i.e. } A(M, \hat{u})^* = \{0\}. \tag{2.3.9}$$

The reason for (2.3.8) is obvious, $\hat{\rho} = 0$ corresponds to pathological types of problems, involving only the constraints of the problem. The reason for (2.3.9) is that this leads to a suprisingly simple form of the set of directions which satisfy (2.2.1) - (2.2.3). For the closed convex cone $K \subset L$ and the bounded linear functional l^* on L, the set

$$K(K, l^*) := K \cap \{l \in L : \langle l^*, l \rangle = 0\}. \tag{2.3.10}$$

is defined. We note that when K is a closed convex cone, then $K(K, l^*)$ is also a closed convex cone.

Lemma 2.13 : *In the terminology of problem (P_1) with $M = U$, when $\hat{l}^*$ is a normal Lagrange multiplier for problem (P_1) at $\hat{u}$ (cf. Theorem 2.10, part(ii)), then the linearizing cone of $S^{-1}(K(K, \hat{l}^*))$ at $\hat{u}$, i.e.*

$$L(S, K(K, \hat{l}^*), \hat{u}), \tag{2.3.11}$$

contains all directions δu such that

$$J'(\hat{u})\delta u = 0, \tag{2.3.12}$$

$$S(\hat{u}) + S'(\hat{u})\delta u \in K, \tag{2.3.13}$$

$$\langle \hat{l}^*, S(\hat{u}) + S'(\hat{u})\delta u \rangle = 0. \tag{2.3.14}$$

Proof : Using Definition 2.8 the inclusion $\delta u \in L(S, K(K, \hat{l}^*), \hat{u})$ is equivalent to

$$S'(\hat{u})\delta u \in C(K(K, \hat{l}^*), S(\hat{u})). \tag{2.3.15}$$

Because K is a cone with vertex at zero, $K(K, \hat{l}^*)$ is also a cone with vertex at zero. Using (2.1.11), (2.3.15) becomes :

$$\exists \lambda \in \mathbb{R}^+ : \lambda S(\hat{u}) + S'(\hat{u})\delta u \in K(K, \hat{l}^*). \tag{2.3.16}$$

Because $\hat{l}^*$ is a normal Lagrange multiplier, the following relations hold :

$$S(\hat{u}) \in K \qquad \langle \hat{l}^*, S(\hat{u}) \rangle = 0, \tag{2.3.17}$$

$$J'(\hat{u}) - \hat{l}^* S'(\hat{u}) = 0. \tag{2.3.18}$$

Combination of (2.3.16), (2.3.17) and the fact that K is a cone gives :

$$S'(\hat{u})\delta u \in K \text{ and } \langle \hat{l}^*, S'(\hat{u})\delta u \rangle = 0, \tag{2.3.19}$$

which proves (2.3.13) and (2.3.14).

(2.3.18) is equivalent to :

$$J'(\hat{u})\delta u = \hat{l}^* S'(\hat{u})\delta u.$$

Combination with (2.3.19) gives (2.3.12).

□

The interpretation of the set $K(K,\hat{l}^*)$ leads us to consider the minimization of the Lagrangian

$$L(u,l^*) := J(u) - l^* S(u), \tag{2.3.20}$$

at $l^* = \hat{l}^*$, over the set $K(K,\hat{l}^*)$, i.e.

$$S(u) \in K(K,\hat{l}^*). \tag{2.3.21}$$

Following the same path as in the previous section, we may derive optimality conditions for the minimization problem corresponding to (2.3.20) - (2.3.21).

As a result of the nonlinearity of the constraint (2.3.21), this derivation involves also a Abadie-type of constraint qualification, which becomes :

$$L(S,K(K,\hat{l}^*),\hat{u}) = T(S^{-1}(K(K,\hat{l}^*),\hat{u})). \tag{2.3.22}$$

Obviously, the first order optimality conditions for this minimization problem will not yield more information about properties of the solution of problem (P_1), than the first order optimality conditions for problem (P_1), stated in the previous section. The first order optimality conditions do show however, that the Lagrange multiplier corresponding to constraint (2.3.21) is zero and hence the minimization of the Lagrangian (2.3.20) seems not to be restricted by the constraint (2.3.21). This leads quite naturally to the consideration of the second Fréchet derivative of the Lagrangian (2.3.20) on the set $K(K,\hat{l}^*)$. In the following theorem second order necessary conditions for optimality for problem (P_1) with $M = U$ are summarized.

Theorem 2.14 : *Let $\hat{u}$ be a (local) solution to problem (P_1) with $M = U$ and let $\hat{l}^*$ be a normal Lagrange multiplier for problem (P_1) with $M = U$. If condition (2.3.22) is satisfied at $(\hat{u},\hat{l}^*)$, then*

$$L''(\hat{u},\hat{l}^*)(\delta u)(\delta u) \geq 0 \quad \textit{for all} \quad \delta u \in L(S,K(K,\hat{l}^*),\hat{u}). \ \dagger \tag{2.3.23}$$

For a proof of this theorem we refer to Hestenes (1975) (see also Maurer et al. (1979)). Note that a more explicit form of the variations δu in (2.3.23) is given in Lemma 2.13.

Using the interpretation of the second Fréchet derivative of the Lagrangian as a bilinear mapping, we see that (2.3.23) states that the second Fréchet derivative of the Lagrangian is positive semi-definite on $L(S,K(K,\hat{l}^*),\hat{u})$, i.e. on the subspace spanned by the linearized constraints at $\hat{u}$.

Theorems 2.10 and 2.14 are involved with *necessary conditions for optimality* for solutions to problem (P_1), i.e. they are of the form

"If $\hat{u}$ is a (local) solution to problem (P_1), then 'certain conditions' must hold."

† The first and second Fréchet derivatives of the Lagrangian L with respect to the argument u and for fixed l^* are denoted L' and L''.

In other words, the (local) optimality property of a solution implies certain conditions. As a consequence of this, we are not sure whether a point $\hat{u}$, which satisfies the necessary conditions for optimality is, or is not, a solution to problem (P_1).

This question leads us to the consideration of conditions for which the implication above is reversed, i.e. conditions which imply optimality. The general form of these conditions is :

"*If 'certain conditions' hold at* $\hat{u}$, *then* $\hat{u}$ *is a local solution to problem* (P_1)."

These conditions are referred to as *sufficient conditions for optimality*.

The ideal situation would be that the conditions of Theorems 2.10 and 2.14, which are necessary for optimality are also sufficient for optimality. However, this is only true for special cases of problem (P_1) and not for the general (nonlinear) problem (P_1).

Sufficient conditions for optimality which are of practical importance involve the second Fréchet derivatives of the mappings involved in the definition of problem (P_1).

The derivation of second order sufficient conditions for optimality in the case of infinite-dimensional space U, turns out to be quite complicated. However, the result, which is stated in the theorem below, has a relatively simple connection with the second order necessary conditions for optimality.

Theorem 2.15 : *Let* $\hat{u}$ *be a point for which* $S(\hat{u}) \in K$ *is satisfied and* $\hat{l}^*$ *be a normal Lagrange multiplier for problem* (P_1) *with* $M = U$ *at the point* $\hat{u}$. *Suppose that condition (2.2.14) is satisfied and that there are a* $\delta > 0$ *and a* $\beta > 0$ *such that*

$$L''(\hat{u}, \hat{l}^*)(\delta u)(\delta u) \geqslant \delta \|\delta u\|^2 \quad \text{for all} \quad \delta u \in \{h \in U : S(\hat{u}) + S'(\hat{u})h \in K \wedge$$

$$\hat{l}^*(S(\hat{u}) + S'(\hat{u})h) \leqslant \beta \|h\|\}. \qquad (2.3.24)$$

then $\hat{u}$ *is a local solution to problem* (P_1) *with* $M = U$.

For a proof of this result the reader should consult Maurer et al. (1979).

A comparison of the condition of Theorems 2.14 and 2.15 reveals that the sufficient conditions are a strengthened form of the full set of necessary conditions. A formal interpretation of Theorem 2.15 is that the second Fréchet derivative of the Lagrangian (2.3.20) must be sufficiently positive definite on a slightly enlarged constraint set.

We note that for finite-dimensional U the condition of Theorem 2.15 may be strengthened to :

"*The second Fréchet derivative of the Lagrangian must be positive definite on* $L(S, K(K, \hat{l}^*), \hat{u})$,"

i.e. the $\geqslant$ sign in (2.3.23) is replaced by $>$ (cf. Maurer et al. (1979), Lemma 5.7).

As in the previous section, we are interested in deriving optimality conditions for problem (EIP), which is essentially a special case of problem (P_1). Therefore we shall apply the results of Theorems 2.14 and 2.15 to this case. Both theorems deal with the case that the constraint set M equals U. Correspondingly, we shall consider problem (EIP) with $A = X$.

The theorem below is a direct consequence of Lemma 2.3 and Theorems 2.14 and 2.15. We note that the Lagrangian for problem (EIP) becomes

$$L(x, y^*, z^*) := \tilde{f}(x) - y^* \tilde{g}(x) - z^* \tilde{h}(x). \tag{2.3.25}$$

Theorem 2.16 :

(i) *Let $\hat{x}$ be a local solution to problem (EIP) with $A = X$, for which both part (i) and (ii) of Theorem 2.12 hold with $\hat{y}^*$ and $\hat{z}^*$. If*

$$R(\tilde{g}'(\hat{x})) = Y, \tag{2.3.26}$$

then

$$L''(\hat{x}, \hat{y}^*, \hat{z}^*)(\delta x)(\delta x) \geqslant 0 \text{ for all } \delta x \in \{\bar{x} \in X : \tilde{g}(\hat{x}) + \tilde{g}'(\hat{x})\bar{x} \in B \wedge \tilde{h}'(\hat{x})\bar{x} = 0 \wedge \hat{y}^*(\tilde{g}(\hat{x}) + \tilde{g}'(\hat{x})\bar{x}) = 0\}. \tag{2.3.27}$$

(ii) *Conversely, if*

$$R(\tilde{h}'(\hat{x})) = Z, \tag{2.3.28}$$

and

$$\exists \bar{x} \in X : \tilde{h}'(\hat{x}) = 0 \wedge \tilde{g}(\hat{x}) + \tilde{g}'(\hat{x})\bar{x} \in \text{int } B, \tag{2.3.29}$$

and $\hat{x}$ satisfies

$$\tilde{g}(\hat{x}) \in B, \tag{2.3.30}$$

$$\tilde{h}(\hat{x}) = 0, \tag{2.3.31}$$

and there exist multipliers $\hat{y}^*$ and $\hat{z}^*$ satisfying

$$\langle \hat{y}^*, y \rangle \geqslant 0 \text{ for all } y \in B, \tag{2.3.32}$$

$$\langle \hat{y}^*, \tilde{g}(\hat{x}) \rangle = 0, \tag{2.3.33}$$

$$L'(\hat{x}, \hat{y}^*, \hat{z}^*) = 0, \tag{2.3.34}$$

and there are a $\delta > 0$ and a $\beta > 0$ such that

$$L''(\hat{x}, \hat{y}^*, \hat{z}^*)(\delta x)(\delta x) \geqslant \delta \|\delta x\|^2 \text{ for all } \delta x \in \{\bar{x} \in X : \tilde{g}(\hat{x}) + \tilde{g}'(\hat{x})\bar{x} \in B \wedge \tilde{h}'(\hat{x})\bar{x} = 0 \wedge \hat{y}^*(\tilde{g}(\hat{x}) + \tilde{g}'(\hat{x})\bar{x}) \leqslant \beta \|x\|\}. \tag{2.3.35}$$

then $\hat{x}$ is a local solution to problem (EIP).

A proof of this theorem is omitted because it follows in all but one aspect directly from Lemma 2.13 and Theorems 2.14 and 2.15. The only aspect which requires some explanation is the constraint qualification (2.3.26). This is a result of the constraint qualification (2.3.22) in Theorem 2.14. One may easily verify that the cone $K(K, \hat{l}^*)$ has no interior when $\hat{l} \neq 0$. A sufficient condition for (2.3.22) to hold in this case is (2.3.26). We note that it is possible to state a less explicit, but stronger result. For our purposes however, (2.3.26) suffices.

3. Optimal control problems with state inequality constraints.

3.1. Statement and discussion of the problem.

In this thesis, the following type of State Constrained Optimal Control Problem (SCOCP) will be considered :

Problem (SCOCP) : *Determine a control function* $\hat{u} \in L_\infty[0,\hat{T}]^m$, *a state trajectory* $\hat{x} \in W_{1,\infty}[0,\hat{T}]^n$ *and a final time* $\hat{T} > 0$, *which minimize the functional*

$$h_0(x(0)) + \int_0^T f_0(x(t),u(t),t)\,dt + g_0(x(T),T), \tag{3.1.1}$$

subject to the constraints :

$$\dot{x}(t) = f(x(t),u(t),t) \qquad \text{a.e. } 0 \leqslant t \leqslant T, \tag{3.1.2}$$

$$D(x(0)) = 0, \tag{3.1.3}$$

$$E(x(T),T) = 0, \tag{3.1.4}$$

$$u(t) \in U \qquad \text{a.e. } 0 \leqslant t \leqslant T, \tag{3.1.5}$$

$$S_1(x(t),u(t),t) \leqslant 0 \qquad \text{a.e. } 0 \leqslant t \leqslant T, \tag{3.1.6}$$

$$S_2(x(t),t) \leqslant 0 \qquad 0 \leqslant t \leqslant T, \tag{3.1.7}$$

where : $h_0 : \mathbb{R}^n \to \mathbb{R}$; $f_0 : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R} \to \mathbb{R}^n$; $g_0 : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}$; $D : \mathbb{R}^n \to \mathbb{R}^c$; $f : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R} \to \mathbb{R}^n$; $E : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}^q$; $S_1 : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R} \to \mathbb{R}^{k_1}$; $S_2 : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}^{k_2}$; $U \subset \mathbb{R}^m$, *is a convex set with nonempty interior.*

$$\text{For all } x \in \mathbb{R}^n, u \in \mathbb{R}^m \quad \text{rank } S_{1u}(x,u,t) = k_1 \quad \text{a.e. } 0 \leqslant t \leqslant T. \dagger \tag{3.1.8}$$

The functions h_0, f_0, g_0, f, D, E S_1 *and* S_2 *are twice continuously differentiable functions with respect to all arguments.*

$$W_{1,\infty}[0,T]^n := \{ x \text{ is an absolute continuous } n\text{-vector function on } [0,T] \text{ with } \dot{x} \in L_\infty[0,T]^n \}.$$

A motivation for problem (SCOCP) is given in the discussion below.

We assume that the dynamic behaviour of the system to be controlled, can be described by a set of ordinary differential equations of the form :

$$\dot{x}(t) = f(x(t),u(t),t) \qquad \text{for all } 0 \leqslant t \leqslant T, \tag{3.1.9}$$

where x is an n-vector function on $[0,T]$ called the *state variable* and u is an m-vector function on $[0,T]$ called the *control variable*.

We are interested in problems where the system is to be controlled from an *initial state* x_0 at time $t = 0$, i.e.

† This condition may be weakened to a more complicated condition, which involves only the gradients of the components of S_1 on intervals where these components are active, i.e. where these components are zero on an interval, along a solution trajectory.

$$x(0) = x_0, \tag{3.1.10}$$

over an interval $[0,T]$. The number T is used to denote the *final time*. We shall assume that T is finite, which means that we are interested in problems with *finite time horizon*.

One of the more difficult technical details of the statement of the problem are the conditions that the control function $u : [0,T] \to \mathbb{R}^m$ must satisfy. In view of the fact that we want to identify the optimal control problem as a specialization of the abstract nonlinear programming problem (EIP), it is desirable to identify u as a vector in a function space. Because u governs the state variable via the right hand side of the set of differential equations (3.1.9), u must be at least integrable (in the sense of Lebesgue) on $[0,T]$. A sufficient condition for this is that u is measurable and essentially bounded on $[0,T]$ (see e.g. Kolmogorov et al. (1961) or Rudin (1976)).
Therefore it is possible to identify u as an element of the space of m-vector functions which are *measurable* and *essentially bounded* on $[0,T]$, which is denoted by $L_\infty[0,T]^m$.

We note that the space $L_\infty[0,T]^m$ is particular well suited for the statement of optimal control problems, which are to be identified as specializations of abstract nonlinear programming problems in Banach space with Fréchet differentiable mappings. This is due to the fact that when more general control functions would be allowed, either the space of control functions is not a Banach space or the mappings involved are not Fréchet differentiable. When the type of control functions would be restricted further, it is possible to identify the optimal control problem as a specialization of problem (EIP) only in the case that the control is assumed to be a continuous function on $[0,T]$. Simple examples exist that show that controls which are solutions to the rather general type of optimal control problems that we want to consider, can be discontinuous.

As a result of the smoothness assumptions on the function f, we have

$$f(x(\cdot),u(\cdot),\cdot) \in L_\infty[0,T]^n,$$

whenever $u \in L_\infty[0,T]^m$ and x is a continuous function on $[0,T]$. Because elements of $L_\infty[0,T]$ which differ on a set of zero Lebesgue measure are regarded as equivalent, the differential equation (3.1.9), which is an equality relation between two vectors in $L_\infty[0,T]$, is allowed to differ on a set of zero Lebesgue measure. We note that because the differential equation must only hold almost everywhere on $[0,T]$, the differential equation is interpreted as the integral equation :

$$x(t) = x(0) + \int_0^t f(x(\tau),u(\tau),\tau)\, d\tau.$$

The state variable x can also be identified as a vector in some function space. Because x is always a continuous function on $[0,T]$, x can be identified as an element of the *space of continuous functions on* $[0,T]$, denoted by $C[0,T]^n$. This space however, contains also vectors that cannot be a solution to any differential equation, because there exist continuous functions which are not the integral of their derivatives. This would complicate the application of the results on optimality conditions, stated in Chapter 2, unnecessary (cf. Section 3.3.1). The space of *absolutely continuous functions on* $[0,T]$ with *measurable* and *essentially bounded (first) time derivatives*, denoted by $W_{1,\infty}[0,T]^n$, is more suitable for our purpose.

As to the explicit dependence of the left hand side of (3.1.9) on the time t, we introduce the following terminology. When f does not depend explicitly on t, the system (3.1.9) is called *autonomous* and when it does *nonautonomous*.

A nonautonomous system may be transformed into an autonomous one by means of an additional state variable. Let y satisfy

$$y(0) = 0,$$

$$\dot{y}(t) = 1 \qquad a.e.\ 0 \leqslant t \leqslant T,$$

then

$$y(t) = t \qquad 0 \leqslant t \leqslant T.$$

Substituting y for t in (3.1.9) yields an autonomous system.

An other distinction is made between *variable final time* problems, i.e. T is not fixed in advance and *fixed final time* problems. It is possible to transform variable time problems into fixed final time problems via a standard approach, which again requires the introduction of an additional state variable (cf. Section 3.3.4).

From a theoretical point of view, there is no objection to the introduction of additional state variables to transform nonautonomous and variable final time problems into autonomous, fixed final time problems. However, in the numerical method to be proposed, all state variables are treated similar and therefore an increase in the dimension of the state vector gives an increase in numerical effort. Because there is no great difficulty in dealing with nonautonomous and variable final time problems directly, they are included in the formulation of problem (SCOCP).

The foregoing discussion focussed on the specification of the differential system. Now we shall consider the specification of the object criterion, which is done by means of a functional which assigns a real value to each triple (x, u, T), called the *objective function*. The following forms are of common use in optimal control theory

$$\int_0^T f_0(x(t), u(t), t)\, dt, \tag{3.1.11}$$

$$g_0(x(T), T), \tag{3.1.12}$$

$$\int_0^T f_0(x(t), u(t), t)\, dt + g_0(x(T), T). \tag{3.1.13}$$

Again from a theoretical point of view, there is no great difference between working with either one of (3.1.11), (3.1.12) or (3.1.13), when the functions f_0 and g_0 are sufficiently smooth. This is because an objective function of the form (3.1.11) can be transformed into the form (3.1.12) and vice versa. From a practical point of view it does matter which form of objective function is used, because the transformation from (3.1.11) to (3.1.12) requires the introduction of an additional state variable, whereas the transformation from (3.1.12) to (3.1.11) may lead to complicated expressions for the objective function. Therefore (3.1.13) is assumed, which covers both the forms (3.1.11) and (3.1.12).

Having discussed the specification of the differential system and the objective function, we now turn to the specification of the constraints, which restrict the solution of the optimal control problems.

In most optimal control problems, there are constraints on the final state of the system, i.e. the state $x(T)$ must satisfy certain conditions. These constraints are called *terminal point constraints*. A general way of specifying these conditions, is by means of a vector function $E : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}^q$, with $q \leqslant n+1$, of the form

$$E(x(T),T) = 0.$$

It is obvious that this formulation includes fixed final time and fixed final state problems.

In most cases the initial state of the system (3.1.9) is known completely and specified in the form of (3.1.10). There are however problems, where the initial state of the system is not specfied completely in advance. To tackle this type of problems the initial state is specified similar to the way in which the terminal state in specified, i.e. using a vector function $D : \mathbb{R}^n \to \mathbb{R}^c$, with $c \leqslant n$ such that,

$$D(x(0)) = 0.$$

Of course the specification (3.1.10) is included in this formulation. A logical extension of (3.1.13) is now to consider an objective function of the form (3.1.1).

Beside terminal point constraints, most optimal control problems include constraints on the control u and the state x, which must hold at all time points of the interval $[0,T]$. A distinction is made between the following types of constraints :

Control constraints :	$u(t) \in U$	*a.e.* $0 \leqslant t \leqslant T$,
Mixed control state constraints :	$S_1(x(t),u(t),t) \leqslant 0$	*a.e.* $0 \leqslant t \leqslant T$,
State constraints :	$S_2(x(t),t) \leqslant 0$	$0 \leqslant t \leqslant T$.

In most cases, control constraints can be written as a set of inequalities and therefore this type of constraints could also be treated as mixed control state constraints.

For example, let $U := \{u : 0 \leqslant u \leqslant \bar{u}\}$. Then the constraint $u \in U$ may be replaced by $S_1(u) = -u(\bar{u}-u) \leqslant 0$.

When optimal control problems are solved analytically, this approach involves unnecessary effort. However, with a numerical solution of the problem, this approach is quite useful, because in a numerical context we need an explicit expression for the set U. Therefore an explicit dependence of the function S_1 on the argument x is not supposed.

A similar argumentation for the state constraints would imply that the state constraints are a subclass of the mixed control state constraints. For the solution of the problem however, it is essential to make the distinction between mixed control state - and state constraints. One might say that a distinction must be made between the *explicit constraints* on the control by way of the *mixed control state constraints* and the *implicit constraints* on the control by way of the *state constraints*. The explicit dependence of the function S_1 on the argument u is certified by means of Assumption (3.1.8).

The functions h_0, f_0, g_0, f, D, E, S_1 and S_2, which define the optimal control problem are called *problem functions*. Most optimal control problems involve problem functions which are at least continuous with respect to their arguments. When we want to identify the problem (SCOCP) as a specialization of the abstract nonlinear programming problem (EIP), we need that the mappings involved in problem (EIP) are at least twice continuously Fréchet differentiable. A requirement for this is that all problem functions are at least twice continuously differentiable with respect to all their arguments (cf. Section 3.2).

If we consider a problem with variable final time for which the control variable and the state variable are to be identified as elements of function spaces, e.g. $u \in L_\infty[0,T]^m$ and $x \in W_{1,\infty}[0,T]^n$, then we have to deal with the technical detail that the function spaces depend on the parameter T, i.e. on the final time. Via this dependence, the functions x and u depend on T. This makes the abstract formulation difficult, if not impossible. Fortunately, it is possible to transform any variable final time problem into a fixed final time problem. Using this transformation approach, optimality conditions for variable final time problems can be derived from the optimality conditions for the transformed fixed final time problem (cf. Section 3.3.4).

3.2. Formulation of problem (SCOCP) as a nonlinear programming problem in Banach spaces.

This section deals with the formulation of problem (SCOCP) as an abstract nonlinear programming problem (EIP). In this formulation, problem (SCOCP) will be treated as an optimal control problem with fixed final time. The optimality conditions for the case that problem (SCOCP) has variable final time will be derived from the optimality conditions for the case of fixed final time (cf. Section 3.3.4).

A basic choice has to be made, as to the manner in which the differential system (3.1.2) is treated. There are two possibilities, either the control variable is considered as the only variable of the optimal control problem, or both the control variable and the state variable are considered as variables of the optimal control problem. In the former approach the state variable is treated as a quantity which depends on u via (3.1.2). Following the latter approach, (3.1.2) enters the formulation of the optimal control problem as an equality constraint. We prefer the latter approach because, as will follow from the discussion in the next section, it leads to a weaker constraint qualification. In addition, the approach extends in a logical way to the numerical method which is described in Chapters 4, 5 and 6.

Thus, we consider in the formulation of problem (EIP) as variables the pair (x,u). The space X becomes the product space of the spaces which contain the variables x and u, i.e.

$$X = W_{1,\infty}[0,T]^n \times L_\infty[0,T]^m. \tag{3.2.1}$$

In the formulation of problem (EIP), the assumption is made that X is a Banach space. We shall show that with the selection of a suitable norm on X this assumption is satisfied. In general, the space X cannot be expected to be a Banach space unless the spaces $W_{1,\infty}[0,T]^n$ and $L_\infty[0,T]^m$ are both Banach spaces.

For every measurable and essentially bounded function $v : [0,T] \to \mathbb{R}^w$, the *$\infty$-norm* is defined by :

$$\|v\|_\infty := \operatorname*{ess\,sup}_{0 \le t \le T} \|v(t)\|, \tag{3.2.2}$$

where $\|\cdot\|$ is the Euclidian vector norm on $\mathbb{R}^w$.

Equipped with the ∞-norm the space $L_\infty[0,T]^m$ is a Banach space.

Analogously, the space $W_{1,\infty}[0,T]^n$ is a Banach space when equipped with the norm

$$\|x\|_{1,\infty} = \max\{\|x\|_\infty, \|\dot{x}\|_\infty\} \quad \textit{for all} \quad x \in W_{1,\infty}[0,T]^n.$$

(cf. Kirsch et al. (1978), p.91-92).

The space X is now a product of Banach spaces for which we may use the following rule to select a norm :

"*X_1 and X_2 are Banach spaces with norms $\|\cdot\|_{X_1}$ and $\|\cdot\|_{X_2}$, the norm on $X_1 \times X_2$ is taken as* $\max\{\|\cdot\|_{X_1}, \|\cdot\|_{X_2}\}$."

With this norm, the space $X_1 \times X_2$ is also a Banach space. Using this rule we obtain as norm on X :

$$\|(x,u)\|_X := \max\{\|x\|_\infty, \|\dot{x}\|_\infty, \|u\|_\infty\}. \qquad (3.2.3)$$

The formulation of the objective function of problem (EIP) follows directly from the objective function of problem (SCOCP).

$$\tilde{f}(x,u) := h_0(x(0)) + \int_0^T f_0(x(t),u(t),t)\,dt + g_0(x(T),T). \qquad (3.2.4)$$

The smoothness assumptions on the problem functions h_0, f_0 and g_0, together with the fact that the norm on the space X is an ∞-norm, yield the following result.

Lemma 3.1 : *Let the functions h_0, f_0 and g_0 satisfy the assumptions of problem (SCOCP) and $\tilde{f} : X \to \mathbb{R}$ be defined by (3.2.4), then the mapping $\tilde{f}$ is twice Fréchet differentiable at all points (x,u) of X and*

$$\tilde{f}'(x,u)(\delta x,\delta u) = h_{0x}(x(0))\delta x(0) + \int_0^T (f_{0x}(x,u,t)\delta x(t) + f_{0u}(x,u,t)\delta u(t))\,dt + g_{0x}(x(T),T)\delta x(T). \qquad (3.2.5)$$

For a proof of this lemma we refer to the proof of Lemma 1.4a, p.94 of Kirsch et al. (1978), who prove that $\tilde{f}$ is once Fréchet differentiable. The second Fréchet differentiability follows from an application of the same lemma to (3.2.5) for fixed $(\delta x, \delta u)$.

The constraints (3.1.2) - (3.1.4) enter the formulation of the abstract problem as equality constraints. This leads to the following formulation of the mapping $\tilde{h}$:

$$\tilde{h}(x,u) := (\dot{x}(\cdot) - f(x(\cdot),u(\cdot),\cdot), D(x(0)), E(x(T),T)). \qquad (3.2.6)$$

To make the formulation of the mapping $\tilde{h}$ complete, we have to identfy the range space Z of $\tilde{h}$, which must be a Banach space. A logical choice for Z is :

$$Z = L_\infty[0,T]^n \times \mathbb{R}^c \times \mathbb{R}^q, \qquad (3.2.7)$$

which equipped with the norm

$$\|(z_1,z_2,z_3)\|_Z = \max\{\|z_1\|_\infty, \|z_2\|, \|z_3\|\} \text{ for all } z_1 \in L_\infty[0,T]^n, z_2 \in \mathbb{R}^c, z_3 \in \mathbb{R}^q, \qquad (3.2.8)$$

is indeed a Banach space.

With regard to the Fréchet differentiability of $\tilde{h}$ we have the following lemma :

Lemma 3.2 : *Let the functions f , D and E satisfy the assumptions of problem (SCOCP) and let the mapping $\tilde{h} : X \to Z$ be defined by (3.2.6), then the mapping $\tilde{h}$ is twice continuously Frèchet differentiable for all (x,u) of X and,*

$$\tilde{h}'(x,u)(\delta x,\delta u) = (\delta\dot{x}(\cdot) - f_x(x(\cdot),u(\cdot),\cdot)\delta x(\cdot) - f_u(x(\cdot),u(\cdot),\cdot)\delta u(\cdot),$$
$$D_x(x(0))\delta x(0) \,,\, E_x(x(T),T)\delta x(T)). \qquad (3.2.9)$$

This lemma is a direct extension of Lemma 1.4b, p.94 of Kirsch et al. (1978).

In the abstract formulation, the inequality constraints of problem (SCOCP) take the form of a required membership of a set A and a restriction of the value of a mapping $\tilde{g}$ to a cone B.

The set A is used to formulate the control constraint (3.1.5) :

$$A := W_{1,\infty}[0,T]^n \times A_u \qquad (3.2.10)$$

where

$$A_u := \{u \in L_\infty[0,T]^m : u(t) \in U \quad a.e. \quad 0 \leqslant t \leqslant T\}. \qquad (3.2.11)$$

Because U is assumed to be a convex set with a nonempty interior, A_u is also a convex set with a nonempty interior.

The mixed control state constraints (3.1.6) and the state constraints (3.1.7) are formulated as :

$$\tilde{g}(x,u) := (S_1(x(\cdot),u(\cdot),\cdot) \,,\, S_2(x(\cdot),\cdot)). \qquad (3.2.12)$$

A logical choice for the range space Y is :

$$Y := L_\infty[0,T]^{k^1} \times C[0,T]^{k^2}. \qquad (3.2.13)$$

Equipped with the norm

$$\|(y_1,y_2)\|_Y := \max\{\|y_1\|_\infty, \|y_2\|_\infty\} \text{ for all } y_1 \in L_\infty[0,T]^{k^1}, y_2 \in C[0,T]^{k^2}. \qquad (3.2.14)$$

To the choice of the range space Y we note that an alternative choice is $L_\infty[0,T]^{k^1} \times W_{1,\infty}[0,T]^{k^2}$. However, the choice (3.2.13) is preferred because the space $C[0,T]^{k^2}$ has a standard representation of the elements of the dual space (cf. Luenberger (1969)). We note that unfortunately, the representation of the elements of the dual space of $L_\infty[0,T]$ is rather complicated and that there seems to be no suitable alternative for the choice of the range space of the operator $S_1(x(\cdot),u(\cdot),\cdot)$. This complicates the application of the optimality conditions, stated in Chapter 2, to the state constrained optimal control problem, as discussed in Section 3.3.2.

Lemma 3.3 : *Let the functions S_1 and S_2 satisfy the assumptions of problem (SCOCP) and the mapping $\tilde{g} : X \to Y$ be defined by (3.2.12), then the mapping $\tilde{g}$ is twice continuously Frèchet differentiable for all (x,u) of X and*

$$\tilde{g}'(x,u)(\delta x,\delta u) = (S_{1x}(x(\cdot),u(\cdot),\cdot)\delta x(\cdot) + S_{1u}(x(\cdot),u(\cdot),\cdot)\delta u(\cdot),$$
$$S_{2x}(x(\cdot),\cdot)\delta x(\cdot)). \qquad (3.2.15)$$

To make the abstract formulation of the inequality constraints complete, we have to specify the cone B, which in the formulation of problem (EIP), is assumed to be closed and

convex, with $0 \in B$ and having nonempty interior.

If we choose B to be :

$$B := B_1 \times B_2, \tag{3.2.16}$$

$$B_1 := \{y_1 \in L_\infty[0,T]^{k_1} : y_{1i}(t) \leqslant 0 \quad a.e. \quad 0 \leqslant t \leqslant T, i = 1,...,k_1\}, \tag{3.2.17}$$

$$B_2 := \{y_2 \in C[0,T]^{k_2} : y_{2i}(t) \leqslant 0 \quad 0 \leqslant t \leqslant T, i = 1,...,k_2\}, \tag{3.2.18}$$

then one can easily verify that the cone B statisfies the assumptions of problem (EIP).

This completes the formulation of the optimal control problem (SCOCP) as a specialization of the abstract nonlinear programming problem (EIP).

3.3. First order optimality conditions for problem (SCOCP).

3.3.1. Regularity conditions for problem (SCOCP).

In view of the application of Theorem 2.12 to the optimal control problem (SCOCP) in the formulation of Section 3.2, we consider the regularity conditions of parts (i) and (ii) of Theorem 2.12.

We start off by noting that throughout this chapter we shall use the following standard result on linear ordinary differential equations (e.g. cf. Hermes et al. (1969), p.36).

Lemma 3.4 : *Let $A(t)$ be an $n \times n$ matrix defined on $[0,T]$ with components $a_{ij} \in L_\infty[0,T]$ (all $i, j = 1,...n$), then for all $h \in L_\infty[0,T]^n$ the ordinary differential equation*

$$\dot{x}(t) - A(t)x(t) = h(t) \quad a.e. \quad 0 \leqslant t \leqslant T, \tag{3.3.1.1}$$

$$x(0) = x_0, \tag{3.3.1.2}$$

has exactly one solution $x \in W_{1,\infty}[0,T]^n$. This solution has the form

$$x(t) = \Phi(t)x_0 + \Phi(t)\int_0^t \Phi^{-1}(s)h(s)\,ds \qquad 0 \leqslant t \leqslant T, \tag{3.3.1.3}$$

where the $n \times n$ matrix Φ is the fundamental matrix solution of (3.3.1.1), i.e. the unique solution to the homogeneous differential equation :

$$\dot{\Phi}(t) - A(t)\Phi(t) = 0, \tag{3.3.1.4}$$

$$\Phi(0) = I. \tag{3.3.1.5}$$

We note that the solution of (3.3.1.1) that satisfies the boundary condition $x(T) = x_T$ has the form :

$$x(t) = \Phi(t)\Phi^{-1}(T)x_T - \Phi(t)\int_t^T \Phi^{-1}(s)h(s)\,ds \qquad 0 \leqslant t \leqslant T. \tag{3.3.1.6}$$

As a first step towards the derivation of regularity conditions for problem (SCOCP), we consider the range of the Fréchet derivative of the mapping $\tilde{h} : X \to Z$, at a solution $(\hat{x}, \hat{u})$ of problem (SCOCP).

Lemma 3.5 :

(i) *Let the functions f, D and E satisfy the assumptions of problem (SCOCP) and let the mapping* $\tilde{h}$ *be defined by (3.2.6), then*

$$R(\tilde{h}'(\hat{x},\hat{u})) = \text{closed}. \tag{3.3.1.7}$$

(ii) *If at* $(\hat{x},\hat{u})$,

$$\text{rank}\,(D_x(\hat{x}(0))) = c, \tag{3.3.1.8}$$

and

$$\text{rank}\,(E_x(\hat{x}(T),T)) = q, \tag{3.3.1.9}$$

then

$$R(\tilde{h}'(\hat{x},\hat{u})) = Z. \tag{3.3.1.10}$$

Proof : Using Lemma 3.4 we first prove that the range of the operator $\tilde{h}_1'(\hat{x},\hat{u}) : X \to L_\infty[0,T]^n$, with

$$\tilde{h}_1'(\hat{x},\hat{u})(\delta x,\delta u) := (\delta\dot{x}(\cdot) - f_x[\cdot]\delta x(\cdot) - f_u[\cdot]\delta u(\cdot)). \quad \dagger \tag{3.3.1.11}$$

is $L_\infty[0,T]^n$. For this purpose we consider the equation

$$\tilde{h}_1'(\hat{x},\hat{u})(\delta x,\delta u) = h, \tag{3.3.1.12}$$

with $h \in L_\infty[0,T]^n$. The range of the mapping $\tilde{h}_1'$ equals $L_\infty[0,T]^n$ if and only if equation (3.3.1.12) has a solution $(\delta x,\delta u) \in X$ for every $h \in L_\infty[0,T]^n$. Using (3.3.1.11) equation (3.3.1.12) is equivalent to :

$$\delta\dot{x} - f_x\delta x - f_u\delta u = h, \tag{3.3.1.13}$$

which has a solution for each $h \in L_\infty[0,T]^n$ by Lemma 3.4. ($\delta x(0)$ and δu can be set to zero.)

Part (i) of the Lemma follows, because the ranges of the operators $D_x(\hat{x}(0))(\cdot) : X \to \mathbb{R}^c$ and $E_x(\hat{x}(T),T)(\cdot) : X \to \mathbb{R}^q$ are always closed, due to the fact that the range spaces of these operators are finite-dimensional.

Part (ii) follows directly from (i) and the fact that (3.3.1.8) and (3.3.1.9) imply

$$R(D_x(\hat{x}(0))) = \mathbb{R}^c,$$

$$R(E_x(\hat{x}(T),T)) = \mathbb{R}^q.$$

□

Part (i) of Lemma 3.5 enables the application of part (i) of Theorem 2.12 to problem (SCOCP) without any additional regularity conditions on the problem. With regard to the result contained in part (ii), we note that this is the weaker form of the constraint qualification we promised at the start of Section 3.2. For if we would have treated x as a quantity dependent on u, condition (3.3.1.10) would require, beside (3.3.1.8) and (3.3.1.9), that the linearized system

† The notation $[\cdot]$ is used to replace $(\hat{x}(\cdot),\hat{u}(\cdot),\cdot)$ or $(\hat{x}(\cdot),\cdot)$.

$$\delta\dot{x} = f_x \delta x + f_u \delta u ,$$

should be completely controllable on $[0,T]$ (cf. Norris (1973)).
We note that we do not need this controllability as a result of the fact that we consider both x and u as variables and that the differential equation was used directly as a constraint, instead of first transforming the differential equation into an integral equation. When both x and u are used as variables, but when the differential equation would first be transformed into an integral equation and x was considered to be an element of the space of continuous functions, then the controllability of the linearized system would also be required (cf. Girsanov (1972), Assumption 9.1).

The theorem below is a specialization of the constraint qualification of part (ii) of Theorem 2.12 for problem (SCOCP).

Theorem 3.6 : *Let $(\hat{x},\hat{u})$ be a solution to problem (SCOCP). When*

$$rank\,(D_x(\hat{x}(0))) = c , \qquad (3.3.1.14)$$

and

$$rank\,(E_x(\hat{x}(T),T)) = q , \qquad (3.3.1.15)$$

and, in addition, there is a pair $(\delta x,\delta u)$ for which †

$$\hat{u}(t) + \delta u(t) \in int\ U \qquad a.e.\ 0 \leqslant t \leqslant T , \qquad (3.3.1.16)$$

$$D_x[0]\delta x(0) = 0, \qquad (3.3.1.17)$$

$$\delta\dot{x}(t) = f_x[t]\delta x(t) + f_u[t]\delta u(t) \qquad a.e.\ 0 \leqslant t \leqslant T , \qquad (3.3.1.18)$$

$$E_x[T]\delta x(T) = 0, \qquad (3.3.1.19)$$

$$S_1[t] + S_{1x}[t]\delta x(t) + S_{1u}[t]\delta u(t) < 0 \qquad a.e.\ 0 \leqslant t \leqslant T , \qquad (3.3.1.20)$$

$$S_2[t] + S_{2x}[t]\delta x(t) < 0 \qquad 0 \leqslant t \leqslant T , \qquad (3.3.1.21)$$

then the regularity constant $\hat{\rho}$ is not zero.

Proof : The hypotheses (3.3.1.14) and (3.3.1.15) imply by Lemma 3.5, (3.3.1.10). Equations (3.3.1.16) - (3.3.1.21) are counterpart to conditions (2.2.19) - (2.2.20) of part (ii) of Theorem 2.12.
□

3.3.2. Representation of the Lagrange multipliers of problem (SCOCP).

In this section we shall consider the representation of the Lagrange multipliers for solutions of problem (SCOCP). In the abstract formulation of problem (EIP) these multipliers are denoted as $\hat{y}^*$ and $\hat{z}^*$. In the case of problem (SCOCP) they can be expressed as elements of function spaces. The major problem we have to deal with is the fact that, the elements of the dual space of $L_\infty[0,T]$ do not admit a simple standard representation.

In establishing a formulation of problem (SCOCP) in the terminology of problem (EIP) (cf. Section 3.2), the range spaces of the constraints, i.e. Y and Z were chosen to be pro-

† We used the notation $[t]$ to replace $(\hat{x}(t),\hat{u}(t),t)$ or $(\hat{x}(t),t)$.

ducts of Banach spaces. A particular choice of the norm on the product spaces was made in such a way as to make the product spaces Banach spaces too. In this case the representation of linear functionals on these product spaces is induced by the components, i.e. when X_1 and X_2 are both Banach spaces and

$$X_s = X_1 \times X_2,$$

then all continuous linear functionals on X_s admit a representation of the form (cf. Porter (1966), p.299) :

$$<x_s^*, x_s> = <x_1^*, x_1> + <x_2^*, x_2>,$$

with $x_1^* \in X_1^*$ and $x_2^* \in X_2^*$.

We shall now develop a representation of the Lagrange multipliers for problem (SCOCP) by considering the products $<\hat{y}^*, \tilde{g}>$ and $<\hat{z}^*, \tilde{h}>$, where $\tilde{g}$ and $\tilde{h}$ are the mappings defined in Section 3.2. Using the fact that Y and Z are product spaces we obtain :

$$<\hat{y}^*, \tilde{g}> = <\hat{\eta}_1, S_1[\cdot]> + <\hat{\xi}, S_2[\cdot]>, \qquad (3.3.2.1)$$

$$<\hat{z}^*, \tilde{h}> = <\hat{\lambda}, \dot{\hat{x}} - f[\cdot]> + <\hat{\sigma}, D(\hat{x}(0))> + <\hat{\mu}, E(\hat{x}(T),T)>, \qquad (3.3.2.2)$$

with : $\hat{\eta}_1 \in (L_\infty[0,T]^{k_1})^*$,
$\hat{\xi} \in (C[0,T]^{k_2})^*$,
$\hat{\lambda} \in (L_\infty[0,T]^n)^*$,
$\hat{\sigma} \in (\mathbb{R}^c)^*$,
$\hat{\mu} \in (\mathbb{R}^q)^*$.

Equations (3.3.2.1) and (3.3.2.2) admit an interpretation of $(\hat{\eta}_1, \hat{\xi}, \hat{\lambda}, \hat{\sigma}, \hat{\mu})$ as Lagrange multipliers associated with a particular constraint (i.e. $\hat{\eta}_1$ is associated with the constraint $S_1(\hat{x}(\cdot), \hat{u}(\cdot), \cdot) \in B_1$).

A representation of the Lagrange multipliers for problem (SCOCP) will be established, once we have a representation for the linear functionals on the right hand side of (3.3.2.1) and (3.3.2.2). These will be considered individually. We start with the representation of the linear functionals which do not pose a problem as they have a standard representation.

Because $\mathbb{R}^c$ and $\mathbb{R}^q$ are Hilbert spaces, the linear functionals on $\mathbb{R}^c$ and $\mathbb{R}^q$ have the form :

$$<\hat{\sigma}, D(\hat{x}(0))> = -\hat{\sigma}^T D(\hat{x}(0)), \qquad (3.3.2.3)$$

$$<\hat{\mu}, E(\hat{x}(T),T)> = -\hat{\mu}^T E(\hat{x}(T),T), \qquad (3.3.2.4)$$

with : $\hat{\sigma} \in \mathbb{R}^c$,
$\hat{\mu} \in \mathbb{R}^q$.

The dual space of $C[0,T]^{k_2}$ is the space $NBV[0,T]^{k_2}$,i.e. the normalized space of k_2-vector functions on $[0,T]$ of bounded variation (cf. Luenberger (1969), p.113-115). The standard representation of these linear functional is given by means of a Stieltjes integral, i.e.

$$<\hat{\xi}, S_2(\hat{x}(\cdot), \cdot)> = -\int_0^T S_2(\hat{x}(t),t)^T \, d\hat{\xi}(t), \qquad (3.3.2.5)$$

with : $\hat{\xi} \in NBV[0,T]^{k_2}$.

We note that the minus signs on the right hand sides of (3.3.2.3) - (3.3.2.5) were chosen in order to obtain the usual form of the minimum principle to be stated in the next section.

The representation of the functionals

$$<\hat{\eta}_1, S_1(x(\cdot),u(\cdot),\cdot)>, \tag{3.3.2.6}$$

and

$$<\hat{\lambda}, \dot{x}(\cdot)-f(x(\cdot),u(\cdot),\cdot)>, \tag{3.3.2.7}$$

is a more difficult problem, because the linear funcionals on $L_\infty[0,T]^{k_1}$ and $L_\infty[0,T]^n$ are elements of $L_\infty[0,T]^*$ and as such admit, in general, only a very complicated representation (cf. Dunford et al. (1958), Ch. IV, Thm. 8.16).

Fortunately, by making use of the fact that $\hat{\eta}_1$ and $\hat{\lambda}$ are Lagrange multipliers for problem (SCOCP) we are able to derive a practically useful representation of the functionals (3.3.2.6) and (3.3.2.7).

We shall first consider the representation of the functional (3.3.2.6). Here we are faced with the difficulty that the constraint $S_1(\hat{x}(\cdot),\hat{u}(\cdot),\cdot)\in B_1$ represents only in part the explicit constraints on the control. The other part is represented by the constraint $\hat{u}\in A_u$, which is a very general representation of a constraint. In order to cope with this difficulty we shall make the following assumption :

Assumption 3.7 : *The set U is of the form :*

$$U = \{u\in \mathbb{R}^m : S_0(u) \leqslant 0\},$$

where $S_0 : \mathbb{R}^m \to \mathbb{R}^{k_0}$ *is a twice continuously differentiable mapping.*

Assumption 3.7 merely states that the control constraints can be transformed into a set of inequalities, i.e.

$$u(t)\in U \qquad a.e.\ \ 0\leqslant t\leqslant T,$$

may be replaced by

$$S_0(u(t)) \leqslant 0 \qquad a.e.\ \ 0\leqslant t\leqslant T.$$

Because we did not make any assumptions about the explicit dependence of $S_1(x,u,t)$ on the argument x, all explicit constraints on the control can be treated in a similar manner. Thus, we end up with one vector function for the constraints on u,

$$S_c(x,u,t) = \begin{bmatrix} S_0(u) \\ S_1(x,u,t) \end{bmatrix} \tag{3.3.2.8}$$

The solution must now satisfy the following constraint :

$$S_c(x(t),u(t),t) \leqslant 0 \qquad a.e.\ \ 0\leqslant t\leqslant T. \tag{3.3.2.9}$$

As we already discussed in Section 3.1, we must futhermore assume that all components of the vector function S_c have an explicit dependence on the argument u.

Assumption 3.8 : *If* $(\hat{x},\hat{u})$ *is a solution to problem (SCOCP) and Assumption 3.7 holds, then* †

$$rank\,(S_{cu}(\hat{x}(t),\hat{u}(t),(t)) = k_0 + k_1 \qquad a.e.\ \ 0 \leqslant t \leqslant T.$$

Assumptions 3.7 and 3.8 enable the derivation of a representation of the linear functional $<\hat{\eta}_1,\cdot>$.

Lemma 3.9 : *Let* $(\hat{x},\hat{u})$ *be a solution to problem (SCOCP) and let in addition Assumptions 3.7 and 3.8 hold, then the linear functional* $<\hat{\eta}_1,\cdot>$, *whose existence is garanteed by Theorem 2.12, has the following representation :*

$$<\hat{\eta}_1, y_1> = -\int_0^T \hat{\eta}_1(t)^T y_1(t)\,dt \quad \textit{for all}\ \ y_1 \in L_\infty[0,T]^{k_1}, \qquad (3.3.2.10)$$

with : $\hat{\eta}_1 \in L_\infty[0,T]^{k_1}$.

Proof : Using the fact that Assumption 3.7 holds, we consider the formulation of problem (SCOCP) with the vector function (3.3.2.8). The corresponding Lagrange multiplier is denoted by $\hat{\eta}_c$.

Using the representation of the Lagrange multipliers discussed earlier in this section, we obtain from part (i) of Theorem 2.12 :

$$\hat{\rho}\tilde{f}'(\hat{x},\hat{u})(\delta x,\delta u) - <\hat{\eta}_c, S_{cx}\delta x + S_{cu}\delta u> - <\hat{\xi}, S_{2x}\delta x> -$$
$$<\hat{\lambda}, \delta\dot{x} - f_x\delta x - f_u\delta u> - <\hat{\sigma}, D_x\delta x(0)> -$$
$$<\hat{\mu}, E_x\delta x(T)> = 0 \quad \textit{for all}\ \ \delta x \in W_{1,\infty}[0,T]^n,\ \delta u \in L_\infty[0,T]^m. \qquad (3.3.2.11)$$

Using the representations (3.3.2.3) - (3.3.2.5) and the result of Lemma 3.1 we obtain :

$$<\hat{\lambda}, \delta\dot{x} - f_x\delta x - f_u\delta u> + <\hat{\eta}_c, S_{cx}\delta x + S_{cu}\delta u> = \hat{\rho}(h_{0x}\delta x(0) +$$
$$\int_0^T (f_{0x}\delta x + f_{0u}\delta u)\,dt + g_{0x}\delta x(T)) + \int_0^T \delta x^T S_{2x}^T\,d\hat{\xi} + \hat{\sigma}^T D_x\delta x(0) +$$
$$\hat{\mu}^T E_x\delta x(T) \quad \textit{for all}\ \ \delta x \in W_{1,\infty}[0,T]^n,\ \delta u \in L_\infty[0,T]^m. \qquad (3.3.2.12)$$

We shall consider (3.3.2.12) using variations $(\delta x,\delta u)$ that satisfy :

$$\delta\dot{x} = f_x\delta x + f_u\delta u \qquad a.e.\ \ 0 \leqslant t \leqslant T,$$
$$\delta x(0) = 0.$$

For these variations the functional $<\hat{\lambda}, \delta\dot{x} - f_x\delta x - f_u\delta u>$ is zero and the right hand side of (3.3.2.12) then gives an explicit relation for the functional $<\hat{\eta}_c, S_{cx}\delta x + S_{cu}\delta u>$.

Next we consider the functions :

$$h(t) = S_{cx}[t]\,\delta x(t) + S_{cu}[t]\,\delta u(t) \qquad a.e.\ \ 0 \leqslant t \leqslant T. \qquad (3.3.2.13)$$

Clearly, $h \in L_\infty[0,T]^{k_0+k_1}$, because $\delta u \in L_\infty[0,T]^m$. Assumption 3.8 ascertains that for every $h \in L_\infty[0,T]^{k_0+k_1}$, there is at least one δu, that satisfies equation (3.3.2.13). To select for each fixed function $h \in L_\infty[0,T]^{k_0+k_1}$ a particular function δu that satisfies (3.3.2.13), we

† In Assumption 3.8 we used S_{cu} to denote the partial derivative of S_c with respect to u.

make use of the pseudoinverse of the matrix $S_{cu}[t]$. Because the matrix $S_{cu}[t]$ is of full row rank, the pseudoinverse of $S_{cu}[t]$ has the form :

$$S_{cu}[t]^+ = S_{cu}[t]^T (S_{cu}[t] S_{cu}[t]^T)^{-1} \qquad a.e. \quad 0 \leqslant t \leqslant T. \qquad (3.3.2.14)$$

The variation δu must therefore satisfy :

$$\delta u(t) = S_{cu}[t]^+ (h(t) - S_{cx}[t]\,\delta x(t)) \qquad a.e. \quad 0 \leqslant t \leqslant T. \qquad (3.3.2.15)$$

Because $(\delta x, \delta u)$ satisfy the linear system, the variations (δx) satisfy :

$$\delta \dot{x} = f_x \delta x + f_u S_{cu}^+ h - f_u S_{cu}^+ S_{cx} \delta x.$$

Using Lemma 3.1 we can write δx dependent on h as :

$$\delta x(t) = \Phi(t) \int_0^t \Phi(s)^{-1} f_u[s]\, S_{cu}[s]^+ h(s)\, ds \qquad 0 \leqslant t \leqslant T, \qquad (3.3.2.16)$$

where Φ is the solution of :

$$\dot{\Phi} - (f_x - f_u S_{cu}^+ S_{cx})\Phi = 0 \qquad \Phi(0) = I. \qquad (3.3.2.17)$$

Rewriting (3.3.2.12) with (3.3.2.15) and (3.3.2.16) yields :

$$<\hat{\eta}_c, h> = \int_0^T (a(t) \int_0^t B(s) h(s)\, ds + c(t) h(t))\, dt + e\,\Phi(T) \int_0^T B(t) h(t)\, dt +$$

$$\int_0^T (\Phi(t) \int_0^t B(s) h(s)\, ds)^T S_{2x}[t]^T\, d\xi(t) \quad \text{for all } h \in L_\infty[0,T]^{k_0+k_1}, \qquad (3.3.2.18)$$

where : $a(t) := \hat{\rho}(f_{0x} - f_{0u} S_{cu}[t]^+ S_{cx})\Phi(t), \qquad 0 \leqslant t \leqslant T$

$B(t) := \Phi(t)^{-1} f_u S_{cu}[t]^+, \qquad 0 \leqslant t \leqslant T$

$c(t) := \hat{\rho} f_{0u} S_{cu}[t]^+, \qquad 0 \leqslant t \leqslant T$

$e := \hat{\rho} g_{0x} + \hat{\mu} E_x.$

Changing the order of integration (cf. Luenberger (1969), p.153-154) :

$$\int_0^T \int_0^t K(t,s)\, ds\, dt = \int_0^T \int_t^T K(s,t)\, ds\, dt,$$

yields :

$$<\hat{\eta}_c, h> = \int_0^T \left\{ \left(\int_t^T a(s)\, ds + e\,\Phi(T) \right) B(t) + c(t) + \int_t^T d\xi(s)^T S_{2x}[s] \Phi(s) B(t) \right\}$$

$$h(t)\, dt \quad \text{for all } h \in L_\infty[0,T]^{k_0+k_1} \qquad (3.3.2.19)$$

The vector function $\hat{\eta}_c : [0,T] \to \mathbb{R}^{k_0+k_1}$ is now defined as :

$$\hat{\eta}_c(t)^T := -\left(\int_t^T a(s)\, ds + \int_t^T d\xi(s)^T S_{2x}[s] \Phi(s) + e\,\Phi(T) \right) B(t)$$

$$+ c(t) \qquad 0 \leqslant t \leqslant T. \qquad (3.3.2.20)$$

And hence :

$$<\hat{\eta}_c, h> = -\int_0^T \hat{\eta}_c(t)^T h(t)\,dt \quad \text{for all } h \in L_\infty[0,T]^{k_0+k_1},$$

which proves (3.3.2.10).

$\hat{\eta}_1 \in L_\infty[0,T]^{k_1}$ follows directly from inspection of the components of (3.3.2.20).

□

The proof of this Lemma is nonconstructive in the sense that we do not obtain a simple relation for $\hat{\eta}_1$, only a representation. For the multiplier $\hat{\lambda}$, we do obtain relations from the derivation of the representation, which follows similar lines as the proof of Lemma 3.9.

Lemma 3.10 : *Let $(\hat{x},\hat{u})$ be a solution to problem (SCOCP) and let, in addition, Assumptions 3.7 and 3.8 hold, then the linear functional $<\hat{\lambda},\cdot>$, whose existence is implied by Theorem 2.12, has the representation :*

$$<\hat{\lambda}, y> = \int_0^T \hat{\lambda}(t)^T y(t)\,dt \quad \text{for all } y \in L_\infty[0,T]^n, \tag{3.3.2.21}$$

with $\hat{\lambda} \in NBV[0,T]^n$, which satisfies

$$\hat{\lambda}(t_1)^T - \hat{\lambda}(t_0)^T = -\int_{t_0}^{t_1} (\hat{\rho} f_{0x}[t] + \hat{\lambda}(t)^T f_x[t] + \hat{\eta}_1(t)^T S_{1x}[t])\,dt$$

$$- \int_{t_0}^{t_1} d\hat{\xi}(t)^T S_{2x}[t] \quad \text{for all } 0 \leqslant t_0 \leqslant t_1 \leqslant T, \tag{3.3.2.22}$$

and

$$\hat{\lambda}(0)^T = -\hat{\rho} h_{0x}[0] - \hat{\sigma}^T D_x[0], \tag{3.3.2.23}$$

$$\hat{\lambda}(T)^T = \hat{\rho} g_{0x}[T] + \hat{\mu}^T E_x[T]. \tag{3.3.2.24}$$

Proof : We use equation (3.3.2.12), with variations $\delta u = 0$ and the representation of $<\hat{\eta}_1,\cdot>$ of Lemma 3.9 :

$$<\hat{\lambda}, \delta\dot{x} - f_x \delta x> = (\hat{\rho} h_{0x} + \hat{\sigma}^T D_x)\delta x(0) + \int_0^T (\hat{\rho} f_{0x} + \hat{\eta}_1^T S_{1x})\delta x(t)\,dt +$$

$$\int_0^T \delta x^T S_{2x}^T\, d\hat{\xi}(t) + (\hat{\rho} g_{0x} + \hat{\mu}^T E_x)\delta x(T) \quad \text{for all } \delta x \in W_{1,\infty}[0,T]^n. \tag{3.3.2.25}$$

Now consider :

$$\delta\dot{x} - f_x \delta x = h \qquad \delta x(0) = 0,$$

which has (by Lemma 3.1) a solution for every $h \in L_\infty[0,T]^n$,i.e.

$$\delta x(t) = \Phi(t) \int_0^t \Phi(s)^{-1} h(s)\,ds, \tag{3.3.2.26}$$

where Φ is as in Lemma 3.1.

Using relation (3.3.2.26) in (3.3.2.25) yields :

$$<\hat{\lambda}, h> = \int_0^T (\hat{\rho} f_{0x} + \hat{\eta}_1^T S_{1x})\Phi(t) \int_0^t \Phi(s)^{-1} h(s)\, ds +$$

$$\int_0^T d\hat{\xi}(t)^T S_{2x}[t]\Phi(t) \int_0^t \Phi(s)^{-1} h(s)\, ds +$$

$$(\hat{\rho} g_{0x} + \hat{\mu}^T E_x)\Phi(T) \int_0^T \Phi(t)^{-1} h(t)\, dt \quad \textit{for all } h \in L_\infty[0,T]^n . \qquad (3.3.2.27)$$

Changing the order of integration in (3.3.2.27) yields :

$$<\hat{\lambda}, h> = \int_0^T \{ \int_t^T (\hat{\rho} f_{0x} + \hat{\eta}_1^T S_{1x})\Phi(s)\, ds + \int_t^T d\hat{\xi}(s)^T S_{2x}[s]\Phi(s)$$

$$+ (\hat{\rho} g_{0x} + \hat{\mu}^T E_x)\Phi(T)\}\Phi(t)^{-1} h(t)\, dt \quad \textit{for all } h \in L_\infty[0,T]^n . \qquad (3.3.2.28)$$

Define now :

$$\hat{\lambda}(t)^T := \{ \int_t^T (\hat{\rho} f_{0x} + \hat{\eta}_1^T S_{1x})\Phi(s)\, ds + \int_t^T d\hat{\xi}(s)^T S_{2x}[s]\Phi(s)$$

$$+ (\hat{\rho} g_{0x} + \hat{\mu}^T E_x)\Phi(T)\}\Phi(t)^{-1} \qquad 0 \leqslant t \leqslant T , \qquad (3.3.2.29)$$

from which (3.3.2.21) directly follows. $\hat{\lambda} \in NBV[0,T]^n$ follows from an inspection of the various components of (3.3.2.29).

We shall next prove relations (3.3.2.22) and (3.3.2.24). Relation (3.3.2.24) follows from (3.3.2.29) for $t = T$. Now consider the product $\hat{\lambda}^T \Phi$:

$$d(\hat{\lambda}(t)^T \Phi(t)) = d\hat{\lambda}(t)^T \Phi(t) + \hat{\lambda}(t)^T \dot{\Phi}(t)\, dt. \qquad (3.3.2.30)$$

Because $\dot{\Phi}$ satisfies :

$$\dot{\Phi} = f_x \Phi,$$

equation (3.3.2.30) becomes

$$d(\hat{\lambda}^T \Phi) = d\hat{\lambda}^T \Phi + \hat{\lambda}^T f_x \Phi\, dt.$$

Using (3.3.2.29) we obtain :

$$d\hat{\lambda}^T \Phi + \hat{\lambda}^T f_x \Phi\, dt = -(\hat{\rho} f_{0x} + \hat{\eta}_1^T S_{1x})\Phi(t)\, dt - d\hat{\xi}(t)^T S_{2x}[t]\Phi(t).$$

Because Φ is invertible this yields :

$$d\hat{\lambda}^T = -(\hat{\rho} f_{0x} + \hat{\lambda}^T f_x + \eta_1^T S_{1x})\, dt - d\hat{\xi}(t)^T S_{2x}[t],$$

which is equivalent to (3.3.2.22), because

$$\int_{t_0}^{t_1} d\hat{\lambda}^T = \hat{\lambda}(t_1)^T - \hat{\lambda}(t_0)^T \quad \textit{for all } 0 \leqslant t_0 \leqslant t_1 \leqslant T.$$

To prove (3.3.2.23), the whole proof should be repreated using variations δx that satisfy :

$$\delta\dot{x} - f_x \delta x = h \qquad \delta x(T) = 0.$$

In this case the variations δx satisfy

$$\delta x(t) = -\Phi(t) \int_t^T \Phi(s)^{-1} h(s)\, ds.$$

The counterpart to (3.3.2.29) becomes

$$\hat{\lambda}(t)^T := -\{\int_0^t (\hat{\rho} f_{0x} + \hat{\eta}_1^T S_{1x})\Phi(s)\, ds + \int_0^t d\hat{\xi}(s)^T S_{2x}[s] + (\hat{\rho} h_{0x} + \hat{\sigma}^T D_x)\}\Phi(t)^{-1} \qquad 0 \leqslant t \leqslant T,$$

which yields (3.3.2.23) for $t = 0$, because $\Phi(0)^{-1} = I$.
□

3.3.3. Local minimum principle.

In this section, the results contained in part (i) of Theorem 2.12, will be expressed in the formulation of problem (SCOCP).

An important role is played by the *Hamiltonian*, which is defined as :

$$H(x,u,\rho,\lambda,t) := \rho f_0(x,u,t) + \lambda^T f(x,u,t). \tag{3.3.3.1}$$

In the theorem below the notation $[t]$ is used to replace $(\hat{x}(t),t)$, $(\hat{x}(t),\hat{u}(t),t)$ or $(\hat{x}(t),\hat{u}(t),\hat{\rho},\hat{\lambda}(t),t)$.

Theorem 3.11 : *If $(\hat{x},\hat{u})$ is a solution to problem (SCOCP) for which Assuptions 3.7 and 3.8 hold, then there exist a real number $\hat{\rho} \geqslant 0$, and vector functions $\hat{\lambda} \in NBV[0,T]^n$, $\hat{\eta}_1 \in L_\infty[0,T]^{k_1}$, $\hat{\xi} \in NBV[0,T]^{k_2}$ and vectors $\hat{\sigma} \in \mathbb{R}^c$, $\hat{\mu} \in \mathbb{R}^q$, not all zero, such that,*

$$\hat{\lambda}(t_1)^T - \hat{\lambda}(t_0)^T = -\int_{t_0}^{t_1} (H_x[t] + \hat{\eta}_1(t)^T S_{1x}[t])\, dt - \int_{t_0}^{t_1} d\hat{\xi}(t)^T S_{2x}[t] \qquad \text{for all } 0 \leqslant t_0 \leqslant t_1 \leqslant T, \tag{3.3.3.2}$$

$$\hat{\lambda}(0)^T = -\hat{\rho} h_{0x}[0] - \hat{\sigma}^T D_x[0], \tag{3.3.3.3}$$

$$\hat{\lambda}(T)^T = \hat{\rho} g_{0x}[T] + \hat{\mu}^T E_x[T], \tag{3.3.3.4}$$

$$(H_u[t] + \hat{\eta}_1(t)^T S_{1u}[t])(u - \hat{u}(t)) \geqslant 0 \qquad \text{for all } u \in U \quad \text{a.e. } 0 \leqslant t \leqslant T, \tag{3.3.3.5}$$

$$\hat{\eta}_1(t) \geqslant 0 \qquad \text{a.e. } 0 \leqslant t \leqslant T, \tag{3.3.3.6}$$

$$\hat{\eta}_{1i}(t) S_{1i}[t] = 0 \qquad \text{a.e. } 0 \leqslant t \leqslant T \quad i = 1,\ldots k_1, \tag{3.3.3.7}$$

$$\hat{\xi}_i(t) = \text{nondecreasing on } [0,T] \qquad i = 1,\ldots k_2, \tag{3.3.3.8}$$

$$\hat{\xi}_i(t) = \text{constant on intervals where } S_{2i}[t] < 0 \quad i = 1,\ldots k_2, \tag{3.3.3.9}$$

Proof : The existence of nontrivial Lagrange multipliers for problem (SCOCP) follows from part (i) of Theorem 2.12 and Lemma 3.5.

Using the representation of the Lagrange multipliers derived in Section 3.3.2, equation (2.2.17) becomes :

$$\hat{\rho}(h_{0x}[0]\delta x(0)+\int_0^T (f_{0x}[t]\delta x+f_{0u}[t]\delta u)\,dt+g_{0x}[T]\delta x(T))-$$

$$\int_0^T \hat{\lambda}(t)^T(\delta\dot{x}-f_x[t]\delta x-f_u[t]\delta u)\,dt+\int_0^T \hat{\eta}_1(t)^T(S_{1x}[t]\delta x+S_{1u}[t]\delta u)\,dt$$

$$+\int_0^T d\hat{\xi}(t)^T S_{2x}[t]\delta x(t)+\hat{\sigma}^T D_x[0]\delta x(0)+\hat{\mu}^T E_x[T]\delta x(T)\geqslant 0$$

$$\text{for all } \delta x\in W_{1,\infty}[0,T]^n,\ \hat{u}+\delta u\in A_u. \qquad (3.3.3.10)$$

Without loss of generality, the variations $(\delta x,0)$, $(0,\delta u)$ may be considered separately, because these variations are independent.

The variations $(\delta x,0)$ were used to derive the representation of the linear functional $<\hat{\lambda},\cdot>$ and hence (3.3.3.2) - (3.3.3.4) follow (cf. Section 3.3.2).

The variations $(0,\delta u)$ yield :

$$\int_0^T(\hat{\rho}f_{0u}[t]+\hat{\lambda}(t)^T f_u[t]+\hat{\eta}_1(t)S_{1u}[t])\delta u\ dt\geqslant 0 \quad \text{for all } \hat{u}+\delta u\in A_u. \qquad (3.3.3.11)$$

Equation (3.3.3.11) is equivalent to (3.3.3.5), because (3.3.3.11) is a supporting functional to the set A_u at the point $\hat{u}$ (cf. Girsanov (1972), p.76-77).

Equation (2.2.16) yields :

$$<\hat{y}^*,y>=-\int_0^T \hat{\eta}_1(t)y_1(t)\,dt-\int_0^T d\hat{\xi}(t)^T y_2(t)\geqslant 0 \quad \text{for all } y_1\in B_1,\ y_2\in B_2.$$

Considering the cases where all components of the vectors y_1 and y_2 are zero except one yields :

$$-\int_0^T \hat{\eta}_{1i}(t)y_{1i}(t)\,dt\geqslant 0 \quad \text{for all } y_{1i}\in L_\infty[0,T]$$

$$\text{with } y_{1i}(t)\leqslant 0 \quad a.e.\ 0\leqslant t\leqslant T \quad i=1,\ldots k_1,$$

and

$$-\int_0^T d\hat{\xi}_i(t)y_{2i}(t)\geqslant 0 \quad \text{for all } y_{2i}\in C[0,T]$$

$$\text{with } y_{2i}(t)\leqslant 0 \qquad 0\leqslant t\leqslant T \quad i=1,\ldots k_2,$$

which imply (3.3.3.6) and (3.3.3.8).

Equation (2.2.15) yields :

$$<\hat{y}^*, \tilde{g}(\hat{x},\hat{u})> = -\int_0^T \hat{\eta}_1(t)^T S_1[t]\,dt - \int_0^T d\hat{\xi}(t)^T S_2[t] = 0. \qquad (3.3.3.12)$$

Because of (3.3.3.6) and (3.3.3.8) and the fact that $S_1[t] \leqslant 0$ *a.e.* $0 \leqslant t \leqslant T$ and $S_2[t] \leqslant 0 \quad 0 \leqslant t \leqslant T$, equations (3.3.3.7) and (3.3.3.9) follow from (3.3.3.12).
□

The result contained in Theorem 3.11 is called a local minimum principle, as a result of equation (3.3.3.5), which implies that the function :

$$(H_u[t] + \hat{\eta}_1(t)^T S_{1u}[t])(u - \hat{u}(t)) \geqslant 0 \qquad (3.3.3.13)$$

is minimized almost everywhere on $[0,T]$ with respect to the argument u over values in the set U.

3.3.4. Minimum principle.

In this section optimality conditions for variable final time problems will be presented. At the same time the results of the previous section will be strengthened in the sense that the local character of the minimization of (3.3.3.13) will be replaced by a pointwise global minimization of the so-called augemented Hamiltonian over the entire set U.

The reason that such a result is desirable is that for *spike variations* (i.e. variations which are only nonzero over a small interval of time), the corresponding variation of the state variables and the objective function will be small. Obviously, spike variations need not be small in the ∞-norm. However, making the interval of time sufficiently small will make these variations comparable to variations which are small in the ∞-norm, but nonzero over a larger interval of time.

Theorem 3.12 : *If $(\hat{x},\hat{u},\hat{T})$ is a solution to problem (SCOCP), for which Assumptions 3.7 and 3.8 hold, then, in addition to (3.3.3.2) - (3.3.3.9) † the following conditions hold,*

$$H[t] = -\hat{\rho} g_{0T}[\hat{T}] - \hat{\mu}^T E_T[\hat{T}] - \int_t^{\hat{T}} (H_t[t] + \hat{\eta}_1(t)^T S_{1t}[t])\,dt$$

$$- \int_t^{\hat{T}} d\hat{\xi}(t)^T S_{2t}[t] \qquad a.e. \quad 0 \leqslant t \leqslant \hat{T}, \qquad (3.3.4.1)$$

and

$$H[t] = \max_{u \in U} H(\hat{x}(t),u,\hat{\rho},\hat{\lambda}(t),t) + \hat{\eta}_1(t)^T S_1(\hat{x}(t),u,t) \quad a.e. \quad 0 \leqslant t \leqslant \hat{T}. \qquad (3.3.4.2)$$

Proof : We shall only outline the main lines of the rigorous proof given by Girsanov (1972), Lectures 13 and 14.

Girsanov considers the case that the mixed control state constraints are not present and that the set of admissible controls U is not necessarily convex, nor is U supposed to have an interior. There is however no great difficulty in treating the present case of mixed control state constraints following entirely the same approach.

The essence of the proof is to admit spike variations on the control in an indirect way, via a *variable time transformation*.

† In these conditions the final time T must be replaced by $\hat{T}$.

This transformation has the following form :

$$t(\tau) := \int_0^{\tau} v(s)\,ds \qquad 0 \leqslant \tau \leqslant 1, \tag{3.3.4.3}$$

$$t(1) = \hat{T}, \tag{3.3.4.4}$$

$$v(\tau) \geqslant 0 \qquad a.e.\ \ 0 \leqslant \tau \leqslant 1. \tag{3.3.4.5}$$

The inverse of this transformation is defined as :

$$\tau(t) := inf\ \{\tau \in [0,1] : t(\tau) = t\ \}. \tag{3.3.4.6}$$

Using this transformation, problem (SCOCP) is transformed to an optimization problem involving the functions $x(\tau)$, $u(\tau)$ and $v(\tau)$, which are functions of the artificial time variable τ. In this transformed problem the function $v(\tau)$ is considered as an additional control variable on [0,1], which is to satisfy the control constraint (3.3.4.5).

In a formal notation the transformed problem is :

$$\underset{x,u,y,v}{Minimize}\ h_0(x(0)) + \int_0^1 f_0(x,u,y)v(\tau)\,d\tau + g_0(x(1),y(1)), \tag{3.3.4.7}$$

subject to :

$$\frac{dx}{d\tau} = v(\tau)f(x,u,y) \qquad a.e.\ \ 0 \leqslant \tau \leqslant 1, \tag{3.3.4.8}$$

$$\frac{dy}{d\tau} = v(\tau) \qquad a.e.\ \ 0 \leqslant \tau \leqslant 1, \tag{3.3.4.9}$$

$$D(x(0)) = 0, \tag{3.3.4.10}$$

$$y(0) = 0, \tag{3.3.4.11}$$

$$E(x(1),y(1)) = 0, \tag{3.3.4.12}$$

$$u(t) \in U \qquad a.e.\ \ 0 \leqslant \tau \leqslant 1, \tag{3.3.4.13}$$

$$v(\tau) \geqslant 0 \qquad a.e.\ \ 0 \leqslant \tau \leqslant 1, \tag{3.3.4.14}$$

$$S_1(x,u,y)v(\tau) \leqslant 0 \qquad a.e.\ \ 0 \leqslant \tau \leqslant 1, \tag{3.3.4.15}$$

$$S_2(x,y) \leqslant 0 \qquad 0 \leqslant \tau \leqslant 1. \tag{3.3.4.16}$$

As a result of the variable time transformation, the transformed problem is autonomous although the original problem can be nonautonomous.

If $v(\tau)$ is considered to be a fixed positive function on [0,1], then problems (SCOCP) and (3.3.4.7) - (3.3.4.16) are equivalent. If $v(\tau)$ is zero over an interval, the state variables x and y will be constant on this interval. On such an interval the value of the control function does not affect the value of the objective function, nor does it involve other constraints than $u \in U$. Following a similar reasoning for the case that $v(\tau)$ is considered to be a variable in the problem (3.3.4.7) - (3.3.4.16), the following result is obtained :

"If $(\hat{x}(t),\hat{u}(t))$ is a solution to problem (SCOCP), then for any function $\hat{v}(\tau)$ satisfying (3.3.4.3) - (3.3.4.5), the triple $(\hat{x}(\tau),\hat{u}(\tau),\hat{v}(\tau))$ is a solution to the transformed problem (3.3.4.7) - (3.3.4.16). The control $\hat{u}(\tau)$ is allowed to have any value satisfying $u \in U$ on intervals where $\hat{v}(\tau)$ is zero."

Because of the assumptions on the differentiability of the problem functions with respect to the argument t (cf. definition of problem (SCOCP)), application of the results of part (i) of Theorem 2.12 on the transformed problem is possible.

Assumptions 3.7 and 3.8 hold for the transformed problem on intervals where $v(\tau) > 0$, whenever these assumptions hold for problem (SCOCP). (Note that the transformed problem contains an additional control v with a constraint $v \geqslant 0$ which is independent of u.) The special form of the constraint (3.3.4.15) was chosen because we do not want to let the constraint $S_1(x,u,t) \leqslant 0$ restrict the choice of the values $\hat{u}(\tau)$ on intervals where $\hat{v}(\tau)$ is zero. As a result of this the regularity Assumption 3.8 does not hold on these intervals, because on these intervals the constraint vanishes completely from the optimization problem. For the representation of the Lagrange multipliers corresponding to the mixed control state constraints this poses no problem, because these Lagrange multiplier may be assigned an arbitrary value on intervals where $\hat{v}(\tau)$ vanishes (the constraints are no longer present on these intervals) and the regularity Assumption 3.8 is only of interest on intervals where $\hat{v}(\tau)$ is nonzero. The Lagrange multipliers correponding to the mixed control state constraints are assigned the following value :

$$\eta_1(\tau) := \eta_1(\tau(t)).$$

The application of the results of part (i) of Theorem 2.12 for variations δx and δu follows similar lines as the previous section. The counterpart to (3.3.3.5) for the additional control variable $v(\tau)$ becomes :

$$(\hat{\rho} f_0(\hat{x}(\tau),\hat{u}(\tau),\hat{y}(\tau)) + \hat{\lambda}(\tau)^T f(\hat{x}(\tau),\hat{u}(\tau),\hat{y}(\tau)) + \hat{\lambda}_y(\tau) +$$

$$\hat{\eta}_1(\tau)^T S_1(\hat{x}(\tau),\hat{u}(\tau),\hat{y}(\tau)))(v - \hat{v}(\tau)) \geqslant 0 \quad \textit{for all } v \geqslant 0 \textit{ a.e. } 0 \leqslant \tau \leqslant 1. \quad \textit{(3.3.4.17)}$$

($\hat{\lambda}_y$ is the adjoint variable associated with (3.3.4.9).)

Because every $\hat{v}(\tau)$ which satisfies (3.3.4.3) - (3.3.4.5) is a solution to the transformed problem, we may consider (3.3.4.17) with $\hat{v}(\tau)$ strictly positive on [0,1]. This implies

$$\hat{\rho} f_0(\hat{x}(\tau),\hat{u}(\tau),\hat{y}(\tau)) + \hat{\lambda}(\tau)^T f(\hat{x}(\tau),\hat{u}(\tau),\hat{y}(\tau)) + \hat{\lambda}_y(\tau) +$$

$$\hat{\eta}_1(\tau)^T S_1(\hat{x}(\tau),\hat{u}(\tau),\hat{y}(\tau)) = 0 \qquad \textit{a.e. } 0 \leqslant \tau \leqslant 1. \qquad \textit{(3.3.4.18)}$$

Alternatively, we may consider functions $\hat{v}(\tau)$ which are zero on intervals. In these cases (3.3.4.17) implies

$$\hat{\rho} f_0(\hat{x}(\tau),\hat{u}(\tau),\hat{y}(\tau)) + \hat{\lambda}(\tau)^T f(\hat{x}(\tau),\hat{u}(\tau),\hat{y}(\tau)) + \hat{\lambda}_y(\tau) +$$

$$\hat{\eta}_1(\tau)^T S_1(\hat{x}(\tau),\hat{u}(\tau),\hat{y}(\tau)) \geqslant 0 \qquad \textit{a.e. on } R_0, \qquad \textit{(3.3.4.19)}$$

where R_0 denotes the set of time points for which $\hat{v}(\tau) = 0$.

The essence of the approach is now that on the set R_0, the values of $\hat{u}(\tau)$, which are restricted to the set U, may still be chosen (they do not affect the value of the object function, nor any of the other constraints). On the set R_0 all other quantities are constant and hence the choice

$$\hat{u}(\tau) := \hat{u}(t(\tau)),$$

yields the equality implied by (3.3.4.18). Therefore $\hat{u}(t(\tau))$ must be a global minimum of

$$\hat{\rho} f_0(\hat{x}(\tau),u,\hat{y}(\tau)) + \hat{\lambda}(\tau)^T f(\hat{x}(\tau),u,\hat{y}(\tau)) + \hat{\eta}_1(\tau)^T S_1(\hat{x}(\tau),u,\hat{y}(\tau)),$$

over the set U.

Of course this reasoning is not a rigorous proof for (3.3.4.2), which should involve a proper choice of the function $\hat{v}(\tau)$ and $\hat{u}(\tau)$ on R_0, that shows that (3.3.4.2) must hold almost everywhere on $[0,\hat{T}]$ and at the same time be a pointwise global minimization (cf. Girsanov (1972) for further details).

Equation (3.3.4.1) is obtained from (3.3.4.18) following the derivation below. Here the use of the variable time transformation (3.3.4.3) - (3.3.4.5) is further superfluous. Therefore we set $\hat{v}(\tau)$ constant on [0,1].

$$\hat{\lambda}_y(t) = -H[t] - \hat{\eta}_1(t)^T S_1[t] = -H[t] \qquad a.e.\ \ 0 \leqslant t \leqslant \hat{T}. \tag{3.3.4.20}$$

Because $\hat{\lambda}_y$ is the adjoint variable corresponding to (3.3.4.9), it satisfies relations similar to (3.3.3.2) - (3.3.3.4):

$$\hat{\lambda}_y(t_1) - \hat{\lambda}_y(t) = -\int_t^{t_1} (H_t[t] + \hat{\eta}_1(t)^T S_{1t}[t])\, dt - \int_t^{t_1} d\hat{\xi}(t)^T S_{2t}[t]$$

$$\text{for all } 0 \leqslant t \leqslant t_1 \leqslant \hat{T}. \tag{3.3.4.21}$$

and

$$\hat{\lambda}_y(\hat{T}) = \hat{\rho} g_{0T}[\hat{T}] + \hat{\mu}^T E_T[\hat{T}]. \tag{3.3.4.22}$$

Taking $t_1 = \hat{T}$ and combination of (3.3.4.21) - (3.3.4.22) with (3.3.4.20) yields (3.3.4.1). □

3.3.5. Smoothness of the multiplier $\hat{\xi}$.

In this section the smoothness of the multiplier $\hat{\xi}$ is considered, which is essential for the practical application of the optimality conditions stated in the previous sections.

Because $\hat{\xi}$ is a function of bounded variation on $[0,\hat{T}]$, it has at most a countable number of discontinuities and its derivative exists almost everywhere on $[0,\hat{T}]$ (cf. Royden (1963), p.86). Hence equation (3.3.3.2) is equivalent to:

$$\dot{\hat{\lambda}}(t)^T = -H_x[t] - \hat{\eta}_1(t)^T S_{1x}[t] - \hat{\eta}_2(t)^T S_{2x}[t] \qquad a.e.\ \ 0 \leqslant t \leqslant \hat{T}. \tag{3.3.5.1}$$

$$\hat{\lambda}(t_j+)^T = \hat{\lambda}(t_j-)^T - \hat{\nu}_j^T S_{2x}[t_j] \quad \textit{at points of discontinuity of } \hat{\xi}. \tag{3.3.5.2}$$

where: $\hat{\eta}_2(t) := \dot{\hat{\xi}}(t)$,
$\hat{\nu}_j := \hat{\xi}(t_j+) - \hat{\xi}(t_j-)$.

The conditions (3.3.3.8) and (3.3.3.9) of Theorem 3.11, i.e. $\hat{\xi}_i$ = *constant* if $S_{2i}[t] < 0$ and $\hat{\xi}_i$ = *nondecreasing on* $[0,\hat{T}]$ are equivalent to the conditions:

$$\hat{\eta}_{2i}(t) = 0 \quad \text{if} \quad S_{2i}[t] < 0, \tag{3.3.5.3}$$

$$\hat{\eta}_{2i}(t) \geqslant 0 \quad \text{if} \quad S_{2i}[t] = 0, \tag{3.3.5.4}$$

and

$$\hat{\nu}_j = 0 \qquad \text{if} \quad S_{2i}[t_j] < 0, \tag{3.3.5.5}$$

$$\hat{\nu}_j \geqslant 0 \qquad \text{if} \quad S_{2i}[t_j] = 0. \tag{3.3.5.6}$$

The application of these optimality conditions is complicated by the fact that we have no information, about the time points t_j at which ξ is possibly discontinuous, on intervals where one or more of the components of S_2 are zero.

Before the main result of this section is stated, some terminology and some definitions are introduced.

Let p_i and l be integers with $1 \leqslant p_i \leqslant l$. Assume that the functions $f(x,u,t)$ and $S_{2i}(x,t)$ are respectively C^l- and C^{p_i}-functions with respect to all arguments. Define the functions (cf. Hamilton (1972)) :

$$F_{2i}^0(x,u,t) := S_{2i}(x,t), \tag{3.3.5.7}$$

$$F_{2i}^j(x,u,t) := \frac{\partial F_{2i}^{j-1}(x,u,t)}{\partial x} f(x,u,t) + \frac{\partial F_{2i}^{j-1}(x,u,t)}{\partial t} \quad j = 1,2,\ldots,p_i. \tag{3.3.5.8}$$

The *order of the state constraint* S_{2i} is p_i, if

$$p_i = \min\{ q \in \mathbb{N} : \exists x_0 \in \mathbb{R}^n \wedge \exists u_0 \in \mathbb{R}^m \wedge \exists t_0 \in \mathbb{R} \left| \frac{\partial F_{2i}^q(x_0,u_0,t_0)}{\partial u} \neq 0 \right| \}.$$

Based on this definition the functions $S_{2i}^j : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}$ for $j = 0,1,\ldots p_i - 1$ and $S_{2i}^{p_i} : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R} \to \mathbb{R}$ are defined as $S_{2i}^j := F_{2i}^j$, for $j = 0,1,\ldots p_i$.

Along a trajectory (x,u) that satisfies the differential system (3.1.2) we have

$$\frac{d^j S_{2i}(x(t),t)}{dt^j} = \begin{cases} S_{2i}^j(x(t),t) & j = 0,1,\ldots,p_i - 1 \\ S_{2i}^{p_i}(x(t),u(t),t) & j = p_i \end{cases} \tag{3.3.5.9}$$

By definition the functions $S_{2i}(x,t)$ do not depend on u explicitly and hence we have $p_i \geqslant 1$ for all $i = 1,\ldots k_2$. A logical extension to the definition of order of a state constraint is, to define *mixed state control constraints* as *state constraints of order zero*.

We now introduce :

$$\tilde{S} : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R} \to \mathbb{R}^{k_1+k_2},$$

$$\tilde{S}(x,u,t) := \begin{bmatrix} S_1(x,u,t) \\ S_2(x,t) \end{bmatrix}, \tag{3.3.5.10}$$

and

$$\tilde{S}^p : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R} \to \mathbb{R}^{k_1+k_2},$$

$$\tilde{S}^{p}(x,u,t) := \begin{vmatrix} S_1(x,u,t) \\ S_{21}^{p_1}(x,u,t) \\ \vdots \\ S_{2k_2}^{p_{k_2}}(x,u,t) \end{vmatrix} \qquad (3.3.5.11)$$

Definition 3.13 : *Let $(\hat{x},\hat{u},\hat{T})$ be a solution to problem (SCOCP) and let*

$$I_i := \{ t \in [0,\hat{T}] : \tilde{S}_i(\hat{x}(t),\hat{u}(t),t) = 0 \} \qquad i = 1,2,\ldots k_1+k_2, \qquad (3.3.5.12)$$

be the set of active points of the state constraint $\tilde{S}_i(x,u,t) \leqslant 0$. With respect to $\tilde{S}_i$, a subinterval $[t_1,t_2] \subset [0,\hat{T}]$, $t_1 < t_2$, is called a boundary interval if $[t_1,t_2] \subset I_i$ and an interior interval if $[t_1,t_2] \cap I_i = \emptyset$. Entry-points respectively exit-points, also called junction points, and contact points, are defined in an obvious way.

The possibilities that $t = 0$ is an entry- or contact point or $t = \hat{T}$ is an exit- or contact point are included. $[t_1,t_2]$ is a boundary interval for $\tilde{S}$ if $[t_1,t_2]$ is a boundary interval for every component $\tilde{S}_i$, $i = 1,\ldots,k_1+k_2$.

For simplicity we shall assume two cases in the sequel, either $[t_1,t_2]$ is an interior interval or $[t_1,t_2]$ is a boundary interval for $\tilde{S}$. Cases where some but not all state constraints are active on an interval $[t_1,t_2]$ are similar to the case that $[t_1,t_2]$ is a boundary interval for $\tilde{S}$. In these cases all assumptions and results correspond to the case that all inactive components of $\tilde{S}$ are omitted completely.

The following regularity condition is of importance :

Assumption 3.14 : *Let the function $\tilde{S}^p : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R} \to \mathbb{R}^{k_1+k_2}$ be defined by (3.3.5.11) and let $(\hat{x},\hat{u},\hat{T})$ be a solution to problem (SCOCP), then*

$$\text{rank } \tilde{S}_u^p(\hat{x}(t),\hat{u}(t),t) = k_1 + k_2 \quad \text{a.e. on } I_1 \cup I_2 \cup \cdots \cup I_{k_1+k_2}. \qquad (3.3.5.13)$$

The following theorem establishes the smoothness of $\hat{\xi}$ on boundary intervals :

Theorem 3.15 : *Let $(\hat{x},\hat{u},\hat{T})$ be a solution to problem (SCOCP) for which Assumptions 3.7, 3.8 and 3.14 hold, and let f_0, f and $\tilde{S}$ be $C^{\bar{p}+l}$-functions ($\bar{p} := \max p_i$) with respect to all arguments and $l \geqslant 0$. Let $[t_1,t_2]$ be a boundary interval. Assume in addition that $\hat{u}(t)$ is a $C^{\bar{p}+l}$-function on $[t_1,t_2]$ with*

$$\hat{u}(t) \in \text{int } U \quad \text{for all } t \in (t_1,t_2). \qquad (3.3.5.14)$$

Then the functions $\hat{\lambda}$ and $\hat{\xi}$ in the adjoint equation (3.3.3.2) are C^{l+1}-functions on (t_1,t_2). In particular the adjoint equation

$$\dot{\hat{\lambda}}(t)^T = -H_x[t] - \hat{\eta}(t)^T \tilde{S}_x[t] \qquad t_1 < t < t_2, \qquad (3.3.5.15)$$

holds, where $\hat{\eta}^T := (\hat{\eta}_1^T,\hat{\xi}^T)$ is a C^l-function.

The proof of this theorem can be found in Maurer (1976,1979), who put the heuristic proof of Jacobson et al. (1971) on a solid base.

The proof is done in two steps. The first step deals with the case of one state constraint and one control. Because of (3.3.5.14) condition (3.3.3.5) becomes ($k_1 = 0$) :

$$H_u[t] = 0 \quad \textit{for all} \quad t_1 < t < t_2.$$

Consideration of the $(\bar{p}-1)$-th time derivative of $H_u[t]$ on (t_1,t_2) yields the result. We note that this approach is essentially based on the smoothness assumption made on the control $\hat{u}(t)$.

The second step deals with the general case of multiple state constraints and multiple controls. The regularity Assumption 3.14 is used to apply the same techniques used in the first step via an elimination process.

Under the hypothesis of Theorem 3.15 we may thus be sure that points of discontinuity of the function $\hat{\xi}$ cannot be interior points of boundary intervals. From (3.3.5.5) we know that these points are also not points of interior intervals. Hence points of discontinuity of $\hat{\xi}$ can only be junction or contact points. At these points equation (3.3.5.2), which is called the 'jump'-condition, must hold.

3.3.6. Alternative formulations of the first order optimality conditions.

This section deals with some alternative formulations of the first order optimality conditions. To simplify things we consider the problem (SCOCP) for the case that there are no mixed control state constraints ($k_1=0$), one state constraint ($k_2=1$) and one control ($m=1$). We note however, that the results of this section can be extended to more general cases in a straightforward manner. Because the manipulations on the state constraints are done for each boundary interval separately, we assume without loss of generality that the set of active points of the state constraint S_2 consists of only one boundary interval $[t_1,t_2]$, with $0<t_1<t_2<\hat{T}$. The order of the state constraint S_2 is denoted by p.

For all $i=0,1,2,\ldots p$ the *augmented Hamiltonian* is defined as :

$$\overline{H}^i(x,u,\hat{\rho},\hat{\lambda}^i,\hat{\eta}^i,t) := \hat{\rho} f_0(x,u,t) + \hat{\lambda}^{iT} f(x,u,t) + \hat{\eta}^i S_2^i(x,u,t), \qquad (3.3.6.1)$$

where the functions S_2^i are defined by (3.3.5.7) - (3.3.5.8).

Setting $\hat{\lambda}^0=\hat{\lambda}$ and $\hat{\eta}^0=\hat{\eta}=\dot{\hat{\xi}}$, Theorems 3.11, 3.12 and 3.15 involve the augmented Hamiltonian for the case $i=0$.

The main result of this section will be a similar statement for all $i=1,\ldots p$. Its statement is simplified by means of the following definitions :

$$\hat{\eta}^1(t) := \begin{cases} \hat{\nu}_2 + \int_t^{t_2} \hat{\eta}^0(\tau)\,d\tau = \hat{\xi}(t_2+) - \hat{\xi}(t) & t_1+ \leqslant t \leqslant t_2- \\ 0 & \textit{elsewhere} \end{cases} \qquad (3.3.6.2)$$

$$\hat{\beta}^1 := \hat{\eta}^1(t_1+) + \hat{\nu}_1 = \hat{\xi}(t_2+) - \hat{\xi}(t_1-), \qquad (3.3.6.3)$$

$$\hat{\eta}^i(t) := \begin{cases} \int_t^{t_2} \hat{\eta}^{i-1}(\tau)\,d\tau & t_1+ \leqslant t \leqslant t_2- \quad i=2,\ldots p \geqslant 2 \\ 0 & \textit{elsewhere} \end{cases} \qquad (3.3.6.4)$$

$$\hat{\beta}^i := \hat{\eta}^i(t_1+) \qquad i=2,...p \geqslant 2. \tag{3.3.6.5}$$

and

$$\hat{\lambda}^i(t)^T := \hat{\lambda}^0(t)^T - \sum_{j=1}^{i} \hat{\eta}^j(t) \frac{\partial S_2^{j-1}[t]}{\partial x} \qquad 0 \leqslant t \leqslant \hat{T} \quad i=1,...,p. \tag{3.3.6.6}$$

With these definitions the following minimum principle holds :

Theorem 3.16 : *Let $(\hat{x},\hat{u},\hat{T})$ be a solution to problem (SCOCP) with $k_1=0$, $k_2=1$ and $m=1$. Suppose that f_0, f and S are C^p-functions and that Assumption 3.14 holds. Assume in addition that the set of active points consists of one boundary interval $[t_1,t_2]$, with $0<t_1<t_2<\hat{T}$ and that $\hat{u}$ is a C^p-function on (t_1,t_2) with*

$$\hat{u}(t) \in \text{int } U \text{ for all } t \in (t_1,t_2). \tag{3.3.6.7}$$

Let $\hat{\rho},\hat{\sigma},\hat{\mu},\hat{\lambda}$ and $\hat{\xi}$ satisfy the conditions of Theorems 3.11, 3.12 and 3.15 and let $\hat{\lambda}^i$ and $\hat{\eta}^i$ be defined by (3.3.6.2) - (3.3.6.6) for all $i=1,...,p$.

Then, for all $i=1,...,p$, the following relations hold :

$$\dot{\hat{\lambda}}^i(t)^T = -\overline{H}_x^i[t] \qquad \text{a.e. } 0 \leqslant t \leqslant \hat{T}, \tag{3.3.6.8}$$

$$\hat{\lambda}^i(0)^T = -\hat{\rho}h_{0x}[0] - \hat{\sigma}^T D_x[0], \tag{3.3.6.9}$$

$$\hat{\lambda}^i(\hat{T})^T = \hat{\rho}g_{0x}[\hat{T}] + \hat{\mu}^T E_x[\hat{T}], \tag{3.3.6.10}$$

$$\hat{\lambda}^i(t_1+)^T = \hat{\lambda}^i(t_1-)^T - \sum_{j=1}^{i} \hat{\beta}^j \frac{\partial S_2^{j-1}[t_1]}{\partial x}, \tag{3.3.6.11}$$

$$\hat{\beta}^j \geqslant 0 \qquad j=1,2,...i, \tag{3.3.6.12}$$

$$\hat{\eta}^j(t) \geqslant 0 \qquad j=1,2,...i \quad t_1<t<t_2, \tag{3.3.6.13}$$

$$\overline{H}^i[t] = \max_{u \in U} \overline{H}^i(\hat{x}(t),u,\hat{\rho},\hat{\lambda}^i(t),\hat{\eta}^i(t),t) \qquad \text{a.e. } 0 \leqslant t \leqslant \hat{T}, \tag{3.3.6.14}$$

$$\overline{H}^i[\hat{T}] = -\hat{\rho}g_{0T}[\hat{T}] - \hat{\mu}^T E_T[\hat{T}], \tag{3.3.6.15}$$

$$\frac{d\overline{H}^i[t]}{dt} = \overline{H}_t^i[t] \qquad \text{a.e. } 0 \leqslant t \leqslant \hat{T}, \tag{3.3.6.16}$$

$$\overline{H}^i[t_1+] = \overline{H}^i[t_1-] + \sum_{j=1}^{i} \frac{\partial S_2^{j-1}[t]}{\partial t}. \tag{3.3.6.17}$$

Proof : The theorem is quite similar to Theorem 5.1 of Maurer (1979), who considered the autonomous case with fixed final time.

The hypotheses are such that the conditions implied by Theorems 3.11, 3.12 and 3.15 hold.

Condition (3.3.6.9) and (3.3.6.10) follow directly from (3.3.3.3) and (3.3.3.4). Taking the time derivative of (3.3.6.6) results in :

$$\dot{\hat{\lambda}}^i = \dot{\hat{\lambda}}^0 - \frac{d}{dt}\left[\sum_{j=1}^{i} \hat{\eta}^j \frac{\partial S_2^{j-1}}{\partial x}\right], \tag{3.3.6.18}$$

and definitions (3.3.6.2) and (3.3.6.4) yield :

$$\dot{\hat{\eta}}^j = -\hat{\eta}^{j-1} \qquad j = 1,2,..p. \tag{3.3.6.19}$$

$$S_2^j = \frac{\partial S_2^{j-1}}{\partial t} + \frac{\partial S_2^{j-1}}{\partial x} f \tag{3.3.6.20}$$

$$\frac{\partial S_2^j}{\partial x} = \frac{\partial^2 S_2^{j-1}}{\partial x \partial t} + \frac{\partial^2 S_2^{j-1}}{\partial x^2} f + \frac{\partial S_2^{j-1}}{\partial x} \frac{\partial f}{\partial x}, \tag{3.3.6.21}$$

$$\frac{d}{dt}\left[\frac{\partial S_2^{j-1}}{\partial x}\right] = \frac{\partial^2 S_2^{j-1}}{\partial t \partial x} + \frac{\partial^2 S_2^{j-1}}{\partial x^2} f = \frac{\partial S_2^j}{\partial x} - \frac{\partial S_2^{j-1}}{\partial x} \frac{\partial f}{\partial x}. \tag{3.3.6.22}$$

Combination of (3.3.6.18) with (3.3.6.19) and (3.3.6.22) gives :

$$\dot{\hat{\lambda}}^i = \dot{\hat{\lambda}}^0 - \sum_{j=1}^{i} \left[-\hat{\eta}^{j-1} \frac{\partial S_2^{j-1}}{\partial x} + \hat{\eta}^j \frac{\partial S_{2_j}}{\partial x} - \hat{\eta}^j \frac{\partial S_2^{j-1}}{\partial x} \frac{\partial f}{\partial x} \right]. \tag{3.3.6.23}$$

Using (3.3.5.1) for $\dot{\hat{\lambda}}^0$ yields (3.3.6.8).

The entry point condition (3.3.6.11) follows from the 'jump'-condition (3.3.5.2) for $t = t_1$, which becomes

$$\hat{\lambda}^0(t_1+) = \hat{\lambda}^i(t_1-) - \hat{\nu}_1 \frac{\partial S_2[t_1]}{\partial x}. \tag{3.3.6.24}$$

Definitions (3.3.6.3), (3.3.6.5) and (3.3.6.6) give :

$$\hat{\lambda}^i(t_1+) = \hat{\lambda}^0(t_1+) - (\hat{\beta}^1 - \hat{\nu}_1) \frac{\partial S_2^{i-1}[t_1]}{\partial x} - \sum_{j=2}^{i} \hat{\beta}^j \frac{\partial S_2^{j-1}[t_1]}{\partial x}. \tag{3.3.6.25}$$

Combination of (3.3.6.24) and (3.3.6.25) give (3.3.6.11).

A similar derivation at $t = t_2$ reveals that for all $i \geqslant 1$, the functions $\hat{\lambda}^i$ are continuous at this point.

Conditions (3.3.6.12) and (3.3.6.13) follow directly from the properties of $\hat{\eta}^0$, $\hat{\nu}_1$ and $\hat{\nu}_2$ and the defining equations (3.3.6.2) - (3.3.6.5).

$$\overline{H}^i(\hat{x}(t),u,\hat{\rho},\hat{\lambda}^i(t),\hat{\eta}^i(t),t) = \hat{\rho} f_0(\hat{x}(t),u,t) + \hat{\lambda}^0(t) f(\hat{x}(t),u,t) -$$

$$\sum_{j=1}^{i} \hat{\eta}^j(t) \frac{\partial S_2^{j-1}[t]}{\partial x} f(\hat{x}(t),u,t) + \hat{\eta}^i(t) S_2^i(\hat{x}(t),u,t).$$

Because,

$$\frac{\partial S_2^{j-1}[t]}{\partial x} f(\hat{x}(t),u,t,) = \begin{cases} S_2^j[t] - \dfrac{\partial S_2^{j-1}[t]}{\partial t} & j = 1,...p-1 \\ S_2^p(\hat{x}(t),u,t) - \dfrac{\partial S_2^{p-1}[t]}{\partial t} & j = p \end{cases} \tag{3.3.6.26}$$

we obtain

$$\overline{H}^i(\hat{x}(t),u,\hat{\rho},\hat{\lambda}^i(t),\hat{\eta}^i(t),t) = H^0(\hat{x}(t),u,\hat{\rho},\hat{\lambda}(t),\hat{\eta}(t),t) +$$

$$\sum_{j=1}^{i} \hat{\eta}^j(t) \frac{\partial S_2^{j-1}[t]}{\partial t} \quad \textit{for all} \quad u \in U. \tag{3.3.6.27}$$

Because the second term does not depend on u, (3.3.6.14) follows directly from (3.3.4.2).

(3.3.6.15) follows from (3.3.4.1) for $t = \hat{T}$ because $\hat{\eta}^j(\hat{T}) = 0$ for all j.

(3.3.6.16) and (3.3.6.17) follow from (3.3.4.1) via a derivation similar to the derivation of (3.3.6.8) and (3.3.6.11).

□

With regard to Definition (3.3.6.4) we note that it implies :

$$\hat{\eta}^i(t_2-) = 0 \quad i = 2,...p. \tag{3.3.6.28}$$

In essence Theorem 3.16 states a minimum principle for each fixed $i \in \{1,...p\}$. From the Definitions (3.3.6.2) - (3.3.6.6) it is clear that the multipliers associated with the various minimum principles for $i = 0,1,...p$ are related. Given a set of multipliers associated with a principle for one specific i, it is possible to obtain the multipliers associated with other minimum principles via either integration or differentiation.

Before this section is finished, we shall make some notes on related results in literature.

For $i = p$ the minimum principle is similar to the conditions given by Bryson et al. (1963). These conditions were derived following an indirect approach. Instead of treating the state constraint direct, the constraint was replaced by :

$$S_2^j(x(t_1),t_1) = 0 \qquad j = 0,1,...p-1, \tag{3.3.6.29}$$

and

$$S_2^0(x(t),u(t),t) = 0 \quad t_1 \leqslant t \leqslant t_2. \tag{3.3.6.30}$$

The conditions given by Bryson et al. however, are somewhat weaker, e.g. they involve (3.3.6.13) only with $j = p$.

This fact was recognized by Jacobson et al. (1971), who were the first to derive the minimum principle for $i = 0$. Later Norris (1973) put the proof of Jacobson et al. on a solid base, except for the results on the smoothness of the multiplier $\hat{\xi}$. These results are due to Maurer (1976,1979). Kreindler (1982) showed that the conditions given by Bryson et al. can be made as strong as the minimum principle for $i = 0$ by augmenting the set of conditions with a number of additional conditions on the multipliers and their derivatives. In fact this yields the minimum principle of Theorem 3.16 for the case $i = p$.

3.4. Solution of some example problems.

In this section we shall give some examples that will be solved using the optimality conditions of the previous sections.

3.4.1. Example 1.

$$\underset{x,u}{\text{Minimize}} \quad \tfrac{1}{2}\int_0^1 u^2(t)\,dt, \tag{3.4.1.1}$$

subject to :

$$\dot{x}_1 = x_2 \qquad 0 \leqslant t \leqslant 1, \tag{3.4.1.2}$$

$$\dot{x}_2 = u \qquad 0 \leqslant t \leqslant 1, \tag{3.4.1.3}$$

$$x_1(0) = 0, \tag{3.4.1.4}$$

$$x_2(0) = 0, \tag{3.4.1.5}$$

$$x_1(1) = 1, \tag{3.4.1.6}$$

$$x_2(1) = 0, \tag{3.4.1.7}$$

$$u(t) - u_{max} \leqslant 0 \qquad 0 \leqslant t \leqslant 1. \tag{3.4.1.8}$$

The problem specified by (3.4.1.1) - (3.4.1.8) is a problem with fixed final time, and fixed initial and terminal state. The constraint (3.4.1.8) is treated as a mixed control state constraint. The control constraint can, in the formulation of problem (SCOCP), be handled in two ways, i.e. by means of the set U or by the constraint function S_1. We shall follow the latter road by setting $S_1 = u - u_{max}$. Because the problem specified by (3.4.1.1) - (3.4.1.8) is a special case of problem (SCOCP), the optimality conditions of Section 3.3.3 can be applied straightforward. Because we have fixed initial and terminal states, the boundary conditions (3.3.3.3) and (3.3.3.4) can be discarded as they only introduce additional multipliers, whose values follow directly from the values of $\lambda(0)$ and $\lambda(T)$.

The Hamiltonian (3.3.3.1) becomes :

$$H(x, u, \rho, \lambda) = \tfrac{1}{2}\rho u^2 + \lambda_1 x_2 + \lambda_2 u. \tag{3.4.1.9}$$

The optimality conditions of Theorem 3.11 take the following form :

$$\dot{\hat{\lambda}}_1 = 0 \qquad a.e.\ 0 \leqslant t \leqslant 1, \tag{3.4.1.10}$$

$$\dot{\hat{\lambda}}_2 = -\hat{\lambda}_1 \qquad a.e.\ 0 \leqslant t \leqslant 1, \tag{3.4.1.11}$$

$$\hat{\rho}\hat{u} + \hat{\lambda}_2 + \hat{\eta}_1 = 0 \qquad a.e.\ 0 \leqslant t \leqslant 1, \tag{3.4.1.12}$$

$$\hat{\eta}_1 \geqslant 0 \qquad a.e.\ 0 \leqslant t \leqslant 1, \tag{3.4.1.13}$$

$$\hat{\eta}_1(\hat{u} - u_{max}) = 0 \qquad a.e.\ 0 \leqslant t \leqslant 1, \tag{3.4.1.14}$$

We shall first consider the regularity of the problem. If there is an interval of nonzero length with $\hat{u}(t) < u_{max}$ then $\hat{\rho} = 1$, because $\hat{\rho} = 0$ would according to (3.4.1.12) imply

$$\hat{\lambda}_2(t) = -\hat{\eta}_1(t).$$

Because, on an interval where $\hat{u}(t) < u_{max}$ we have

$$\hat{\eta}_1(t) = 0,$$

the zero solution would follow for $\lambda_2(t)$ and $\lambda_1(t)$, and that would contradict the main

statement of the theorem.

The situations $\hat{u}(t) < u_{max}$ and $\hat{u}(t) \leqslant u_{max}$ (i.e. equality holds on a nonzero interval), are considered separately.

In the case that

$$\hat{u}(t) < u_{max} \qquad 0 \leqslant t \leqslant 1. \qquad (3.4.1.15)$$

condition (3.4.1.14) implies

$$\hat{\eta}_1(t) = 0 \qquad 0 \leqslant t \leqslant 1. \qquad (3.4.1.16)$$

substitution into (3.4.1.12) yields :

$$\hat{u}(t) = -\hat{\lambda}_2(t) \qquad 0 \leqslant t \leqslant 1. \qquad (3.4.1.17)$$

$\hat{\lambda}_2(t)$ follows from (3.4.1.10) and (3.4.1.11) as

$$\hat{\lambda}_1(t) = \hat{\lambda}_1 = constant \qquad 0 \leqslant t \leqslant 1, \qquad (3.4.1.18)$$

$$\hat{\lambda}_2(t) = \hat{\lambda}_2(0) - \hat{\lambda}_1 t \qquad 0 \leqslant t \leqslant 1. \qquad (3.4.1.19)$$

Substitution of the control (3.4.1.17) in (3.4.1.2) and (3.4.1.3) and integration using the boundary conditions (3.4.1.4) and (3.4.1.5) yields :

$$\hat{x}_2(t) = -\hat{\lambda}_2(0)t + \tfrac{1}{2}\hat{\lambda}_1 t^2 \qquad 0 \leqslant t \leqslant 1, \qquad (3.4.1.20)$$

$$\hat{x}_1(t) = -\tfrac{1}{2}\hat{\lambda}_2(0)t^2 + \tfrac{1}{6}\hat{\lambda}_1 t^3. \qquad 0 \leqslant t \leqslant 1. \qquad (3.4.1.21)$$

The numerical values of $\hat{\lambda}_2(0)$ and $\hat{\lambda}_1$ are determined from the boundary conditions (3.4.1.6) and (3.4.1.7) :

$$\hat{\lambda}_2(0) = -6, \qquad (3.4.1.22)$$

$$\hat{\lambda}_1 = -12. \qquad (3.4.1.23)$$

This solution is only a candidate for the solution if $u_{max} > 6$, i.e. in the situation that the control constraint is not active at any time point (cf. Figure 3.1).

In the case that $u_{max} < 6$, the situation is a little more complicated. Based on the unconstrained solution we may guess that the constraint is active over an interval $[0, t_1]$ and inactive over the interval $(t_1, 1]$.

Conditions (3.4.1.10) - (3.4.1.14) imply in this case :

$$\hat{u}(t) = \begin{cases} u_{max} & 0 \leqslant t \leqslant t_1 \\ -\hat{\lambda}_2(0) + \hat{\lambda}_1 t & t_1 < t \leqslant 1 \end{cases} \qquad (3.4.1.24)$$

$$\hat{\eta}_1(t) = \begin{cases} -\hat{\lambda}_2(t) - u_{max} & 0 \leqslant t \leqslant t_1 \\ 0 & t_1 < t \leqslant 1 \end{cases} \qquad (3.4.1.25)$$

Substitution of the control (3.4.1.24) in (3.4.1.2) and (3.4.1.3) and integration using the boundary conditions (3.4.1.4) and (3.4.1.5) yields :

$$\hat{x}_2(t) = \begin{cases} u_{max} t & 0 \leqslant t \leqslant t_1 \\ u_{max} t_1 - \hat{\lambda}_2(0)(t - t_1) + \tfrac{1}{2}\lambda_1(t^2 - t_1^2) & t_1 < t \leqslant 1 \end{cases} \qquad (3.4.1.26)$$

$$\hat{x}_1(t) = \begin{cases} \frac{1}{2} u_{max} t^2 & 0 \leqslant t \leqslant t_1 \\ \frac{1}{2} u_{max} t_1^2 + (u_{max} t_1 + \hat{\lambda}_2(0) t_1 - \frac{1}{2} \hat{\lambda}_1 t_1^2)(t - t_1) - & \\ \frac{1}{2} \hat{\lambda}_2(0)(t^2 - t_1^2) + \frac{1}{6} \hat{\lambda}_1 (t^3 - t_1^3) & t_1 < t \leqslant 1 \end{cases} \qquad (3.4.1.27)$$

The boundary conditions (3.4.1.6) and (3.4.1.7) are satisfied when $\hat{\lambda}_2(0)$ and $\hat{\lambda}_1$ are :

$$\hat{\lambda}_1 = \frac{-12 + 6 u_{max} t_1}{(1 - t_1)^3}, \qquad (3.4.1.28)$$

$$\hat{\lambda}_2(0) = \frac{-6(1 + t_1) + u_{max} t_1 (t_1^2 + t_1 + 4)}{(1 - t_1)^3}. \qquad (3.4.1.29)$$

Combination of (3.4.1.25) and (3.4.1.24) with (3.4.1.13) yields

$$\hat{\eta}_1 \geqslant 0 \qquad 0 \leqslant t \leqslant t_1, \qquad (3.4.1.30)$$

and (3.4.1.14)

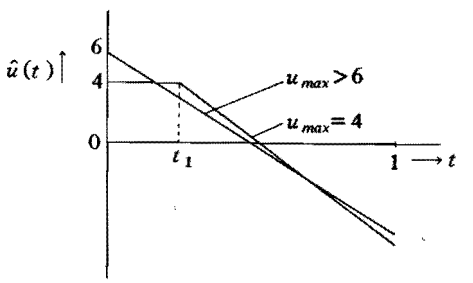
$$\hat{u}(t) \leqslant u_{max} \qquad t_1 < t \leqslant 1, \qquad (3.4.1.31)$$

results in the condition

$$\hat{\lambda}_2(t) \leqslant - u_{max} \qquad 0 \leqslant t \leqslant t_1, \qquad (3.4.1.32)$$

and

$$\hat{\lambda}_2(t) \geqslant - u_{max} \qquad t_1 < t \leqslant 1. \qquad (3.4.1.33)$$



Solution of Example 1 for $u_{max} > 6$ and $u_{max} = 4$.
Figure 3.1

Because $\hat{\lambda}_2$ must be continuous on [0,1] as a result of the fact that there are no state constraints of order higher than zero, we must have

$$\hat{\lambda}_2(t_1) = - u_{max}, \qquad (3.4.1.34)$$

and hence

$$\hat{u}(t_1 +) = u_{max}, \qquad (3.4.1.35)$$

i.e. the control must also be continuous at $t = t_1$.

With (3.4.1.18), (3.4.1.28) and (3.4.1.29), equation (3.4.1.35) may be solved for t_1 :

$$t_1 = \frac{1}{2}\left|\frac{6}{u_{max}} - 1\right|. \qquad (3.4.1.36)$$

For $2 < u_{max} < 6$ we have $0 < t_1 < 1$. For $u_{max} \leqslant 2$ the problem has no solution because there is no feasible control for which the boundary conditions (3.4.1.6) and (3.4.1.7) can be satisfied. In Figure 3.1 the optimal control $\hat{u}(t)$ is presented for two values of u_{max}.

An alternative method for the determination of the time point t_1 is to use condition (3.3.4.1), which states for this autonomous problem that the Hamiltonian must be constant on [0,1] and hence

$$H[t_1+] = H[t_1-]. \qquad (3.4.1.37)$$

A simple derivation shows that this conditions implies that the control must be continuous at $t = t_1$ and hence the same result follows.

3.4.2. Example 2

$$\underset{x,u}{\text{Minimize}} \quad \frac{1}{2}\int_0^1 u^2(t)\,dt\,, \qquad (3.4.2.1)$$

subject to :

$$\dot{x}_1 = x_2 \qquad 0 \leqslant t \leqslant 1, \qquad (3.4.1.2)$$

$$\dot{x}_2 = u \qquad 0 \leqslant t \leqslant 1, \qquad (3.4.2.3)$$

$$x_1(0) = 0, \qquad (3.4.2.4)$$

$$x_2(0) = 0, \qquad (3.4.2.5)$$

$$x_1(1) = 1, \qquad (3.4.2.6)$$

$$x_2(1) = 0, \qquad (3.4.2.7)$$

$$x_2(t) - x_{2,max} \leqslant 0 \qquad 0 \leqslant t \leqslant 1. \qquad (3.4.2.8)$$

This problem is similar to the problem of Example 1, except for the constraint (3.4.2.8), which is now a state constraint of first order.

The optimality conditions of Theorem 3.11 combined with the smoothness results of Section 3.3.5 take the following form :

$$\dot{\hat{\lambda}}_1 = 0 \qquad a.e.\ 0 \leqslant t \leqslant 1, \qquad (3.4.2.9)$$

$$\dot{\hat{\lambda}}_2 = -\hat{\lambda}_1 - \hat{\eta}_2 \qquad a.e.\ 0 \leqslant t \leqslant 1, \qquad (3.4.2.10)$$

$$\hat{\rho}u + \hat{\lambda}_2 = 0 \qquad a.e.\ 0 \leqslant t \leqslant 1, \qquad (3.4.2.11)$$

$$\hat{\eta}_2 \geqslant 0 \qquad a.e.\ 0 \leqslant t \leqslant 1, \qquad (3.4.2.12)$$

$$\hat{\eta}_2(\hat{x}_2 - x_{2,max}) = 0 \qquad a.e.\ 0 \leqslant t \leqslant 1, \qquad (3.4.2.13)$$

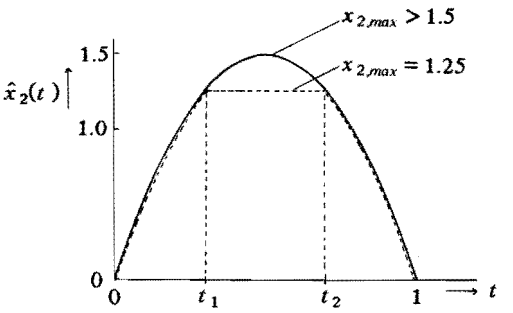
$$\hat{\lambda}_2(t_i+) = \hat{\lambda}(t_i-) - \nu_i \qquad \textit{at junction or contact points } t_i, \qquad (3.4.2.14)$$

$$\nu_i \geqslant 0 \qquad \textit{at junction or contact points } t_i. \qquad (3.4.2.15)$$

We note that the hypotheses of Theorem 3.16 are fullfilled because on boundary intervals the control $\hat{u}(t)$ is zero and hence at least once differentiable with respect to t.

As with Example 1, a simple derivation shows that if there is an interval of nonzero length on which $\hat{x}_2 < x_{2,max}$ then the regularity constant $\hat{\rho}$ must be nonzero.

The unconstrained solution of the problem, i.e. if $\hat{x}_2(t) < x_{2,max}$ is identical with the one derived in the previous section. The state variable $\hat{x}_2$ corresponding to this solution is given in Figure 3.2.



Solution of Example 2 for $x_{2,max} > 1.5$ and $x_{2,max} = 1.25$.
Figure 3.2

For $x_{2,max} < 1.5$ the solution, if it exists, will be constrained by the state constraint (3.4.2.8).

Considering this case we assume that the set of active points of the state constraint (3.4.2.8), consists of one interval $[t_1, t_2]$, with $0 < t_1 < t_2 < 1$.

The functions S_2^i defined by (3.3.5.7) and (3.3.5.8) are :

$$S_2^0 = x_2 - x_{2,max}, \tag{3.4.2.16}$$

$$S_2^1 = u, \tag{3.4.2.17}$$

and hence the constraint is of first order.

On the interval $[t_1, t_2]$ the control is determined by

$$S_2^p(x, u) = 0,$$

which yields in the present case

$$\hat{u}(t) = 0 \qquad t_1 \leqslant t \leqslant t_2, \tag{3.4.2.18}$$

and hence

$$\hat{\lambda}_2(t) = 0 \qquad t_1 \leqslant t \leqslant t_2. \tag{3.4.2.19}$$

combination with (3.4.2.10) yields :

$$\hat{\eta}_2(t) = -\hat{\lambda}_1 \qquad t_1 \leqslant t \leqslant t_2. \tag{3.4.2.20}$$

Using (3.4.2.10), (3.4.2.14) and (3.4.2.19) we obtain

$$\hat{\lambda}_2(t) = \begin{cases} \hat{\nu}_1 + \hat{\lambda}_1(t_1 - t) & 0 \leqslant t < t_1 \\ 0 & t_1 < t < t_2 \\ -\hat{\nu}_2 - \hat{\lambda}_1(t - t_2) & t_2 < t \leqslant 1 \end{cases} \tag{3.4.2.21}$$

With (3.4.2.11) the control becomes :

$$\hat{u}(t) = \begin{cases} -\hat{\nu}_1 - \hat{\lambda}_1(t_1 - t) & 0 \leqslant t < t_1 \\ 0 & t_1 < t < t_2 \\ \nu_2 + \lambda_1(t - t_2) & t_2 < t \leqslant 1 \end{cases} \qquad (3.4.2.22)$$

Using the boundary conditions (3.4.2.4) and (3.4.2.5) integration yields :

$$\hat{x}_2(t) = \begin{cases} -\hat{\nu}_1 t - \hat{\lambda}_1(-\frac{1}{2}t^2 + t_1 t) & 0 \leqslant t \leqslant t_1 \\ x_{2,max} & t_1 \leqslant t \leqslant t_2 \\ x_{2,max} + \hat{\nu}_2(t - t_2) + \frac{1}{2}\lambda_1(t - t_2)^2 & t_2 \leqslant t \leqslant 1 \end{cases} \qquad (3.4.2.23)$$

$$\hat{x}_1(t) = \begin{cases} -\frac{1}{2}\hat{\nu}_1 t^2 - \hat{\lambda}_1(-\frac{1}{6}t^3 + \frac{1}{2}t_1 t^2) & 0 \leqslant t \leqslant t_1 \\ -\frac{1}{2}\hat{\nu}_1 t_1^2 - \hat{\lambda}_1(-\frac{1}{6}t_1^3 + \frac{1}{2}t_1^2) + x_{2,max}(t - t_1) & t_1 \leqslant t \leqslant t_2 \\ -\frac{1}{2}\hat{\nu}_1 t_1^2 - \hat{\lambda}_1(-\frac{1}{6}t_1^3 + \frac{1}{2}t_1^2) & \\ + x_{2,max}(t - t_1) + \hat{\nu}_2\frac{1}{2}(t - t_2)^2 + \hat{\lambda}_1\frac{1}{6}(t - t_2)^3 & t_2 \leqslant t \leqslant 1 \end{cases} \qquad (3.4.2.24)$$

The multipliers $\hat{\nu}_1$, $\hat{\nu}_2$ and $\hat{\lambda}_1$ follow from the boundary conditions (3.4.2.6) (3.4.2.7) and the condition that the state variable x_2 is continuous at the point t_1.

$$\hat{\lambda}_1 = \frac{1 - \frac{1}{2}x_{2,max}(1 - t_1 - t_2)}{\frac{1}{4}t_1^2 - \frac{1}{3}t_1^3 - \frac{1}{4}(1 - t_2)^2 + \frac{1}{6}(1 - t_2)^3}. \qquad (3.4.2.25)$$

$$\hat{\nu}_1 = \frac{-x_{2,max} - \frac{1}{2}\hat{\lambda}_1 t_1^2}{t_1}. \qquad (3.4.2.26)$$

$$\hat{\nu}_2 = \frac{-x_{2,max} - \frac{1}{2}\hat{\lambda}_1(1 - t_2)^2}{(1 - t_2)^2}. \qquad (3.4.2.27)$$

The time points t_1 and t_2 may be determined as follows

$$x_2(t) \leqslant x_{2,max} \qquad 0 \leqslant t \leqslant t_1 \wedge t_2 \leqslant t \leqslant 1, \qquad (3.4.2.28)$$

and

$$\hat{\nu}_1 \geqslant 0, \qquad (3.4.2.29)$$

$$\hat{\nu}_2 \geqslant 0, \qquad (3.4.2.30)$$

$$\hat{\eta}_2(t) \geqslant 0 \qquad t_1 \leqslant t \leqslant t_2. \qquad (3.4.2.31)$$

Consider the state variable x_2 on $[0, t_1]$,

$$\hat{x}_2(t) = -(\hat{\nu}_1 + \hat{\lambda}_1 t_1)t + \frac{1}{2}\hat{\lambda}_1 t^2 \qquad 0 \leqslant t \leqslant t_1. \qquad (3.4.2.32)$$

Thus

$$\dot{\hat{x}}_2(t) = -(\hat{\nu}_1 + \hat{\lambda}_1 t_1) + \lambda_1 t \qquad 0 \leqslant t \leqslant t_1, \qquad (3.4.2.33)$$

$$\ddot{\hat{x}}_2(t) = \hat{\lambda}_1 \qquad 0 \leqslant t \leqslant t_1. \qquad (3.4.2.34)$$

At the point

$$\bar{t} = \frac{\hat{\nu}_1}{\hat{\lambda}_1} + t_1, \tag{3.4.2.35}$$

the state variable x_2 has an extreme point. Because of (3.4.2.20) and (3.4.2.31) we have

$$\hat{\lambda}_1 \leqslant 0. \tag{3.4.2.36}$$

Thus x_2 has a maximum at $\bar{t}$. Because of (3.4.2.28) this maximum cannot be a point of the interval $[0,t_1)$ and hence either

$$\frac{\hat{\nu}_1}{\hat{\lambda}_1} + t_1 < 0, \tag{3.4.2.27}$$

or

$$\frac{\hat{\nu}_1}{\hat{\lambda}_1} + t_1 \geqslant 0. \tag{3.4.2.28}$$

Using (3.4.2.26) it follows that (3.4.2.37) cannot hold. Because of (3.4.2.29) and (3.4.2.36), in the case of (3.4.2.38) it must be

$$\hat{\nu}_1 = 0. \tag{3.4.2.39}$$

A similar derivation on the interval $[t_2,1]$ yields

$$\hat{\nu}_2 = 0. \tag{3.4.2.40}$$

Using (3.4.2.25) - (3.4.2.27), (3.4.2.39) and (3.4.2.40), it is possible to determine t_1 and t_2 as

$$t_1 = \frac{3}{2}\frac{x_{2,max} - 1}{x_{2,max}}, \tag{3.4.2.41}$$

$$t_2 = 1 - t_1. \tag{3.4.2.42}$$

As with the previous example, an alternative method is to use condition (3.3.4.1), i.e.

$$H[t_i +] = H[t_i -] \qquad i = 1,2. \tag{3.4.2.43}$$

A simple derivation shows in this case that (3.4.2.39) and (3.4.2.40) must hold.

4. Sequential quadratic programming in function spaces.

In this chapter a first step is taken towards a numerical solution of problem (SCOCP). In Section 4.1 we shall present the method in the abstract terminology of problem (EIP) of Chapter 2. Section 4.2 deals with the application of the method to optimal control problems. The formulation follows from the interpretation of problem (SCOCP) as a specialization of the abstract problem (EIP). A number of details concerning the application of the abstract method to the problem (SCOCP) are discussed in Section 4.3. An outline of the implementation of the method is given in Section 4.4.

4.1. Description of the method in terms of nonlinear programming in Banach spaces.

The method that is proposed in this section for the solution of the abstract optimization problem (EIP) is a generalization of a certain sequential quadratic programming method for the solution of finite-dimensional nonlinear programming problems. For a description of various of these sequential quadratic programming methods we refer to Bertsekas (1982), Gill et al. (1981), Han (1976), Powell (1978, 1980), Schittkowski (1980, 1981), Stoer (1984), Tapia (1974a, 1974b, 1977, 1978).

4.1.1. Motivation for sequential quadratic programming methods.

In this section we shall give a motivation for the use of sequential quadratic programming methods by considering the solution of problem (EIP) stated in Section 2.1 :

Problem (EIP): *Given Banach spaces X, Y and Z, twice continuously Fréchet differentiable mappings $\tilde{f} : X \to \mathbb{R}$, $\tilde{g} : X \to Y$ and $\tilde{h} : X \to Z$, a convex set $A \subset X$ having a nonempty interior, and a closed convex cone $B \subset Y$ with $0 \in B$ and having a nonempty interior, then find an $\hat{x} \in A$, such that $\tilde{g}(\hat{x}) \in B$ and $\tilde{h}(\hat{x}) = 0$, and that*

$$\tilde{f}(\hat{x}) \leqslant \tilde{f}(x) \quad \text{for all} \quad x \in A \cap \tilde{g}^{-1}(B) \cap \mathrm{N}(\tilde{h}).$$

In the sequel we shall assume that in the formulation of problem (EIP), the set A is the entire space X, i.e. $A = X$. This is done because in a numerical method the more explicit formulation of inequality constraints of the form $\tilde{g}(x) \in B$ is required.

Sequential quadratic programming methods (SQP-methods) are based on the observation that 'near' the solution, the original problem may be replaced by a suitable quadratic programming problem. SQP-methods make use of the sequential solution of quadratic subproblems, to generate directions of search. Along these directions better approximations to the solution are determined.

The motivation for the quadratic subproblems follows directly from the second order sufficient conditions for optimality discussed in Section 2.3. It may be deduced from Theorem 2.16 that the Lagrangian $L(x, \hat{y}^*, \hat{z}^*)$ has a local minimum in the subspace spanned by the linearized constraints, at a point $(\hat{x}, \hat{y}^*, \hat{z}^*)$ for which the sufficient conditions for optimality of part (ii) of Theorem 2.16 hold.

This observation is the motivation for the idea to calculate a direction of search for the improvement of the current estimate x_i of the solution by solving the linearly constrained subproblem :

$$\underset{\Delta x_i}{\textit{Minimize}} \quad L(x_i+\Delta x_i, y_i^*, z_i^*).$$

$$\textit{subject to}: \tilde{g}(x_i) + \tilde{g}'(x_i)(\Delta x_i) \in B,$$

$$\tilde{h}(x_i) + \tilde{h}'(x_i)(\Delta x_i) = 0.$$

where $\tilde{g}$ and $\tilde{h}$ are as defined in problem (EIP) and y_i^* and z_i^* are estimates of the Lagrange multipiers $\hat{y}^*$ and $\hat{g}^*$.

What is obtained is a linearly constrained minimization problem with a nonlinear objective function, which may be approximated by a second order expansion at $x = x_i$.

$$L(x_i+\Delta x_i, y_i^*, z_i^*) \sim L(x_i, y_i^*, z_i^*) + \tilde{f}'(x_i)(\Delta x_i) - y_i^*\tilde{g}'(x_i)(\Delta x_i) - z_i^*\tilde{h}'(x_i)(\Delta x_i) + \frac{1}{2}L''(x_i, y_i^*, z_i^*)(\Delta x_i)(\Delta x_i).$$

Based on this expansion the following linearly constrained quadratic subproblem is constructed for the calculation of a direction of search Δx_i.

Problem (EIQP):

$$\underset{\Delta x_i}{\textit{Minimize}} \quad \tilde{f}'(x_i)(\Delta x_i) + \frac{1}{2}L''(x_i, y_i^*, z_i^*)(\Delta x_i)(\Delta x_i). \tag{4.1.1.1}$$

$$\textit{subject to}: \tilde{g}(x_i) + \tilde{g}'(x_i)(\Delta x_i) \in B, \tag{4.1.1.2}$$

$$\tilde{h}(x_i) + \tilde{h}'(x_i)(\Delta x_i) = 0. \tag{4.1.1.3}$$

In this problem formulation the term $(y_i^*\tilde{g}(x_i) + z_i^*\tilde{h}(x_i))(\Delta x_i)$ is omitted. The reason for this is that we want to obtain a quadratic subproblem which, at the optimal point $\hat{x}$, has the same Lagrange multipliers as the original problem. When the term $(y_i^*\tilde{g}(x_i) + z_i^*\tilde{h}(x_i))(\Delta x_i)$ would not have been omitted, then the Lagrange multipliers of the subproblem at the point x_i would have been $\hat{y}^* - y_i^*$ and $\hat{z}^* - z_i^*$, which would have meant that the Lagrange multipliers of the subproblem would have converged to zero as $x_i \to \hat{x}$. Because the Lagrange multipliers of the subproblem play an important part in the determination of the set of active constraints, this is an undesirable phenomenon. With the modification mentioned above the Lagrange multipliers obtained via the solution of problem (EIQP) may be used as new estimates of the Lagrange multipliers $\hat{y}^*$ and $\hat{z}^*$ of the original problem.

An alternative motivation for the subproblems follows from the application of Newton's method to the first order necessary conditions for optimality. Consider thereto problem (EIP) without the constraint $\tilde{g}(x) \in B$. Assuming that the hypotheses of part (ii) of Theorem 2.12 hold, the first order necessary conditions for optimality imply that at a point $\hat{x}$, there exists a $\hat{z}^* \in Z^*$, such that

$$F(\hat{x}, \hat{z}^*) = 0, \tag{4.1.1.4}$$

where the operator $F : X \times Z^* \to X^* \times Z$ is defined by :

$$F(x, z^*) := \begin{vmatrix} \tilde{f}'(x) - z^*\tilde{h}'(x) \\ \tilde{h}(x) \end{vmatrix} \tag{4.1.1.5}$$

The method of Newton applied to (4.1.1.4) requires the iterative solution of :

$$F(x_i, z_i^*) + F'(x_i, z_i^*)(\Delta x_i, \Delta z_i^*) = 0, \tag{4.1.1.6}$$

or, equivalently,

$$\tilde{f}'(x_i) - z_i^* \tilde{h}'(x_i) + L''(x_i, z_i^*)(\Delta x_i) - \Delta z_i^* \tilde{h}'(x_i) = 0,$$

$$\tilde{h}(x_i) + \tilde{h}'(x_i)(\Delta x_i) = 0.$$

Setting :

$$z_{i+1}^* := z_i^* + \Delta z_i^*,$$

yields :

$$L''(x_i, z_i^*)(\Delta x_i) - z_{i+1}^* \tilde{h}'(x_i) = -\tilde{f}'(x_i), \tag{4.1.1.7}$$

$$\tilde{h}'(x_i)(\Delta x_i) = -\tilde{h}(x_i). \tag{4.1.1.8}$$

When the multiplier z_{i+1}^* is interpreted as a Lagrange multiplier, then the equations (4.1.1.7) - (4.1.1.8) constitute precisely the first order necessary conditions for optimality of :

Problem (EQP) :

$$\underset{\Delta x_i}{Minimize} \quad \tilde{f}'(x_i)(\Delta x_i) + \tfrac{1}{2} L''(x_i, z_i^*)(\Delta x_i)(\Delta x_i), \tag{4.1.1.9}$$

$$subject\ to : \tilde{h}(x_i) + \tilde{h}'(x_i)(\Delta x_i) = 0. \tag{4.1.1.10}$$

The extension of the method of Newton to nonlinear programming problems with inequality constraints is not straightforward. To investigate this consider instead of (4.1.1.4) the inequality (inclusion in a positive cone) :

$$F(\hat{x}, \hat{y}^*) \in C, \tag{4.1.1.11}$$

where the operator $F : X \times Y^* \to X^* \times Y \times Y^* \times \mathbf{R}$ is defined by :

$$F(x, y^*) := \begin{bmatrix} \tilde{f}'(x) - y^* \tilde{g}(x) \\ \tilde{g}(x) \\ y^* \\ y^* \tilde{g}(x) \end{bmatrix}, \tag{4.1.1.12}$$

and

$$C := \{0\} \times B \times B^+ \times \{0\}, \tag{4.1.1.13}$$

with

$$B^+ := \{y^* \in Y^* : \langle y^*, y \rangle \geqslant 0 \ \ for\ all \ \ y \in B\}. \tag{4.1.1.14}$$

Similar to the case of equality constraints, the inclusion (4.1.1.11) constitutes the first order necessary conditions for optimality for problem (EIP) under the assumption that the regularity constant $\hat{\rho}$ may be set equal to one. A generalization of Newton's method to (4.1.1.11) implies the solution of :

$$F(x_i, y_i^*) + F'(x_i, y_i^*)(\Delta x_i, \Delta y_i^*) \in C, \tag{4.1.1.15}$$

or, equivalently using $y_{i+1}^* := y_i^* + \Delta y_i^*$.

$$L''(x_i, y_i^*)(\Delta x_i) - y_{i+1}^* \tilde{g}'(x_i) = -\tilde{f}'(x_i), \tag{4.1.1.16}$$

$$\tilde{g}(x_i) + \tilde{g}'(x_i)(\Delta x_i) \in B, \tag{4.1.1.17}$$

$$y_{i+1}^* \in B^+, \tag{4.1.1.18}$$

$$\langle y_{i+1}^*, \tilde{g}(x_i) + \tilde{g}'(x_i)(\Delta x_i) \rangle - \Delta y_i^* \tilde{g}'(x_i)(\Delta x_i) = 0. \tag{4.1.1.19}$$

The conditions (4.1.1.16) - (4.1.1.19) are not necessary conditions for optimality of any (sub)problem as in the equality constrained case. However, if we replace (4.1.1.19) by

$$\langle y_{i+1}^*, \tilde{g}(x_i) + \tilde{g}'(x_i)(\Delta x_i) \rangle = 0, \tag{4.1.1.20}$$

then conditions (4.1.1.16), (4.1.1.17), (4.1.1.18) and (4.1.1.20) are the first order necessary conditions for optimality of :

Problem (IQP) :

$$\underset{\Delta x_i}{\text{Minimize}} \quad \tilde{f}'(x_i)(\Delta x_i) + \tfrac{1}{2} L''(x_i, y_i^*)(\Delta x_i)(\Delta x_i), \tag{4.1.1.21}$$

$$\text{subject to : } \tilde{g}(x_i) + \tilde{g}'(x_i)(\Delta x_i) \in B. \tag{4.1.1.22}$$

Summarizing the discussion sofar, we gave a motivation for an algorithm which makes use of directions of search calculated via the solution of problem ((E)IQP), either as a minimization of the Lagrangian in the subspace spanned by the linearized constraints, or as a Newton-like method applied to the first order necessary conditions for optimality. We note that in the discussion of the algorithm, implicitly the assumption was made that at every point (x_i, y_i^*, z_i^*) the problem ((E)IQP) has a solution which satisfies the sufficient conditions for optimality of Theorem 2.16.

4.1.2. Active set strategies and merit function.

In this section we shall consider some algorithmic options for SQP-methods for the solution of problem (EIP).

There are essentially two ways in which inequality constraints of the form $\tilde{g}(x) \in B$ may be handled. One way is to use in each iteration of the method an estimate of that part of the constraints which is active at the solution. This estimate is called the *working set* and is updated before each iteration. The constraints in the working set together with the equality constraints define a nonlinear programming problem with only equality constraints. Application of the SQP-method to this problem requires in each iteration the solution of a problem of the type (EQP), i.e. a quadratic programming problem with linear equality constraints. A strategy which is used to determine the working set is called an *active set strategy*. In the case of SQP with equality constrained subproblems the active set strategy is based on an estimate of the solution of the original problem. The second way to handle the inequality constraints $\tilde{g}(x) \in B$ is to solve the problem (EIQP) as a quadratic programming problem with linear equality and inequality constraints. The major problem in a solution procedure of problem (EIQP) is again the determination of the active set, i.e. that part of the constraints $\tilde{g}(x_i) + \tilde{g}'(x_i)(\Delta x_i) \in B$ which are satisfied as equalities at the solution point. Thus in this case the active set strategy is part of the quadratic programming algorithm that calculates the solution of the subproblem (EIQP).

We note one essential difference between the two methods. With the first method the active set strategy focusses directly on the active set of the original (nonlinear) problem

whereas with the second method the active set strategy is used to determine the active set of problem (EIQP).

The discussion in the previous section focussed on the motivation for the calculation of directions of search via the solution of a quadratic programming problem. The derivation of this quadratic programming problem is entirely based on linearization arguments that hold only in a neighborhood of a solution $(\hat{x}, \hat{y}^*, \hat{z}^*)$. Hence it must be assumed that the current iterate (x_i, y_i^*, z_i^*) is 'sufficiently close' to the solution. For a practical procedure this assumption is too restrictive. Fortunately it is possible to 'globalize' the method proposed, by means of a *merit function*. This is a function which assigns a real value to each triple $(x, y^*, z^*) \in X \times Y^* \times Z^*$, and which has the property that it has a minimum at the point $(\hat{x}, \hat{y}^*, \hat{z}^*)$. Using the direction of search Δx_i and the Lagrange multipliers $(\bar{y}^*, \bar{z}^*)$ obtained via the solution of the problem (EIQP), the current iterate (x_i, y_i^*, z_i^*) is, at each iteration, modified such that the merit function is minimized along the direction of search $(\Delta x_i, \bar{y}^* - y_i^*, \bar{z}^* - z_i^*)$, i.e.

$$M\{\alpha_i\} = \min_{\alpha > 0} M\{\alpha\},$$

where M denotes the merit function and the notation $\{\alpha\}$ is used to replace $(x_i + \alpha \Delta x_i,\ y_i^* + \alpha(\bar{y}^* - y_i^*),\ z_i^* + \alpha(\bar{z}^* - z_i^*))$.

The parameter α_i is called the *step size*.

We note that in order to preserve the excellent local convergence properties of Newton's method, the merit function must have the property that in a neighborhood of the solution, the step size α_i converges to one.

4.1.3. Abstract version of the algorithm.

Based on the sequential solution of quadratic programming problems (EIQP) we are led to the following algorithm :

Algorithm 4.1 :

(0) *Set* $x_0 := $ *given value*; $i := 0$;

(i) *Calculate first order Lagrange multiplier estimates* (y_i^*, z_i^*) *as the multipliers corresponding to the solution of :*

$$\underset{d}{\text{Minimize}} \quad \tilde{f}'(x_i)(d) + \tfrac{1}{2} \langle Gd, d \rangle,$$

$$\text{subject to : } \tilde{g}(x_i) + \tilde{g}'(x_i)(d) \in B,$$

$$\tilde{h}(x_i) + \tilde{h}'(x_i)(d) = 0.$$

where $G : X \times X \to \mathbb{R}$ *is a positive definite mapping used to imitate an inner product in the Banach space* X, *as* $(x \mid y) = \langle Gx, y \rangle$. †

(ii) *Calculate the Hessian of the Lagrangian at* x_i

$$L''(x_i, y_i^*, z_i^*) := \tilde{f}''(x_i) - y_i^* \tilde{g}''(x_i) - z_i^* \tilde{h}''(x_i).$$

(iii) *Calculate second order Lagrange multiplier estimates* $(\bar{y}^*, \bar{z}^*)$ *and the Newton direction* d_N *as the solution of :*

$$\underset{d}{\text{Minimize}} \quad \tilde{f}'(x_i)(d) + \tfrac{1}{2} < L''(x_i, y_i^*, z_i^*)d, d >,$$

$$\text{subject to : } \tilde{g}(x_i) + \tilde{g}'(x_i)(d) \in B,$$

$$\tilde{h}(x_i) + \tilde{h}'(x_i)(d) = 0.$$

(iv) *If* $\|d_N\| \leqslant \epsilon$ *then ready.*

(v) *Calculate a step size* α_i *such that*

$$M\{\alpha_i\} = \min_{\alpha > 0} M\{\alpha\},$$

and set

$$x_{i+1} := x_i + \alpha_i d_N,$$

$$y_{i+1}^* := y_i^* + \alpha_i(\bar{y}^* - y_i^*),$$

$$z_{i+1}^* := z_i^* + \alpha_i(\bar{z}^* - z_i^*).$$

(vi) $i := i + 1$
goto (ii).

The algorithm above is based on the sequential solution of quadratic programming problems with equality and inequality constraints (EIQP). A similar algorithm follows for the case that the calulation of the direction of search is based on the solution of quadratic programming problems with only equality constraints (EQP). In this case the active set strategy is to be performed at the point of step (ii).

† The mapping G can be chosen the identy operator in Hilbert spaces. Using the interpretation of the mapping G as an imitation of an inner product, the solution d of step (i) has the interpretation of a generalized projection of the negative gradient on the subspace spanned by the linearized constraints.

4.2. Application of the method to optimal control problems.

4.2.1. Formulation of the problems (EIQP/SCOCP) and (EQP/SCOCP).

In this section we shall consider the formulation of the problems (EIQP/SCOCP) and (EQP/SCOCP) which are the specializations of the problems (EIQP) and (EQP) for the state constrained optimal control problems (SCOCP). From Section 3.1 we recall

Problem (SCOCP) : *Determine a control function* $\hat{u} \in L_\infty[0,T]^m$, *a state trajectory* $\hat{x} \in W_{1,\infty}[0,T]^n$ *and a final time* $\hat{T} > 0$, *which minimize the functional*

$$h_0(x(0)) + \int_0^T f_0(x(t),u(t),t)\,dt + g_0(x(T),T),$$

subject to the constraints :

$$\dot{x}(t) = f(x(t),u(t),t) \quad a.e.\ 0 \leqslant t \leqslant T,$$

$$D(x(0)) = 0,$$

$$E(x(T),T) = 0,$$

$$u(t) \in U \quad a.e.\ 0 \leqslant t \leqslant T,$$

$$S_1(x(t),u(t),t) \leqslant 0 \quad a.e.\ 0 \leqslant t \leqslant T,$$

$$S_2(x(t),t) \leqslant 0 \quad 0 \leqslant t \leqslant T,$$

where : $h_0 : \mathbb{R}^n \to \mathbb{R}$; $f_0 : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R} \to \mathbb{R}^n$; $g_0 : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}$; $D : \mathbb{R}^n \to \mathbb{R}^c$; $f : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R} \to \mathbb{R}^n$; $E : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}^q$; $S_1 : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R} \to \mathbb{R}^{k_1}$; $S_2 : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}^{k_2}$; $U \subset \mathbb{R}^m$, *is a convex set with nonempty interior.*

For all $x \in \mathbb{R}^n, u \in \mathbb{R}^m$ *rank* $S_{1u}(x,u,t) = k_1$ *a.e.* $0 \leqslant t \leqslant T$.

The functions h_0, f_0, g_0, f, D, E S_1 *and* S_2 *are twice continuously differentiable functions with respect to all arguments.*

For the sake of brevity we shall consider fixed final time problems, because variable final time problems can be transformed into fixed final time problems (cf. Section 3.3.4).

The assumption that, in the formulation of problem (EIP), the set A is the entire space X, becomes in the formulation of problem (SCOCP) :

$$U = L_\infty[0,T]^m. \qquad (4.2.1.1)$$

This will be assumed in the sequel without any further reference.

To denote the variables in the current approximation to the solution of problem (SCOCP) we shall use the notation $x^i(t)$, $u^i(t)$, $\lambda^i(t)$, $\eta_1^i(t)$, $\xi^i(t)$, $\eta_2^i(t)$, ν_j^i, σ^i and μ^i. The notation $[t]$ is used to replace argument lists involving $x^i(t)$, $u^i(t)$, $\lambda^i(t)$, $\eta_1^i(t)$, $\xi^i(t)$, σ^i and μ^i, e.g. $[t] \equiv (x^i(t),u^i(t))$.

For the formulation of the problems (EIQP) and (EQP) an expression for the second Fréchet differential of the Lagrangian is required.

Lemma 4.2 : *Under the assumptions given in the formulation of problem (SCOCP), the Lagrangian is twice continuously Fréchet differentiable for all* $x_i \in W_{1,\infty}[0,T]^n$, $u^i \in L_\infty[0,T]^m$, $\lambda^i \in NBV[0,T]^n$, $\eta_1^i \in L_\infty[0,T]^{k_1}$, $\xi^i \in NBV[0,T]^{k_2}$, $\sigma^i \in \mathbb{R}^c$, $\mu^i \in \mathbb{R}^q$ *and*

$$L''(x^i,u^i,\lambda^i,\eta_1^i,\xi^i,\sigma^i,\mu^i)(\delta x_1,\delta u_1)(\delta x_2,\delta u_2) = \delta x_1(0)^T(h_{0xx}[0]+\sigma * D_{xx}[0])\delta x_2(0)$$

$$+ \int_0^T [\delta x_1(t)^T \ \delta u_1(t)^T] \begin{bmatrix} H_{xx}[t]+\eta_1^i(t)*S_{1xx}[t] & H_{xu}[t]+\eta_1^i(t)*S_{1xu}[t] \\ H_{ux}[t]+\eta_1^i(t)*S_{1ux}[t] & H_{uu}[t]+\eta_1^i(t)*S_{1uu}[t] \end{bmatrix} \begin{bmatrix} \delta x_2(t) \\ \delta u_2(t) \end{bmatrix} dt$$

$$+ \int_0^T \delta x_1(t)^T (d\,\xi^i(t)*S_{2xx}[t])\delta x_2(t) +$$

$$\delta x_1(T)^T(g_{0xx}[T]+\mu * E_{xx}[T])\delta x_2(T). \quad \dagger \qquad (4.2.1.2)$$

where the Hamiltonian $H(x,u,\lambda,t)$ *is defined by :*

$$H(x,u,\lambda,t) := f_0(x,u,t) + \lambda^T f(x,u,t).$$

A proof of this lemma is not given here as it follows in a straightforward fashion from the application of Lemma 1.4a, p.94 of Kirsch et al. (1978) to the first Fréchet differential of the Lagrangian.

In the sequel we shall occasionally use the pair η_2^i and ν^j instead of the multiplier ξ^i. The multiplier η_2^i represents the time derivative of ξ^i whenever it exists and the multipliers ν^j represent the discontinuities of the multiplier ξ^i at time points t_j, i.e.

$$\eta_2^i(t) := \dot{\xi}^i(t) \qquad a.e. \quad 0 \leqslant t \leqslant T, \qquad (4.2.1.3)$$

and

$$\nu^j := \xi^i(t_j+) - \xi^i(t_j-). \qquad (4.2.1.4)$$

The specialization of problem (EIQP) for problem (SCOCP) follows directly from Lemma 4.2 and the abstract formulation of problem (SCOCP) as given in Section 3.2.

† The notation $\bar{a} * M$ is used to denote the tensor product of a vector $\bar{a}$ with a block matrix M. The interpretation of this product is that for instance $\sigma * D_{xx}[0]$ is the Hessian of the functional $\sigma^T D(x)$ with respect to x for fixed σ at $\hat{x}(0)$.

Problem (EIQP/SCOCP):

$$\underset{d_x,\,d_u}{\text{Minimize}} \quad h_{0x}[0]d_x(0) + \int_0^T (f_{0x}[t]d_x(t) + f_{0u}[t]d_u(t))dt + g_{0x}[T]d_x(T) +$$

$$\tfrac{1}{2}d_x(0)^T M_1 d_x(0) + \tfrac{1}{2}\int_0^T [d_x(t)^T \; d_u(t)^T] \begin{bmatrix} M_2[t] & M_3[t] \\ M_3[t]^T & M_4[t] \end{bmatrix} \begin{bmatrix} d_x(t) \\ d_u(t) \end{bmatrix} dt$$

$$+ \tfrac{1}{2}\sum_j d_x(t_j)^T M_6[t_j]d_x(t_j) + \tfrac{1}{2}d_x(T)^T M_5 d_x(T), \qquad (4.2.1.5)$$

$$\text{subject to}: \dot{d}_x = f_x[t]d_x + f_u[t]d_u + f[t] - \dot{x}^i(t) \quad \text{a.e. } 0 \leqslant t \leqslant T, \qquad (4.2.1.6)$$

$$D[0] + D_x[0]d_x(0) = 0, \qquad (4.2.1.7)$$

$$E[T] + E_x[T]d_x(T) = 0, \qquad (4.2.1.8)$$

$$S_1[t] + S_{1x}[t]d_x + S_{1u}[t]d_u \leqslant 0 \quad \text{a.e. } 0 \leqslant t \leqslant T, \qquad (4.2.1.9)$$

$$S_2[t] + S_{2x}[t]d_x \leqslant 0 \quad 0 \leqslant t \leqslant T, \qquad (4.2.1.10)$$

$$\text{where}: M_1 := h_{0xx}[0] + \sigma * D_{xx}[0], \qquad (4.2.1.11)$$

$$M_2[t] := f_{0xx}[t] + \lambda^i * f_{xx}[t] + \eta_1^j * S_{1xx}[t] + \eta_2^j * S_{2xx}[t], \qquad (4.2.1.12)$$

$$M_3[t] := f_{0xu}[t] + \lambda^i * f_{xu}[t] + \eta_1^j * S_{1xu}[t], \qquad (4.2.1.13)$$

$$M_4[t] := f_{0uu}[t] + \lambda^i * f_{uu}[t] + \eta_1^j * S_{1uu}[t], \qquad (4.2.1.14)$$

$$M_5 := g_{0xx}[T] + \mu * E_{xx}[T], \qquad (4.2.1.15)$$

$$M_6[t_j] := \nu_j^j * S_{2xx}[t_j] \text{ for all } j. \qquad (4.2.1.16)$$

The statement of problem (EQP/SCOCP) requires the introduction of the following somewhat complicated terminology.

Recall the definition (3.3.5.10) of the vector function $\tilde{S}(x,u,t)$ which contains all control and state constraints. With every component $\tilde{S}_l$ $(l = 1,...k_1+k_2)$ a set $W_l \subset [0,T]$ is associated, which is the collection of all time points for which the constraint $\tilde{S}_l$ is supposed to hold as equality. The set W_l is called the *working set* of $\tilde{S}_l$.

The sets W_l consist of m_l^b *boundary intervals* $[t^l_{2j-1}, t^l_{2j}]$ $(j = 1,2,...m_l^b)$ and m_l^c *contact points* $t^l_{2m_l^b+j}$ $(j = 1,2,...m_l^c)$.

$I(t)$ is used to denote the *index set of active constraints* at the time point t, i.e.

$$I(t) := \{l : 1 \leqslant l \leqslant k_1+k_2 \wedge t \in W_l\} \quad \text{for all} \quad 0 \leqslant t \leqslant T.$$

$\bar{k}(t)$ denotes the number of constraints in the working set, i.e. the number of indices in the set $I(t)$.

Elements of the index set $I(t)$ are referred to as $i_1, i_2, ...etc.$, i.e.

$$I(t) = \{i_1, i_2, ..., i_{\bar{k}(t)}\}.$$

The state constraints of the subproblem (EQP/SCOCP) follow from the linearization of the constraints

$$\tilde{S}_l(x(t),u(t),t) = 0 \qquad a.e.\ \ t \in W_l,\ l = 1,2,\ldots k_1+k_2. \tag{4.2.1.17}$$

which (along $(x^i(t),u^i(t))$ are given by :

$$\tilde{S}_l[t] + \tilde{S}_{lx}[t]d_x(t) + \tilde{S}_{lu}[t]d_u(t) = 0 \qquad a.e.\ \ t \in W_l,\ l = 1,2,\ldots k_1+k_2. \tag{4.2.1.18}$$

The $\bar{k}(t)$-vector $R[t]$ is used to denote all constraints in the working set at time point t in a compact way, i.e.

$$R_l[t] := \tilde{S}_{i_l}[t] \qquad l = 1,2,\ldots \bar{k}(t), \quad 0 \leqslant t \leqslant T. \tag{4.2.1.19}$$

The linearization of the state constraints is denoted by

$$R[t] + R_x[t]d_x(t) + R_u[t]d_u(t) = 0 \qquad 0 \leqslant t \leqslant T. \tag{4.2.1.20}$$

(We note that when $\bar{k}(t)$ is zero, then $R[t]$ has dimension zero and hence, at these time points, there is no constraint on d_x and d_u). With the terminology introduced above, problem (EQP/SCOCP) becomes :

Problem (EQP/SCOCP) :

$$\underset{d_x,d_u}{\text{Minimize}} \quad h_{0x}[0]d_x(0) + \int_0^T (f_{0x}[t]d_x(t) + f_{0u}[t]d_u(t))dt + g_{0x}[T]d_x(T) +$$

$$\tfrac{1}{2}d_x(0)^T M_1 d_x(0) + \tfrac{1}{2}\int_0^T [d_x(t)^T\ d_u(t)^T] \begin{bmatrix} M_2[t] & M_3[t] \\ M_3[t]^T & M_4[t] \end{bmatrix} \begin{bmatrix} d_x(t) \\ d_u(t) \end{bmatrix} dt$$

$$+ \tfrac{1}{2}\sum_j d_x(t_j)^T M_6[t_j]d_x(t_j) + \tfrac{1}{2}d_x(T)^T M_5 d_x(T), \tag{4.2.1.21}$$

$$\textit{subject to :}\ \dot{d}_x = f_x[t]d_x + f_u[t]d_u + f[t] - \dot{x}^i(t) \qquad a.e.\ \ 0 \leqslant t \leqslant T, \tag{4.2.1.22}$$

$$D[0] + D_x[0]d_x(0) = 0, \tag{4.2.1.23}$$

$$E[T] + E_x[T]d_x(T) = 0, \tag{4.2.1.24}$$

$$R[t] + R_x[t]d_x + R_u[t]d_u = 0 \qquad a.e.\ \ 0 \leqslant t \leqslant T, \tag{4.2.1.25}$$

where the matrices M_1, M_2, M_3, M_4, M_5, M_6 are defined by (4.2.1.11) - (4.2.1.16).

4.2.2. Active set strategies for problem (SCOCP).

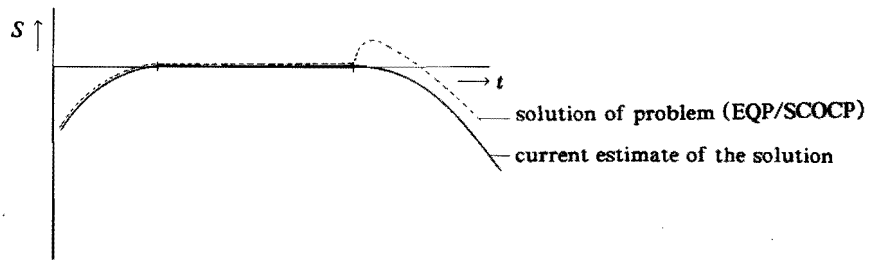
Most solution procedures for the solution of optimal control problems involving constraints on the control and/or state consist of two stages. In the first stage the *structure of the solution* is determined, i.e. the sequence of time intervals on which the constraints are active and inactive on $[0,T]$. In addition to the (estimated) structure of the solution, this stage yields also a rough approximation to the solution. In the second stage, the exact solution is determined using the results of the first stage. In this section an argumentation for and definition of the two stages will be given.

Consideration of the SQP-methods described in Section 4.1 for the solution of problem (SCOCP) yields the sequential solution of problems of the type (EIQP/SCOCP) or (EQP/SCOCP). In the case that problem (EIQP/SCOCP) has a unique solution for which the sufficient conditions for optimality of Theorem 2.16 are satisfied, the main problem of obtaining the solution of problem (EIQP/SCOCP) is the determination of the set of active points of the state constraints. For if this set is available, then the solution of problem

(EIQP/SCOCP) can be determined as the solution of problem (EQP/SCOCP) using the set of active points as working set. The solution of problem (EQP/SCOCP) can be obtained as the solution of a linear multipoint boundary value problem (cf. Section 5.1), which admits more or less standard numerical solution procedures. Unfortunately, there are no standard procedures for the solution of problems of the type (EIQP/SCOCP), or more specifically for the determination of the active set of this type of problems. As a first step towards a solution procedure, we consider a general procedure for the solution of the finite-dimensional counterpart of problem (EIQP/SCOCP), which is reviewed in Appendix A. This method has the following characteristics :

1) The method has an iterative nature using as candidates for the solution, solutions to quadratic programming problems with only linear equality constraints.
2) The iterates are all feasible points, i.e. the complete set of inequality constraints of the quadratic programming problem are satisfied at each iteration.
3) The active set strategy consists of addition of constraints to the working set whenever the step size is restricted (i.e. when one or more constraints become violated at the candidate solution point), or the (possible) deletion of constraints from the working set whenever the direction of search becomes zero (i.e. the minimum in the current subspace is achieved) and the Lagrange multiplier corresponding to the constraint has a wrong sign.

It is not possible to apply the method to the solution of problem (EIQP/SCOCP) without adaptation. The reason for this is the infinite-dimensional nature of the constraints (4.2.1.9) - (4.2.1.10) In fact the constraints (4.2.1.9) - (4.2.1.10) represent a k_1+k_2 set of constraints at each time point t. As a result of this it is likely that during the execution of the method the stepsize becomes zero, because any nonzero step would lead to a violation of the constraint (cf. Figure 4.1) and hence the method would fail to converge.



Infeasible direction of search.
Figure 4.1

We recall that if it would be possible to solve problem (EIQP/SCOCP) at each iteration of Algorithm 4.1, then ultimately (assuming convergence) the solution of problem (SCOCP) would be obtained. In that case, the stucture of the solution would follow simply via an inspection of the set of active points. However, because problem (EIQP/SCOCP) cannot be solved easily, the solution process is broken into the two stages mentioned earlier, the first being the determination of an estimate of the set of active points of the state constraints.

Having this goal in mind we consider the replacement of problem (EIQP/SCOCP) by a simpler problem such that the solution of this problem is an approximation to the solution

of problem (EIQP/SCOCP). Therefore the grids Δ^1 and Δ^2 are introduced as :

$$\Delta^j := (\bar{t}_0^j, \bar{t}_1^j, \ldots, \bar{t}_{\bar{p}_j}^j) \qquad j=1,2 \qquad (4.2.2.1)$$

and

$$\Delta := \Delta^1 \times \Delta^2, \qquad (4.2.2.2)$$

where the (time) points $\bar{t}_l^j \in [0,T]$ satisfy :

$$0 \leqslant \bar{t}_0^j \leqslant \bar{t}_1^j \ldots\ldots \leqslant \bar{t}_{\bar{p}_j}^j \leqslant T \qquad j=1,2. \qquad (4.2.2.3)$$

Problem (EIQP/SCOCP) is now replaced by a similar linear-quadratic optimal control problem, where the junction and contact points of the constraints (4.2.1.9) and (4.2.1.10) are restricted to the grids Δ^1 and Δ^2 respectively. The problem (EIQP/SCOCP) with *junction and contact points restricted to the grid* Δ is called problem (EIQP/SCOCP/Δ).

Presumably, if the grid Δ is sufficiently 'fine', then the solution of problem (EIQP/SCOCP/Δ) will be an approximation to the solution of problem (EIQP/SCOCP). Assuming that the SQP-method converges with the direction of search obtained via the solution of problem (EIQP/SCOCP/Δ), the structure of the solution of problem (SCOCP) will be obtained as the structure of the converged solution.

The definition of problem (EIQP/SCOCP/Δ) will now be made more explicit. By restricting the junction and contact points to a finite set of points, the problem (EIQP/SCOCP) is in fact replaced by a minimization problem over a set of problems (EQP/SCOCP) where the working set must be chosen according to the restriction that the junction and contact points are points of the grid Δ.

Definition 4.3 : *Given a pair of functions $d_x \in W_{1,\infty}[0,T]^n$ and $d_u \in PC[0,T]^m$, the sets of boundary points of the constraints (4.2.1.9) and (4.2.1.10) with respect to the grid Δ (defined by (4.2.2.1) - (4.2.2.3)) are defined as follows :*

$J_B^{1l}(d_x, d_u, \Delta^1)$ is the union of the intervals $[\bar{t}_r^1 \leqslant t \leqslant \bar{t}_{r+1}^1]$ $(r=0,1,\ldots,\bar{p}_1-1)$ for which :

$$S_{1l}[\bar{t}_r^1+] + S_{1l,x}[\bar{t}_r^1+]d_x(\bar{t}_r^1) + S_{1l,u}[\bar{t}_r^1+]d_u(\bar{t}_r^1+) = 0 \qquad (4.2.2.4)$$

and

$$S_{1l}[\bar{t}_{r+1}^1-] + S_{1l,x}[\bar{t}_{r+1}^1-]d_x(\bar{t}_{r+1}^1) + S_{1l,u}[\bar{t}_{r+1}^1-]d_u(\bar{t}_{r+1}^1-) = 0. \qquad (4.2.2.5)$$

$J_B^{2l}(d_x, \Delta^2)$ is the union of the intervals $[\bar{t}_r^2 \leqslant t \leqslant \bar{t}_{r+1}^2]$ $(r=0,1,\ldots,\bar{p}_2-1)$ for which

$$S_{2l}[\bar{t}_r^2] + S_{2l,x}[\bar{t}_r^2]d_x(\bar{t}_r^2) = 0 \qquad (4.2.2.6)$$

and

$$S_{2l}[\bar{t}_{r+1}^2] + S_{2l,x}[\bar{t}_{r+1}^2]d_x(\bar{t}_{r+1}^2) = 0, \qquad (4.2.2.7)$$

The definition of problem (EIQP/SCOCP/Δ) is stated as a combination of problems (EIQP/SCOCP) and (EQP/SCOCP), and uses the sets of boundary points as working sets.

Problem (EIQP/SCOCP/Δ) : *Determine, if it exists, a control function* $\hat{d}_u \in PC[0,T]^m$, *and a state trajectory* $\hat{d}_x \in W_{1,\infty}[0,T]^n$, *which minimize the functional*

$$h_{0x}[0]d_x(0) + \int_0^T (f_{0x}[t]d_x(t) + f_{0u}[t]d_u(t))dt + g_{0x}[T]d_x(T) +$$

$$\tfrac{1}{2} d_x(0)^T M_1 d_x(0) + \tfrac{1}{2} \int_0^T [d_x(t)^T \; d_u(t)^T] \begin{bmatrix} M_2[t] & M_3[t] \\ M_3[t]^T & M_4[t] \end{bmatrix} \begin{bmatrix} d_x(t) \\ d_u(t) \end{bmatrix} dt$$

$$+ \tfrac{1}{2} \sum_j d_x(t_j)^T M_6[t_j] d_x(t_j) + \tfrac{1}{2} d_x(T)^T M_5 d_x(T). \qquad (4.2.2.8)$$

subject to :

$$\dot{d}_x = f_x[t]d_x + f_u[t]d_u + f[t] - \dot{x}^i(t) \quad a.e. \quad 0 \leqslant t \leqslant T, \qquad (4.2.2.9)$$

$$D[0] + D_x[0]d_x(0) = 0, \qquad (4.2.2.10)$$

$$E[T] + E_x[T]d_x(T) = 0. \qquad (4.2.2.11)$$

$$S_{1l}[t] + S_{1l,x}[t]d_x(t) + S_{1l,u}[t]d_u(t) = 0$$

$$\text{for all } t \in J_B^{1l}(d_x, d_u, \Delta^1), \quad l = 1,2,\ldots,k_1, \qquad (4.2.2.12)$$

$$S_{2l}[t] + S_{2l,x}[t]d_x(t) = 0 \quad \text{for all } t \in J_B^{2l}(d_x, \Delta^2), \quad l = 1,2,\ldots,k_2, \qquad (4.2.2.13)$$

$$S_1[\bar{t}_r^1+] + S_{1,x}[\bar{t}_r^1+]d_x(\bar{t}_r^1) + S_{1,u}[\bar{t}_r^1+]d_u(\bar{t}_r^1+) \leqslant 0 \quad r = 0,1,\ldots,\bar{p}_1 - 1, \qquad (4.2.2.14)$$

$$S_1[\bar{t}_r^1-] + S_{1,x}[\bar{t}_r^1-]d_x(\bar{t}_r^1) + S_{1,u}[\bar{t}_r^1-]d_u(\bar{t}_r^1-) \leqslant 0 \quad r = 1,\ldots,\bar{p}_1, \qquad (4.2.2.15)$$

$$S_2[\bar{t}_r^2] + S_{2,x}[\bar{t}_r^2]d_x(\bar{t}_r^2) \leqslant 0 \quad r = 0,1,\ldots,\bar{p}_2. \qquad (4.2.2.16)$$

where the matrices $M_1, M_2, M_3, M_4, M_5, M_6$ *are defined by (4.2.1.11) - (4.2.1.16).*

The definition above shows that restricting the junction and contact points of problem (EIQP/SCOCP) to the grid Δ is not equivalent to replacing the constraints (4.2.1.9) - (4.2.1.10) by a finite set of inequalities, because on boundary intervals the constraints are still to be satisfied as equalities.

The method for the solution of problem (EIQP/SCOCP/Δ), is essentially an adaptation of a certain method for the solution of finite-dimensional quadratic programming problems. The adaptation of the method for the solution of problem (EIQP/SCOCP/Δ) is discussed in detail in Section 5.2.

The first stage of the method is completed once the direction of search is 'sufficiently' small. At this point the structure of the solution of problem (SCOCP) is estimated as the structure of the current iterate. Because the junction and contact points were in the first stage, restricted to a (fixed) finite set of points, it is not likely that the current iterate is a 'good' approximation to the solution.

Therefore a second stage is started, such that in each iteration one or more junction and/or contact points are shifted. The amount of shift required for each point is determined using the violation of the constraints (4.2.1.9) - (4.2.1.10) on interior intervals and the sign information of the Lagrange multipliers on boundary intervals. The techniques used, are essentially strategies which focus on the active set of the original (nonlinear) problem (SCOCP). These techniques are described in Section 5.3. When one or more junction and/or contact points are shifted, a direction of search is calculated via the solution of

problem (EQP/SCOCP). Contrary to the first stage, the second stage is thus based on the sequential solution of quadratic programming problems with only equality constraints.

4.3. Further details of the algorithm.

In step (i) of the abstract Algorithm 4.1 use is made of a mapping G to imitate an inner product in the Banach space X. In the application of the algorithm to problem (SCOCP), we take G such that $<G(x_1,u_1), (x_2,u_2)>$ resembles the L_2-inner product, i.e.

$$<G(x_1,u_1), (x_2,u_2)> \;:=\; \int_0^T (x_1(t)^T x_2(t) + u_1(t)^T u_2(t))\, dt$$

$$\text{for all } x_1,x_2 \in W_{1,\infty}[0,T]^n,\ u_1,u_2 \in L_\infty[0,T]^m. \qquad (4.3.1)$$

With this choice, step (i) of Algorithm 4.1 involves the solution of problem (EIQP/SCOCP/Δ) with $M_1=0$, $M_2[t]=I_n$, $M_3[t]=0$, $M_4[t]=I_m$, $M_5=0$ and $M_6[t]=0$.

In the first stage of the method, the step size α_i is determined using a merit function. Essentially this merit function is a combination of the objective function and a *penalty term*, which is some measure for the constraint violation. The direction of search (which was motivated only by linearization arguments) will, in general, not give a decrease of both the objective function and the penalty term. Decreasing both terms simultaneously can be conflicting goals. In these cases the merit function provides a balance between achieving either of these goals, with the intension that in each iteration progress towards a solution point is made.

We shall now give a formal motivation of the merit function that is used in the current implementation of the method. Recent literature on SQP methods indicate that there are various alternatives to this choice. We do not intend to give a complete survey of possible choices for the merit function; for this we refer to Bertsekas (1982), Fletcher (1981, 1983) and Gill et al. (1984). To the particular choice made in this section we note that, contrary to other choices of merit functions, it allows a rather complete convergence analysis in the finite-dimensional case (cf. Schittkowski (1981)).

A merit function should satisfy the following requirements :

1) The solution of the original problem should be a (local) minimum of the merit function.
2) In combination with the direction of search, it should always be possible to choose a step size, such that the merit function is decreased.
3) The merit function should not inhibit convergence of the step size to one, in a neighborhood of a solution point.

For problems with only equality constraints, a suitable choice of the merit function is the so-called *augmented Lagrangian:*

$$M(x,\lambda;\rho) := \tilde{f}(x) + \lambda^T \tilde{h}(x) + \tfrac{1}{2}\rho \|\tilde{h}(x)\|^2, \qquad (4.3.2)$$

where λ is an estimate for the Lagrange multiplier corresponding to the equality constraint and $\rho > 0$ is a penalty constant.

A motivation for this choice of merit function is that the Lagrangian has a minimum in the tangent subspace of the linearized constraints at a solution point (assuming that the sufficient conditions for optimality of Theorem 2.16 hold at this point). The penalty term

is added to extend this feature to a larger set, outside the tangent subspace of the linearized constraints.
For a 'sufficiently high' value of ρ, the merit function (4.3.2) satisfies the requirements 1) - 3) in the case of finite-dimensional nonlinear programming.

For the extension of this merit function to include also inequality constraints we first consider the finite-dimensional case of one scalar function $g : X \to \mathbf{R}$, which defines the constraint :

$$g(x) \leqslant 0. \tag{4.3.3}$$

The augmented Lagrangian is defined in this case as : (e.g. cf. Bertsekas (1982)) :

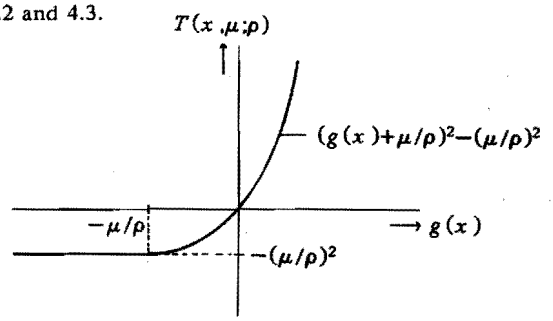
$$M(x,\mu;\rho) := \tilde{f}(x) + \mu\bar{g}(x,\mu;\rho) + \tfrac{1}{2}\rho\bar{g}(x,\mu;\rho)^2, \tag{4.3.4}$$

where : $\bar{g}(x,\mu;\rho) := max\{g(x), -\mu/\rho\}$.

A simple analysis of the penalty term

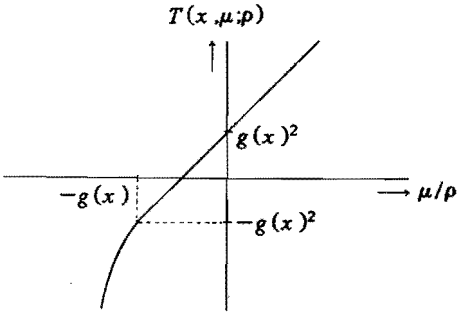
$$T(x,\mu;\rho) := \frac{2\mu}{\rho}\bar{g}(x,\mu;\rho) + \bar{g}(x,\mu;\rho)^2, \tag{4.3.5}$$

yields the Figures 4.2 and 4.3.



$T(x,\mu;\rho$ considered as a function of x for fixed μ.
Figure 4.2

$$T(x,\mu;\rho) = \begin{cases} (g(x)+\mu/\rho)^2 - (\mu/\rho)^2 & g(x) \geqslant -\mu/\rho \\ -(\mu/\rho)^2 & g(x) < -\mu/\rho \end{cases} \tag{4.3.6}$$



$T(x,\mu;\rho$ considered as a function of μ for fixed x.
Figure 4.3

$$T(x,\mu;\rho) = \begin{cases} g(x)^2+2g(x)\mu/\rho & \mu/\rho \geq -g(x) \\ -(\mu/\rho)^2 & \mu/\rho < -g(x) \end{cases} \tag{4.3.7}$$

Figures 4.2 and 4.3 show that $T(x,\mu;\rho)$ is continuously differentiable with respect to both x and μ, whenever $g(x)$ is continuously differentiable with respect to x.

A similar approach to problem (SCOCP) yields the following merit function : †

$$M(x,u,\lambda,\eta_1,\xi,\sigma,\mu;\rho) := h_0(x(0)) + \sigma^T D(x(0)) + \int_0^T (f_0(x,u,t) -$$

$$\lambda^T(\dot{x}-f(x,u,t)) + \sum_{l=1}^{k_1} \eta_{1l}\bar{S}_{1l}(x,u,\eta_{1l},t;\rho) + \sum_{l=1}^{k_2} \eta_{2l}\bar{S}_{2l}(x,\eta_{2l},t;\rho)\,)\,dt +$$

$$\sum_j \sum_{l=1}^{k_2} \nu_{jl}\bar{S}_{2l}(x,\nu_{jl},t_j;\rho) + g_0(x(T),T) + \mu^T E(x(T),T) +$$

$$\tfrac{1}{2}\rho\left\{ \int_0^T (\|\dot{x}-f(x,u,t)\|^2 + \sum_{l=1}^{k_1} \bar{S}_{1l}(x,u,\eta_{1l},t;\rho)^2 + \sum_{l=1}^{k_2} \bar{S}_{2l}(x,\eta_{2l},t;\rho)^2)\,dt + \right.$$

$$\left. \sum_j \sum_{l=1}^{k_2} \bar{S}_{2l}(x,\nu_{jl},t_j;\rho)^2 + \|D(x(0))\|^2 + \|E(x(T),T)\|^2 \right\}. \tag{4.3.8}$$

with :

$$\bar{S}_{1l}(x,u,\eta_{1l},t;\rho) := max\,\{S_{1l}(x,u,t)\,,\,-\eta_{1l}/\rho\}. \tag{4.3.9}$$

$$\bar{S}_{2l}(x,\eta_{2l},t;\rho) := max\,\{S_{2l}(x,t)\,,\,-\eta_{2l}/\rho\}. \tag{4.3.10}$$

We note that the inequality constraints are incorporated in the merit function similar to the finite-dimensional approach, using the smooth penalty terms $T(x,\mu;\rho)$. As a result of this the merit function (4.3.8) is Fréchet differentiable and has therefore essentially the same properties as its finite-dimensional counterpart.

We now consider the actual determination of the step size α_i, which must be calculated such that the merit function is minimized along the direction of search. To this end various strategies may be used. (For a survey on methods for step size determination we refer

† Again we use η_{2l} and ν_{jl} to denote the time derivative and 'jumps' of the multiplier ξ_l (cf. (4.2.1.3) - (4.2.1.4)).

to Gill et al. (1981) and Bertsekas (1982).) We mention :

1) Exact line minimization, i.e.

$$\alpha_i = arg\left[\min_{\alpha>0} M\{\alpha\}\right], \tag{4.3.11}$$

where $\{\alpha\}$ was used to replace $(x^i+\alpha d_x^i,\ u^i+\alpha d_u^i,\ \lambda^i+\alpha(\bar{\lambda}^i-\lambda^i),\ \eta_1^i+\alpha(\bar{\eta}_1^i-\eta_1^i),\ \xi^i+\alpha(\bar{\xi}^i-\xi^i),\ \sigma^i+\alpha(\bar{\sigma}^i-\sigma^i),\ \mu^i+\alpha(\bar{\mu}^i-\mu^i))$.

2) Approximate line minimization. As an example we mention the Armijo step size rule, i.e. given scalars $\beta\in(0,1)$ and $\epsilon\in(0,\frac{1}{2})$ determine the step size α as

$$\alpha=\beta^k$$

where k is the smallest nonnegative integer that satisfies

$$M\{0\} - M\{\beta^k\} \geq -\epsilon\beta^k M'\{0\}(d_x^i, d_u^i, \bar{\lambda}^i-\lambda^i, \bar{\eta}_1^i-\eta_1^i, \bar{\xi}^i-\xi^i, \bar{\sigma}^i-\sigma^i, \bar{\mu}^i-\mu^i). \tag{4.3.12}$$

The choice as to which strategy is followed is not critical for Newton-like methods (exact second derivatives are used), because it is not important that the exact minimum is achieved along the direction of search. When the solution is approached, the step size α_i will converge to one anyway. In a numerical implementation the approximate line minimization tends to be more efficient, because the number of evaluations of the function $M\{\alpha\}$ is less. Therefore the Armijo rule is used in the first stage of the method in the current implementation.

Because in the second stage of the method, the current iterate $(x^i, u^i, \lambda^i, \eta_1^i, \xi^i, \sigma^i, \mu^i)$ is supposed to be 'sufficiently' close to the solution a step size procedure is omitted. The complete method may be summarized as follows :

Algorithm 4.4 :

(0) Δ, *and* (x_0, u_0) *given.*
$i := 0$.

Stage 1 : steps (i) - (vi)

(i) *Calculate first order Lagrange multiplier estimates* $(\lambda^0, \eta_1^0, \xi^0, \sigma^0, \mu^0)$ *as the multipliers corresponding to the solution of problem (EIQP/SCOCP/*Δ*) with the matrices* $M_1=0$, $M_2[t]=I_n$, $M_3[t]=0$, $M_4[t]=I_m$, $M_5=0$, $M_6[t]=0$.

(ii) *Calculate the matrices* M_j *(j=1,2,...,6) corresponding to (4.2.1.11) - (4.2.1.16).*

(iii) *Calculate the Newton direction* (d_x^i, d_u^i) *and second order Lagrange multiplier estimates* $(\bar{\lambda}^i, \bar{\eta}_1^i, \bar{\xi}^i, \bar{\sigma}^i, \bar{\mu}^i)$ *as the solution of problems (EIQP/SCOCP/*Δ*) (using the matrices* M_j *determined in the previous step).*

(iv) *If* $\|(d_x^i, d_u^i)\|_X \leq \epsilon_1$ *then goto (vii).*

(v) *Given scalars* $\beta\in(0,1)$ *and* $\epsilon\in(0,\frac{1}{2})$ *determine the step size* α_i *as*

$$\alpha_i = \beta^k,$$

where k is the smallest nonnegative integer that satisfies

$$M\{0\} - M\{\beta^k\} \geqslant -\epsilon\beta^k M'\{0\}(d_x^i, d_u^i, \bar{\lambda}^i - \lambda^i, \bar{\eta}_1^i - \eta_1^i, \bar{\xi}^i - \xi^i, \bar{\sigma}^i - \sigma^i, \bar{\mu}^i - \mu^i),$$

and set :

$$x^{i+1} := x^i + \alpha_i d_x^i,$$

$$u^{i+1} := u^i + \alpha_i d_u^i,$$

$$\lambda^{i+1} := \lambda^i + \alpha_i(\bar{\lambda}^i - \lambda^i),$$

$$\eta_1^{i+1} := \eta_1^i + \alpha_i(\bar{\eta}_1^i - \eta_1^i),$$

$$\xi^{i+1} := \xi^i + \alpha_i(\bar{\xi}^i - \xi^i),$$

$$\sigma^{i+1} := \sigma^i + \alpha_i(\bar{\sigma}^i - \sigma^i),$$

$$\mu^{i+1} := \mu^i + \alpha_i(\bar{\mu}^i - \mu^i).$$

(vi) $i := i+1$,
goto (ii).

Stage 2 : steps (vii) - (xii)

(vii) *Use* $(x^i, u^i, \lambda^i, \eta_1^i, \xi^i, \sigma^i \mu^i)$ *to determine working sets* W_j *for the constraints* $\tilde{S}_j$.

(viii) *Calculate the matrices* M_j *(j=1,2,...,6) corresponding to (4.2.1.11) - (4.2.1.16).*

(ix) *Calculate the Newton direction* (d_x, d_u) *and second order Lagrange multiplier estimates* $(\bar{\lambda}^i, \bar{\eta}_1^i, \bar{\xi}^i, \bar{\sigma}^i, \bar{\mu}^i)$ *as the solution of problem (EQP/SCOCP). (Using the working sets determined in step (vii) and the matrices* M_j *determined in step (viii).)*

(x) *If* $\|(d_x^i, d_u^i)\|_X \leqslant \epsilon_2$ *then ready.*

(xi) *Set :*

$$x^{i+1} := x^i + d_x^i,$$

$$u^{i+1} := u^i + d_u^i,$$

$$\lambda^{i+1} := \bar{\lambda}^i,$$

$$\eta_1^{i+1} := \bar{\eta}_1^i,$$

$$\xi^{i+1} := \bar{\xi}^i,$$

$$\sigma^{i+1} := \bar{\sigma}^i,$$

$$\mu^{i+1} := \bar{\mu}^i.$$

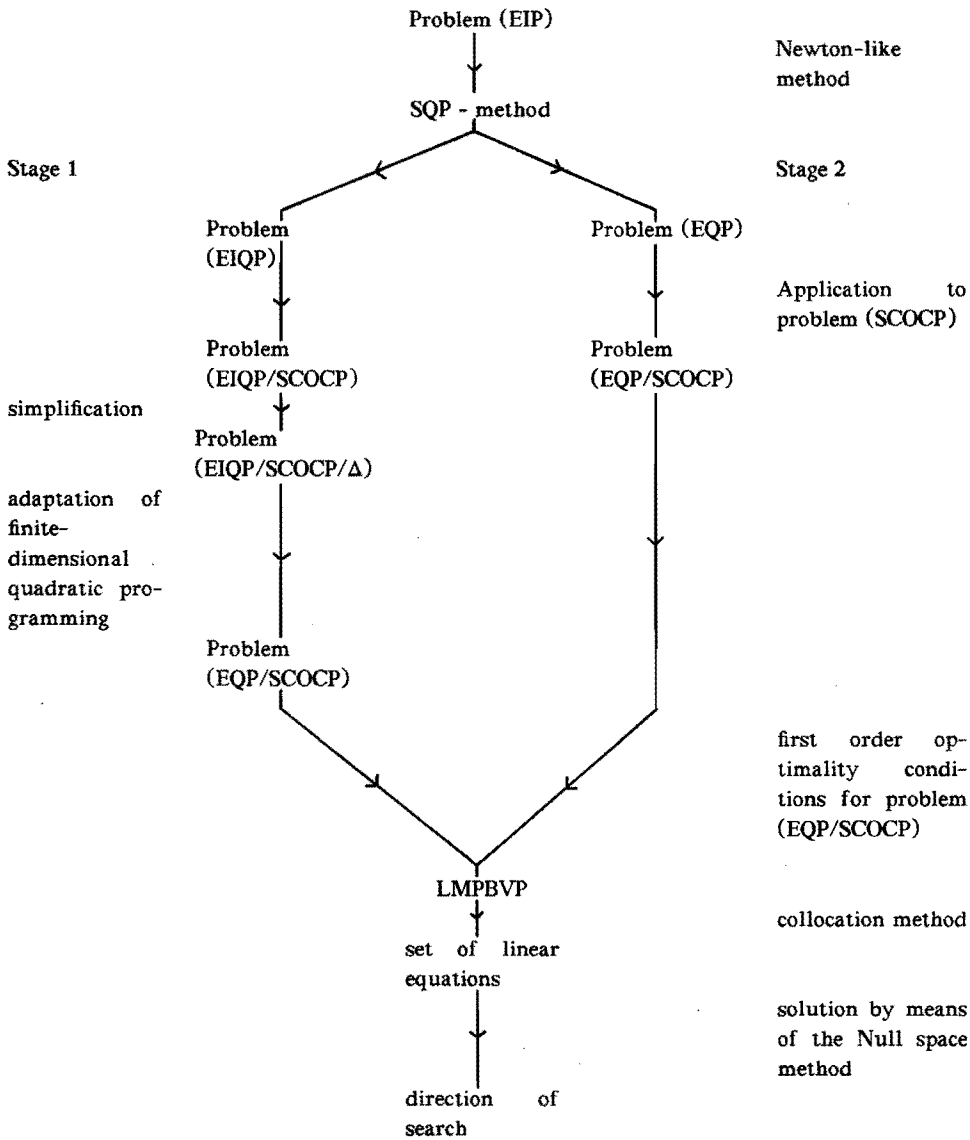
(xii) $i := i+1$,
goto (vii).

4.4. Outline of the implementation of the method.

In this section an outline of the implementation of the method will be given. This outline may serve as a guide for the Chapters 5 and 6, which deal with the most important aspects of the implementation of Algorithm 4.4. In Chapter 5 the solution of the subproblems (EQP/SCOCP) and (EIQP/SCOCP/Δ) and the active set strategy used in the second stage of the algorithm are discussed. Chapter 6 deals with a discussion on the numerical implementation of the method, which essentially comes down to the numerical solution of a linear multipoint boundary value problem.

One of the most important aspects of the method is the calculation of a direction of search. With the SQP-method of Algorithm 4.1 the direction of search is determined either as the solution of problem (EIQP) or as the solution of problem (EQP), which in the application of the method to problem (SCOCP) become problems (EIQP/SCOCP) and (EQP/SCOCP). Because problem (EIQP/SCOCP) cannot be solved easily, the solution process is split up into two stages. In the first stage the structure of the solution is determined, whereas in the second stage the actual solution is determined. The first stage of the solution process requires the solution of problem (EIQP/SCOCP/Δ) which is a simplification of problem (EIQP/SCOCP). Extension of the ideas of finite-dimensional quadratic programming to the solution of problem (EIQP/SCOCP/Δ) requires also the solution of problem (EQP/SCOCP), for the calculation of a direction of search (cf. Section 5.2). Application of the first order optimality conditions to problem (EQP/SCOCP) yields a linear multipoint boundary value problem (LMPBVP) (cf. Section 5.1). The numerical solution of this linear multipoint boundary value problem is done by means of a collocation method (cf. Section 6.1). This collocation method yields a set of linear equations. The numerical solution of the set of equations several methods may be used (cf. Section 6.2). In the current numerical implementation of the method the so-called Null space method is used, which finally yields the direction of search.

In the scheme below the various relations between the problems are summarized.



Scheme for the calculation of the direction of search

5. Solution of the subproblems and determination of the active set.

This chapter deals with three different aspects of the method presented in the previous chapter. In Section 5.1 the solution of the subproblem (EQP/SCOCP) is considered. Section 5.2 deals with a method for the solution of subproblem (EIQP/SCOCP/Δ). This method, which is essentially an adaptation of a common method for the solution of finite-dimensional quadratic programming problems, requires the repeated solution of problem (EQP/SCOCP). The active set strategy which is used in the second stage of the method is described in Section 5.3. The direction of search in this second stage is again determined as the solution of problem (EQP/SCOCP).

5.1. Solution of problem (EQP/SCOCP).

In view of the solution of problem (EQP/SCOCP) this section deals with optimality conditions for optimal control problems with state equality constraints. These conditions do not follow directly from Chapter 3, because there only state inequality constraints were considered. The results contained in this section will show that there is a basic difference between the optimality conditions for optimal control problems with state equality constraints and optimal control problems with state inequality constraints.

For the sake of clarity, we shall first consider optimality conditions for a problem (ESCOCP), which is similar to problem (SCOCP) but contains only state equality constraints. This approach will enable us to make use of most aspects of the formulation of problem (SCOCP) as an abstract nonlinear programming problem in Banach spaces. One may easily verify that problem (EQP/SCOCP) is a special case of problem (ESCOCP).

Problem (ESCOCP): *Determine a control function* $\hat{u} \in L_\infty[0,T]^m$ *and a state trajectory* $\hat{x} \in W_{1,\infty}[0,T]^n$, *which minimize the functional*

$$h_0(x(0)) + \int_0^T f_0(x(t),u(t),t)\,dt + g_0(x(T)), \tag{5.1.1}$$

subject to the constraints :

$$\dot{x}(t) = f(x(t),u(t),t) \qquad a.e.\ \ 0 \leqslant t \leqslant T, \tag{5.1.2}$$

$$D(x(0)) = 0, \tag{5.1.3}$$

$$E(x(T)) = 0, \tag{5.1.4}$$

$$S_{1l}(x(t),u(t),t) = 0 \qquad a.e.\ \ t \in W_l,\ l = 1,2..k_1, \tag{5.1.5}$$

$$S_{2l}(x(t),t) = 0 \qquad t \in W_{k_1+l},\ l = 1,2...k_2, \tag{5.1.6}$$

where : $h_0 : \mathbb{R}^n \to \mathbb{R}$; $f_0 : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R} \to \mathbb{R}^n$; $g_0 : \mathbb{R}^n \to \mathbb{R}$; $D : \mathbb{R}^n \to \mathbb{R}^c$; $E : \mathbb{R}^n \to \mathbb{R}^q$; $f : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R} \to \mathbb{R}^n$; $S_1 : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R} \to \mathbb{R}^{k_1}$; $S_2 : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}^{k_2}$;

$$\text{For all } x \in \mathbb{R}^n, u \in \mathbb{R}^m \text{ rank } S_{1u}(x,u,t) = k_1 \ \ a.e.\ \ 0 \leqslant t \leqslant T. \tag{5.1.7}$$

The functions h_0, f_0, g_0, f, D, E S_1 *and* S_2 *are twice continuously differentiable functions with respect to all arguments.*

The sets W_j *are closed subsets of the interval* $[0,T]$.

5.1.1. Optimality conditions for problem (ESCOCP).

Similar to the approach in Chapter 3, problem (ESCOCP) is considered as a special case of problem (EIP). The difference between the formulations of problem (SCOCP) and (ESCOCP) as special cases of problem (EIP) are the definition of the mapping $\tilde{h}$ and the fact that the constraint $\tilde{g}(x,u) \in B$ is not present at all in the latter case.

We shall first consider two special cases of problem (ESCOCP), the first one being the case of only mixed control state constraints, and the other one being the case of a single state constraint (with order greater than zero).

In the first case the mapping $\tilde{h}$ is defined as :

$$\tilde{h}(x,u) := (\dot{x}(\cdot) - f(x(\cdot),u(\cdot),\cdot), D(x(0)), E(x(T)), S_1(x(\cdot),u(\cdot),\cdot)). \qquad (5.1.1.1)$$

The range space of $\tilde{h}$ is :

$$Z = L_\infty[0,T]^n \times \mathbb{R}^c \times \mathbb{R}^q \times \prod_{l=1}^{k_1} L_\infty(W_l), \qquad (5.1.1.2)$$

with

$$\prod_{l=1}^{k_1} L_\infty(W_l) = L_\infty(W_1) \times L_\infty(W_2) \times \ldots\ldots L_\infty(W_{k_1}). \qquad (5.1.1.3)$$

The spaces $L_\infty(W_l)$ are spaces of measurable and essentially bounded functions on W_l equipped with the norm :

$$\|v\|_{\infty,W_l} := \operatorname*{ess\,sup}_{t \in W_l} \|v(t)\|. \qquad (5.1.1.4)$$

The spaces $L_\infty(W_l)$ are Banach spaces (cf. Kantorovitch et al. (1982)).

The Fréchet differentiability of $\tilde{h}$ follows directly from Lemmas 3.2 and 3.3 and the Fréchet differential is given by :

$$\tilde{h}'(x,u)(\delta x,\delta u) = (\delta\dot{x}(\cdot) - f_x \delta x(\cdot) - f_u \delta u(\cdot), D_x \delta x(0), E_x \delta x(T), S_{1x}\delta x(\cdot) + S_{1u}\delta u(\cdot)). \qquad (5.1.1.5)$$

The hypothesis *rank* $S_{1u} = k_1$ implies

$$R(S_{1x}\delta x(\cdot) + S_{1u}\delta u(\cdot)) = \prod_{l=1}^{k_1} L_\infty(W_l). \qquad (5.1.1.6)$$

Thus for the mapping $\tilde{h}$ defined by (5.1.1.1) the hypotheses of part (i) of Lemma 3.5 hold and hence there exist nontrivial Lagrange multipliers for problem (ESCOCP) with $k_2 = 0$.

Using a derivation similar to the proof of Lemma 3.9 a representation for the linear functional $\langle \hat{\eta}_1, \cdot \rangle$ may be derived as :

$$\langle \hat{\eta}_1, y_1 \rangle = -\sum_{l=1}^{k_1} \int_{W_l} \hat{\eta}_{1l}(t) y_{1l}(t)\, dt \quad \text{for all } y_{1l} \in L_\infty(W_l) \quad l = 1,2..k_1 \qquad (5.1.1.7)$$

with $\hat{\eta}_{1l} \in L_\infty(W_l)$ $l = 1,2..k_1$.

To simplify notation the domain of definition of the multipliers $\hat{\eta}_{1l}$ is extended to the entire interval $[0,T]$ as :

$$\hat{\eta}_{1l}(t) := 0 \quad \text{for all } t \in [0,T] \setminus W_l \quad l = 1,2..k_1, \tag{5.1.1.8}$$

which yields the notation :

$$<\hat{\eta}_1, y_1> = -\int_0^T \hat{\eta}_1(t)^T y_1(t)\, dt \quad \text{for all } y_1 \in L_\infty[0,T]^{k_1}. \tag{5.1.1.9}$$

With a representation of the linear functional $<\hat{\lambda}, \cdot>$ as given by Lemma 3.10 we thus have the following optimality conditions :

Lemma 5.1 : *If $(\hat{x}, \hat{u})$ is a solution to problem (ESCOCP) with $k_2 = 0$, then there exist a real number $\hat{\rho} \geqslant 0$, and vector functions $\hat{\lambda} \in NBV[0,T]^n$, $\hat{\eta}_1 \in L_\infty[0,T]^{k_1}$, and vectors $\hat{\sigma} \in \mathbb{R}^c$, $\hat{\mu} \in \mathbb{R}^q$, not all zero, such that,*

$$\dot{\hat{\lambda}}(t)^T = -H_x[t] - \hat{\eta}_1(t)^T S_{1x}[t] \quad \text{a.e. } 0 \leqslant t \leqslant T, \tag{5.1.1.10}$$

$$\hat{\lambda}(0)^T = -\hat{\rho} h_{0x}[0] - \hat{\sigma}^T D_x[0], \tag{5.1.1.11}$$

$$\hat{\lambda}(T)^T = \hat{\rho} g_{0x}[t] + \hat{\mu}^T E_x[T], \tag{5.1.1.12}$$

$$H_u[t] + \hat{\eta}_1(t)^T S_{1u}[t] = 0 \quad \text{a.e. } 0 \leqslant t \leqslant T, \tag{5.1.1.13}$$

$$\hat{\eta}_{1l}(t) = 0 \quad \text{for all } t \in [0,T] \setminus W_l \quad l = 1,2..k_1. \tag{5.1.1.14}$$

A proof of this lemma is omitted as it is a direct anologue to the proof of Theorem 3.11.

We next turn to the second special case of problem (ESCOCP), i.e. we assume that instead of mixed control state constraints there is (only) one state equality constraint ($k_1 = 0, k_2 = 1$) of the form :

$$S_2(x(t), t) = 0 \qquad t_1 \leqslant t \leqslant t_2, \tag{5.1.1.15}$$

i.e. $W = [t_1, t_2]$, with $0 < t_1 < t_2 < T$.

In a similar treatment, the mapping $\tilde{h}$ would now be defined as :

$$\tilde{h}(x,u) := (\dot{x}(\cdot) - f(x(\cdot), u(\cdot), \cdot), D(x(0)), E(x(T)), S_2(x(\cdot), \cdot)), \tag{5.1.1.16}$$

with

$$Z = L_\infty[0,T]^n \times \mathbb{R}^c \times \mathbb{R}^q \times C[t_1, t_2]. \tag{5.1.1.17}$$

This mapping $\tilde{h}$ is again Fréchet differentiable by Lemmas 3.2 and 3.3. In contrast to the situation considered above the range of the mapping $\tilde{h}'$ is not closed, because the range of

$$S_{2x}(x(\cdot), \cdot)(\delta x(\cdot))$$

is not closed and hence nontrivial Lagrange multipliers need not exist.

We note that this is a consequence of the fact that the range space of the operator is $C[t_1, t_2]$. When the range space would have been chosen to be $W_{1,\infty}[0,T]$ then the range of the operator would have been closed. Unfortunately, this space has no standard representation for the elements of the dual space and hence it is not a simple task to derive optimality conditions via this road.

Instead of the approach suggested by (5.1.1.16), we may replace the state equality constraint (5.1.1.15) by *interior point constraints* of the form :

$$S_2^j(x(t_1,t_1) = 0 \qquad j=0,1,...p-1, \tag{5.1.1.18}$$

and the *mixed control state constraint* :

$$S_2^g(x(t),u(t),t) = 0 \qquad a.e. \quad t_1 \leqslant t \leqslant t_2, \tag{5.1.1.19}$$

where p is the order of the state constraint S_2 and the functions S_2^j are defined by (3.3.5.7) - (3.3.5.8).

The mapping $\tilde{h}$ becomes :

$$\begin{aligned}\tilde{h}(x,u) := \; & (\dot{x}(\cdot) - f(x(\cdot),u(\cdot),\cdot), D(x(0)), E(x(T)), \\ & N(x(t_1),t_1), S_2^g(x(\cdot),u(\cdot),\cdot)),\end{aligned} \tag{5.1.1.20}$$

where

$$N(x,t) := \begin{vmatrix} S_2(x,t) \\ S_2^1(x,t) \\ \vdots \\ S_2^{g-1}(x,t) \end{vmatrix}. \tag{5.1.1.21}$$

with range space :

$$Z = L_\infty[0,T]\times\mathbb{R}^c\times\mathbb{R}^q\times\mathbb{R}^p\times L_\infty[t_1,t_2]. \tag{5.1.1.22}$$

The regularity of $\tilde{h}$ follows from the lemma below.

Lemma 5.2 : *Let the functions f,D,E and S_2 satisfy the assumptions of problem (ESCOCP) with $k_1=0$ and $k_2=1$ and let the functions f and S_2 be p-times differentiable with respect to all arguments. Let the mapping $\tilde{h}$ be defined by (5.1.1.20) - (5.1.1.22). Assume that*

$$S_{2u}^g(\hat{x}(t),\hat{u}(t),t) \neq 0 \qquad a.e. \text{ on } [t_1,t_2], \tag{5.1.1.23}$$

then

$$R(\tilde{h}'(\hat{x},\hat{u})) = \text{closed}.$$

Furthermore if, in addition, at $(\hat{x},\hat{u})$

$$\begin{aligned} &\text{rank } D_x(\hat{x}(0)) = c \\ &\text{rank } E_x(\hat{x}(T)) = q \\ &\text{rank } N_x(\hat{x}(t_1),t_1)) = p \end{aligned}$$

then

$$R(\tilde{h}'(\hat{x},\hat{u})) = Z. \tag{5.1.1.24}$$

The proof follows from the same arguments as the proof of Lemma 3.5. Condition (5.1.1.23) is used to establish

$$R(S_{2x}^g\delta x(\cdot) + S_{2u}^g\delta u(\cdot)) = L_\infty[t_1,t_2]. \tag{5.1.1.25}$$

Using an approach similar to Subsections 3.3.2 and 3.3.3 we obtain the following optimality conditions :

Lemma 5.3 : *If* $(\hat{x},\hat{u})$ *is a solution to problem (ESCOCP), with* $k_1=0$, $k_2=1$, $W=[t_1,t_2]$ *and if the functions* S_2 *and* f *are p-times differentiable with respect to all arguments and*

$$S^g_{2u}(\hat{x}(t),\hat{u}(t),t) \neq 0 \qquad a.e. \quad t_1 \leqslant t \leqslant t_2, \tag{5.1.1.26}$$

then, there exist a real number $\hat{\rho} \geqslant 0$, *and functions* $\hat{\lambda} \in NBV[0,T]^n$, $\hat{\gamma} \in L_\infty[0,T]$ *and vectors* $\hat{\sigma} \in \mathbb{R}^c$, $\hat{\mu} \in \mathbb{R}^q$ *and numbers* $\hat{\beta}^j$ $(j=1,...p)$, *not all zero, such that*

$$\hat{\lambda}(t)^T = -H_x[t] - \hat{\gamma}(t)S^g_{2x}[t] \qquad a.e. \quad 0 \leqslant t \leqslant T, \tag{5.1.1.27}$$

$$\hat{\lambda}(0)^T = -\hat{\rho}h_{0x}[0] - \hat{\sigma}^T D_x[0], \tag{5.1.1.28}$$

$$\hat{\lambda}(T)^T = \hat{\rho}g_{0x}[T] + \hat{\mu}^T E_x[T], \tag{5.1.1.29}$$

$$H_u[t] + \hat{\gamma}(t)S^g_{2u}[t] = 0 \qquad a.e. \quad 0 \leqslant t \leqslant T, \tag{5.1.1.30}$$

$$\hat{\lambda}(t_1+)^T = \hat{\lambda}(t_1-)^T - \sum_{j=1}^{p} \hat{\beta}^j S^{j-1}_{2x}[t_1], \tag{5.1.1.31}$$

$$\hat{\gamma}(t) = 0 \qquad for\ all\ 0 \leqslant t < t_1 \ \text{ and } \ t_2 < t \leqslant T. \tag{5.1.1.32}$$

Because the approach of replacing (5.1.1.15) by (5.1.1.18) - (5.1.1.19) is quite similar to the approach of Bryson et al. (1963), it is not surprising that the optimality conditions of Lemma 5.3 are quite similar to the results contained in Theorem 3.16 for the case $i=p$ The difference are the relations that, by definition, the multipliers $\hat{\eta}^j$ must satisfy, i.e. (3.3.6.5) and (3.3.6.28). In the present case, the multiplier $\hat{\gamma}$ need not satisfy these relations. (Obviously, if t_1 and t_2 are chosen to coincide with the true entry- and exit points of the inequality constrained problem, we shall have $\hat{\eta}^p(t)=\hat{\gamma}(t)$).

Up to this point, it is still not clear why the approach using (5.1.1.16) was not feasible. To investigate this we consider the Lagrangian :

$$\begin{aligned} L := \rho(h_0(x(0)) + \int_0^T f_0(x(t),u(t),t)\,dt + g_0(x(T))) - \\ \int_0^T \lambda(t)^T(\dot{x}(t) - f(x(t),u(t),t))\,dt + \sigma^T D(x(0)) + \mu^T E(x(T)) + \\ \int_0^T \gamma(t)S^g_2(x(t),u(t),t)\,dt + \sum_{j=1}^{p} \beta^j S^{j-1}_2(x(t_1),t_1), \end{aligned} \tag{5.1.1.33}$$

which has a stationary point at $(\hat{x},\hat{u},\hat{\lambda},\hat{\sigma},\hat{\mu},\hat{\gamma},\hat{\beta}^j)$. Assuming that the multiplier $\hat{\gamma}$ is sufficiently smooth, we consider the integration by parts of the term :

$$A := \int_0^T \hat{\gamma}(t)S^g_2[t]\,dt + \sum_{j=1}^{p} \hat{\beta}^j S^{j-1}_2[t_1], \tag{5.1.1.34}$$

which yields :

$$A = \int_{t_1+}^{t_2-} \hat{\gamma}(t)\,dS^{g-1}_2[t] + \sum_{j=1}^{p} \hat{\beta}^j S^{j-1}_2[t_1], \tag{5.1.1.35}$$

$$A = \int_{t_1+}^{t_2-} (-1)\hat{\gamma}(t)S_2^{g-1}[t]\,dt + \hat{\gamma}(t_2-)S_2^{g-1}[t_2] - \hat{\gamma}(t_1+)S_2^{g-1}[t_1] + \sum_{j=1}^{p} \hat{\beta}^j S_2^{j-1}[t_1]. \qquad (5.1.1.36)$$

Continuing this integration by parts we obtain after p times :

$$A = \int_{t_1+}^{t_2-} \hat{\eta}_0(t)S_2[t]\,dt + \sum_{j=1}^{p} \left[\hat{\nu}_1^{j-1}S_2^{j-1}[t_1] + \hat{\nu}_2^{j-1}S_2^{j-1}[t_2]\right], \qquad (5.1.1.37)$$

with :

$$\hat{\nu}_1^{j-1} := \hat{\beta}^j + (-1)^{p-j}\frac{d^{p-j+1}\hat{\gamma}}{dt^{p-j+1}}(t_1+) \qquad j=1,....p, \qquad (5.1.1.38)$$

$$\hat{\nu}_2^{j-1} := -(-1)^{p-j}\frac{d^{p-j+1}\hat{\gamma}}{dt^{p-j+1}}(t_2-) \qquad j=1,....p, \qquad (5.1.1.39)$$

$$\hat{\eta}_0(t) := (-1)^p\frac{d^p\hat{\gamma}}{dt^p}(t) \qquad t_1+ \leqslant t \leqslant t_2-. \qquad (5.1.1.40)$$

And hence, at the optimal point $(\hat{x},\hat{u},\hat{\lambda},\hat{\sigma},\hat{\mu},\hat{\gamma},\hat{\beta}^j)$ the Lagrangian may be expressed as :

$$L = \hat{\rho}\left[h_0[0] + \int_0^T f_0[t]\,dt + g_0[T]\right] - \int_0^T \hat{\lambda}(t)^T(\dot{\hat{x}} - f[t])\,dt + \hat{\sigma}^T D[0] + \hat{\mu}^T E[T] + \int_0^T \hat{\eta}_0 S_2[t]\,dt + \sum_{j=1}^{p}(\hat{\nu}_1^{j-1}S_2^{j-1}[t_1] + \hat{\nu}_2^{j-1}S_2^{j-1}[t_2]). \qquad (5.1.1.41)$$

We observe that expression (5.1.1.41) is in fact the Lagrangian belonging to the abstract formulation of the problem based on (5.1.1.16) augmented with entry- and exit point constraints of the form :

$$\begin{aligned} S_2^j(x(t_1),t_1) &= 0 \qquad j=0,1,....,p-1 \\ S_2^j(x(t_2),t_2) &= 0 \qquad j=0,1,....,p-1 \end{aligned} \qquad (5.1.1.42)$$

This reveals that the approach following (5.1.1.16) was not feasible because the state equality constraints require in general, additional entry and exit point constraints of the form (5.1.1.42). When the entry- and exit point are such that they coincide with the entry- and exit point of the corresponding inequality constrained problem, then these constraints are no longer necessary and hence the multipliers $\hat{\nu}_1^j$ and $\hat{\nu}_2^j$ $(j=1,...p-1)$ will automatically be zero. We note however, that the inclusion of the constraints (5.1.1.42) in the formulation of problem (ESCOCP) would still leave the question about the closedness of the range of the operator $\tilde{h}'$ open, with the approach following (5.1.1.16).

The formulation of the optimality conditions of Lemma 5.3 will be used for the solution of problem (EQP/SCOCP), whereas the alternative formulation of the Lagrangian (5.1.1.41) will be used to derive an active set strategy for problem (EIQP/SCOCP/Δ).

Extension of the previous results to the general case of problem (ESCOCP) is straightforward. We note that to derive a representation for the linear functionals $<\hat{\eta}_1,\cdot>$ and $<\hat{\gamma},\cdot>$ the matrix of the partial derivatives of the active mixed control state constraints

with respect to u, consisting of rows of the matrices S_{1u} and S^{p}_{2u} is required to be of full row rank.

5.1.2. Optimality conditions for problem (EQP/SCOCP).

In this section optimality conditions for problem (EQP/SCOCP) are considered. Because problem (EQP/SCOCP) is a special case of problem (ESCOCP) these conditions will follow from the previous section. However for problem (ESCOCP) the optimality conditions involve the functions S^{j}_{2} as defined by (3.3.5.7) - (3.3.5.8).
To apply the optimality conditions of problem (ESCOCP) to problem (EQP/SCOCP) the counterpart to the functions S^{j}_{2} must be determined for problem (EQP/SCOCP).

The state constraints of problem (EQP/SCOCP) are considered individually and are denoted by :

$$\tilde{T}_l(d_x,t) := S_{2l}(x^i(t),t) + S_{2lx}(x^i(t),t)d_x \qquad l=1,2,...k_2. \tag{5.1.2.1}$$

To the notation $S_{2l}(x^i(t),t)$ we note that this function is considered to be a function of time only, in contrast to the notation $S_{2l}(x,t)$ where S_{2l} is considered as a function of x and t.

The partial derivative of (5.1.2.1) with respect to the argument t becomes :

$$\frac{\partial \tilde{T}_l(d_x,t)}{\partial t} = S_{2lx}(x^i(t),t)\dot{x}^i(t) + S_{2lt}(x^i(t),t) + \dot{x}^i(t)^T S_{2lxx}(x^i(t),t)d_x + S_{2lxt}(x^i(t),t)d_x. \tag{5.1.2.2}$$

In the formulation of problem (EQP/SCOCP) Definitions (3.3.5.7) - (3.3.5.8) become :

$$\tilde{T}^j_l := \begin{cases} \tilde{T}_l & j=0 \\ \dfrac{\partial \tilde{T}^{j-1}_l}{\partial t} + \dfrac{\partial \tilde{T}^{j-1}_l}{\partial x}(f_x d_x + f_u d_u + f - \dot{x}^i) & j=1,....p_l \end{cases} \tag{5.1.2.3}$$

where p_l is the order of the state constraint (5.1.2.1).

Lemma 5.4 : *Let the functions S^{j}_{2l} be defined by (3.3.5.7) - (3.3.5.8) and let the functions $\tilde{T}_l$ be defined by (5.1.2.1). If*

$$S^{j}_{2lxx} = 0 \qquad \text{for all } j=0,1,...p_l-1 \quad l=1,.....k_2, \tag{5.1.2.4}$$

then the functions defined by (5.1.2.3) satisfy :

$$\tilde{T}^j_l = \begin{cases} S^{j}_{2l}(x^i(t),t) + S^{j}_{2lx}(x^i(t),t)d_x & j=0,1.....p_l-1 \quad l=1,2,...k_2 \\ S^{p_l}_{2l}(x^i(t),u^i(t),t) + S^{p_l}_{2lx}(x^i(t),u^i(t),t)d_x & \\ + S^{p_l}_{2lu}(x^i(t),u^i(t),t)d_u & j=p_l \quad l=1,2....k_2 \end{cases} \tag{5.1.2.5}$$

Proof : (5.1.2.5) is proved by induction. For $j=0$ equation (5.1.2.5) is true by Definitions (5.1.2.1) and (5.1.2.3).
Now suppose (5.1.2.5) holds for some j. with $0 \leqslant j < p_l$. By definition

$$\tilde{T}_l^{j+1} = \frac{\partial \tilde{T}_l^j}{\partial t} + \frac{\partial \tilde{T}_l^j}{\partial x}(f_x d_x + f_u d_u + f - \dot{x}^i), \tag{5.1.2.6}$$

using (5.1.2.5) we obtain

$$\frac{\partial \tilde{T}_l^j}{\partial t} = S_{2lx}^j \dot{x}^i(t) + S_{2lt}^j + \dot{x}^i(t)^T S_{2lxx}^j d_x + S_{2lxt}^j d_x \tag{5.1.2.7}$$

and

$$\frac{\partial \tilde{T}_l^j}{\partial d_x} = S_{2lx}^j. \tag{5.1.2.8}$$

Combination of (5.1.2.6), (5.1.2.7) and (5.1.2.8) gives

$$\tilde{T}_l^{j+1} = S_{2lx}^j \dot{x}^i(t) + S_{2lt}^j + \dot{x}^i(t)^T S_{2lxx}^j d_x + S_{2lxt}^j d_x + S_{2lx}^j f_x d_x + S_{2lx}^j f_u d_u + S_{2lx}^j (f - \dot{x}^i(t)). \tag{5.1.2.9}$$

We now use the special structure of the functions S_{2l}^j, induced by the Definition (3.3.5.8), i.e.

$$S_{2l}^{j+1} = S_{2lt}^j + S_{2lx}^j f \tag{5.1.2.10}$$

and hence

$$S_{2lx}^{j+1} = S_{2lxt}^j + S_{2lxx}^j f + S_{2lx}^j f_x, \tag{5.1.2.11}$$

$$S_{2lu}^{j+1} = S_{2lx}^j f_u \tag{5.1.2.12}$$

(We note that for (5.1.2.12) use is made of the fact that $j < p_l$.)
Substitution of (5.1.2.10) - (5.1.2.12) in (5.1.2.9) yields :

$$\tilde{T}_l^{j+1} = S_{2l}^{j+1} + (\dot{x}^i(t) - f)^T S_{2xx}^j d_x + S_{2lx}^{j+1} d_x + S_{2lu}^{j+1} d_u. \tag{5.1.2.13}$$

By definition, if $j < p_l - 1$, the term S_{2lu}^{j+1} is zero.
To make the induction step complete use is made of the hypothesis (5.1.2.4).
□

Lemma 5.4 provides quite a simple expression for the functions $\tilde{T}_l^j$ which, along a trajectory (d_x, d_u) of (4.2.1.22) may be considered as the time derivative of this constraint. The hypotheses of Lemma 5.4 state however, that the state constraints S_2 must be linear functions in the variable x. In practice this is quite a heavy assumption. Fortunately, it is possible to tranform any problem (SCOCP) which does not satisfy (5.1.2.4) in a way such that, for the transformed problem condition (5.1.2.4) will hold, i.e. such that the transformed problem has only linear state constraints. This transformation is outlined in Appendix B.

In the sequel we shall always assume that condition (5.1.2.4) is satisfied, because it gives the simple expressions of the functions $\tilde{T}_{2l}^j$. As a consequence of this the matrix M_6, in the object functions of problems (EQP/SCOCP) and (EIQP/SCOCP) will be zero (cf. (4.2.1.16)).

In the general case of problem (EQP/SCOCP) the regularity conditions (5.1.7) and (5.1.1.23) require some modification. This is due to the fact that, in the formulation of problem (EQP/SCOCP), it is allowed that boundary arcs of various constraints coincide or overlap.

Using a notation similar to (4.2.1.19) - (4.2.1.20) the $\bar{k}(t)$-vector is defined as :

$$R_l^p[t] := \tilde{S}^p_{i_l} \qquad l = 1,2,...\bar{k}(t), \qquad 0 \leqslant t \leqslant T, \tag{5.1.2.14}$$

where $\tilde{S}^p$ is defined by (3.3.5.11) and the indices of the active constraints i_l are elements of the index set $I(t)$.

A straightforward generalization of (5.1.7) and (5.1.1.23) is that the rank of the matrix $R_u^p[t]$ must be $\bar{k}(t)$. This is a consequence of the fact that in the approach of Section 5.1.1 the state constraints are transformed into the mixed control state constraints

$$R^p[t] + R_x^p[t]\dot{d}_x(t) + R_u^p[t]\dot{d}_u(t) = 0 \qquad a.e. \quad 0 \leqslant t \leqslant T. \tag{5.1.2.15}$$

We are now ready to state the optimality conditions for problem (EQP/SCOCP) which follow directly as a generalization of the results contained in Section 5.1.1.

Theorem 5.5 : *Let $(\bar{d}_x, \bar{d}_u)$ be a solution to problem (EQP/SCOCP) and assume*

$$rank\ R_u^p[t] = \bar{k}(t) \qquad a.e. \quad 0 \leqslant t \leqslant T, \tag{5.1.2.16}$$

and

$$S^j_{2lxx}[t] = 0 \qquad for\ all\ \ j = 0,1,...p_l - 1, \quad l = 1,2,...k_2, \tag{5.1.2.17}$$

then there exist a real number $\bar{\rho} \geqslant 0$, and vector functions $\bar{\lambda} \in NBV[0,T]^n$, $\bar{\eta} \in L_\infty[0,T]^{k_1+k_2}$ and vectors $\bar{\sigma} \in \mathbb{R}^c$, $\bar{\mu} \in \mathbb{R}^q$, and numbers $\bar{\beta}^k_{lj}$, $\bar{v}_{lj}$, not all zero, such that,

$$\dot{\bar{\lambda}}(t)^T = -\bar{\lambda}(t)^T f_x[t] - \bar{\eta}(t)^T \tilde{S}^p_x[t] - \bar{\rho} f_{0x}[t]$$
$$- \bar{\rho}\dot{\bar{d}}_x(t)^T M_2[t] - \bar{\rho}\dot{\bar{d}}_u(t)^T M_3[t] \qquad a.e. \quad 0 \leqslant t \leqslant T, \tag{5.1.2.18}$$

$$\bar{\lambda}(0)^T = -\bar{\rho} h_{0x}[0] - \bar{\sigma}^T D_x[0] - \bar{\rho}\dot{\bar{d}}_x(0)^T M_1, \tag{5.1.2.19}$$

$$\bar{\lambda}(T)^T = \bar{\rho} g_{0x}[T] + \bar{\mu}^T E_x[T] + \bar{\rho}\dot{\bar{d}}_x(T)^T M_5, \tag{5.1.2.20}$$

$$\bar{\lambda}(t)^T f_u[t] + \bar{\eta}(t)^T \tilde{S}^p_u[t] + \bar{\rho} f_{0u}[t] +$$
$$\bar{\rho}\dot{\bar{d}}_x(t)^T M_3[t] + \bar{\rho}\dot{\bar{d}}_u(t)^T M_4[t] = 0 \quad a.e. \quad 0 \leqslant t \leqslant T, \tag{5.1.2.21}$$

$$\bar{\eta}_i(t) = 0 \quad \text{if } l \notin I(t) \quad 0 \leqslant t \leqslant T. \tag{5.1.2.22}$$

At an entry point t^l_{2j-1} of the state constraint $\tilde{T}_l$ the multiplier $\bar{\lambda}$ satisfies :

$$\bar{\lambda}(t^l_{2j-1}+) = \bar{\lambda}(t^l_{2j-1}-) - \sum_{k=1}^{p_l} \bar{\beta}^k_{lj} S^k_{2lx}{}^{-1}[t^l_{2j-1}]. \tag{5.1.2.23}$$

At a contact point $t^l_{2m_l^p+j}$ of the state constraint $\tilde{T}_l$ the multiplier $\bar{\lambda}$ satisfies :

$$\bar{\lambda}(t^l_{2m_l^p+j}+) = \bar{\lambda}(t^l_{2m_l^p+j}-) - \bar{v}_{lj} S_{2lx}[t^l_{2m_l^p+j}]. \tag{5.1.2.24}$$

We remind the reader to Definition (3.3.5.11) of $\tilde{S}^p$ and that in the formulation of problem (EQP/SCOCP) the notation $[t]$ is used to replace argument lists involving $x^i(t)$, $u^i(t)$, $\lambda^i(t)$, etc.

With respect to Theorem 5.5 we note that it does not explicitly include the case of coinciding entry - and contact points. In these cases however, the jump conditions (5.1.2.23) and (5.1.2.24) are generalized in a straightforward manner.

5.1.3. Linear multipoint boundary value problem for the solution of problem (EQP/SCOCP).

When it is assumed that problem (EQP/SCOCP) has a solution for which the regularity constant $\bar{\rho}$ may be set nonzero, then, under certain hypotheses, the solution of problem (EQP/SCOCP) can be obtained as the solution of a linear multipoint boundary value problem.

Theorem 5.6 : *If problem (EQP/SCOCP) has a solution for which the regularity constant $\bar{\rho}$ may be set nonzero, and*

$$S^l_{2lxx}[t] = 0 \quad \text{for all } j = 1,2,\ldots,p_l-1,\ l = 1,2,\ldots,k_2,\ 0 \leqslant t \leqslant T, \tag{5.1.3.1}$$

and

$$\operatorname{rank} \begin{bmatrix} M_4[t] & R^p_u[t]^T \\ R^p_u[t] & 0 \end{bmatrix} = m + \bar{k}(t) \quad \text{a.e. } 0 \leqslant t \leqslant T, \tag{5.1.3.2}$$

then the solution of problem (EQP/SCOCP) can be obtained as the solution of the following set of equations :

$$\begin{bmatrix} \dot{\bar{d}}_x(t) \\ \dot{\bar{\lambda}}(t) \end{bmatrix} = \begin{bmatrix} f_x[t] & 0 \\ -M_2[t] & -f_x[t]^T \end{bmatrix} \begin{bmatrix} \bar{d}_x(t) \\ \bar{\lambda}(t) \end{bmatrix} + \begin{bmatrix} f_u[t] & 0 \\ -M_3[t]^T & -\tilde{S}^p_x[t]^T \end{bmatrix} \begin{bmatrix} \bar{d}_u(t) \\ \bar{\eta}(t) \end{bmatrix} + \begin{bmatrix} f[t] - \dot{x}^i(t) \\ -f_{0x}[t]^T \end{bmatrix} \quad \text{a.e. } 0 \leqslant t \leqslant T, \tag{5.1.3.3}$$

$$\begin{bmatrix} R^p_x[t] & 0 \\ M_3[t]^T & f_u[t]^T \end{bmatrix} \begin{bmatrix} \bar{d}_x(t) \\ \bar{\lambda}(t) \end{bmatrix} + \begin{bmatrix} R^p_u[t] & 0 \\ M_4[t] & \tilde{S}^p_u[t]^T \end{bmatrix} \begin{bmatrix} \bar{d}_u(t) \\ \bar{\eta}(t) \end{bmatrix} = - \begin{bmatrix} R^p[t] \\ f_{0u}[t]^T \end{bmatrix} \quad \text{a.e. } 0 \leqslant t \leqslant T, \tag{5.1.3.4}$$

$$\bar{\eta}_l(t) = 0 \qquad \text{if } l \notin I(t) \quad 0 \leqslant t \leqslant T, \tag{5.1.3.5}$$

$$\begin{bmatrix} D_x[0] & 0 \\ M_1 & I \end{bmatrix} \begin{bmatrix} \bar{d}_x(0) \\ \bar{\lambda}(0) \end{bmatrix} + \begin{bmatrix} 0 \\ D_x[0]^T \end{bmatrix} \bar{\sigma} = - \begin{bmatrix} D[0] \\ h_{0x}[0]^T \end{bmatrix}, \tag{5.1.3.6}$$

$$\begin{bmatrix} E_x[T] & 0 \\ M_5 & -I \end{bmatrix} \begin{bmatrix} \bar{d}_x(T) \\ \bar{\lambda}(T) \end{bmatrix} + \begin{bmatrix} 0 \\ E_x[T]^T \end{bmatrix} \bar{\mu} = - \begin{bmatrix} E[T] \\ g_{0x}[T] \end{bmatrix}, \tag{5.1.3.7}$$

$$\bar{\lambda}(t^l_{2j-1}+) = \bar{\lambda}(t^l_{2j-1}-) - \sum_{k=1}^{p_l} \bar{\beta}^k_{lj} S^k_{2lx}{}^{-1}[t^l_{2j-1}]^T \qquad j = 1,2,\ldots,m^b_l \quad l = 1,2,\ldots,k_2, \tag{5.1.3.8}$$

$$S^k_{2l}[t^l_{2j-1}] + S^k_{2lx}[t^l_{2j-1}]\bar{d}_x(t^l_{2j-1}) = 0$$

$$k = 0,1,\ldots,p_l-1 \quad j = 1,2,\ldots,m^b_l \quad l = 1,2,\ldots,k_2, \tag{5.1.3.9}$$

$$\bar{\lambda}(t^l_{2m^b_l+j}+) = \bar{\lambda}(t^l_{2m^b_l+j}-) - \bar{v}_{lj} S_{2lx}[t^l_{2m^b_l+j}]^T \qquad j = 1,2,\ldots,m^c_l \quad l = 1,2,\ldots,k_2, \tag{5.1.3.10}$$

$$S_{2l}[t^l_{2m^b_l+j}] + S_{2lx}[t^l_{2m^b_l+j}]\bar{d}_x(t^l_{2m^b_l+j}) = 0 \qquad j = 1,2,\ldots,m^c_l \quad l = 1,2,\ldots,k_2. \tag{5.1.3.11}$$

The theorem follows directly from the combination of the constraints of problem (EQP/SCOCP) and the optimality conditions of Theorem 5.5. The system of equations of

Theorem 5.6 becomes a standard linear multipoint boundary value problem, when (5.1.3.4) and (5.1.3.5) are used to eliminate the control $\bar{d}_u$ and the multiplier $\bar{\eta}$ from (5.1.3.3). This is possible as a result of assumption (5.1.3.2) and hence $\bar{d}_u$ and $\bar{\eta}$ satisfy an equation of the form

$$\begin{bmatrix} \bar{d}_u(t) \\ \bar{\eta}(t) \end{bmatrix} = A(t) \begin{bmatrix} \bar{d}_x(t) \\ \bar{\lambda}(t) \end{bmatrix} + b(t) \qquad a.e. \quad 0 \leqslant t \leqslant T. \tag{5.1.3.12}$$

Equations (5.1.3.6) and (5.1.3.7) constitute boundary equations for the differential equation (5.1.3.1) (combined with (5.1.3.12)), whereas (5.1.3.8) and (5.1.3.9) constitute interior point conditions.

5.2. Solution of problem (EIQP/SCOCP/Δ).

This section deals with a method for the solution of problem (EIQP/SCOCP/Δ). The main problem we are faced with is the determination of the active set of constraints, because once this set is known, the solution of problem (EIQP/SCOCP/Δ) may be obtained as the solution of problem (EQP/SCOCP). Problem (EQP/SCOCP) may be solved via the solution of the linear multipoint boundary value problem discussed in Section 5.1.3. For simplicity we shall assume, throughout this section, that problem (EIQP/SCOCP/Δ) has a unique solution.

The method for the solution of problem (EIQP/SCOCP/Δ) that is proposed in this section, is an adaptation of a well known method for the solution of finite-dimensional quadratic programming problems, which has the following characteristics (cf. Appendix A) :

1) The method has an iterative nature, using as candidates for the solution, solutions to quadratic programming problems with only equality constraints.
2) The iterates are all *feasible points*, i.e. the complete set of inequality constraints of the quadratic programming problem are satisfied during each iteration.
3) The active set strategy consists of the addition of constraints to the working set whenever the step size α_i is restricted, or the (possible) deletion of constraints from the working set, whenever the direction of search becomes zero and one or more Lagrange multipliers have a wrong sign.

Essentially each iteration of the method consists of the following three steps :

(i) Calculation of a direction of search.

(ii) Calculation of a step size.

(iii) Updating the working set (active set strategy).

One iteration of the method for the solution of problem (EIQP/SCOCP/Δ) consists essentially of the same steps (i), (ii) and (iii). The adaptation of these steps will be considered individually.

In steps (i) and (ii) the working set, i.e. the current estimate for the active set of constraints in a solution point of problem (EIQP/SCOCP/Δ), is kept fixed and given a working set :

$$W := W_1 \times W_2 \times \ldots\ldots \times W_{k_1+k_2},$$

a solution of problem (EQP/SCOCP), denoted $(\bar{d}_x^i, \bar{d}_u^i)$, is a (new) candidate for the solution of problem (EIQP/SCOCP/Δ). This is because the definition of problems

(EQP/SCOCP) and (EIQP/SCOCP/Δ) show that when the working set of problem (EQP/SCOCP) is the active set of problem (EIQP/SCOCP/Δ) then the solutions of both problems are the same. Hence an obvious choice for the direction of search in the ith iteration, denoted $(\Delta d_x^i, \Delta d_u^i)$ is :

$$\Delta d_x^i := \bar{d}_x^i - d_x^i, \tag{5.2.1}$$

$$\Delta d_u^i := \bar{d}_u^i - d_u^i, \tag{5.2.2}$$

where (d_x^i, d_u^i) denotes the iterate in the ith iteration. (We note that this choice is entirely analoguous to the finite-dimensional case).

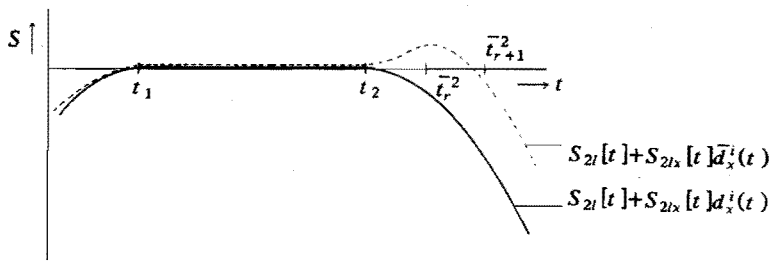
Now the determination of the step size α_i is considered, i.e.

$$d_x^{i+1} := d_x^i + \alpha_i \Delta d_x^i, \tag{5.2.3}$$

$$d_u^{i+1} := d_u^i + \alpha_i \Delta d_u^i. \tag{5.2.4}$$

In the finite-dimensional case the step size α_i is chosen so that the objective function is minimized along the direction of search subject to the restriction that (d_x^{i+1}, d_u^{i+1}) must be a feasible point of the constraints of the problem.

We shall show that in the case of problem (EIQP/SCOCP/Δ) such a choice is not always possible (cf. Figure 5.1).



Feasible point and infeasible direction of search.
Figure 5.1

The case considered is of a state constraint which has a working set $W_l = [t_1, t_2]$. The solution of problem (EQP/SCOCP), i.e. $\bar{d}_x^i$, is not a feasible point of the state constraint, because

$$S_{2l}[\bar{t}_r^2] + S_{2lx}[\bar{t}_r^2]\bar{d}_x^i(\bar{t}_r^2) > 0 \tag{5.2.5}$$

For the values $\alpha \in [0, \bar{\alpha})$, with

$$\bar{\alpha} := \frac{S_{2l}[\bar{t}_r^2] + S_{2lx}[\bar{t}_r^2]d_x^i(\bar{t}_r^2)}{S_{2lx}[\bar{t}_r^2]\Delta d_x^i(\bar{t}_r^2)}, \tag{5.2.6}$$

the objective function is as a function of α decreasing and the points

$$d_x(t;\alpha) := d_x^i(t) + \alpha \Delta d_x^i(t), \tag{5.2.7}$$

$$d_u(t;\alpha) := d_u^i(t) + \alpha \Delta d_u^i(t), \tag{5.2.8}$$

are feasible, because for $\alpha \in [0, \bar{\alpha})$

$$S_{2l}[\bar{t}_r^2] + S_{2lx}[\bar{t}_r^2]d_x(\bar{t}_r^2;\alpha) < 0. \qquad (5.2.9)$$

However, the point $(d_x(t;\bar{\alpha}),d_u(t;\bar{\alpha}))$ is not feasible, because

$$S_{2l}[t] + S_{2lx}[t]d_x(t;\bar{\alpha}) \neq 0 \quad \text{for all } t_2 < t < \bar{t}_r^2. \qquad (5.2.10)$$

In spite of this fact, we still would like to choose the step size $\alpha_i := \bar{\alpha}$, because the objective function is as a function of α decreasing on $[0,\bar{\alpha})$ and $(d_x(t;\bar{\alpha}),d_u(t;\bar{\alpha}))$ is 'almost' feasible. We now define :

Definition 5.7 : *A pair of functions $d_x \in W_{1,\infty}[0,T]^n$ and $d_u \in PC[0,T]^m$ are called Δ-feasible with respect to the constraints of problem (EIQP/SCOCP/Δ) if they satisfy :*

$$\dot{d}_x = f_x[t]d_x + f_u[t]d_u + f[t] - \dot{x}^i(t) \quad \text{a.e. } 0 \leqslant t \leqslant T, \qquad (5.2.11)$$

$$D[0] + D_x[0]d_x(0) = 0, \qquad (5.2.12)$$

$$E[t] + E_x[T]d_x(T) = 0, \qquad (5.2.13)$$

$$S_1[\bar{t}_r^1-] + S_{1x}[\bar{t}_r^1-]d_x(\bar{t}_r^1) + S_{1u}[\bar{t}_r^1-]d_u(\bar{t}_r^1-) \leqslant 0 \quad \text{for all } r = 1,2,\ldots\bar{p}_1, \qquad (5.2.14)$$

$$S_1[\bar{t}_r^1+] + S_{1x}[\bar{t}_r^1+]d_x(\bar{t}_r^1) + S_{1u}[\bar{t}_r^1+]d_u(\bar{t}_r^1+) \leqslant 0 \quad \text{for all } r = 0,1,\ldots\bar{p}_1-1, \qquad (5.2.15)$$

$$S_2[\bar{t}_r^2] + S_{2x}[\bar{t}_r^2]d_x(\bar{t}_r^2) \leqslant 0 \quad \text{for all } r = 0,1,\ldots\bar{p}_2. \qquad (5.2.16)$$

It is obvious that when (d_x^i,d_u^i) is Δ-feasible, and strict inequality holds for all constraints (5.2.14) - (5.2.16) which are not in the working set in iteration i, then it is always possible to select a nonzero step size α_i such that (d_x^{i+1},d_u^{i+1}) is also Δ-feasible. Thus contrary to the finite-dimensional case, the iterates (d_x^i,d_u^i) are in general not feasible, but only Δ-feasible, i.e. the state equality constraints may be violated at interior points of boundary intervals. On the other hand they will always be satisfied at junction and contact points (at all grid points).

As a consequence of this the direction of search consists of two components. A *range space component*, which is a result of the constraint violation (i.e. to restore (d_x^i,d_u^i) from Δ-feasible to feasible) and a *null space component*, which is the actual direction of descent of the objective function in the tangent subspace of the constraints. †

We now turn to the active set strategy, i.e. how the working set is modified in each iteration. This active set strategy is performed after the step size α_i has been determined.

Similar to the finite-dimensional case a constraint is added to the working set when the step size α_i is restricted by one or more constraints. Considering the example of Figure 5.1, the interval $(t_2,\bar{t}_r^2]$ is added to the working set. In the finite-dimensional case, only one constraint is added to the working set in order to maintain that the constraint matrix remained of full row rank. In the present case however, an infinite number of constraints are added to the working set, because the constraint must hold at all time points of the interval $(t_2,\bar{t}_r^2]$. In a numerical setting this may in fact cause trouble, i.e. a matrix of constraint normals, which approximates the constraints of problem (EQP/SCOCP) may become rank deficient as a result of the addition of several constraints in one iteration (cf. Appendix E4).

† We note that with the method of Appendix A, only the null space component needs to be computed, because the range space component is always zero. This fact was used in the replacement of (A13) - (A15) by (A16) - (A18).

In the case that the step size α_i is restricted by more than one constraint, i.e. equality holds for several constraints (5.2.14) - (5.2.16) at the point (d_x^{i+1}, d_u^{i+1}), which were strict inequalities at the point (d_x^i, d_u^i), then only one such constraint is added to the working set. This strategy is similar to the (conservative) strategy of the method for the finite-dimensional case and is followed in the hope to circumvent problems of rank deficiency mentioned above. Using this strategy it is possible that the step size α_i becomes occasionally zero, because a constraint which is satisfied as an equality at the point (d_x^i, d_u^i) is not necessarily in the working set.

We now turn to the subject of deleting constraints from the working set, when the direction of search has become zero.

First we note that when the direction of search has become zero, then (d_x^i, d_u^i) must be a solution to problem (EQP/SCOCP) with the current working set and hence a feasible point of the constraints.

In the finite-dimensional case, only one constraint, which has a Lagrange multiplier with a wrong sign, is deleted from the working set. The situation of the present case however, is considerably more complex. Reasons for this are, that it seems not possible to derive optimality conditions for problem (EIQP/SCOCP/Δ) using the theory contained in Chapter 2, and the fact that the state constraints of order greater than zero represent implicit constraints on the control.

The elimination of constraints from the working set, takes in the present case the form of the elimination of time points or time intervals from one of the working sets W_l^{i-1}, i.e. the working sets which were used in the previous iteration for the constraints (cf. (4.2.1.18)) :

$$\tilde{S}_l[t] + \tilde{S}_{lx}[t]d_x(t) + \tilde{S}_{lu}[t]d_u(t) = 0 \qquad t \in W_l^{i-1} \quad l = 1,\ldots,k_1+k_2. \tag{5.2.17}$$

The determination as to which point(s) can be deleted from the working sets is based on the Lagrange multipliers $(\bar{\eta}, \bar{\beta}_{lr}^k, \bar{v}_{lr})$, which are obtained as the solution of the linear multipoint boundary value problem (5.1.3.3) - (5.1.3.11).

The first k_1 components of the vector $\bar{\eta}$ are the Lagrange multipliers associated with the mixed control state constraints

$$S_{1l}[t] + S_{1lx}[t]d_x(t) + S_{1lu}[t]d_u(t) = 0 \qquad t \in W_l^{i-1} \quad l = 1,\ldots,k_1. \tag{5.2.18}$$

The last k_2 components of the vector $\bar{\eta}$ are Lagrange multipliers associated with the constraints :

$$S_{2l}^{p_l}[t] + S_{2lx}^{p_l}[t]d_x(t) + S_{2lu}^{p_l}[t]d_u(t) = 0 \qquad t \in W_{k_1+l}^{i-1} \quad l = 1,\ldots,k_2, \tag{5.2.19}$$

which may formally be interpreted as the p_l th time derivatives of the state constraints :

$$S_{2l}[t] + S_{2lx}[t]d_x(t) = 0 \qquad t \in W_l^{i-1} \quad l = 1,\ldots,k_2. \tag{5.2.20}$$

The multipliers $\bar{\beta}_{lr}^k$ are Lagrange multipliers associated with the entry point constraints at $\bar{t}_r^2$, i.e.

$$S_{2l}^{k-1}[\bar{t}_r^2] + S_{2lx}^{k-1}[\bar{t}_r^2]d_x(\bar{t}_r^2) = 0 \quad k = 1,\ldots,p_l. \tag{5.2.21}$$

The multipliers $\bar{v}_{lr}$ are Lagrange multipliers associated with the interior point constraints at $\bar{t}_r^2$, i.e.

$$S_{2l}[\bar{t}_r^2] + S_{2lx}[\bar{t}_r^2]d_x(\bar{t}_r^2) = 0. \tag{5.2.22}$$

The actual determination as to which point(s) are deleted from the working sets is based on the signs of the Lagrange multipliers corresponding to the state constraint(s). For the mixed control state constraints (5.2.18) and the interior point constraints (5.2.22), the Lagrange multipliers are directly available (i.e. the first k_1 components of the vector $\bar{\eta}$ and the multipliers $\bar{v}_{lr}$). For boundary intervals of the state constraints (5.2.20) the Lagrange multipliers may be obtained from the last k_2 components of the vector $\bar{\eta}$ and the numbers $\bar{\beta}_{lr}^k$ as :

$$\bar{\eta}_{0l}(t) := (-1)^{p_l} \frac{d^{p_l}\bar{\eta}_{k_1+l}(t)}{dt^{p_l}}. \tag{5.2.23}$$

The Lagrange multipliers associated with entry - and exit point constraints of the form (t_1 is an entry point and t_2 is an exit point) :

$$S_{2l}^{k-1}[t_1] + S_{2lx}^{k-1}[t_1]d_x(t_1) = 0 \quad k = 1,\ldots,p_l. \tag{5.2.24}$$

$$S_{2l}^{k-1}[t_2] + S_{2lx}^{k-1}[t_2]d_x(t_2) = 0 \quad k = 1,\ldots,p_l. \tag{5.2.25}$$

are respectively :

$$\bar{v}_{l1}^{k-1} := \bar{\beta}_{l1}^k + (-1)^{p_l-k} \frac{d^{p_l-k+1}\bar{\eta}_{k_1+l}}{dt^{p_l-k+1}}(t_1+) \qquad k = 1,\ldots,p_l, \tag{5.2.26}$$

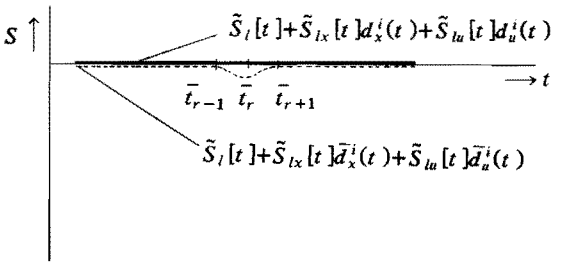
$$\bar{v}_{l2}^{k-1} := -(-1)^{p_l-k} \frac{d^{p_l-k+1}\bar{\eta}_{k_1+l}}{dt^{p_l-k+1}}(t_2-) \qquad k = 1,\ldots,p_l. \tag{5.2.27}$$

The active set strategy consists of the elimination of one time point or one time interval from the working set and is based on these multipliers.

The criteria which are used to delete time points from the working sets may now be summarized as follows (these rules are in fact based on the more rigorous results contained in Appendix C) :

Case 1 : Boundary intervals of mixed control state constraints.
It is supposed that the Lagrange multiplier corresponding to this constraint, $\bar{\eta}_l$, is continuous on boundary intervals. If at some grid point $\bar{t}_r^1$, the multiplier $\bar{\eta}_l$ is strictly negative, then the interval $(\bar{t}_{r-1}^1, \bar{t}_{r+1}^1)$ can be deleted from the working set, provided $|\bar{t}_{r-1}^1 - \bar{t}_{r+1}^1|$ is 'sufficiently' small (Lemma C2). The results in Appendix C do not give any information about how small the interval must be. Fortunately, for the specific numerical implementation of the method, it can be shown that the numerical approximations to the multipliers $\bar{\eta}_l$ are also Lagrange multipliers of a certain finite-dimensional quadratic programming problem (cf. Section 6.1.2). Therefore, for any mixed control state constraint with a Lagrange multiplier having wrong sign at a grid point $\bar{t}_r^1$, the interval $(\bar{t}_{r-1}^1, \bar{t}_{r+1}^1)$ may be deleted from the working set.

Case 2 : Contact points of state constraints (order ≥ 1).
If the Lagrange multiplier $\bar{v}_{lr}$ associated with the interior point constraint at $\bar{t}_r^2$ is strictly negative, then this contact point can be deleted from the working set (Lemma C6).

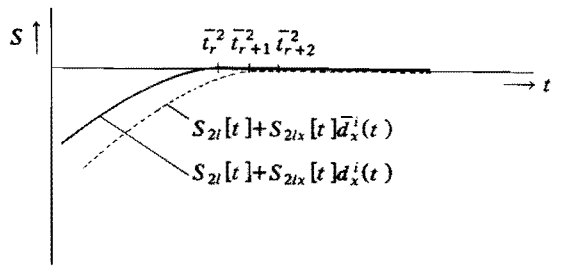


Cases 1 and 3.
Figure 5.2

Case 3 : Interior points of boundary intervals of state constraints (order ≥ 1). When the multiplier $\bar{\eta}_{0l}$ is strictly negative at a grid point $\bar{t}_r^2$ which is also an interior point of a boundary arc, then the interval $(\bar{t}_{r-1}^2, \bar{t}_{r+1}^2)$ can be deleted from the working set, provided $|\bar{t}_{r+1}^2 - \bar{t}_{r-1}^2|$ is sufficiently small (Lemma C5).

Case 4 : Entry- and exit points of boundary intervals of first order state constraints.
To each entry- and exit point of a first order state constraint, one multiplier $\bar{\nu}_{lr}^0$ is associated. If the multiplier $\bar{\nu}_{lr}^0$ is strictly negative, then the boundary interval can be reduced, provided the interval which is eliminated from the working set is sufficiently small (Lemma C4).

Case 5 : Entry- and exit points of second order state constraints.
For the sake of brevity we consider only the case of an entry point, because the case of an exit point is quite similar.
To each entry point of a second order state constraint two multipliers are associated, i.e. $\bar{\nu}_{lr}^0$ and $\bar{\nu}_{lr}^1$.



Cases 4 and 5.1.
Figure 5.3

Case 5.1 : If

$$\bar{\nu}_{lr}^{0} - \frac{\bar{\nu}_{lr}^{1}}{\bar{t}_{r+1}^{2} - \bar{t}_{r}^{2}} < 0,$$

then, the interval $[\bar{t}_r^2, \bar{t}_{r+1}^2)$ can be eliminated from the working set, provided this interval is sufficiently small. In this case the entry point $\bar{t}_r^2$ is eliminated itself and the boundary arc is thus reduced (Lemma C3, part (i)).

Case 5.2 : If

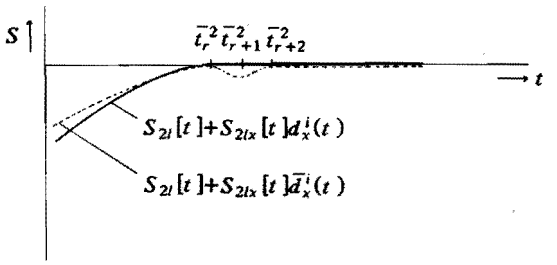
$$\bar{\nu}_{lr}^{0} - \frac{\bar{\nu}_{lr}^{1}}{\bar{t}_{r+1}^{2} - \bar{t}_{r}^{2}} > 0,$$

and

$$\bar{\nu}_{lr}^{1} < 0,$$

and, $\bar{t}_{r+1}^2$ is not an exit point, then the interval $(\bar{t}_r^2, \bar{t}_{r+2}^2)$ can be eliminated from the working set, provided $|\bar{t}_{r+2}^2 - \bar{t}_r^2|$ is sufficiently small (Lemma C3, part (ii)).

In this case the entry point becomes a contact point and the boundary arc is reduced.



Case 5.2
Figure 5.4

The various cases are visualized in Figures 5.2 - 5.4.

From the rules stated above it becomes clear that, when the multipliers $(\bar{\lambda}, \bar{\eta}, \bar{\sigma}, \bar{\mu}, \bar{\beta}_{lr}^{l}, \bar{\nu}_{lr})$ satisfy the conditions (5.2.28) - (5.2.33), then no time points will be deleted from the working set.

$$\bar{\eta}_l(\bar{t}_r^1) \geqslant 0 \qquad \text{for all } r = 0,1,\ldots\bar{p}_1,\ l = 1,\ldots k_1, \qquad (5.2.28)$$

$$\bar{v}_{lr} \geqslant 0 \qquad \text{for all contact points} \qquad (5.2.29)$$

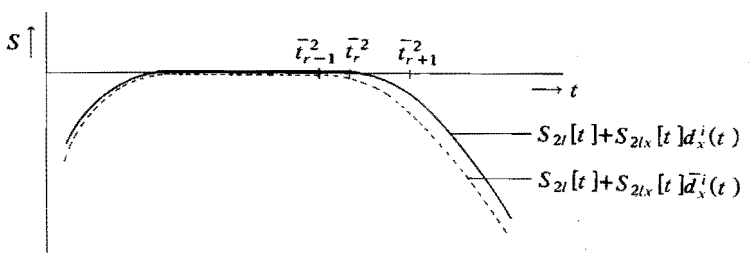
$$\bar{\eta}_{0l}(\bar{t}_r^2) \geqslant 0 \qquad \text{for all interior points of boundary intervals of state constraints} \qquad (5.2.30)$$

$$\bar{\nu}_{lr}^0 \geqslant 0 \qquad \text{for all entry- and exit points of first order state constraints} \qquad (5.2.31)$$

$$\bar{\nu}_{lr}^0 - \frac{\bar{\nu}_{lr}^1}{\bar{t}_{r+1}^2 - \bar{t}_r^2} \geqslant 0 \text{ and } \bar{\nu}_{lr}^1 \geqslant 0 \qquad \text{for all entry points of second order state constraints} \qquad (5.2.32)$$

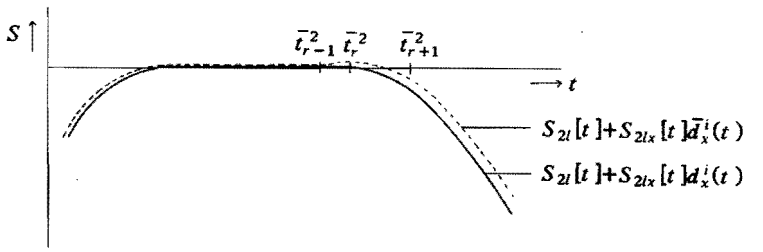
$$\bar{\nu}_{lr}^0 + \frac{\bar{\nu}_{lr}^1}{\bar{t}_r^2 - \bar{t}_{r-1}^2} \geqslant 0 \text{ and } \bar{\nu}_{lr}^1 \leqslant 0 \qquad \text{for all exit points of second order state constraints} \qquad (5.2.33)$$

On the other hand, if the multipliers do not satisfy these conditions then improvement of the objective function can be made by deleting time points from the working set. However, in the case that a time interval is eliminated from the working set (cases 3 - 5), a Δ-feasible direction of search can only be garantueed if the interval is 'sufficiently small'.
If the junction and contact points are restricted to an a priori chosen and fixed grid, this condition may not always be satisfied. Both situations are depicted in figures 5.5 and 5.6.



Δ-feasible direction of search.
Figure 5.5

A possible remedy for this problem is to check whether the direction of search is or is not Δ-feasible and in the case that the direction of search is not Δ-feasible to adjust the grid Δ. We note that up to this point the grid Δ was treated as though it is specified in advance and kept fixed throughout the first stage of Algorithm 4.4. An advantage of this remedy is that after the grid Δ is modified properly, it is possible to continue the algorithm and to



Infeasible direction of search.
Figure 5.6

stop only at a point (d_x^i, d_u^i) which is a solution to problem (EQP/SCOCP) and for which the Lagrange multipliers, corresponding, to the solution of problem (EQP/SCOCP) satisfy the conditions (5.2.28) - (5.2.33).

The strategy which is used to modify the grid Δ is essentially motivated by the same arguments as the rules for the active set strategy. The following cases may be distinguished :

1) An interval interior of a boundary arc was eliminated from the working set. In this case the grid Δ is 'too course' and the grid may be adjusted by inserting additional grid points in the interval which was deleted.

2) An entry- or exit point was eliminated from the working set. In this case, it is sufficient to shift the grid point which was deleted from the working set. The actual time point to which the grid point is shifted is simply determined by reducing the corresponding interval with a constant factor.

The algorithm outlined above may be summarized as follows :

Algorithm 5.8 :

(0) *Given a Δ-feasible pair (d_x^0, d_u^0).*
$i := 0$.

(i) *If (d_x^i, d_u^i) is feasible, the direction of search $(\Delta d_x^{i-1}, \Delta d_u^{i-1})$ was zero and the Lagrange multipliers corresponding to the solution of problem (EQP/SCOCP) satisfy the conditions (5.2.28) - (5.2.33), then ready.*

(ii) *Calculate a Δ-feasible direction of search $(\Delta d_x^i, \Delta d_u^i)$.*

(iia) *Calculate a direction of search $(\Delta d_x^i, \Delta d_u^i)$, based on the solution of problem (EQP/SCOCP).*

(iib) *If the direction of search is not feasible for the constraint which was deleted from the working set in the iteration i-1, then "Modify the grid Δ" and goto (iia).*

(iii) *If $\|(\Delta d_x^i, \Delta d_u^i)\| = 0$ then goto (vii).*

(iv) *Calculate a step size α_i and set*

$$d_x^{i+1} := d_x^i + \alpha_i \Delta d_x^i,$$

$$d_u^{i+1} := d_u^i + \alpha_i \Delta d_u^i .$$

(v) *If the step size α_i was restricted by one or more constraints, add one of these constraints to the working set.*

(vi) $i := i+1$.
goto (i)

(vii) *Check signs of multipliers and, if possible, delete a constraint from the working set.*
goto (ii).

In Algorithm 5.8 it is assumed that an initial Δ-feasible point (d_x^0, d_u^0) is available in step (0). In general however, as in the case of finite-dimensional quadratic programming, such a point is not available.

With finite-dimensional quadratic programming an initial feasible point may be computed using a phase 1 - simplex procedure (cf. Gill et al. (1981)). This phase 1 - simplex procedure may be started with an arbitrary point and generates directions of search for a linear programming problem by means of a simplex strategy.

A related method is to make use of an algorithm which is essentially similar to Algorithm 5.8. As with the phase 1 - simplex procedure, the constraints of the problem are put in the objective function when they are violated at the current point, or treated as constraints when they are satisfied at the current point. The objective function for the linear programming problem takes the following form :

$$\bar{f}(d_x, d_u) := \sum_{l=1}^{k_1+k_2} \int_{W_l^+} (\tilde{S}_{lx}[t]d_x + \tilde{S}_{lu}[t]d_u)\, dt , \qquad (5.2.34)$$

where

$$W_l^+ := \{t \in [0,T] : \tilde{S}_l[t] + \tilde{S}_{lx}[t]d_x^i(t) + \tilde{S}_{lu}[t]d_u^i(t) > 0\} \qquad (5.2.35)$$

Instead of using the simplex technique for the generation of a direction of search, a direction of search can be determined as the solution of a quadratic programming problem (i.e. as in Algorithm 5.8), this is done by means of augmenting the objective function (5.2.34) with the term

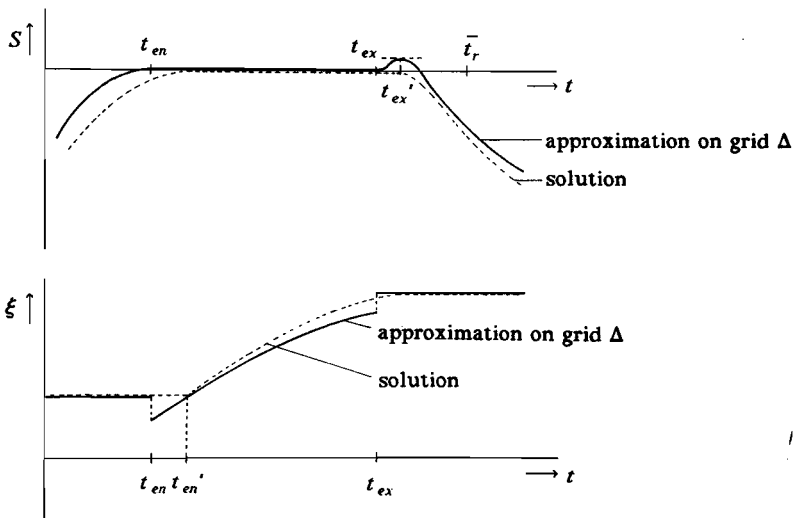
$$\tfrac{1}{2}\left\{x(0)^T x(0) + \int_0^T (x(t)^T x(t) + u(t)^T u(t))\, dt + x(T)^T x(T)\right\}. \qquad (5.2.36)$$

The solution of the resulting quadratic programming problem has the interpretation of the negative gradient of the objective function (5.2.34) projected on the subspace of feasible points.

The starting point of this procedure is in general arbitrary. A plausible choice is to take the solution of problem (EQP/SCOCP) with the last working set which was used in the previous iteration of Algorithm 4.4. When this point is feasible with respect to the constraints of problem (EIQP/SCOCP/Δ), then Algorithm 5.8 is started at this point and when it is not feasible, then the point is used as a starting point of the phase 1 procedure outlined above.

5.3. Determination of the active set of problem (SCOCP).

This section deals with the active set strategy that is to be executed in step (vii) of Algorithm 4.4 and which plays a key role in the second stage of the method. Obviously, convergence of the first stage of the method is assumed and hence an estimate of the solution of problem (SCOCP) together with estimates for the Lagrange multipliers are available. Assuming the direction of search became zero in the last iteration of the first stage, these estimates have the interpretation of an approximation to the solution on the grid Δ as depicted in Figure 5.7 for a scalar state constraint.



Solution of first stage and exact solution.
Figure 5.7

In general this approximation will not satisfy the constraints nor the optimality conditions of problem (SCOCP) completely. As an example consider Figure 5.7, the state constraint is violated just after the constraint switches from active to inactive and the multiplier ξ is not nondecreasing because it has a negative jump at t_{en}.

Using the active set strategy described in this section, the entry- , exit- and contact points are adjusted, in order to make convergence to a point which satisfies the constraints and the optimality conditions of problem (SCOCP), possible.

As the example of Figure 5.7 already indicates, the adjustment of the junction and contact points has a local character and hence the adjustment of the different junction and contact point is done completely independent of each other.

In this section we shall consider only those cases where junction and contact points of different constraints do not coincide. A strategy for more general cases is still to be investigated. Two different strategies for the computation of the actual amount of shift of the junction and contact points are described in Subsections 5.3.1 and 5.3.2. (For a more detailed treatment we refer to Souren (1986).)

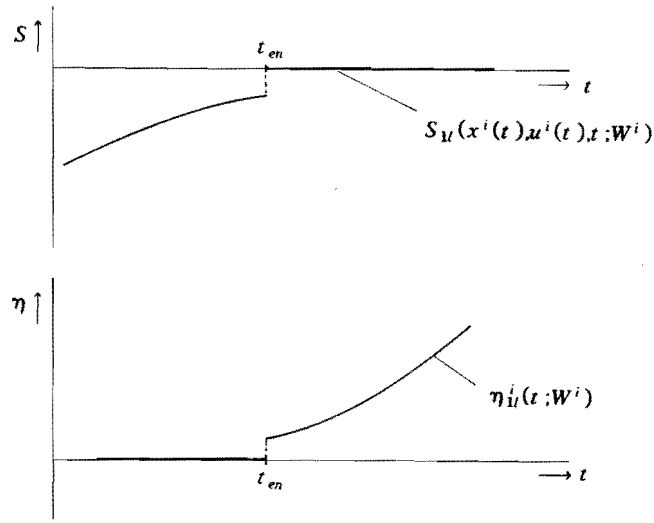
5.3.1. Determination of the junction and contact points based on the Lagrange multipliers.

One way to adjust the junction and contact points is based on the violation of the constraints and the conditions that the Lagrange multipliers corresponding to the state constraints must satisfy. This method was in fact already outlined in Figure 5.7.
The entry point t_{en} is shifted to the point t_{en}', where $\xi(t_{en}) = \xi(t_{en}')$, i.e. $\xi(t) - \xi(t_{en}) < 0$ on (t_{en}, t_{en}') and $\xi(t) - \xi(t_{en}) > 0$ on $(t_{en}', T]$. The exit point t_{ex} is shifted to a point where

$$\frac{d^p S}{dt^p}(x(t_{ex}'), u(t_{ex}'), t_{ex}') = 0.$$

Similar to the description of the active set strategy in Section 5.2 a number of different cases may be distinguished.

Case 1 : Entry- and exit point of boundary intervals of mixed control state constraints. We shall only consider the case of an entry point, because exit points are treated similarly. The situation which is likely to occur in the optimal point is depicted in Figure 5.8.



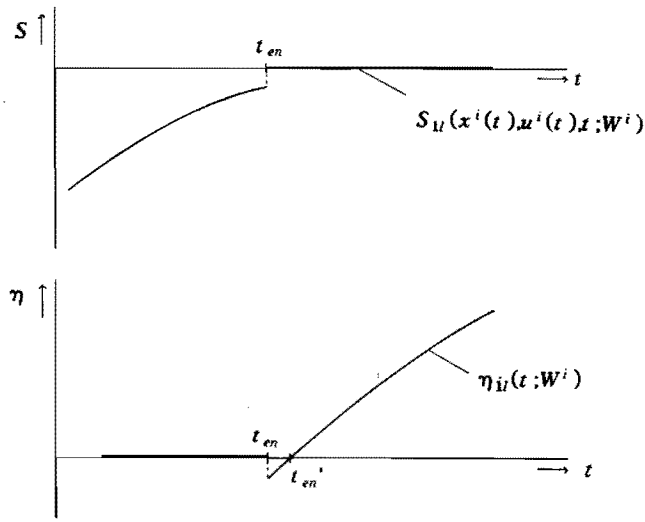
Solution.
Figure 5.8

If the structure of the solution is correct, but the entry point of the constraint is not correct, then one of the two situations depicted in figures 5.9 and 5.10, will arise. †

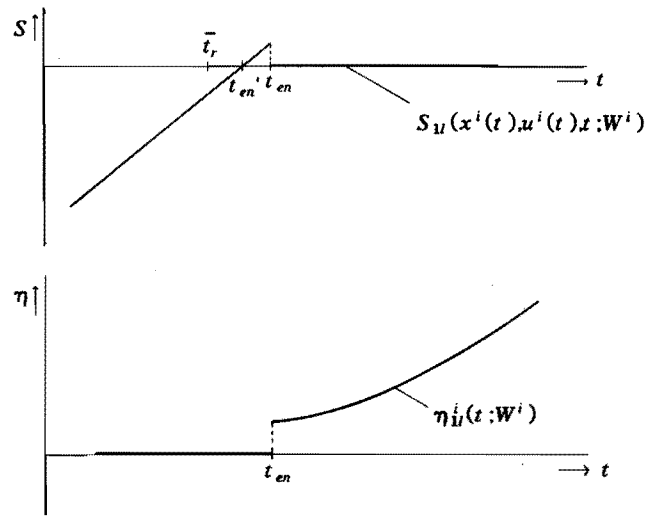
† $S_{1l}(x^i(t), u^i(t), t; W^i)$ and $\eta^i_{1l}(t; W^i)$ denote the value of the mixed control state constraint S_{1l} and the multiplier η_{1l} along the current approximation to the solution in iteration i, with working set W^i.

In the case depicted in Figure 5.9, a new estimate for the entry point, t_{en}' is determined as :

$$\eta^i(t_{en}';W^i) = 0. \qquad (5.3.1.1)$$



Adjustment of entry point based on multiplier η^i.
Figure 5.9



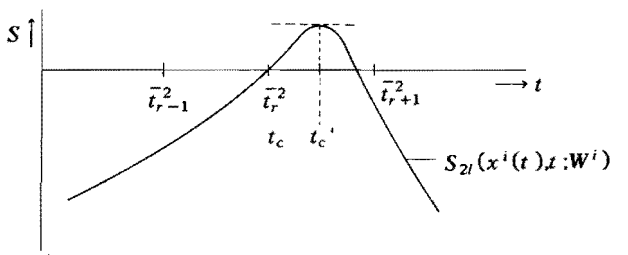
Adjustment of entry point based on constraint violation.
Figure 5.10

In the case depicted in Figure 5.10, a new estimate for the entry point, t_{en}' is determined as :

$$S_{1l}(x^i(t_{en}'),u^i(t_{en}'),t_{en}';W^i) = 0. \qquad (5.3.1.2)$$

Case 2 : Contact points of state constraints (order $\geqslant$ 1). The situation which will occur when the value of the contact point is not correct, is depicted in Figure 5.11. In this case the new estimate of the contact point, t_c' satisfies :

$$\frac{dS_{2l}(x^i(t_c'),t_c';W^i)}{dt} = 0. \qquad (5.3.1.3)$$



Adjustment of contact points.
Figure 5.11

Case 3 : Entry- and exit point of boudary intervals of state constraints (order $\geqslant$ 1). This case is depicted in Figure 5.7. In the case that there is a violation of the constraint (near the exit point in Figure 5.7), then the strategy is similar to the case of a mixed control state constraint with $S_{1l}(x^i(t),u^i(t),t;W^i)$ replaced by $S_{2l}^{p_l}(x^i(t),u^i(t),t;W^i)$, i.e. the p_l-th time derivative of the state constraint. In the case that the Lagrange multiplier ξ_l^i is not nondecreasing on [0,T], then the junction points are adjusted as depicted in Figure 5.7.

We note that in the actual implementation of the method the "nondecreasing" condition for ξ_l^i is expressed in terms of the multipliers β^1 and η^1 as defined by (3.3.6.2) - (3.3.6.3). For first order state constraints this means that directly use is made of the multiplier γ that is associated with the mixed control state constraint $S_{2l}^{p_l}$ (cf. Lemma 5.3).

Following the strategy outlined above, the junction and contact points are adjusted using the following scheme :

$$t_{en}' := \Phi(W^i), \qquad (5.3.1.4)$$

because the solution of problem (EQP/SCOCP), which is used as a direction of search in the second stage of Algorithm 4.4, is governed by the working set W^i. Assuming that shifting junction and contact points gives only local variations in the solution we replace (5.3.1.4) by :

$$t_{en}' := \Phi(t_{en}), \qquad (5.3.1.5)$$

which reveals that the iteration process is essentially a fixed point iteration. When Φ is a

smooth function we shall have linear convergence if $\Phi'(\hat{t}_{en}) \neq 0$ ($\hat{t}_{en}$ denotes the optimal entry point) and quadratic convergence if $\Phi'(\hat{t}_{en}) = 0$. If $\Phi'(\hat{t}_{en}) \neq 0$ and $\Phi'(\hat{t}_{en}) \neq 1$ then the rate of convergence of the iteration process may be improved by modification of (5.3.1.5) to a secant iteration process.

5.3.2. Determination of the junction and contact points based on the Hamiltonian.

An alternative way to adjust the junction and contact points is based on the results contained in Theorem 3.12, which state that for all junction and contact points $\tilde{t}$ the following jump condition must hold :

$$H[\tilde{t}+] = H[\tilde{t}-] - d\,\xi(\tilde{t})^T S_{2t}[\tilde{t}]. \qquad (5.3.2.1)$$

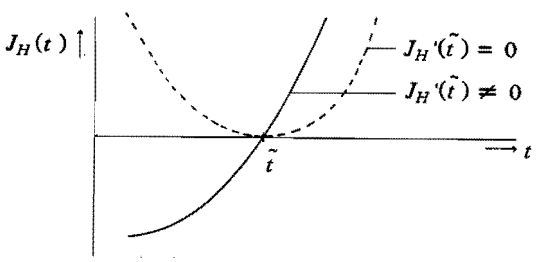
Given an approximation to the solution $(x^i, u^i, \lambda^i, \eta_1^i, \xi^i, \sigma^i, \mu^i)$ we now define for each junction and contact point : †

$$J_H(\tilde{t}) := H[\tilde{t}+] - H[\tilde{t}-] + d\,\xi(\tilde{t})^T S_{2t}[\tilde{t}], \qquad (5.3.2.2)$$

and we consider the equation

$$J_H(\tilde{t}) = 0, \qquad (5.3.2.3)$$

where $\tilde{t}$ is a junction or a contact point. Equation (5.3.2.3) may be solved via a standard strategy, which determines a zero of a nonlinear function of one variable. The iterates of such a strategy will be used for the working sets for successive iterations of Algorithm 4.4. This strategy will in general yield good results, provided $J_H'(\tilde{t}) \neq 0$. Unfortunately, practical examples exist for which $J_H'(\tilde{t}) = 0$ (cf. Figure 5.12). This is a serious drawback for the use of this technique in a general solution for problem (SCOCP).



Defect of jump condition v.s. junction or contact point.
Figure 5.12

† Note that in (5.3.2.1) straight brackets were used to replace argumentlists involving the solution of problem (SCOCP) and in (5.3.2.2) these brackets were used to replace argumentlists involving the current iterate.

6. Numerical implementation of the method.

This chapter deals with the most important aspect of the numerical implementation of the method, i.e. the numerical solution of the linear multipoint boundary value problem, which is to be solved in order to obtain a numerical approximation to the solution of problem (EQP/SCOCP). Section 6.1 deals with a motivation for the choice of the integration method and an inspection of the set of linear equations to be solved. The solution of this set of equations is considered in more detail in Section 6.2. The truncation errors of the integration method are considered in Section 6.3. For the sake of completeness, a number of computational details of rather specialized nature are given in Appendices D and E.

6.1. Numerical solution of problem (EQP/SCOCP).

6.1.1. Solution of the linear multipoint boundary value problem.

From Theorem 5.6 we recall that the solution of problem (EQP/SCOCP) can be obtained as the solution of the following linear multipoint boundary value problem †:

$$\begin{bmatrix} \dot{d}_x \\ \dot{\lambda} \end{bmatrix} = \begin{bmatrix} f_x & 0 \\ -M_2 & -f_x^T \end{bmatrix} \begin{bmatrix} d_x \\ \lambda \end{bmatrix} + \begin{bmatrix} f_u & 0 \\ -M_3 & -(R_x^p)^T \end{bmatrix} \begin{bmatrix} d_u \\ \eta_I \end{bmatrix} + \begin{bmatrix} f - \dot{x}^i \\ -f_{0x} \end{bmatrix} \quad 0 \leqslant t \leqslant T \quad (6.1.1.1)$$

$$0 = \begin{bmatrix} R_x^p & 0 \\ M_3^T & f_u^T \end{bmatrix} \begin{bmatrix} d_x \\ \lambda \end{bmatrix} + \begin{bmatrix} R_u^p & 0 \\ M_4 & (R_u^p)^T \end{bmatrix} \begin{bmatrix} d_u \\ \eta_I \end{bmatrix} + \begin{bmatrix} R^p \\ f_{0u} \end{bmatrix} \quad 0 \leqslant t \leqslant T, \quad (6.1.1.2)$$

$$\begin{bmatrix} D_x[0] & 0 \\ M_1 & I \end{bmatrix} \begin{bmatrix} d_x(0) \\ \lambda(0) \end{bmatrix} + \begin{bmatrix} 0 \\ D_x[0]^T \end{bmatrix} \sigma = - \begin{bmatrix} D[0] \\ h_{0x}[0]^T \end{bmatrix}, \quad (6.1.1.3)$$

$$N_x[\tilde{t}_j] d_x(\tilde{t}_j) = - N[\tilde{t}_j], \quad (6.1.1.4)$$

$$\lambda(\tilde{t}_j +) = \lambda(\tilde{t}_j -) - N_x[\tilde{t}_j]^T \chi_j, \quad (6.1.1.5)$$

$$\begin{bmatrix} E_x[T] & 0 \\ M_5 & -I \end{bmatrix} \begin{bmatrix} d_x(T) \\ \lambda(T) \end{bmatrix} + \begin{bmatrix} 0 \\ E_x[T]^T \end{bmatrix} \mu = - \begin{bmatrix} E[T] \\ g_{0x}[T]^T \end{bmatrix}. \quad (6.1.1.6)$$

where η_I denotes the $\bar{k}(t)$-vector of components of the multiplier η corresponding to the active constraints. The matrices $N_x[\tilde{t}_j]$ and the vectors $N[\tilde{t}_j]$ represent the interior point constraints (5.1.3.9) and (5.1.3.11). The vectors χ_j contain the multipliers $\bar{\beta}_{lj}^k$ and $\bar{v}_{lj}$. The notation $\tilde{t}_j$ is used for the junction and contact points, in order to simplify notation.

The set of equations (6.1.1.1) - (6.1.1.6) can be transformed into a standard linear multipoint boundary value problem, by means of substitution of

$$\begin{bmatrix} d_u \\ \eta_I \end{bmatrix} = - \begin{bmatrix} R_u^p & 0 \\ M_4 & (R_u^p)^T \end{bmatrix}^{-1} \left(\begin{bmatrix} R_x^p & 0 \\ M_3^T & f_u^T \end{bmatrix} \begin{bmatrix} d_x \\ \lambda \end{bmatrix} + \begin{bmatrix} R^p \\ f_{0u} \end{bmatrix} \right), \quad (6.1.1.7)$$

into (6.1.1.1) and elimination of the vectors σ, χ_j and μ using:

$$\sigma = - (D_x[0]^T)^+ (h_{0x}[0]^T - \lambda(0) - M_1 d_x(0)), \quad (6.1.1.8)$$

† Obviously, it is assumed that the hypotheses of Theorem 5.6 hold.

$$\chi_j = (N_x[\tilde{t}_j]^T)^+(\lambda(\tilde{t}_j-) - \lambda(\tilde{t}_j+)), \qquad (6.1.1.9)$$

$$\mu = -(E_x[T]^T)^+(g_{0x}[T]^T + \lambda(T) - M_5 d_x(T)). \qquad (6.1.1.10)$$

Substitution of respectively (6.1.1.8) in (6.1.1.3), (6.1.1.9) in (6.1.1.5), and (6.1.1.10) in (6.1.1.6) yields a set of $2n$ boundary conditions and $2n$ interior point conditions at each point $\tilde{t}_j$. †

Equations (6.1.1.1) - (6.1.1.6) can thus be transformed into :

$$\dot{v}(t) = A_1[t]v(t) + B_1[t]w(t) + c_1[t] \quad a.e. \quad 0 \leqslant t \leqslant T, \qquad (6.1.1.11)$$

$$0 = A_2[t]v(t) + B_2[t]w(t) + c_2[t] \quad a.e. \quad 0 \leqslant t \leqslant T, \qquad (6.1.1.12)$$

$$K_0 v(0) + l_0 = 0, \qquad (6.1.1.13)$$

$$K_j^+ v(\tilde{t}_j+) + K_j^- v(\tilde{t}_j-) + l_j = 0 \quad all\ j, \qquad (6.1.1.14)$$

$$K_T v(T) + l_T = 0. \qquad (6.1.1.15)$$

For the numerical solution of ordinary boundary value problems two types of methods may be distinguished :

1) Shooting methods.
   For linear boundary value problems these methods are called methods of particular solutions. Of practical importance are multiple shooting methods. With these methods the entire interval [0,T] is divided into a number of subintervals. The values of the vector v are estimated at one side of the subinterval and the values on the other side of the subinterval are obtained as the solution of an initial value problem. The solution obtained in this way will not be continuous on boundary points of successive subintervals, nor will it satisfy the boundary - and interior point conditions. Using the defect of the boundary -, interior point - and continuity conditions of a number of solutions with different initial values of the vector v, it is possible to compute the correct initial values of v (cf. Stoer et al. (1980) and Miele et al. (1968)).

2) Approximation methods.
   With these methods the time functions v are approximated using a finite-dimensional base. The equations (6.1.1.11) - (6.1.1.15) yield in a way dependent on the actual method, a set of linear equations. This usually large and sparse system of equations may be solved using sparse matrix techniques.

In the implementation of Algorithm 4.4 an approximation method is chosen in favour of a shooting method, because of the following arguments :

a) For shooting methods usually a Runge-Kutta like integration method is used, in order to allow control of the truncation error in solving the initial value problem. Because the right hand side of (6.1.1.11) dependents on the current estimate $(x^i, u^i, \lambda^i, \eta_1^i, \xi^i)$ of the solution of problem (SCOCP), some kind of interpolation of the time functions $(x^i, u^i, \lambda^i, \eta_1^i, \xi^i)$ is required. Practical experience showed that this may cause problems (cf. Souren (1984)). With an approximation method these problems are circumvented by the use of a fixed step integration method. ‡

† In addition to equations (6.1.1.4) - (6.1.1.5) use is made of the condition $d_x(\tilde{t}_j+) = d_x(\tilde{t}_j-)$.
‡ For the implementation of the active set strategies, discussed in Sections 5.2 and 5.3, an interpolation scheme for the time functions is required anyway.

b) At every time point where the right hand side of (6.1.1.11) is to be evaluated, the equation

$$B_2[t]w = -c_2[t] - A_2[t]v,$$

must be solved for w.
It is considered an advantage of approximation methods that the equations (6.1.1.11) and (6.1.1.12) can be treated similar.

c) The actual implementation of the particular approximation method chosen can be linked directly to the solution of a large, sparse quadratic programming problem. This allows a more or less standard numerical approach (cf. Section 6.2).

Within the class of approximation methods a distinction can be made between finite difference methods (with extrapolation) and collocation methods. It can be shown that for higher order methods, collocation methods using polynomials of order ≥ 2 are more efficient than finite difference methods (cf. Souren (1986)). Therefore only methods of this type will be considered here.

The time functions are approximated using piecewise polynomials on [0,T], i.e. given a grid

$$0 = t_0 < t_1 < \ldots\ldots t_{p-1} < t_p = T, \qquad (6.1.1.16)$$

the function $v(t)$ is approximated using l th-degree polynomials on (t_r, t_{r+1}). For each time function this yields $l+1$ coefficients on each subinterval (t_r, t_{r+1}). One of these coefficients will be determined by the fact that the function v must be continuous at the points t_r (or must satisfy equation (6.1.1.15)). The other l coefficients are determined by the condition that the differential equation must be satisfied at l distinct points on the interval (t_r, t_{r+1}). These points are called the collocation points, which are defined using l numbers ρ_i which satisfy :

$$0 \leq \rho_1 < \rho_2 < \ldots\ldots < \rho_l \leq 1. \qquad (6.1.1.17)$$

The collocation points on (t_r, t_{r+1}) are defined by :

$$\tau_{lr+i} := t_r + \rho_i h_r \qquad i = 1,\ldots\ldots l \qquad r = 0,1,\ldots\ldots,p-1, \qquad (6.1.1.18)$$

where

$$h_r := t_{r+1} - t_r. \qquad (6.1.1.19)$$

Because the approximating functions are polynomials on the intervals (t_r, t_{r+1}) the time points $\tilde{t}_j$, where (6.1.1.15) must be satisfied, can only be points of the grid (6.1.1.16). This yields automatically the grid Δ^2 (cf. (4.2.2.1) - (4.2.2.3)), i.e. the grid to which the junction and contact points of the state constraints with order ≥ 1 of problem (EIQP/SCOCP) are restricted during the first stage of the solution process. The grid Δ^1, i.e. the grid to which the junction and contact points of the mixed control state constraints are restricted may be chosen to be all collocation points. The reason for this is that in the collocation method these constraints enter the formulation only at these points, i.e. only the values of the mixed control state constraints on the collocation points are required (see description of the collocation scheme below).

The collocation scheme is governed by the actual parameterization scheme used for the finite-dimensional representation of the (approximating) time functions. There are two obvious alternatives to this parameterization :

1) The truncated power base is used to represent the time functions on the interval (t_r, t_{r+1}), i.e.

$$v(t) = \sum_{i=1}^{l} v_{r,i}(t - t_r)^i \qquad t_r \leqslant t < t_{r+1}.$$

In this case the coefficients of the polynomials, $v_{r,i}$, are used as parameters in the collocation scheme.

2) The values of the time functions v on the grid points and the collocation points, i.e. $v(t_r)$, $v(\tau_{lr+1}),\ldots,v(\tau_{lr+l})$ are used directly as parameters in the collocation scheme.

The second parameterization scheme was actually chosen. A motivation for this may be that the truncated power base is not always a suitable base for piecewise polynomial interpolation (cf. de Boor (1978)). † The derivation of the collocation scheme based on this second scheme is done via the application of implicit Runge-Kutta schemes to the boundary value problem (see also Weiss (1974)). To this end the following quantities are defined, using numbers ρ_i that satisfy (6.1.1.17) :

$$\omega_{jk} := \int_0^{\rho_j} L_k(s)\, ds \qquad j = 1,\ldots,l \quad k = 1,\ldots,l\ , \tag{6.1.1.20}$$

where

$$L_k(s) := \prod_{\substack{i=1 \\ i \neq k}}^{l} \frac{(s - \rho_i)}{(\rho_k - \rho_i)} \quad 0 \leqslant s \leqslant 1. \tag{6.1.1.21}$$

The weights ω_{jk} lead to the following set of quadrature rules :

$$\int_0^{\rho_j} \phi(s)\, ds \sim \sum_{k=1}^{l} \omega_{jk}\, \phi(\rho_k). \tag{6.1.1.22}$$

In case that $\rho_1 > 0$ and $\rho_l < 1$ in the collocation method, the introduction of l additional weights is necessary :

$$\bar{\omega}_k := \int_0^1 L_k(s)\, ds. \tag{6.1.1.23}$$

The weights $\bar{\omega}_k$ are also used in a quadrature formula :

$$\int_0^1 \phi(s)\, ds \sim \sum_{k=1}^{l} \bar{\omega}_k\, \phi(\rho_k). \tag{6.1.1.24}$$

Depending on whether $\rho_1 = 0$ or $\rho_1 > 0$, and whether $\rho_l = 1$ or $\rho_l < 1$, different collocation schemes will follow.

Up till this point the numbers ρ_i were treated as arbitrary fixed quantities. However, the actual choice of these numbers is still left open. These numbers may be chosen such that the order of the quadrature formulas (6.1.1.24) is maximized. In addition, one is able to fix ρ_1 to zero and/or ρ_l to one. When ρ_1 and ρ_l are not fixed, this maximization yields the Gaussian quadrature formulas, where $\rho_1 > 0$ and $\rho_l < 1$ (cf. Stoer et al. (1980), p.142-151).

† We note that this base was used in an earlier implementation of the method (cf. de Jong et al. (1985)). Numerical evidence also pointed out that the second base is a better choice.

The numbers ρ_i, which define (collocation) points on the interval [0,1], are called the Gauss points. When either ρ_1 or ρ_l is fixed, the points ρ_j become the so-called Radau points and when both ρ_1 and ρ_l are fixed then the so-called Lobatto points follow. It can be shown that usually the Lobatto points are the most efficient from a numerical point of view (cf. Weiss (1974)). However, for the specific case considered here, the use of Gauss points seems to have a significant advantage over the use of Lobatto points. The reason for this is that, using the Gauss points, the set of linear equations that results from the collocation method applied to the specific linear multipoint boundary value problem (6.1.1.1) - (6.1.1.6) can be tranformed into a symmetric indefinite system, which allows a solution procedure that makes efficiently use of this structure. This transformation seems not possible when the Lobatto points are used. Therefore the Gauss points are used in the current implementation of the method.

The collocation scheme follows from the approximation of the integral equations which follow from (6.1.1.11) as :

$$v(\tau_{lr+i}) = v(t_r+) + \int_{t_r+}^{\tau_{lr+i}} \Big[A_1[s]v(s) + B_1[s]w(s) + c_1[s]\Big]\, ds$$

$$i = 1,\ldots,l \qquad r = 0,1,\ldots,p-1. \tag{6.1.1.25}$$

$$v(t_{r+1}-) = v(t_r+) + \int_{t_r+}^{t_{r+1}-} \Big[A_1[s]v(s) + B_1[s]w(s) + c_1[s]\Big]\, ds$$

$$r = 0,1,\ldots,p-1. \tag{6.1.1.26}$$

Approximation of (6.1.1.25) - (6.1.1.26) using (6.1.1.22) and (6.1.1.24) yields the following set of linear equations :

$$v(\tau_{lr+i}) = v(t_r+) + h_r \sum_{k=1}^{l} \omega_{ik} \Big[A_1[\tau_{lr+k}]v(\tau_{lr+k}) + B_1[\tau_{lr+k}]w(\tau_{lr+k}) + c_1[\tau_{lr+k}]\Big]$$

$$i = 1,\ldots,l \qquad r = 0,1,\ldots,p-1, \tag{6.1.1.27}$$

$$v(t_{r+1}-) = v(t_r+) + h_r \sum_{k=1}^{l} \bar{\omega}_k \Big[A_1[\tau_{lr+k}]v(\tau_{lr+k}) + B_1[\tau_{lr+k}]w(\tau_{lr+k}) + c_1[\tau_{lr+k}]\Big]$$

$$r = 0,1,\ldots,p-1. \tag{6.1.1.28}$$

The vector w is determined by the algebraic equation (6.1.1.12) almost everywhere on [0,T]. For the numerical solution of (6.1.1.27) - (6.1.1.28) the value of this vector is only required at the collocation points, this yields the following equations :

$$0 = A_2[\tau_{lr+i}]v(\tau_{lr+i}) + B_2[\tau_{lr+i}]w(\tau_{lr+i}) + c_2[\tau_{lr+i}]$$

$$i = 1,\ldots,l \qquad r = 0,1,\ldots,p-1\,. \tag{6.1.1.29}$$

At every grid point t_r $(r = 1,\ldots,p-1)$ an equation of the form (6.1.1.15) holds, because either t_r coincides with one of the time points $\tilde{t}_j$ or the $v(t)$ must be continuous at t_r, in which case (6.1.1.15) holds with $K_j^+ = I$, $K_j^- = -I$, and $l_j = 0$. † Combination of (6.1.1.13)

† At this point it is assumed that $t = 0$ and $t = T$ are not junction and contact points. Generalization to this case may be done by taking the boundary - and interior point conditions together.

- (6.1.1.15) with (6.1.1.27) - (6.1.1.29) yields a sparse set of linear equations that can be solved using sparse matrix techniques.

We note that combination of (6.1.1.15) and (6.1.1.28) allows the elimination of either $v(t_r -)$ or $v(t_r +)$ from the set of linear equations.

6.1.2. Inspection of the collocation scheme.

In this section the set of linear equations that follows from the collocation method applied to the linear multipoint boundary value problem for the solution of problem (EQP/SCOCP) will be considered in more detail.

In Section 6.1.1 the collocation method was outlined using the compact formulation, of the linear multipoint boundary value problem, of equations (6.1.1.11) - (6.1.1.15). For the implementation of the collocation method use is made of the structure of the equations (6.1.1.1) - (6.1.1.6) which is hidden by the more compact formulation. To outline the essence of the approach, equations (6.1.1.1) and (6.1.1.2) are rewritten as :

$$\begin{bmatrix} -\dot{\lambda} \\ 0 \end{bmatrix} = \begin{bmatrix} M_2 & M_3 \\ M_3^T & M_4 \end{bmatrix} \begin{bmatrix} d_x \\ d_u \end{bmatrix} + \begin{bmatrix} f_x^T & (R_x^p)^T \\ f_u^T & (R_u^p)^T \end{bmatrix} \begin{bmatrix} \lambda \\ \eta_I \end{bmatrix} + \begin{bmatrix} f_{0x} \\ f_{0u} \end{bmatrix} \quad 0 \leqslant t \leqslant T. \qquad (6.1.2.1)$$

$$\begin{bmatrix} \dot{d}_x \\ 0 \end{bmatrix} = \begin{bmatrix} f_x & f_u \\ R_x^p & R_u^p \end{bmatrix} \begin{bmatrix} d_x \\ d_u \end{bmatrix} \qquad 0 \leqslant t \leqslant T. \qquad (6.1.2.2)$$

Here a distinction is made between the equations due to the constraints of problem (EQP/SCOCP), i.e. (6.1.2.2) and the equations which result from the optimality conditions for problem (EQP/SCOCP), i.e. (6.1.2.1). The main result of this section will be that the linear equations that follow from the collocation method applied to the equations (6.1.1.1) - (6.1.1.6) can be transformed into a set of linear equations of the form

$$\begin{bmatrix} M & C^T \\ C & 0 \end{bmatrix} \begin{bmatrix} d \\ \zeta \end{bmatrix} = \begin{bmatrix} -c \\ b \end{bmatrix} \qquad (6.1.2.3)$$

where the submatrices C and M are sparse and banded.

For the solution of the collocation scheme effective use of the special structure of the system (6.1.2.3) is possible.

As a first step towards the transformation outlined above we consider the linear equations due to the constraints of problem (EQP/SCOCP), that arise in the collocation scheme, in more detail. To the notation we note that d_x^r denotes the approximation to $d_x(t_r)$, $d_x^{r,i}$ denotes the approximation to $d_x(\tau_{lr+i})$ and $d_u^{r,i}$ denotes the approximation to $d_u(\tau_{lr+i})$. ‡

$$d_x^{r,i} = d_x^r + h_r \sum_{k=1}^{l} \omega_{ik} \left[f_x[\tau_{lr+k}] d_x^{r,k} + f_u[\tau_{lr+k}] d_u^{r,k} + e[\tau_{lr+k}] \right],$$

$$i = 1,\ldots,l \qquad r = 0,1,\ldots,p-1, \qquad (6.1.2.4)$$

‡ In the collocation method we must also have $d_x(t_r +) = d_x(t_r -)$.

$$d_x^{r+1} = d_x^r + h_r \sum_{k=1}^{l} \bar{\omega}_k \left[f_x[\tau_{lr+k}] d_x^{r,k} + f_u[\tau_{lr+k}] d_u^{r,k} + e[\tau_{lr+k}] \right],$$

$$r = 0,1,\ldots,p-1. \qquad (6.1.2.5)$$

$$R_x^p[\tau_{lr+i}] d_x^{r,i} + R_u^p[\tau_{lr+i}] d_u^{r,i} = -R^p[\tau_{lr+i}] \quad i = 1,\ldots,l \quad r = 0,1,\ldots,p-1, \qquad (6.1.2.6)$$

$$D_x[t_0] d_x(t_0) = -D[t_0], \qquad (6.1.2.7)$$

$$N_x[t_r] d_x(t_r) = -N[t_r] \qquad r = 1,\ldots,p-1, \qquad (6.1.2.8)$$

$$E_x[t_p] d_x(t_p) = -E[t_p], \qquad (6.1.2.9)$$

where : $e[t] := f[t] - \dot{x}^i(t) \qquad 0 \leqslant t \leqslant T.$

Using the notation introduced below, the equations (6.1.2.4) - (6.1.2.9) may be written in the matrix notation :

$$Cd = b, \qquad (6.1.2.10)$$

where :

$$C := \begin{vmatrix} C_1 & & & \\ & C_2 & & \\ & & \ddots & \\ & & & C_p \end{vmatrix} \qquad (6.1.2.11)$$

The submatrices C_r $(r = 0,1,\ldots,p-1)$ consist of :

$$C_r := \begin{vmatrix} K_r & 0 & . & . & . & . & 0 \\ N_x[t_r] & 0 & . & . & . & . & 0 \\ 0 & R_x^p[\tau_{lr+1}] & R_u^p[\tau_{lr+1}] & 0 & . & . & 0 \\ . & 0 & 0 & . & . & . & . \\ . & . & . & . & . & . & . \\ 0 & 0 & 0 & . & . & . & R_u^p[\tau_{lr+l}] \\ I & G_{11r} & H_{11r} & G_{12r} & . & . & H_{1lr} \\ I & G_{21r} & H_{21r} & . & . & . & . \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ I & G_{l1r} & H_{l1r} & . & . & . & H_{llr} \\ I & \bar{G}_{1r} & \bar{H}_{1r} & . & . & . & \bar{H}_{lr} \end{vmatrix} \qquad (6.1.2.12)$$

$$C_p := \begin{vmatrix} -I \\ E_x[T] \end{vmatrix}, \qquad (6.1.2.13)$$

with :

$$K_r := \begin{cases} D_x[0] & r=0 \\ -I & r=1,\ldots,p-1 \end{cases} \qquad (6.1.2.14)$$

$$G_{ijr} := \begin{cases} h_r \omega_{ij} f_x[\tau_{lr+j}] & i \neq j \\ h_r \omega_{ij} f_x[\tau_{lr+j}] - I & i = j \end{cases} \quad i=1,\ldots l \quad j=1,\ldots l \quad r=0,1,\ldots p-1, \qquad (6.1.2.15)$$

$$\bar{G}_{ir} := h_r \bar{\omega}_i f_x[\tau_{lr+i}] \quad i=1,\ldots l \quad r=0,1,\ldots p-1, \qquad (6.1.2.16)$$

$$H_{ijr} := h_r \omega_{ij} f_u[\tau_{lr+i}] \quad i=1,\ldots l \quad j=1,\ldots l \quad r=0,1,\ldots p-1, \qquad (6.1.2.17)$$

$$\bar{H}_{ir} := h_r \bar{\omega}_i f_u[\tau_{lr+i}] \quad i=1,\ldots l \quad r=0,1,\ldots p-1, \qquad (6.1.2.18)$$

The vectors d and b have the following components :

$$d := \begin{bmatrix} d_x^0 \\ \cdot \\ \cdot \\ \cdot \\ d_x^r \\ d_x^{r,1} \\ d_u^{r,1} \\ \cdot \\ \cdot \\ d_x^{r,l} \\ d_u^{r,l} \\ \cdot \\ \cdot \\ \cdot \\ d_x^p \end{bmatrix} \qquad b := - \begin{bmatrix} D[0] \\ R^p[\tau_1] \\ \cdot \\ \cdot \\ \sum_{k=1}^{l} \bar{\omega}_k e[\tau_{l(r-1)+k}] \\ N[t_r] \\ R^p[\tau_{lr+1}] \\ \cdot \\ \cdot \\ R^p[\tau_{lr+l}] \\ \sum_{k=1}^{l} \omega_{1k} e[\tau_{lr+k}] \\ \cdot \\ \cdot \\ \sum_{k=1}^{l} \omega_{lk} e[\tau_{lr+k}] \\ \sum_{k=1}^{l} \bar{\omega}_k e[\tau_{lr+k}] \\ \cdot \\ \cdot \\ E[T] \end{bmatrix} \qquad (6.1.2.19)$$

The optimality conditions for problem (EQP/SCOCP) are treated in a similar way. As a notation we use $\lambda^{r,+}$ to denote the approximation to $\lambda(t_r+)$, $\lambda^{r,-}$ for the approximation to $\lambda(t_r-)$, $\lambda^{r,i}$ for the approximation to $\lambda(\tau_{lr+i})$ and $\eta_I^{r,i}$ for the approximation to $\eta_I(\tau_{lr+i})$. The collocation method applied to the optimality conditions for problem (EQP/SCOCP) yield the following equations :

$$\lambda^{r,i} = \lambda^{r,+} - h_r \sum_{k=1}^{l} \omega_{ik} (M_2[\tau_{lr+k}]d_x^{r,k} + f_x[\tau_{lr+k}]^T \lambda^{r,k} + M_3[\tau_{lr+k}]d_u^{r,k} + R_x^p[\tau_{lr+k}]^T \eta_I^{lr+k} + f_{0x}[\tau_{lr+k}]) \qquad i=1,\ldots,l \quad r=0,1,\ldots,p-1, \tag{6.1.2.20}$$

$$\lambda^{r+1,-} = \lambda^{r,+} - h_r \sum_{k=1}^{l} \bar{\omega}_k (M_2[\tau_{lr+k}]d_x^{r,k} + f_x[\tau_{lr+k}]^T \lambda^{r,k} + M_3[\tau_{lr+k}]d_u^{r,k} + R_x^p[\tau_{lr+k}]^T \eta_I^{lr+k} + f_{0x}[\tau_{lr+k}]) \qquad r=0,1,\ldots,p-1, \tag{6.1.2.21}$$

$$M_3[\tau_{lr+i}]^T d_x^{r,i} + f_u[\tau_{lr+i}]^T \lambda^{r,i} + M_4[\tau_{lr+i}]d_u^{lr+i} + R_u^T[\tau_{lr+i}]^T \eta_I^{lr+i} = -f_{0u}[\tau_{lr+i}]^T \qquad i=1,\ldots,l \quad r=0,1,\ldots,p-1, \tag{6.1.2.22}$$

$$M_1 d_x^0 + \lambda^{0,+} + D_x[0]^T \sigma = -h_{0x}[0]^T. \tag{6.1.2.23}$$

$$\lambda^{r,+} = \lambda^{r,-} - N_x[t_r]^T \chi_r \qquad r=1,\ldots,p-1, \tag{6.1.2.24}$$

$$M_5 d_x^p - \lambda^{p,-} + E_x[T]^T \mu = -g_{0x}[T]^T. \tag{6.1.2.25}$$

To equations (6.1.2.23) and (6.1.2.25) we note that $\lambda^{0,+}$ denotes the approximation to $\lambda(0)$ and $\lambda^{p,-}$ denotes the approximation to $\lambda(T)$.

Now the variables $\zeta_{r,k}$ and θ_{lr+i} are introduced as :

$$\sum_{k=1}^{l} \frac{\omega_{ki}}{\bar{\omega}_i} \zeta_{r,k} := \lambda^{r,i} - \lambda^{r+1,-} \qquad r=0,1,\ldots,p-1, \tag{6.1.2.26}$$

$$\theta_{lr+i} := h_r \bar{\omega}_i \eta_I^{lr+i} \qquad i=1,\ldots,l \quad r=0,1,\ldots,p-1. \tag{6.1.2.27}$$

Equations (6.1.2.20) - (6.1.2.25) can be transformed into the form :

$$Md + C^T \zeta = -c, \tag{6.1.2.28}$$

provided the weights ω_{ij} and $\bar{\omega}_i$ satisfy the condition :

$$\frac{\omega_{ij}}{\bar{\omega}_j} + \frac{\omega_{ji}}{\bar{\omega}_i} = 1 \qquad i=1,\ldots,l \quad j=1,\ldots,l. \tag{6.1.2.29}$$

The matrix M has the following block structure :

$$M := \begin{vmatrix} M_1 & & & & & & \\ & M^{0,1} & & & & & \\ & & M^{0,l} & & & & \\ & & & 0 & & & \\ & & & & M^{1,1} & & \\ & & & & & \ddots & \\ & & & & & & M^{p-1,l} \\ & & & & & & & M_5 \end{vmatrix} \qquad (6.1.2.30)$$

with

$$M^{r,i} := h_r \bar{\omega}_i \begin{bmatrix} M_2[\tau_{lr+i}] & M_3[\tau_{lr+i}] \\ M_3[\tau_{lr+i}]^T & M_4[\tau_{lr+i}] \end{bmatrix}. \qquad (6.1.2.31)$$

The components of the vectors ζ and c are:

$$\zeta := \begin{vmatrix} \sigma \\ \theta_1 \\ \vdots \\ \lambda^{r,-} \\ \chi_r \\ \theta_{lr+1} \\ \vdots \\ \theta_{lr+l} \\ \zeta_{r,1} \\ \vdots \\ \zeta_{r,l} \\ \lambda^{r+1,-} \\ \vdots \\ \lambda^{p,-} \\ \mu \end{vmatrix} \qquad c := \begin{vmatrix} h_{0x}[0]^T \\ h_0 \bar{\omega}_1 f_{0x}[\tau_1]^T \\ \vdots \\ 0 \\ h_r \bar{\omega}_1 f_{0x}[\tau_{lr+1}]^T \\ h_r \bar{\omega}_1 f_{0u}[\tau_{lr+1}]^T \\ \vdots \\ h_r \bar{\omega}_l f_{0x}[\tau_{lr+l}]^T \\ h_r \bar{\omega}_l f_{0u}[\tau_{lr+l}]^T \\ 0 \\ \vdots \\ g_{0x}[T]^T \end{vmatrix} \qquad (6.1.2.32)$$

We note that the transformation of (6.1.2.20) - (6.1.2.25) into (6.1.2.28) and vice versa is somewhat lengthy and essentially follows similar lines as the proof of Theorem 2.1 of Weiss (1974). Condition (6.1.2.29) has been verified for the case that the points ρ_i are the Gauss points.

The full set of linear equations to be solved, in order to obtain a numerical approximation for the solution of the linear multipoint boundary value problem can thus be transformed into :

$$\begin{bmatrix} M & C^T \\ C & 0 \end{bmatrix} \begin{bmatrix} d \\ \zeta \end{bmatrix} = \begin{bmatrix} -c \\ b \end{bmatrix} \tag{6.1.2.33}$$

which consistute precisely the first order necessary conditions for optimality for the following quadratic programming problem :

$$\underset{d}{\textit{Minimize}} \quad c^T d + \tfrac{1}{2} d^T M d \,, \tag{6.1.2.34}$$

$$\textit{subject to} : Cd = b \,, \tag{6.1.2.35}$$

provided the matrix M is positive definite on the null space of the matrix C. This shows in fact that the solution of the set of linear equations, which follows from the collocation method applied to the linear multipoint boundary value problem, which was obtained from the combination of the constraints and the optimality conditions of problem (EQP/SCOCP), is essentially the same as the solution of a certain quadratic programming problem which can be obtained as a certain finite-dimensional approximation to problem (EQP/SCOCP).

We note that when the points ρ_i are the Lobatto points, then a similar transformation seems no longer possible, which argues in favour of the use of Gauss points, because in this case it is possible to use the special structure of (6.1.2.33) in the numerical solution of the set of linear equations.

6.2. Numerical solution of the collocation scheme.

In this section the numerical solution of the collocation scheme will be considered. From the previous section we recall that the collocation scheme allows the following compact formulation :

$$\begin{bmatrix} M & C^T \\ C & 0 \end{bmatrix} \begin{bmatrix} d \\ \zeta \end{bmatrix} = \begin{bmatrix} -c \\ b \end{bmatrix}. \tag{6.2.1}$$

where the matrices C and M are sparse and banded. When M is semi-definite then this system is regular if and only if both the submatrices $(M \; C^T)$ and C have full row rank. Throughout this section we shall assume that at least the matrix C has full row rank.

As a notation we shall use $\bar{n}$ as the dimension of the vectors d and c, and $\bar{m}$ as the dimension of the vectors ζ and b. The matrices C and M are thus respectively $\bar{m} \times \bar{n}$ and $\bar{n} \times \bar{n}$ matrices.

6.2.1. Consideration of various alternative implementations.

We shall first consider three alternatives for the numerical solution of the system of linear equations (6.2.1) individually.

Method 1 : Direct solution of the collocation scheme.

The left hand side of (6.2.1) contains a symmetric indefinite $(\bar{n}+\bar{m})\times(\bar{n}+\bar{m})$ matrix. The matrices C and M are both banded. Using suitable column and row permutations, the matrix

$$\begin{bmatrix} M & C^T \\ C & 0 \end{bmatrix},$$

can be transformed into a banded (symmetric indefinite) system. † The resulting banded system may be solved by determination of a suitable factorization of the matrix, making use of its sparsity and symmetry, followed by the solution of a number of triangular systems (cf. Golub et al. (1983), p.100).

We note that for the factorization the submatrix M need not be invertible. (As an example consider the special case of a linear program, i.e. $M=0$ and $\bar{m}=\bar{n}$.)

Method 2 : Range space methods.

When the matrix M is invertible, another solution procedure is possible. System (6.2.1) then yields :

$$\hat{d} = -M^{-1}(c + C^T\hat{\xi}), \qquad (6.2.1.1)$$

and

$$(CM^{-1}C^T)\hat{\xi} = -(CM^{-1}c + b). \qquad (6.2.1.2)$$

where $\hat{d}$ and $\hat{\xi}$ are used to denote solutions of system (6.2.1).

If the matrix C is of full row rank, then also the left hand side of (6.2.1.2) is invertible, and hence $\hat{\xi}$ can be obtained as

$$\hat{\xi} = -(CM^{-1}C^T)^{-1}(CM^{-1}c + b). \qquad (6.2.1.3)$$

Combination of (6.2.1.3) and (6.2.1.1) yields :

$$\hat{d} = -(I+C^T(CM^{-1}C^T)^{-1}C)M^{-1}c - C^T(CM^{-1}C^T)^{-1}b. \qquad (6.2.1.4)$$

The method requires the determination of suitable factorizations of the matrices M and $(CM^{-1}C^T)$. Once these factorizations are determined, (6.2.1.3) and (6.2.1.4) can readily be solved. Because in the present case the matrix M is not invertible, this method is not applicable for the solution of the collocation scheme. (The matrix M is a block diagonal matrix with a number of zero blocks on the diagnonal, cf. equation (6.1.2.30)).

Method 3 : Null space methods.

A third alternative to the solution of the system (6.2.1) is to split the solution vector d into two parts, i.e.

$$d = d_R + d_N, \qquad (6.2.1.5)$$

where d_R is the component of d in the range space of the matrix C^T such that

† Essentially this yields a system similar to the one given by de Jong et al. (1985), who considered a implementation of the collocation method based on an other parameterization scheme.

$$Cd_R = b, \tag{6.2.1.6}$$

and d_N is the component in the null space of the matrix C, i.e.

$$Cd_N = 0. \tag{6.2.1.7}$$

Let Y be an $\bar{n}\times\bar{m}$ matrix whose columns are a base for the range space of the matrix C^T and Z an $\bar{n}\times(\bar{n}-\bar{m})$ matrix whose columns are a base for the null space of the matrix C, i.e. $Y^TZ=0$ and $CZ=0$, then d, as any vector $d\in\mathbb{R}^{\bar{n}}$, can also be written as :

$$d = Yd_y + Zd_z = d_R + d_N, \tag{6.2.1.8}$$

with : $d_R = Yd_y$,
$d_N = Zd_z$.

If the matrix C has full row rank, then the rows of the matrix C and the columns of the matrix Y are both a base for the range space of the matrix C^T, so the matrix (CY) is regular. Hence the range space solution part d_R can uniquely be determined from

$$(CY)d_y = b. \tag{6.2.1.9}$$

Combination of the upperpart of equation (6.2.1) with (6.2.1.8) gives :

$$MZd_z + C^T\zeta = -c - MYd_y, \tag{6.2.1.10}$$

and premultiplication with Z^T yields :

$$(Z^TMZ)d_z = -Z^Tc - Z^TMYd_y. \tag{6.2.1.11}$$

When the matrix (Z^TMZ) is regular, then a unique null space solution component d_z will exist.

The Lagrange multipliers $\hat{\zeta}$ may be obtained using the upperpart of (6.2.1) premultiplied by Y^T, i.e.

$$Y^TM\hat{d} + Y^TC^T\hat{\zeta} = -Y^Tc, \tag{6.2.1.12}$$

or, equivalently

$$(CY)^T\hat{\zeta} = -Y^T(c + M\hat{d}). \tag{6.2.1.13}$$

Observing that (CY) is regular yields that (6.2.1.13) can be solved.

Obviously, a practical implementation of the Null space method requires the determination of the matrices Y and Z. We shall mention two alternatives.

Let the matrix C be partitioned such that

$$C = [B\ S]. \tag{6.2.1.14}$$

where B is an $(\bar{m}\times\bar{m})$ regular matrix and S an $\bar{m}\times(\bar{n}-\bar{m})$ matrix. Then Y and Z can be taken as :

$$Y = C^T, \tag{6.2.1.15}$$

$$Z = \begin{bmatrix} -B^{-1}S \\ I \end{bmatrix}. \tag{6.2.1.16}$$

Using (6.2.1.14), (6.2.1.15) and (6.2.1.16) one easily verifies that

$$CY = CC^T, \tag{6.2.1.17}$$

and

$$CZ = 0 \tag{6.2.1.18}$$

The method based on this choice is called the *Null space method based on variable reduction*.

An alternative representation is based on the LQ-factorization of the matrix C, i.e.

$$C = [L\ 0]Q^T, \tag{6.2.1.19}$$

where L is an $(\bar{m}\times\bar{m})$ regular lowertriangular matrix and Q an $(\bar{n}\times\bar{n})$ orthogonal matrix.

If the matrices Y and Z are respectively chosen as the first $\bar{m}$ and last $\bar{n}-\bar{m}$ columns of Q, i.e.

$$Q = [Y\ Z], \tag{6.2.1.20}$$

then

$$CY = L, \quad C^T = YL^T, \tag{6.2.1.21}$$

and

$$CZ = 0. \tag{6.2.1.22}$$

Because Q is an orthogonal matrix the matrices Y and Z satisfy :

$$Y^TY = I_{\bar{m}}, \tag{6.2.1.23}$$

$$Y^TZ = 0, \tag{6.2.1.24}$$

$$Z^TZ = I_{\bar{n}-\bar{m}}, \tag{6.2.1.25}$$

where $I_{\bar{m}}$ and $I_{\bar{n}-\bar{m}}$ denote the $\bar{m}\times\bar{m}$ and $(\bar{n}-\bar{m})\times(\bar{n}-\bar{m})$ identy matrix.

The *Null space method based on LQ-factorization of the matrix C* requires thus the solution of :

$$Ld_y = b, \tag{6.2.1.26}$$

$$(Z^TMZ)d_z = -Z^T(c + MY\hat{d}_y), \tag{6.2.1.27}$$

$$d = Z\hat{d}_z + Y\hat{d}_y, \tag{6.2.1.28}$$

$$L^T\zeta = -Y^T(c + M\hat{d}). \tag{6.2.1.29}$$

Considering the various methods for the solution of the collocation scheme mentioned above, we notice that in general, Null space methods have an advantage over the direct solution of the system (6.2.1) (i.e. method 1), because instead of the solution of an $(\bar{n}+\bar{m})\times(\bar{n}+\bar{m})$ system, these methods require the solution of two systems of smaller dimension, i.e. $\bar{m}\times\bar{m}$ and $(\bar{n}-\bar{m})\times(\bar{n}-\bar{m})$. From (6.2.1.26) - (6.2.1.29) we recall that with the Null space method the computation of the solution $\hat{d}$ and of the Lagrange multiplier vector $\hat{\zeta}$ are done separately. Because these quantities are used in different steps of Algorithm 5.8, they need never be computed unnecessary, as is the case with the first method, i.e. the direct solution of (6.2.1).

The implementation of the Null space method was done using the LQ-factorization of the matrix C. This choice was made in view of the condition of the matrix $Z^T MZ$. This condition is of great importance for the amount of effort necessary for the solution of system (6.2.1.27), because this system is solved using an iterative method. The motivation that in general, this is never a bad choice is based on the following reasoning. Suppose the Null space method is implemented with an arbitrary matrix $\tilde{Z}$, then $\tilde{Z}$ can be written as :

$$\tilde{Z} = ZW,$$

where Z is the matrix consisting of the last $\bar{n}-\bar{m}$ columns of the matrix Q and W is an $(\bar{n}-\bar{m})\times(\bar{n}-\bar{m})$ regular scaling matrix. It may be verified that the condition number of the matrix $\tilde{Z}^T M\tilde{Z}$ satisfies : †

$$\kappa(\tilde{Z}^T M\tilde{Z}) \leqslant \kappa(Z^T MZ)\kappa^2(W),$$

which indicates that the condition number of the matrix W may destroy the condition of the matrix $\tilde{Z}^T M\tilde{Z}$ compared to the condition number of the matrix $Z^T MZ$.

We also note that a much stronger motivation for the use of the LQ-factorization would have been possible when the matrix M would have been positive definite (cf. Gill et al. (1974b)). For in that case it is possible to show that the LQ-factorization is the optimal choice with respect to the minimization of the condition number of the matrix $\tilde{Z}^T MZ$.

6.2.2. Numerical solution of the collocation scheme by means of the Null space method based on LQ-factorization.

The equations involved in the numerical solution of the collocation scheme by means of the Null space method are recapitulated below :

$$C = [L\ 0]Q^T, \tag{6.2.2.1}$$

$$Q = [Y\ Z], \tag{6.2.2.2}$$

$$Ld_y = b, \tag{6.2.2.3}$$

$$(Z^T MZ)d_z = -Z^T(c + MY\hat{d}_y), \tag{6.2.2.4}$$

$$d = Y\hat{d}_y + Z\hat{d}_z, \tag{6.2.2.5}$$

$$L^T\zeta = -Y^T(c + M\hat{d}). \tag{6.2.2.6}$$

Systems (6.2.2.3) and (6.2.2.6) are respectively lowertriangular and uppertriangular systems. Their solution is quite standard and is done respectively by forward elimination and back substitution (e.g. cf. Golub et al. (1983), p.52). The two major problems in the solution of the collocation scheme via (6.2.2.1) - (6.2.2.6) are the LQ-factorization of the matrix C and the solution of system (6.2.2.4).

The LQ-factorization of the matrix C is done by means of Householder transformations. Because the matrix C is large and sparse (banded), it is advantageous to modify the usual orthogonalization procedure for dense matrices (e.g. cf. Golub et al. (1983) or Lawson et al. (1974)) following the ideas of Reid (1967). The LQ-factorization procedure yields the

† $\kappa(W)$ denotes the condition number of the matrix W, i.e. $\|W\|\|W^{-1}\|$, where we use the 2-norm for the matrix norms.

matrix L explicitly, which is beside lowertriangular also banded, and the matrix Q implicitly, as a product of Householder transformations. The vectors which define these Householder transformations require essentially the same amount of storage as the (sparse) matrix C. It can be shown that the matrix Q is, in general, a dense matrix, and hence it is not efficient to form the matrix Q explicit. A more detailed description of the LQ-factorization process is given in Appendix D.

As a result of the fact that the matrix Q is available in factored form, i.e. as a product of Householder transformations, it is possible to compute matrix-vector products of the form Qd and $Q^T d$. Hence Yd_y, Zd_z, $Y^T d$ and $Z^T d$ can also be computed because :

$$Q\begin{bmatrix} d_y \\ 0 \end{bmatrix} = Yd_y, \tag{6.2.2.7}$$

$$Q\begin{bmatrix} 0 \\ d_z \end{bmatrix} = Zd_z, \tag{6.2.2.8}$$

$$Q^T d = \begin{bmatrix} Y^T d \\ Z^T d \end{bmatrix}. \tag{6.2.2.9}$$

The product $(Z^T MZ)d_z$ can thus be computed as :

$$Z^T \cdot (M \cdot (Z \cdot d_z)). \tag{6.2.2.10}$$

To form the matrix $Z^T MZ$ explicitly, the columns may be generated by computation of the vectors

$$Z^T MZe_j \qquad j = 1,\ldots,\bar{n}-\bar{m}, \tag{6.2.2.11}$$

where e_j is the jth column of the $(\bar{n}-\bar{m})\times(\bar{n}-\bar{m})$ identy matrix, i.e. $I_{\bar{n}-\bar{m}}$. The product (6.2.2.10) is thus to be evaluated $(\bar{n}-\bar{m})$ times. When the matrix $Z^T MZ$ is positive definite (which is true in most of the cases considered here), then the solution of equation (6.2.2.4) can be obtained using Cholesky factorization. The numerical effort to solve (6.2.2.4) after $Z^T MZ$ has been formed is thus approximately $(\bar{n}-\bar{m})^3/6$ flops. † An alternative way is to solve (6.2.2.4) by means of an iterative method. In many cases, a suitable iterative method for the solution of a large sparse system is the linear conjugate gradient method (cf. Golub et al. (1983)). This method requires in most cases less than $\bar{n}-\bar{m}$ iterations. Each iteration involves one evaluation of the matrix-vector product (6.2.2.10) and approximately $5(\bar{n}-\bar{m})$ flops. Thus the linear conjugate gradient method requires only $5(\bar{n}-\bar{m})^2$ flops, in addition to at most $\bar{n}-\bar{m}$ evaluations of a matrix-vector product (6.2.2.10). This clearly argues in favour of the solution of (6.2.2.4) by means of the linear conjugate gradient method.

An alternative motivation for the use of an iterative method follows from the consideration of the dimensions of the matrix $Z^T MZ$, which are equal to the dimension of the null space of the matrix C. ‡ An upperbound for the dimension of the null space of the matrix C is obtained from the case that the working sets W_j of problem (EQP/SCOCP) are empty (i.e. no active state constraints). In this case the row dimension of the matrix C is (cf.

† flops is an abbreviation of floating point operations.
‡ We note that because the matrix Z consists of columns of the dense matrix Q, the matrix $Z^T MZ$ will also be dense.

Section 6.1.2) :

$$\bar{m} = c + n(l+1)p + q.$$

The dimension of the vector d is :

$$\bar{n} = n(l+1)p + mlp + n,$$

and hence, the dimension of the null space of C is in this case :

$$\bar{n} - \bar{m} = mlp + n - c - q.$$

Practical cases are $l \geqslant 2$ (at least two collocation points per grid interval), $p \geqslant 25$ (at least 25 grid points), $m \geqslant 1$ (at least one control variable), $n - c - q = 0$ (e.g. $c = n$, $q = 0$, initial state completely specified and terminal state free). This yields as an optimistic upperbound for the dimension of the null space 50 and hence $Z^T MZ$ can be a dense 50×50 matrix, which indicates that in 'normal' cases the matrix $Z^T MZ$ can be quite large.

The linear conjugate gradient method is recapitulated below, from Golub et al. (1983), for the solution of the equation

$$Gp = -g. \qquad (6.2.2.12)$$

Algorithm 6.1 (p,g,G,ϵ)

Initialize

$p_0 := 0$
$r_0 := -g$
$i := 1$
$\beta_1 := 0$

Do linear conjugate gradient steps untill the required accuracy is achieved.

while $\|r_{i-1}\| / \|g\| > \epsilon$

do

$\beta_i := r_{i-1}^T r_{i-1} / r_{i-2}^T r_{i-2}$ $\quad (i > 1)$
$u_i := r_{i-1} + \beta_i u_{i-1}$ $\quad (\beta_1 = 0)$
$\alpha_i := r_{i-1}^T r_{i-1} / (u_i^T G u_i)$
$p_i := p_{i-1} + \alpha_i u_i$
$r_i := r_{i-1} - \alpha_i G u_i$
$i := i + 1$

od

A formal motivation for this algorithm may be found in the unconstrained minimization of the functional

$$\phi(p) := g^T p + \tfrac{1}{2} p^T G p. \qquad (6.2.2.13)$$

using directions of search u_i and step sizes α_i. The vector r_i satisfies :

$$r_i = -g - Gp_i. \qquad (6.2.2.14)$$

and

$$\nabla \phi(p_i) = g + Gp_i = r_i. \qquad (6.2.2.15)$$

The linear conjugate gradient algorithm has at least a linear rate of convergence, with

convergence factor

$$\frac{\sqrt{\kappa(G)} - 1}{\sqrt{\kappa(G)} + 1}, \tag{6.2.2.16}$$

where $\kappa(G) := \|G\| \cdot \|G^{-1}\|$ is the condition number of (6.2.2.12).

In order to obtain satisfactory convergence properties, the condition number must be close to unity. This leads to the consideration of scaling methods in order to improve the condition of (6.2.2.12). The development of scaling methods is difficult because the matrix G is not explicitly available. Hence the application of the usual scaling methods for iterative methods, seems not possible. Fortunately, the Null space method allows the simultaneous application of the two strategies outlined below. Experiments with the implementation of the method show that these strategies do in fact yield a significant improvement with respect to the amount of numerical effort.

Scaling of the collocation scheme.

The collocation scheme is transformed into :

$$D_1^T M D_1 q + D_1^T C^T \zeta = - D_1^T c, \tag{6.2.2.17}$$

$$C D_1 q = b, \tag{6.2.2.18}$$

where D_1 is a regular scaling matrix. The solution of the collocation scheme using the Null space method is in this case computed from

$$C D_1 = [L \; 0] Q^T, \tag{6.2.2.19}$$

$$Q = [Y \; Z], \tag{6.2.2.20}$$

$$L q_y = b, \tag{6.2.2.21}$$

$$(Z^T D_1^T M D_1 Z) q_z = - Z^T D_1^T (c + M Y \hat{q}_y), \tag{6.2.2.22}$$

$$d = D_1 Y \hat{q}_y + D_1 Z \hat{q}_z, \tag{6.2.2.23}$$

$$L^T \zeta = - Y^T D_1^T (c + M \hat{d}). \tag{6.2.2.24}$$

The scaling matrix D_1 must be chosen in a way that

$$\kappa(Z^T D_1^T M D_1 Z) \; \textit{is small.} \tag{6.2.2.25}$$

Unfortunately, there is no general rule that can be used for the choice of the scaling method. A method that works well in many practical cases is to choose D_1 as a diagonal matrix with elements such that the diagonal elements of the matrix $D_1 M D_1$ are all equal to one. In our case however, the diagonal elements of M can also be negative or zero. Therefore the diagonal elements of D_1 are chosen to be :

$$(D_1)_{ii} := \frac{1}{max\{\epsilon_D, \sqrt{|M_{ii}|}\}}, \tag{6.2.2.26}$$

where ϵ_D is a small quantity.

Preconditioning of the linear conjugate gradient algorithm.

A second scaling strategy is based on preconditioning of the linear conjugate gradient method, which means that the so-called preconditioned equation

$$D_2^{-1}GD_2^{-1}\bar{q}_z = -D_2^{-1}g. \qquad (6.2.2.27)$$

is solved rather than equation (6.2.2.22) (cf. Golub et al. (1983)). Here D_2 is a nonsingular, symmetric scaling matrix. Once the solution of (6.2.2.27) $\bar{q}_z$ is determined, the solution of (6.2.2.12) follows as :

$$\hat{q}_z = D_2^{-1}\bar{q}_z. \qquad (6.2.2.28)$$

With this preconditioning strategy, the linear conjugate gradient algotithm becomes :

Algorithm 6.2 (q,g,G,D_2,ϵ)

Initialize

$q_0 := 0$
$r_0 := -g$
$i := 1$
$\beta_1 := 0$

Do linear conjugate gradient steps untill the required accuracy is achieved.

while $\|r_{i-1}\|/\|g\| > \epsilon$

do

Solve $(D_2)^2 z_{i-1} = r_{i-1}$
$\beta_i := z_{i-1}^T r_{i-1}/z_{i-2}^T r_{i-2}$ $\quad (i>1)$
$u_i := z_{i-1} + \beta_i u_{i-1}$ $\quad (\beta_1 = 0)$
$\alpha_i := z_{i-1}^T r_{i-1}/(u_i^T G u_i)$
$q_i := q_{i-1} + \alpha_i u_i$
$r_i := r_{i-1} - \alpha_i G u_i$
$i := i + 1$

od

The main problem in making a specific choice for the matrix D_2 is again that the elements of the matrix G are not explicitly available, because the matrix G is only available in the factored form $G = Z^T D_1 M D_1 Z$. As with the previous strategy, the scaling matrix D_2 must be chosen so that

$$\kappa(D_2^{-1}GD_2^{-1}) \textit{ is small.} \qquad (6.2.2.29)$$

We adopted the strategy given by Nash (1984, 1985), who shows that the elements of the matrix G may be approximated using quasi-Newton updates of the matrix G. We note that, neglecting the influence of roundoff errors, the matrix G will follow from this update process after $\bar{n}-\bar{m}$ iterations of the linear conjugate gradient method. The quasi-Newton updates may be computed during the linear conjugate gradient method with very little effort, because most of the quantities used are already available, as is revealed by the update formula

$$B_0 := I \qquad (6.2.2.30)$$

$$B_{i+1} := B_i - \frac{r_{i-1}r_{i-1}^T}{u_i^T r_{i-1}} + \frac{(Gu_i)(Gu_i)^T}{u_i^T G u_i}. \qquad (6.2.2.31)$$

An important advantage of the form of the update formula (6.2.2.31) is that the elements of the quasi-Newton updates B_{i+1} can be computed individually and hence it is also

possible to compute the update only partly. In the implementation of the method this update scheme is used to obtain an approximation to the diagonal of the matrix G.

During one execution of the linear conjugate gradient method, an approximation of the diagnonal of the matrix G is developed, using (6.2.2.30) and (6.2.2.31). When the linear conjugate gradient method is called again, and there have been no modifications in the working set since the last call of this algorithm, then the approximation to the diagonal of the matrix G developed during the previous call is used as a preconditioner, i.e. as $(D_2)^2$. Otherwise this scaling strategy is not used.

We now turn to an other aspect of the solution of the collocation scheme. During the execution of Algorithm 5.8, the collocation scheme is solved in order to obtain a direction of search for the improvement $\hat{d}-\bar{d}$ of the current estimate $\bar{d}$ of the solution of problem (EIQP/SCOCP/Δ).
This improvement may be obtained directly as the solution of :

$$Lq_y = b - C\bar{d}, \tag{6.2.2.32}$$

$$(Z^T D_1 M D_1 Z)q_z = - Z^T D_1(c + M(\bar{d}+Y\hat{q}_y)), \tag{6.2.2.33}$$

$$\hat{d}-\bar{d} = D_1 Y\hat{q}_y + D_1 Z\hat{q}_z. \tag{6.2.2.34}$$

The advantage of the use of (6.2.2.32) - (6.2.2.34) is revealed by the situation

$$\hat{d} - \bar{d} = 0, \tag{6.2.2.35}$$

i.e. $\bar{d}$ is already the solution of the collocation scheme, which yields a direction of search of zero. In this case, the linear conjugate gradient algorithm will require no iterations at all, because the right hand side of (6.2.2.33) is zero.

Up till this point it was implicitly assumed that the matrix $Z^T D_1 M D_1 Z$ is always positive definite. Cases where the matrix $Z^T D_1 M D_1 Z$ is indefinite correspond to those cases where, similar to the case of finite-dimensional quadratic programming, the problem (EQP/SCOCP) has no bounded solution. In these cases it suffices that the direction of search in Algorithm 5.8 is a direction of negative curvature. When $Z^T D_1 M D_1 Z$ is indefinite, then it is likely (cf. Nash (1983)) that during the execution of the linear conjugate gradient algorithm, the vector u_i becomes a direction of negative curvature, i.e.

$$u_i^T G u_i = u_i^T (Z^T D_1 M D_1 Z) u_i < 0. \tag{6.2.2.36}$$

Because this quantity is already necessary in the linear conjugate gradient method, it is rather simple to detect this situation. In this case it may be advantageous to stop the linear conjugate gradient algorithm and to use u_i as a direction of search in Algorithm 5.8. Because if u_i is a direction of negative curvature, then so is $-u_i$, the algorithm can thus always be terminated with a vector q which satisfies :

$$g^T q < 0, \tag{6.2.2.37}$$

i.e. q is beside a direction of negative curvature also a direction of descent of the function (6.2.2.13).

The following lemma establishes that u_i is always the vector which satisfies (6.2.2.37).

Lemma 6.3 : *The vectors u_i determined by Algorithm 6.2 satisfy :*

$$g^T u_i < 0. \tag{6.2.2.38}$$

Proof : The vectors r_i and z_i satisfy (cf. Golub et al. (1983), p.374) :

$$r_j^T z_i = 0 \qquad i \neq j. \tag{6.2.2.39}$$

Now consider

$$g^T u_i = g^T(z_{i-1} + \beta u_{i-1}). \tag{6.2.2.40}$$

Because $r_0 = -g$ this yields :

$$g^T u_i = -r_0^T z_{i-1} + \beta_i g^T u_{i-1}. \tag{6.2.2.41}$$

Using (6.2.2.39) we obtain :

$$g^T u_i = \begin{cases} -r_0^T z_0 = -r_0^T (D_2)^{-2} r_0 & i = 1 \\ \beta_i g^T u_{i-1} & i > 1 \end{cases} \tag{6.2.2.42}$$

For $i = 1$ the result follows from the positive definiteness of D_2. For $i > 2$ the result follows from an induction argument, because $\beta_i > 0$ for all i.

□

6.3. Truncation errors of the collocation method.

This section is devoted to the estimation of the truncation errors which deteriorate the direction of search in the numerical implementation of the method.

The truncation errors associated with the solution of the collocation method are considered by de Boor et al. (1973) and Weiss (1974). To apply their results to the collocation method described in this chapter, we make use of the abstract notation of the linear multipoint boundary value problem of Section 6.1.1, i.e.

$$\dot{v}(t) = A_1[t]v(t) + B_1[t]w(t) + c_1[t] \qquad a.e.\ \ 0 \leqslant t \leqslant T. \tag{6.3.1}$$

$$0 = A_2[t]v(t) + B_2[t]w(t) + c_2[t] \qquad a.e.\ \ 0 \leqslant t \leqslant T. \tag{6.3.2}$$

$$K_0 v(0) + l_0 = 0, \tag{6.3.3}$$

$$K_j^+ v(\tilde{t}_j +) + K_j^- v(\tilde{t}_j -) + l_j = 0 \quad all\ j, \tag{6.3.4}$$

$$K_T v(T) + l_T = 0. \tag{6.3.5}$$

The time function $w(t)$ may be eliminated using (6.3.2), i.e.

$$w(t) = -B_2[t]^{-1}(A_2[t]v(t) + c_2[t]) \qquad a.e.\ \ 0 \leqslant t \leqslant T. \tag{6.3.6}$$

Combination with (6.3.1) yields :

$$\dot{v}(t) = (A_1[t] - B_1[t]B_2[t]^{-1}A_2[t])v(t) + (c_1[t] - B_1[t]B_2[t]^{-1}c_2[t]). \tag{6.3.7}$$

For the derivation of results on the accuracy of the numerical approximation obtained from the collocation method, it is assumed that the coefficients of the matrix $A_1[t] - B_1[t]B_2[t]^{-1}A_2[t]$ and the vector $c_1[t] - B_1[t]B_2[t]^{-1}c_2[t]$ are at least $(l+1)$-times continuously differentiable on the grid intervals (t_r, t_{r+1}). For simplicity we shall also assume that the grid points are uniformly distributed on $[0,T]$, i.e. $t_{r+1} - t_r = h = T/p$

$(r = 0,1,...p-1)$.

The exact solution of the linear multipoint boundary value problem will be denoted by $\hat{v}(t)$ and the solution obtained from the collocation method by $v(t)$. At grid points t_r where $v(t)$ is continuous, i.e. grid points that do not coincide with one of the points $\tilde{t}_j$ the following result holds for sufficiently small h (cf. de Boor et al. (1973) or Weiss (1974)) :

$$\| v(t_r) - \hat{v}(t_r) \| \leq C_r h^{2l}. \tag{6.3.8}$$

At points $\tilde{t}_j$ where $v(t)$ is possibly discontinuous, a result similar to (6.3.8) holds for both $v(\tilde{t}_j -)$ and $v(\tilde{t}_j +)$.

We note that both de Boor et al. (1973) and Weiss (1974) consider two point boundary value problems and assume that the right hand side of (6.3.7) is sufficiently smooth on the entire interval $[0,T]$. These results can be adapted to the present case of piecewise smooth coefficients following the approach of Keller (1969). The extension to multipoint boundary value problems is straightforward.

At the collocation points the numerical approximation to the solution obtained from the collocation method is less accurate compared to the accuracy of the numerical approximation at the grid points, i.e. for sufficiently small h :

$$\| v(\tau_{lr+i}) - \hat{v}(\tau_{lr+i}) \| \leq C_{r,i} h^{l+1} \qquad i = 1,...,l \quad r = 0,1,...p-1. \tag{6.3.9}$$

Numerical evidence led Souren (1986) to believe that the truncation errors in v have a maximum at the collocation points and hence :

$$\| v(t) - \hat{v}(t) \|_\infty \leq C_g h^{l+1}. \tag{6.3.10}$$

where $C_g = \max_{r,i} C_{r,i}$.

From (6.3.9) and (6.3.6) we obtain for the numerical approximation to the time functions $w(t)$ at the collocation points :

$$\| w(\tau_{lr+i}) - \hat{w}(\tau_{lr+i}) \| \leq D_{r,i} h^{l+1}. \tag{6.3.11}$$

For the derivation of the results stated above, it was assumed that the right hand side of the differential equation (6.3.7) is sufficiently smooth on the grid intervals (t_r, t_{r+1}). The actual structure of the linear multipoint boundary value problem given by (6.1.1.1) - (6.1.1.6) reveals that this condition is satisfied when the problem functions of problem (SCOCP) are sufficiently smooth and that, in addition, the number of components of the vector R^p must be constant on the grid intervals (t_r, t_{r+1}). This last condition is equivalent to the condition that all junction and contact points of all constraints must coincide with grid points t_r, i.e. constraints are to be taken active and inactive per entire grid interval. For state constraints this condition is automatically satisfied, as a result of the fact that at junction and contact points, also interior point conditions of the type (6.3.4) must be fulfilled. For mixed control state constraints the condition mentioned above is not automatically satisfied, because these constraints are (at least in the first stage of the method) taken active and inactive per collocation point and not per entire grid interval. Taking mixed control state constraints active (inactive) per entire grid interval would take with the collocation method the form of taking these constraints active (inactive) at all collocation points of the grid interval. In the first stage of the method (Algorithm 4.4)

the accuracy of the direction of search obtained is not very important and hence the mixed control state constraints may well be taken active and inactive per collocation point. (This simplifies the active set strategy for these constraints as mentioned in Section 5.2). In the second stage of the method, the accuracy of the direction of search is important and hence in this stage the mixed control state constraints are taken active and inactive per entire grid interval.

Based on the results given above the truncation errors of each of the time functions, i.e. $v(t)$ or $w(t)$ may be estimated numerically by assuming the following model for the approximation obtained from the collocation method :

$$\theta(t;h) = \hat{\theta}(t) + C(t)h^k + o(h^k), \tag{6.3.12}$$

where $\theta(t;h)$ denotes either one of the time functions $v(t)$ or $w(t)$ obtained with the collocation method as a numerical approximation to the solution of the linear multipoint boundary value problem with grid intervals of size h. Let $\theta(t;\alpha h)$ and $\theta(t;\beta h)$ be solutions to the linear multipoint boundary value problem with grid intervals of the size αh and βh $(0<\beta<\alpha<1)$, then using the solutions $\theta(t;h)$, $\theta(t;\alpha h)$ and $\theta(t;\beta h)$, the constant $C(t)$ and the order of the integration scheme k may be determined for each time point. Define :

$$\Gamma(t) := \frac{\theta(t;\alpha h)-\theta(t;\beta h)}{\theta(t;h)-\theta(t;\alpha h)}. \tag{6.3.13}$$

The order of the integration scheme k may be obtained as the solution of the equation

$$(1+\Gamma(t))\alpha^k - \beta^k = \Gamma(t), \tag{6.3.14}$$

the constant $C(t)$ follows as :

$$C(t) = \frac{\theta(t;h)-\theta(t;\alpha h)}{(1+\alpha^k)h^k}. \tag{6.3.15}$$

The model (6.3.12) implies that, if h is small enough, either :

$$\theta(t;h) > \theta(t;\alpha h) > \theta(t;\beta h), \tag{6.3.16}$$

or

$$\theta(t;h) < \theta(t;\alpha h) < \theta(t;\beta h), \tag{6.3.17}$$

Because the numerical solutions of the linear multipoint boundary value problems contain beside truncation errors, also roundoff errors, both the conditions (6.3.16) - (6.3.17) may fail to hold. Hence the constant $C(t)$ and the order k can only be determined when $\Gamma(t)>0$ and when $|\theta(t;h)-\theta(t;\alpha h)|$ and $|\theta(t;\alpha h)-\theta(t;\beta h)|$ have significant digits.

An alternative for the estimation of the truncation errors is to make use of the a priori knowledge on the value of the order of the integration method k, in which case only the solutions $\theta(t;h)$ and $\theta(t;\alpha h)$ of the linear multipoint boundary value problem are needed. A drawback to this alternative is that we have no information on the validity of the estimates obtained.

A drawback of the procedure outlined above is that the solutions $\theta(t;\alpha h)$ and $\theta(t;\beta h)$ the linear multipoint boundary value problem must be solved using a 'finer' grid (e.g. $\alpha=\frac{1}{2}$ and $\beta=\frac{1}{3}$). In cases where it is sufficient to have only a rough estimate for the truncation error, a similar procedure can be used with $\beta>\alpha>1$.

7. Numerical solution of some problems.

In this chapter the numerical results of the solution of some example problems will be given. First in Section 7.1 the instationary dolphin flight of a glider, subject to various constraints, is considered. The unconstrained instationary dolphin flight has recently been a quite popular benchmark for testing numerical methods for the solution of optimal control problems (cf. de Jong (1985), Lorentz (1985)). Next in Section 7.2 the reentry manoever of an Apollo capsule is considered. This problem† is much more difficult to solve as a result of the fact that the solution trajectory of the differential equations depend in an extremely sensitive way on the initial data. We quote Stoer et al. (1980), p. 496 :

> "The solution has moving singularities which lie in an immediate neighborhood of the initial point of integration. This sensitivity is a consequence of the effect of atmospheric forces, and the physical interpretation of the singularity is a 'crash' of the capsule or a 'hurling' back into space. As can be shown by an a posteriori calculation, there exist differentiable solutions of the optimal control problem for an extremely narrow domain of boundary data, which is the mathematical formulation of the danger involved in the reentry manoever."

Finally in Section 7.3 the optimal control of two (dynamically) independent servo systems along a prespecified geometric path is considered. The optimal control is subject to both constraints on the accelerations and the velocities of the servos. The major difficulty with these problems is the determination of the correct structure of the solution ‡.

7.1. Instationary dolphin flight of a glider.

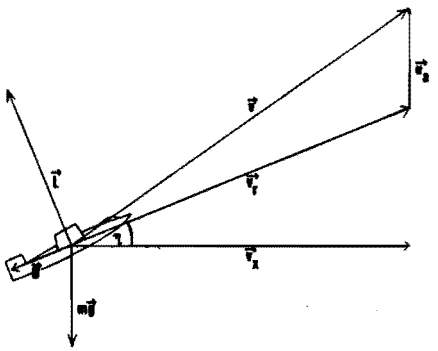
7.1.1. Statement and solution of the unconstrained problem.

A glider, which is flying through an area with a variable vertical velocity of air (a thermal), is modelled as a point mass m that experiences a gravity force mg, a lift force L perpendicular to the velocity relative to the air, v_r and a drag force D opposite to the velocity v_r. The variables of the problem are depicted in Figures 7.1 and 7.2 (for more details see de Jong (1985) or Lorentz (1985)). The relative velocity vector makes an angle η relative to the horizontal plane. The motion of the glider is restricted to the vertical plane. The vertical moving air mass (the thermal) is assumed to have a horizontal extent of $5R$. The upward wind velocity u_a is given as a function of the horizontal distance x, from the start of the flight at the left end of the thermal as :

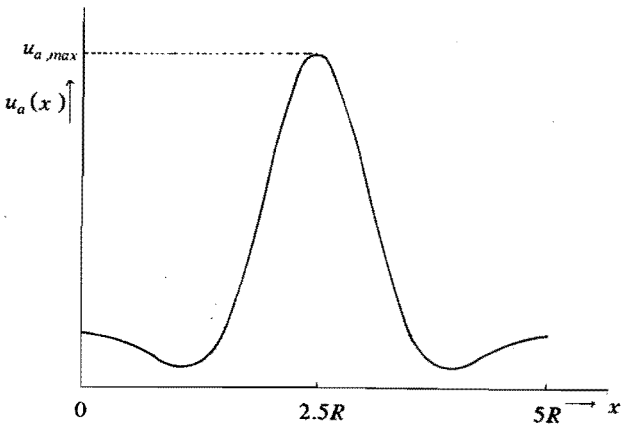
$$u_a(x) = u_{a\,max}\left[1-\left[\frac{x}{R}-2.5\right]^2\right]e^{-\left|\frac{x}{R}-2.5\right|^2} \qquad \text{for all } 0 \leqslant x \leqslant 5R. \qquad (7.1.1.1)$$

The objective of the problem is to control the glider from $x=0$ to $x=5R$, such that the 'relative' flight time is minimal, where the relative time is defined as the sum of the time required to fly from $x=0$ to $x=5R$ and the time necessary to regain the lost altitude at a specified constant rate of climb z.

† This problem was suggested as a benchmark by Dr. K.H. Well of DFVLR.
‡ The sequence in which the different constraints are active and inactive.



Velocities, forces and angle.
Figure 7.1



Upward wind velocity.
Figure 7.2

The mathematical formulation of the optimal control problem is :

$$\underset{u}{\text{minimize}} \quad \int_0^{5R} \frac{1}{v_x}\left(1-\frac{v_y}{z}\right) dx\,. \tag{7.1.1.2}$$

$$\text{subject to :} \quad \frac{dv_x}{dx} = \left(-L\sin\eta - D\cos\eta\right)/mv_x \qquad 0 \leqslant x \leqslant 5R\,, \tag{7.1.1.3}$$

$$\frac{dv_y}{dx} = \left(L\cos\eta - D\cos\eta - mg\right)/mv_x \qquad 0 \leqslant x \leqslant 5R\,, \tag{7.1.1.4}$$

$$v_x(0) = v_x(T) = v_{x,Mc}\,. \tag{7.1.1.5}$$

$$v_y(0) = v_y(T) = v_{y,Mc}\,. \tag{7.1.1.6}$$

$$where: L = \frac{\rho S v_r^2}{2} u, \qquad (7.1.1.7)$$

$$D = \frac{\rho S v_r^2}{2} \sum_{i=0}^{4} u^i, \qquad (7.1.1.8)$$

$$v_r = \sqrt{v_x^2 + (v_y - u_a)^2}, \qquad (7.1.1.9)$$

$$\eta = arctan \frac{v_y - u_a}{v_x}. \qquad (7.1.1.10)$$

In the formulation above the distance x is used as independent variable, which is derived from the formulation based on the time t by making use of :

$$\frac{dv_x}{dt} = \frac{dv_x}{dx}\frac{dx}{dt} = \frac{dv_x}{dx} v_x, \qquad (7.1.1.11)$$

$$\frac{dv_y}{dt} = \frac{dv_y}{dx}\frac{dx}{dt} = \frac{dv_y}{dx} v_x. \qquad (7.1.1.12)$$

The problem (7.1.1.2) - (7.1.1.10) was solved using the following constants :

$\rho = 1.13\ kg/m^3$	$k_0 = 0.0118$	$u_{a,max} = 5\ m/s$
$g = 9.80665\ m/s^2$	$k_1 = -0.0254$	$v_{x,Mc} = 41.631\ m/s$
$m = 346.5\ kg$	$k_2 = 0.0770$	$v_{y,Mc} = -1.344\ m/s$
$S = 10.5\ m^2$	$k_3 = -0.0540$	$z = 2\ m/s$
$R = 100\ m$	$k_4 = 0.0166$	

Table 7.1 : constants used in the numerical example.

The starting trajectory for the numerical solution procedure, was given by :

$$v_x(x) = v_{x,Mc} \qquad 0 \leqslant x \leqslant 5R, \qquad (7.1.1.13)$$

$$v_y(x) = v_{y,Mc} \qquad 0 \leqslant x \leqslant 5R, \qquad (7.1.1.14)$$

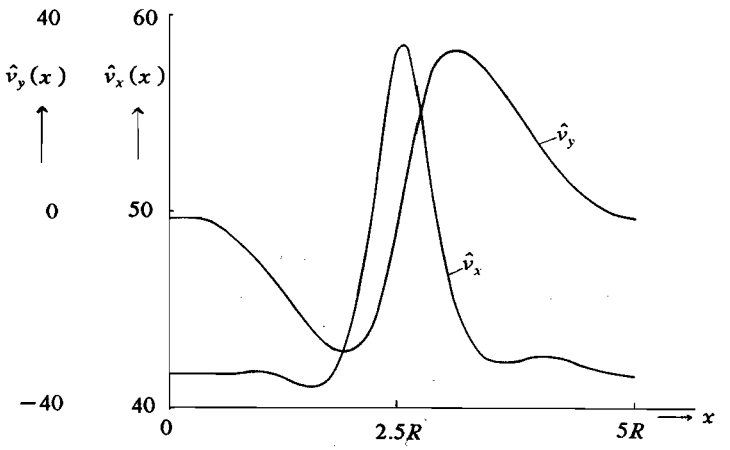
$$u(x) = 0.3041737 \qquad 0 \leqslant x \leqslant 5R. \qquad (7.1.1.15)$$

This trajectory is in fact a flight along a straight line from $x = 0$ to $x = 5R$, which is obviously the solution when there is no thermal present (i.e. $u_{a,max} = 0$).

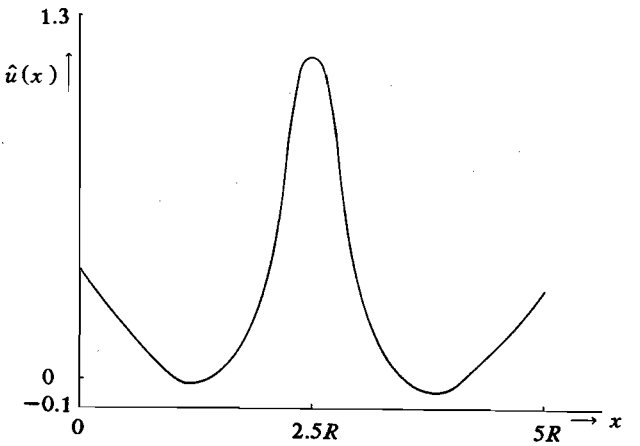
For the numerical solution of the problem an equidistant grid was used for the collocation method and the problem was solved using $p = 20$, $p = 40$, $p = 50$, $p = 80$, $p = 160$ and $l = 2$, $l = 3$. Recall that p is the number of grid intervals and l is the order of the polynomials used for the state variables.

The solution trajectory of the optimal control problem (7.1.1.2) - (7.1.1.10) for the case to which the numerical values in Table 7.1 apply, is given in the Figures 7.3 and 7.4. The convergence history of the solution process, corresponding to the case $p = 50$, $l = 2$ is given in Appendix F, Table F1.

The value of the objective function corresponding to the numerical solutions of the different values for p and l, is given in Table 7.2.



Solution of unconstrained glider problem, state variables.
Figure 7.3



Solution of unconstrained glider problem, control variable.
Figure 7.4

l	$p=20$	$p=40$	$p=80$	$p=160$	order
2	7.30220430969	7.30227700518‡	7.30239324467‡	7.30240227507‡	3.5
3	7.30222852296‡	7.30239951524‡	7.30240286698‡	7.30240286637	5.7

Table 7.2 : values of objective function and estimated order.

In the most right column of Table 7.2 an estimate of the order of the integration method is given, which is based on the values obtained for the objective function. (Theoretically this exponent should be $\geqslant 2l$, cf. Section 6.3).

‡ These values were used for the calculation of the order.

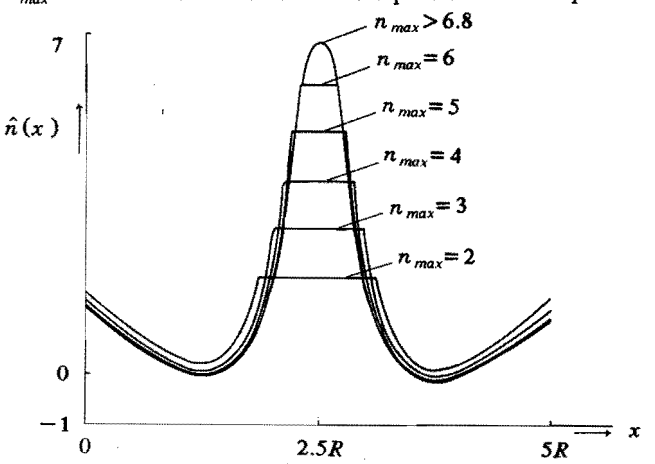
7.1.2. Restriction on the acceleration (mixed control state constraint).

The acceleration of the glider is the quotient of the lift force and the mass, i.e. L/m. Because the glider pilot cannot endure great accelerations, a constraint of the form :†

$$n = \frac{L}{mg} = \frac{\rho S v_r^2 u}{mg} \leqslant n_{max}, \qquad (7.1.2.1)$$

is necessary. Because (7.1.2.1) contains both the state variables (v_r depends on v_x and v_y) and the control variable u, this constraint is a mixed control state constraint.

In Figure 7.5 the normal load factor corresponding to the solution $\hat{n}(x)$ is given for various values of n_{max}. In the numerical solution of the problem we used $p = 20$ and $l = 2$.



Normal load factor $\hat{n}$ for various values of n_{max}.

Figure 7.5

The convergence history of the solution process corresponding to the case $n_{max} = 4$ is given in Appendix F, Table F2.

7.1.3. Restriction on the velocity (first order state constraint).

In many practical cases the velocity of the glider must stay below a certain limit. This yields, in the formulation of the optimal control problem the following state constraint :

$$v_r(x) \leqslant v_{max} \qquad 0 \leqslant x \leqslant 5R, \qquad (7.1.3.1)$$

which states that the relative velocity of the glider is not to exceed the limit v_{max}. Using (7.1.1.9) we obtain :

$$S_2(v_x, v_y, x) = v_x^2 + (v_y - u_a)^2 - v_{max}^2 \qquad 0 \leqslant x \leqslant 5R, \qquad (7.1.3.2)$$

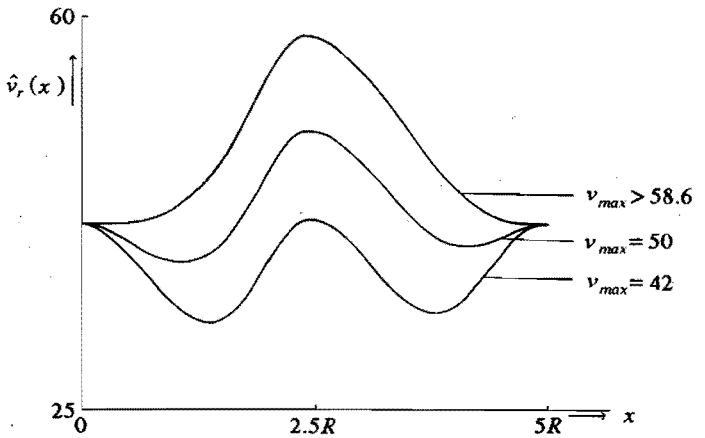
Differentiating (7.1.3.2), to the independent variable x yields the function S_2^1 (this function is defined by (3.3.5.7) - (3.3.5.8)) :

† In most aerospace control applications, the acceleration is limited to 4-6g.

$$S_2^1(v_x, v_y, u, x) = 2v_x \frac{dv_x}{dx} + 2(v_y - u_a)\left|\frac{dv_y}{dx} - \frac{du_a}{dx}\right| \qquad 0 \leqslant x \leqslant 5R. \qquad (7.1.3.3)$$

Substituting the equations of motion of the glider into (7.1.3.3) reveals that the function S_2^1 contains the control explicitly and hence the constraint is of first order. †

For values of $v_{max} > 58.6$ m/s the constraint (7.1.3.2.) is inactive on the entire interval of control. For values $v_{max} \leqslant 58.6$ m/s the state constraint has a contact point near $x = 2.5R$. In Figure 7.6 the velocity $\hat{v}_r(x)$ is given for three different values of v_{max}.



Glider velocity $\hat{v}_r$ for various values of v_{max}.

Figure 7.6

The convergence history corresponding to the case $v_{max} = 50$ is given in Appendix F, Table F3.

7.1.4. Restriction on the altitude (second order state constraint).

The solution trajectory of the unconstrained glider problem reveals that the glider dives first towards the earth and then regains altitude in the second half of the interval, as a result of the thermal. In many cases however, the glider is not allowed to flight below a certain altitude. The altitude of the glider is determined by :

$$\frac{dy}{dx} = \frac{v_y}{v_x} \qquad 0 \leqslant x \leqslant 5R, \qquad (7.1.4.1)$$

$$y(0) = y_0, \qquad (7.1.4.2)$$

where y_0 is the altitude at the initial point $x = 0$. (In the implementation the actual value of y_0, which is arbitrary, was set to zero.)

The altitude constraint which states that the glider may never fly below a certain limit becomes :

† We note that the state constraint (7.1.3.2) does not satisfy (5.1.2.4) and hence in the implementation, the constraint is transformed using the technique outlined in Appendix B.

$$y(x) \geq y_{min} \qquad 0 \leq x \leq 5R, \tag{7.1.4.3}$$

or, in the terminology of problem (SCOCP),

$$S_2(y) = -y + y_{min} \leq 0 \tag{7.1.4.4}$$

Differentiating to the independent variable x and substituting the equations of motion of the glider yields (for a formal definition cf. (3.3.5.7) - (3.3.5.8)) :

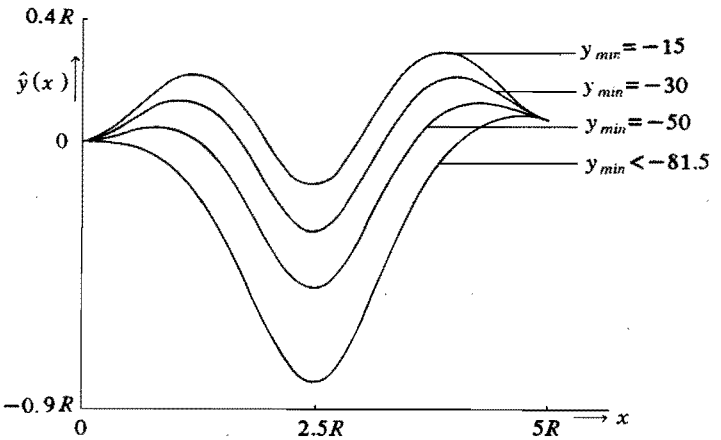
$$S_2^1(v_x, v_y) = -\frac{v_y}{v_x} \tag{7.1.4.5}$$

and

$$S_2^2(v_x, v_y, u, x) = \frac{v_y}{v_x^2}\frac{dv_x}{dx} - \frac{1}{v_x}\frac{dv_y}{dx}, \tag{7.1.4.6}$$

which reveals that the state constraint (7.1.4.4) is of second order.

For values $y_{min} < -81.5$ m the state constraint (7.1.4.4) is inactive during the entire flight. For values $y_{min} \geq -81.5$ m, the state constraint has, similar to the constraint on the velocity, a contact point near $x = 2.5R$. In Figure 7.7 the altitude $\hat{y}(x)$ is given for four different values of y_{min}.



Glider altitude $\hat{y}$ for various values of y_{min}.

Figure 7.7

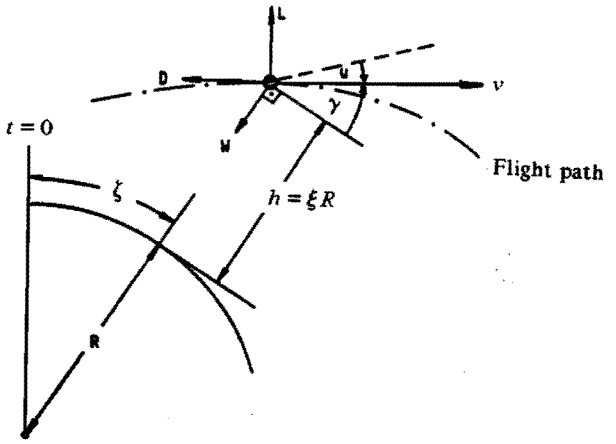
The convergence history corresponding to the case $y_{min} = -30$ is given in Appendix F, Table F4.

7.2. Reentry manoever of an Apollo capsule.

7.2.1. Description of the problem.

The problem deals with the reentry manoever of an Apollo capsule to the earth atmosphere, which is depicted in Figure 7.8.

The space vehicle is modelled as a point mass, subject to a lift force L, a drag force D and a gravity force W. The state variables are the velocity v, the flight-path angle γ, the



Variables of the Apollo reentry problem.
Figure 7.8

normalized altitude $\xi = h/R$ and the distance on the earth's surface ζ. These state variables satisfy the following set of differential equations :

$$\dot{v} = -\frac{S}{2m}\rho v^2 C_D(u) - \frac{g_0 \sin\gamma}{(1+\xi)^2} \qquad (7.2.1.1)$$

$$\dot{\gamma} = \frac{S}{2m}\rho v C_L(u) + \frac{v\cos\gamma}{R(1+\xi)} - \frac{g_0\cos\gamma}{v(1+\xi)^2} \qquad (7.2.1.2)$$

$$\dot{\xi} = \frac{v}{R}\sin\gamma \qquad (7.2.1.3)$$

$$\dot{\zeta} = \frac{v}{1+\xi}\cos\gamma \qquad (7.2.1.4)$$

where : R = earth's radius ($209.0352\ 10^5 ft$),
$\rho = \rho_0 e^{-\beta R\xi}$ = atmospheric density ($\rho_0 = 2.3769\ 10^{-3} slug/ft^3$ and $\beta = 1/0.235\ 10^{-5} ft^{-1}$),
g_0 = gravitational acceleration ($23.2172\ 10^{-4}\ 10^5 ft/s^2$),
$C_D(u) = C_{D0} + C_{DL}\cos u$ = aerodynamical drag coefficient,
$C_L(u) = C_{L0}\sin u$ = aerodynamical lift coefficient,
u = angle of attack = control variable,
S/m = frontal area / mass of vehicle.

The constants C_{D0}, C_{DL}, C_{L0} and S/m differ for the problems discussed in following sections.

7.2.2. Solution of the unconstrained reentry problem.

The flight path of the Apollo capsule is for the problem discussed in this section governed by the differential equations (7.2.1.1) - (7.2.1.4) with the following numerical constants $C_{D0} = 0.88$, $C_{DL} = 0.52$, $C_{DL} = -0.505$ and $S/m = 50000\ 10^{-5} ft^2/slug$.

During the reentry manoever the total stagnation point convective heating per unit area, given by

$$J = \int_0^T 10v^3\sqrt{\rho}\, dt\,, \tag{7.2.2.1}$$

must be minimized.

The reentry manoever is started at the following initial point :

$$v(0) = 0.35\ 10^5 ft/s \tag{7.2.2.2}$$

$$\gamma(0) = -5.75° \frac{\pi}{180°} \tag{7.2.2.3}$$

$$\xi(0) = 4/R \tag{7.2.2.4}$$

$$\zeta(0) = 0 \tag{7.2.2.5}$$

and at the (variable) final time, the following terminal point conditions must be satisfied :

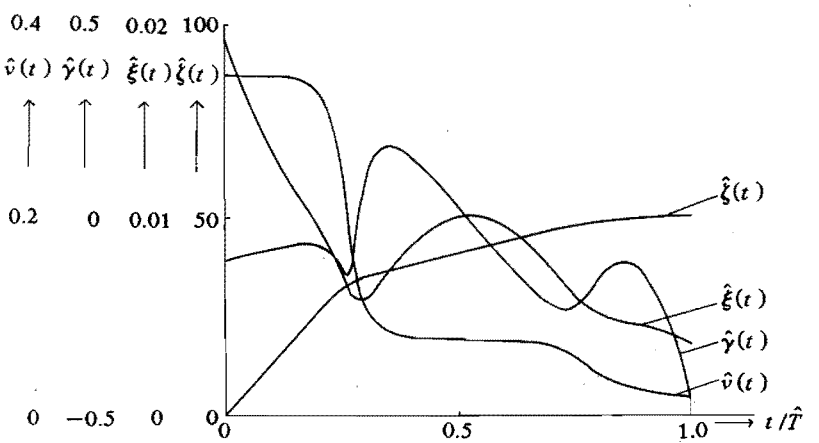
$$v(T) = 0.0165\ 10^5 ft/s \tag{7.2.2.6}$$

$$\gamma(T) = free \tag{7.2.2.7}$$

$$\xi(T) = 0.75530/R \tag{7.2.2.8}$$

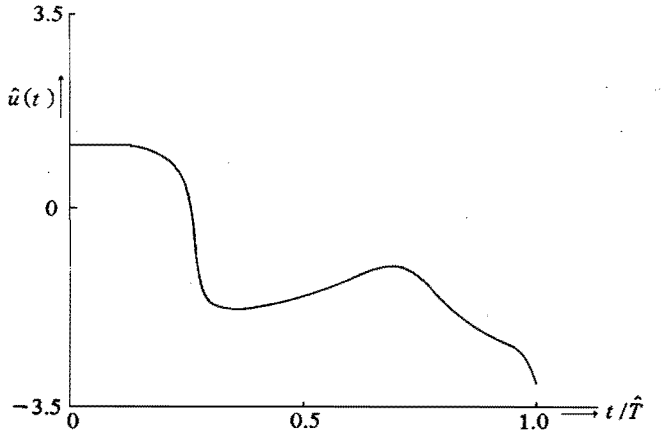
$$\zeta(T) = 51.6912\ 10^5 ft \tag{7.2.2.9}$$

As a starting trajectory the data given by Bals ((1983), Table 17) were used.



Solution of the unconstrained reentry problem, state variables.
Figure 7.9

In Figure 7.9 the state variable histories corresponding to the numerical solution of the problem are given. Figure 7.10 shows the optimal control history. The convergence history corresponding to the numerical solution of the problem is given in Appendix F, Table F5.



Solution of unconstrained reentry problem, control variable.
Figure 7.10

7.2.3. Restriction on the acceleration (mixed control state constraint).

The flight path of the Apollo capsule is for the problem discussed in this section governed by the differential equations (7.2.1.1) - (7.2.1.3)† with the following numerical constants $C_{D0} = 1.174$, $C_{DL} = -0.9$, $C_{L0} = 0.6$, $S/m = 53200\ 10^5 ft^2/slug$.

The optimal control of the reentry manoever should be such that the velocity at the (variable) final time T is maximized, i.e. the functional

$$J = -v(T), \tag{7.2.3.1}$$

must be minimized.

The reentry manoever is started at the initial point :

$$v(0) = 0.36\ 10^5 ft/s \tag{7.2.3.2}$$

$$\gamma(0) = -8.1° \frac{\pi}{180°} \tag{7.2.3.3}$$

$$\xi(0) = 4/R \tag{7.2.3.4}$$

After the reentry manoever the state variable γ and ξ should satisfy :

$$\gamma(T) = 0 \tag{7.2.3.5}$$

$$\xi(T) = 2.5/R \tag{7.2.3.6}$$

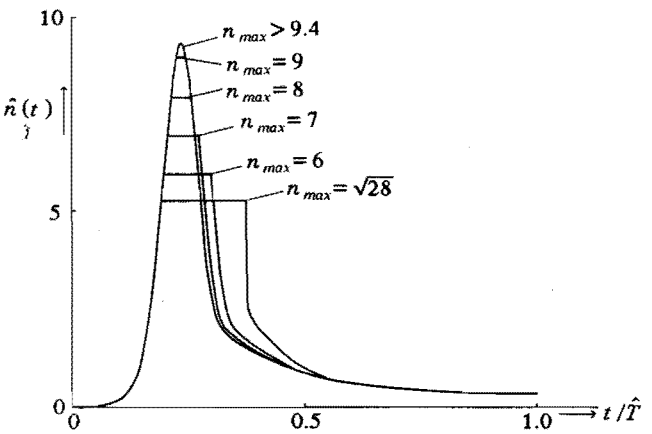
During the reentry manoever the total acceleration of the vehicle should be bounded to values which are bearable by the astronauts. In the formulation of the optimal control problem this yields the following mixed control state constraint :

† Because there is no terminal point constraint for the state variable ζ and this variable is not present in the equations (7.2.1.1) - (7.2.1.3), this variable may be omitted completely.

$$\frac{S\rho v}{2m}\sqrt{C_L(u)^2 + C_D(u)^2} \leqslant n_{max}g \qquad (7.2.3.7)$$

As with the glider problem of Section 7.1.2, the maximum normal load factor n_{max} is normally a value between 4 and 6.

The problem was solved for a number of different values of n_{max}. For each of these runs the data given by Bals ((1983), Table 14) were used, as a starting trajectory, which is an estimate of the solution of the reentry problem when no constraints are present. The maximum acceleration which arises during the reentry problem when no acceleration constraint is taken into account is $9.4g$. Thus for values of n_{max} smaller than 9.4 the optimal control will be restricted by the mixed control state constraint (7.2.3.7).



Normal load factor $\hat{n}$ for various values of n_{max}.

Figure 7.11

The normal load factor $\hat{n}(t)$ is given in Figure 7.11 for values of $n_{max} = 9,8,7,6,\sqrt{28}$. For values lower than $\sqrt{28}$ no convergence could be achieved. These results are similar to those of Gillessen (1974). Probably there is no feasible control of the reentry manoever possible for values lower than $\sqrt{28}$ and with the boundary conditions (7.2.3.3) - (7.2.3.7). The convergence history of the case $n_{max} = 6$ is given in Appendix F, Table F6.

7.2.4. Restriction on the altitude (second order state constraint).

The reentry manoever of the Apollo capsule is now considered, subject to a restriction on the altitude (cf. Bals (1983), Gillessen (1974), Hiltman (1983)).

An inspection of the solution of the unconstrained reentry problem discussed in Section 7.2.2 shows that after the vehicle has dived into the earth's atmosphere, the altitude of the vehicle ξ is again increased, in order to minimize the heating of the front shield of the vehicle. As a result of this increase in altitude the movement of the vehicle will be directed from the earth for some time. This is a dangerous situation because during this movement directed from the earth, small errors in the control of the vehicle may lead to 'hurling' back to space. In order to decrease this danger, a constraint on the altitude ξ is added, once the first altitude minimum is passed. The constraint is thus of the following form :

$$S_2 = \xi(t) \leqslant \xi_{max} \qquad \alpha T \leqslant t \leqslant T, \tag{7.2.4.1}$$

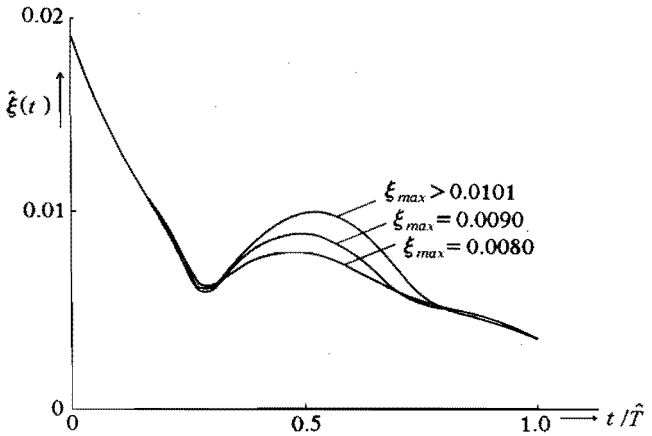
where α is an a priori specified quantity (actual value $\alpha = 0.3$).

An inspection of the functions S_2^1 and S_2^2 obtained from (7.2.4.1), (7.2.1.1) - (7.2.1.3) via differentiation to the time yields that the state constraint (7.2.4.1) is of second order.

For the remaining details the problem is similar to the problem discussed in Section 7.2.2, except for the final state of γ, which should satisfy :

$$\gamma(T) = -26.237124° \frac{\pi}{180°} \tag{7.2.4.2}$$

As a starting trajectory the data given by Bals ((1983), Table 17) were used to solve the unconstrained problem, which corresponds to the case $\xi_{max} > 0.0101$. Using each time the solution obtained for the previous value of ξ_{max} as an initial estimate, the value of ξ_{max} was decreased successively to 0.0090 and 0.0080. For values lower than $\xi_{max} = 0.0080$ no convergence of the method could be achieved. This was due to the fact that the step size became very small and hence there was no longer progress towards a solution point.



Relative altitude $\hat{\xi}$ for various values of ξ_{max}.

Figure 7.12

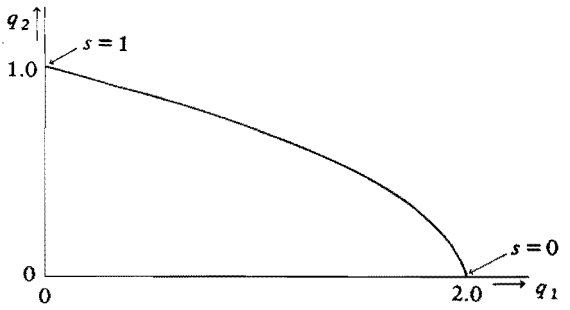
In Figure 7.12 the altitude $\hat{\xi}(t)$ is shown for the values $\xi_{max} > 0.0101$ and $\xi_{max} = 0.0090, 0.0080$. The convergence history corresponding to the case $\xi_{max} = 0.0090$ is given in Appendix F, Table F7.

7.3. Optimal control of servo systems along a prespecified path, with contraints on the acceleration and the velocity.

In this section the optimal control of two dynamically independent servo systems, along a prespecified path is considered subject to constraints on the accelerations and the velocities of the individual servo systems.

7.3.1. Statement of the problem.

The optimal control problem to be considered is a special case of the problem outlined in Section 1.2, namely the case of two dynamically independent servo systems q_1 and q_2, which are to be controlled along a path $\Upsilon(s)$ (depicted in Figure 7.13) from the point $s=0$ to the point $s=1$.



Path $\Upsilon(s)$.
Figure 7.13

The dynamic behaviour of the servo systems is supposed to be described by the following differential equations :

$$J_i \ddot{q}_i(t) = F_i(t) \qquad 0 \leqslant t \leqslant T \qquad i = 1,2 \tag{7.3.1.1}$$

To control the system along the path, the servo position coordinates q_1 and q_2 must satisfy :

$$q_i(t) = \Upsilon_i(s(t)) \qquad 0 \leqslant t \leqslant T \qquad i = 1,2 \tag{7.3.1.2}$$

The optimal control problem is now, as in Section 1.2, to find a twice differentiable function $s : [0,T] \rightarrow [0,1]$, such that constraints of the type

$$|\dot{q}_i(t)| \leqslant V_{max,i} \qquad 0 \leqslant t \leqslant T \qquad i = 1,2 \tag{7.3.1.3}$$

and

$$|F_i(t)| \leqslant F_{max,i} \qquad 0 \leqslant t \leqslant T \qquad i = 1,2 \tag{7.3.1.4}$$

are satisfied and that in addition the following objective function is minimized (for fixed $c \geqslant 0$) :

$$T + \tfrac{1}{2} c \int_0^T \ddot{s}(t)^2 \, dt. \tag{7.3.1.5}$$

(The final time T is supposed to be variable.)

As in Section 1.2 it is possible to eliminate the coordinates q_i and the forces F_i completely from the statement of the optimal control problem, using (7.3.1.1) - (7.3.1.4). The state constraints (7.3.1.3) become :

$$|\Upsilon_i'(s(t))\dot{s}(t)| \leqslant V_{max,i} \qquad 0 \leqslant t \leqslant T \qquad i = 1,2 \tag{7.3.1.6}$$

Because the movement along the curve directed from the point $s=0$ to the point $s=1$ corresponds with $\dot{s}(t) > 0$, it is likely that the solution of the optimal control problem

will (automatically) satisfy the condition :

$$|\dot{s}(t)| \geqslant 0 \qquad 0 \leqslant t \leqslant T. \tag{7.3.1.7}$$

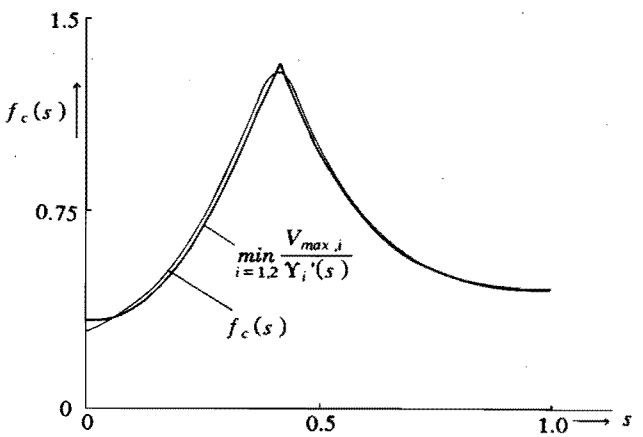
Under the assumption that this condition is satisfied, the constraints (7.3.1.6) may be rewritten as :

$$\dot{s}(t) \leqslant \min_{i=1,2} \frac{V_{max,i}}{|\Upsilon_i'(s(t))|} \qquad 0 \leqslant t \leqslant T. \tag{7.3.1.8}$$

As will follow from the exact statement of the optimal control problem given below, the constraint (7.3.1.8) is a state constraint of order one. Instead of using the nonsmooth form (7.3.1.8) for the state constraint, the problem is simplified by using a smooth approximation to the right hand side of (7.3.1.8). The constraint (7.3.1.8) is now replaced by :

$$\dot{s}(t) \leqslant f_c(s(t)) \qquad 0 \leqslant t \leqslant T. \tag{7.3.1.9}$$

In Figure 7.14 both the right hand side of (7.3.1.8) and the function $f_c(s)$ are plotted as a function of the variable s, for the path of Figure 7.13.



State constraint function $f_c(s)$.

Figure 7.14

For the approximation f_c the smoothing spline of Schoenberg and Reinsch is used (cf. de Boor (1978)).

Using relation (7.3.1.1) and the second time derivative of (7.3.1.2), the constraints (7.3.1.4) become :

$$|\Upsilon_i'(s(t))\ddot{s}(t) + \Upsilon_i''(s(t))\dot{s}(t)^2| \leqslant A_{max,i} \qquad 0 \leqslant t \leqslant T \qquad i = 1,2. \tag{7.3.1.10}$$

with :

$$A_{max,i} := \frac{F_{max,i}}{J_i} \qquad i = 1,2. \tag{7.3.1.11}$$

The optimal control problem involves the selection of a twice differentiable function $s:[0,T] \rightarrow [0,1]$ and a final time $T > 0$, that satisfy the constraints (7.3.1.9) and (7.3.1.11).

Because the motion starts at $s=0$ and ends at $s=1$, we must also have $s(0)=0$ and $s(T)=1$.

For the sake of completeness we will now give a formal statement of the optimal control problem, in the form which is used in combination with the numerical implementation of the method.

The relative path position s is formally denoted by x_1. An artificial state variable is used for the value of the state constraint (7.3.1.9), i.e.

$$x_2(t) = \dot{s}(t) - f_c(s(t)) \qquad 0 \leqslant t \leqslant T. \tag{7.3.1.12}$$

The numerical implementation of the method is done for optimal control problems on the fixed final time interval [0,1], therefore the optimal control problem must be transformed to this interval using a transformation

$$t = \tau T \qquad 0 \leqslant \tau \leqslant 1. \tag{7.3.1.13}$$

The variable T has the form of a parameter in the transformed optimal control problem, which is formally taken into account using a state variable x_3 that satisfies :

$$\dot{x}_3(\tau) = 0 \qquad 0 \leqslant \tau \leqslant 1. \tag{7.3.1.14}$$

The second derivative of the relative path position playes the role of the control variable and is therefore denoted by u.

The optimal control problem may now formally by stated as :

$$\underset{x,u}{\textit{minimize}}\; x_3(1) + \tfrac{1}{2} c \int_0^1 u(\tau)^2 \, d\tau, \tag{7.3.1.15}$$

subject to :

$$\dot{x}_1 = x_3(x_2 + f_c(x_1)) \qquad 0 \leqslant \tau \leqslant 1, \tag{7.3.1.16}$$

$$\dot{x}_2 = x_3(u - f_c'(x_1)(x_2 + f_c(x_1))) \qquad 0 \leqslant \tau \leqslant 1, \tag{7.3.1.17}$$

$$\dot{x}_3 = 0 \qquad 0 \leqslant \tau \leqslant 1, \tag{7.3.1.18}$$

$$x_1(0) = 0, \tag{7.3.1.19}$$

$$x_2(0) = -f_c(0), \tag{7.3.1.20}$$

$$x_1(1) = 1, \tag{7.3.1.21}$$

$$x_2(1) = -f_c(1), \tag{7.3.1.22}$$

$$x_2 \leqslant 0 \qquad 0 \leqslant \tau \leqslant 1, \tag{7.3.1.23}$$

$$Y_i'(x_1)u + Y_i''(x_1)(x_2 + f_c(x_1))^2 - A_{max,i} \leqslant 0 \qquad i = 1,2 \quad 0 \leqslant \tau \leqslant 1, \tag{7.3.1.24}$$

$$-Y_i'(x_1)u - Y_i''(x_1)(x_2 + f_c(x_1))^2 - A_{max,i} \leqslant 0 \qquad i = 1,2 \quad 0 \leqslant \tau \leqslant 1. \tag{7.3.1.25}$$

7.3.2. Numerical results of the servo problem.

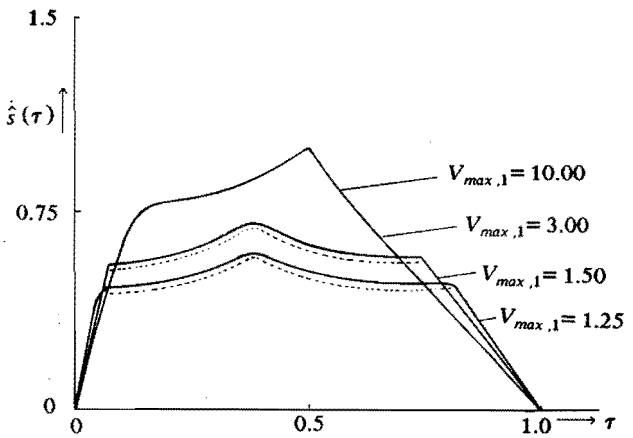
The problem described in the previous section was solved for a number of different values of the maximum servo velocities and accelerations and for different values of the parameter c which defines the objective function.

The numerical solutions discussed in this section were obtained using an equidistant grid of 20 points in the first stage of the method (i.e. $p = 20$). For the approximations to the state variables quadratic polynomials were used on the grid intervals (i.e. $l = 2$).

The maximum velocities and accelerations of the servo system with index 2 were taken dependent on the values of the servo system with index 1 in the following way :

$$V_{max,2} = \tfrac{1}{2} V_{max,1},$$

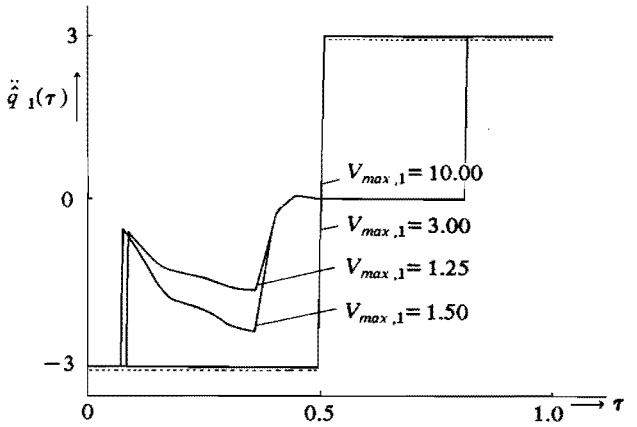
$$A_{max,2} = 2A_{max,1}.$$



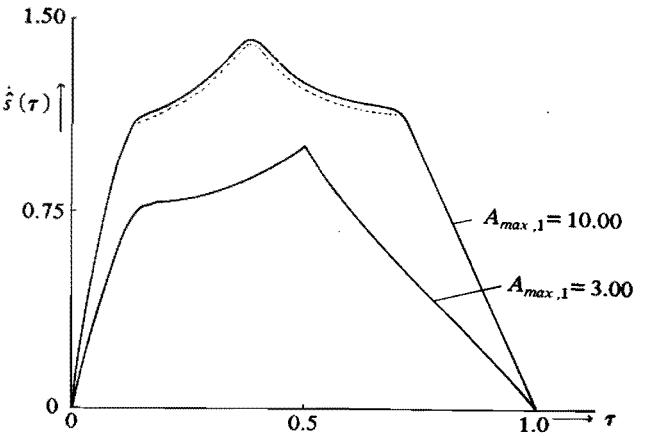
Path velocity for various values of $V_{max,1}$.
Figure 7.15

The first case to be considered is the case that the parameter c and the maximum acceleration $A_{max,1}$ are kept fixed ($c = 10^{-2}$ and $A_{max,1} = 3$). In Figures 7.15 and 7.16 the path velocities $\hat{s}$ and the accelerations $\ddot{\hat{q}}_1$ which are numerical solutions to the problem for the cases that $V_{max,1} = 10$, $V_{max,1} = 3$, $V_{max,1} = 1.5$ and $V_{max,1} = 1.25$, are given. The dotted lines indicate when a constraint is active on either the path velocity or on the acceleration of the servo with index 1.

From Figures 7.15 and 7.16 we note that the solutions corresponding to the cases $V_{max,1} = 10$ and $V_{max,1} = 3$ are indentical, which is a result of the fact that in these cases the velocity constraint (7.3.1.10) is not active at all. In these cases the constraint on the acceleration is almost always active. When the maximum velocity is decreased to $V_{max,1} = 1$, then the acceleration constraint is only active part of the time and the constraint on the path velocity is active over some period of time. When the maximum velocity is further decreased, the velocity constraint becomes active over a longer period of time.



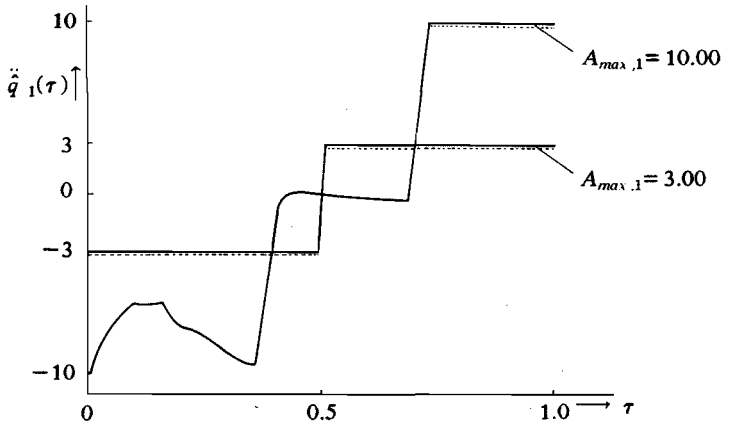
Acceleration of servo 1 for various values of $V_{max,1}$.
Figure 7.16



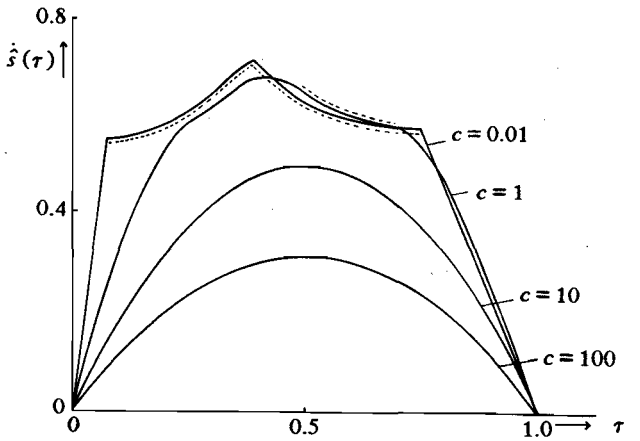
Path velocity for various values of $A_{max,1}$.
Figure 7.17

The second case to be considered is the case in which the maximum velocity is kept fixed and where the maximum acceleration is varied ($c = 10^{-2}$, $V_{max,1} = 3$). The path velocities $\dot{s}$ and maximum accelerations $\ddot{q}$ which are numerical solutions to the problem are given in Figures 7.17 and 7.18 for the cases $A_{max,1} = 10$ and $A_{max,1} = 3$. The solution corresponding to the case $A_{max,1} = 3$ is again of the bang-bang type. In this case the acceleration constraint (7.3.1.10) is almost always active. When the acceleration constraint is increased, the velocity constraint becomes active.

The last case that is considered is the case where the maximum velocities and accelerations are kept fixed and where the parameter c is varied ($V_{max,1} = 1.5$ and $A_{max,1} = 3$). In Figure 7.19 the path velocities corresponding to the numerical solutions of the problem for the cases $c = 0.01$, $c = 1$, $c = 10$ and $c = 100$ are given. The solutions corresponding to the cases $c = 10$ and $c = 100$ are unconstrained solutions, i.e. no constraints are active at all.



Acceleration of servo 1 for various values of $A_{max,1}$.
Figure 7.18



Path velocity for various values of c.
Figure 7.19

In Table F8, Appendix F the convergence history corresponding to the case $V_{max,1} = 1.5$, $A_{max,1} = 3$ and $c = 1$ is given.

8. Evaluation and final remarks.

8.1. Relations between the SQP method in function space and some other methods.

The SQP method in function space, described in the previous chapters for the solution of state constrained optimal control problems, is essentially a method based on the abstract formulation of the state constrained optimal control problems in infinite-dimensional function spaces. The method consists of two stages. In the first stage the optimal control problem is approximately solved using a fixed step integration scheme. Hence, the first stage yields a rough approximation to the solution and a good estimate for the structure of the solution. The problem is solved more accurately in the second stage, which determines the exact locations of the junction and contact points of the state constraints. In the numerical context this means that during the second stage the integration step is adjusted in a neighborhood of junction and contact points. The first stage was developed by extension of the ideas of finite-dimensional sequential quadratic programming, based on the use of inequality constrained subproblems, to the abstract formulation. The second stage is based on a similar extension of the ideas of finite-dimensional sequential quadratic programming to the abstract formulation, but now based on the use of equality constrained subproblems.

A method which is strongly related to the first stage of the SQP method in function space, is the method which converts the optimal control problem into a finite-dimensional mathematical programming problem. This is done by approximating the control and the state functions using piecewise polynomial functions. The polynomial coefficients that are associated with this approximation become the variables in the mathematical programming problem. The finite-dimensional mathematical programming problem is then solved using a general purpose nonlinear programming method. Methods of this type are called *methods of direct discretization*. As we are interested in the relation between the SQP method in function space and methods of direct discretization, it will be assumed in the sequel, that a sequential quadratic programming method is used to solve the finite-dimensional nonlinear programming problem. Before we consider the relation between the SQP method in function space and methods of direct discretization, we will outline two specific methods of direct discretization (cf. Kraft (1980, 1984)).

One way to convert an optimal control problem into a nonlinear programming problem is to approximate the control $u(t)$ by means of a spline function on $[0,T]$ (cf. de Boor (1978)). Thereto a grid is chosen and the values of the control on the grid points, which are called the *spline knots*, are the variables of the nonlinear programming problem. The state variables of the system, $x(t)$, are treated as quantities dependent on the control u and may, at any time point, be obtained as the numerical solution of an initial value problem. With this type of method, gradients are usually obtained via numerical differentiation.

A refinement of this method, which significantly improves the accuracy of the solution obtained, is to take the spline knots also as variables of the nonlinear programming problem, i.e. the control is approximated by a spline function on a variable grid.

Another way to convert an optimal control problem into a nonlinear programming problem is to approximate, not only the control, but also the state by means of spline functions. The differential system

$$\dot{x}(t) = f(x(t),u(t),t) \qquad 0 \leqslant t \leqslant T, \tag{8.1.1}$$

is then converted into a number of equality constraints

$$\dot{\tilde{x}}(\tau_j) = f(\tilde{x}(\tau_j),\tilde{u}(\tau_j),\tau_j) \qquad j = 1,..... \tag{8.1.2}$$

which state that for the finite-dimensional approximation $(\tilde{x},\tilde{u})$, the differential system must be satisfied at the (collocation) points τ_j.

We note that the second method is in fact a refinement of the first method, as the second method is equivalent to the first method, when the implicit Runge-Kutta scheme, discussed in Section 6.1.1, is used as the integration scheme.

With both methods, state constraints can be treated in essentially three ways :

1) by taking care of them via penalty terms in the objective function.

2) via conversion into inequality constraints of the type :

$$y(0) = 0, \tag{8.1.3}$$

$$\dot{y}(t) = max\ \{0,S(x(t),t)\}, \tag{8.1.4}$$

$$y(T) \leqslant y_T, \tag{8.1.5}$$

where y_T is a 'small' quantity.

3) by replacing them by a finite number of inequalities of the form

$$S(x(t_j),t_j) \leqslant 0 \qquad j = 1,.... \tag{8.1.6}$$

where the points t_j are a finite subset of points of $[0,T]$.

Experience shows that the approaches 1) and 2), which are essentially similar, yielding relatively inefficient procedures with relatively inaccurate solutions (cf. Well (1983)). It is obvious that with the third approach the state constraints may be violated at all points, except at the time points t_j. According to the terminology of Kraft (1984), the state constraints are treated as a 'soft' constraints with the third approach.

For problems without state constraints of order $\geqslant 1$, the first stage of the SQP method in function space is equivalent to the method of direct discretization that is based on the conversion of the optimal control problem into a nonlinear programming problem in following way :

The state function is approximated using lth order piecewise polynomials on the intervals defined by

$$0 = t_0 < t_1 < .... < t_p = T, \tag{8.1.7}$$

which are continuous at the points t_r $(r = 1,....,p-1)$. The control is analogously approximated by means of $(l-1)$th order piecewise polynomials on the same intervals (t_r,t_{r+1}) $(r = 0,....,p-1)$. The differential system is replaced by a finite number of equality constraints :

$$\dot{x}(\tau_{lr+i}) = f(x(\tau_{lr+i}),u(\tau_{lr+i}),\tau_{lr+i}) \quad i = 1,....,l \quad r = 0,1,....,p-1, \tag{8.1.8}$$

where the collocation points τ_{lr+i} are as defined in Section 6.1.1. The mixed control state constraints are replaced by a finite number of inequality constraints :

$$S_1(x(\tau_{lr+i}),u(\tau_{lr+i}),\tau_{lr+i}) \leqslant 0 \quad i=1,...,l \quad r=0,...,p-1, \tag{8.1.9}$$

and the boundary conditions at $t=0$ and $t=T$ remain :

$$D(x(t_0)) \quad = 0, \tag{8.1.10}$$

$$E(x(t_p),t_p) = 0. \tag{8.1.11}$$

The objective function is approximated as a finite sum by means of the quadrature rule (6.1.1.24), i.e.

$$h_0(x(t_0)) + \sum_{r=0}^{p-1} h_r \sum_{i=1}^{l} \overline{\omega}_i f_0(x(\tau_{lr+i}),u(\tau_{lr+i}),\tau_{lr+i}) + g_0(x(t_p),t_p). \tag{8.1.12}$$

The connection between the two methods (i.e. the first stage of the SQP method in function space and the method of direct discretization) is revealed by the special structure of the collocation scheme for the linear multipoint boundary value problem, that follows from the necessary conditions for optimality for problem (EQP/SCOCP). This special structure indicates that the collocation equations are essentially equivalent to the necessary conditions for optimality for the quadratic programming problem obtained from problem (EQP/SCOCP) via the above mentioned discretization (cf. Section 6.1.2). Observing that the linear multipoint boundary value problem may be obtained via a Newton approach from the nonlinear multipoint boundary value problem, that follows from the optimality conditions for problem (SCOCP), yields the connection with the corresponding nonlinear programming problem.

Similar to the case of mixed control state constraints, it follows for problems with state constraints of order $\geqslant 1$, that when these constraints are replaced by interior point constraints of the form

$$S(x(t_j),t_j) \leqslant 0 \quad r=0,1,...p, \tag{8.1.13}$$

then the first stage of the SQP method in function space and the method of direct discretization using the approach 3) for the state constraints, are again equivalent.

However, in the case of the SQP method in function space, the state constraints of order $\geqslant$ 1 are, on boundary intervals replaced by the conditions

$$S^j(x(t_r),t_r) = 0 \quad j=0,1,...p_s, \tag{8.1.14}$$

at the entry points, and the conditions

$$S^{p_s}(x(\tau_{lr+i}),u(\tau_{lr+i}),\tau_{lr+i}) = 0, \tag{8.1.15}$$

at all collocation points, interior to boundary intervals. (p_s is the order of the state constraint). This is an essential difference between both methods, because a similar approach seems for direct discretization methods not possible. This is a result of the fact that the active set strategy discussed in Section 5.2, is entirely based on the special, infinite-dimensional relationship between the interior point constraints (8.1.14) and the mixed control state constraints (8.1.15).

The advantage of the SQP method in function space, compared to the methods of direct discretization, is that boundary intervals are approximated directly, instead of replacing them by of a number of interior point constraints. In the terminology of Kraft (1984), the state constraints are with the SQP method in function space, treated as 'hard' constraints.

Therefore, in general, the solution obtained from the first stage of the SQP method in function space will be a better approximation to the exact solution of the problem (SCOCP), than the solution obtained from the direct discretization methods, where the state constraints are treated as 'soft' constraints.

In practice, direct discretization methods are often used for the same purpose as the first stage of the SQP method in function space, i.e. to obtain the structure of the solution and a rough estimate of the solution of the optimal control problem. The solution of the optimal control problem can thereafter be obtained more accurately using, for instance, a method for the solution of the nonlinear multipoint boundary value problem which may be derived from the necessary conditions for optimality for the problem (SCOCP) (cf. Bock (1983), Bulirsch (1983), Maurer (1974, 1975)). This approach is essentially similar to the stage 1 - stage 2 approach of the SQP method in function space, where the second stage is started with the solution and the Lagrange multipliers obtained from the first stage. In this context, a serious disadvantage of the direct discretization methods is that the Lagrange multipliers obtained, corresponding to the solution of the nonlinear programming problem, cannot be used as estimates for the Lagrange multipliers in the second stage. This is due to the fact that the state constraints are treated differently in the first and the second stage. With the function space approach, the state constraints are treated similarly in both stages and hence the Lagrange multipliers obtained from the first stage can be used directly as estimates for the Lagrange multipliers in the second stage.

The second stage of the SQP method in function space can be compared with the 'multiple shooting' approach. With this approach the optimality conditions for problem (SCOCP) are used to derive a multipoint boundary value problem, which is solved using a multiple shooting method. The control and the Lagrange multipliers corresponding to the state constraints are eliminated analytically. In general, the junction and contact points of the state constraints are not known a priori and in addition, the right hand side of the set of differential equations and the adjoint variable may be discontinuous at these points. Therefore use is made of so-called switching functions which are used to locate these points, i.e. a zero of a switching function coincides with a junction or contact point. The general form of the multipoint boundary value problem is thus :

$$\dot{y} = F(y,t,z(y,t)) \qquad 0 \leqslant t \leqslant T, \tag{8.1.16}$$

$$G(y(0),y(T)) = 0, \tag{8.1.17}$$

$$H(y(\tilde{t}_j),\tilde{t}_j) = 0 \qquad \text{for all } \tilde{t}_j \tag{8.1.18}$$

At the junction and contact points one of the switching functions z_i has an isolated zero, i.e.

$$z_i(y(\tilde{t}_j),\tilde{t}_j) = 0 \tag{8.1.19}$$

The second stage of the function space method consists of the calculation of a direction of search based on the numerical solution of problem (EQP/SCOCP) and of the active set strategy described in Section 5.3. Without the active set strategy the second stage of the method solves, in fact, a nonlinear multipoint boundary value problem, where the control and the Lagrange multipliers corresponding to the state constraints are not eliminated as with the multiplie shooting approach, but which are determined by nonlinear algebraic equations. The active set strategy of Section 5.3 plays a role similar to the switching

function concept, as it is (only) used to determine the exact location of the junction and contact points. With the multiple shooting approach a thorough understanding of the first and second order conditions for optimality, for state constrained optimal control problem, is required and the actual conversion of the optimal control problem into a nonlinear multipoint boundary value problem in general, involves considerable work. With the SQP method in function space the problem functions are the only ones used and hence no conversion is required.

Reviewing the SQP method in function space in the context of Section 1.4, the first stage of the method is essentially a direct method and is therefore likely to have a relatively large region of convergence and which yields a relatively inaccurate solution. The second stage is essentially an indirect method, which has a relatively small region of convergence and which yields a relatively accurate solution. In the first stage of the method the structure of the solution is determined. The second stage requires, as all indirect methods, the structure of the solution and a relatively good estimate of the solution as an initial starting point. Because the first stage yields both the structure of the solution and an approximation to the solution, the second stage is automatically started with the structure and the solution obtained from the first stage. The entire method may thus be viewed as a method which combines the merits of both a direct and an indirect method.

8.2. Final remarks.

The results contained in Chapters 2 and 3 show that, at present, the optimality conditions for state constrained optimal control problems can be derived rigorously from a number of rather basic results on optimality in abstract vector spaces. Refinements dealing with the continuity of the Lagrange multipliers at junction points can be derived from these optimality conditions (e.g. cf. Maurer (1977)). An inspection of these refinements shows however, that they need not hold for the optimal control problems with state equality constraints, as considered in Section 5.1. Because the SQP method in function space requires both the solution of problems with state equality and state inequality constraints, it seems that these results have no application for the method presented in the thesis.

The SQP method in function space is essentially a Newton-like method applied to the first order necessary conditions for optimality. For the SQP method in abstract vector spaces, derived in Section 4.1, convergence results similar to those given by Kantorivich et al. (1982), can be stated. In applying the SQP method to state constrained optimal control problems several heuristic steps were taken. These heuristic adaptations of the SQP method, complicate the derivation of convergence results greatly. Because the solution method for the subproblem (EIQP/SCOCP/Δ) is also a heuristic adaptation of a method for which convergence results can be derived, it is quite likely that the method converges, but it seems hard to derive strict convergence results.

The main problem, in the derivation of the convergence results mentioned above, is the fact that with the SQP method in function space, boundary arcs of state constraints of order $\geqslant 1$, are treated as 'hard' constraints. We note that finite-dimensional sequential quadratic programming methods allow a rather complete convergence analysis (cf. Schittkowski (1981)). Hence in the case that the SQP method in function space is equivalent to a method of direct discretization (as outlined in the previous section) the convergence results for the method of direct discretization using finite-dimensional sequential quadratic programming, will also hold for the SQP method in function space. Also in

this case the solution method for the subproblem (EIQP/SCOCP/Δ) is identical with the quadratic programming method reviewed on Appendix A, which allows a standard convergence analysis.

The numerical results on the solution of some benchmark problems, given in Chapter 7, show that the method can indeed be used for the solution of state constrained optimal control problems. The Apollo reentry problems are the most difficult problems which are currently solved with the method. The sensitivity of the problem results in a relative ill-conditioning of the matrices, which determine the subproblems to be solved. A difficulty that had to faced in addition to the ill-conditioning of the matrices, was the fact that the subproblems were unbounded below (indefinite projected Hessian of the Lagrangian) except in a very small neighborhood of the solution. We note that the modifications which were implemented in order to overcome these problems led to a significant improvement in the implementation of the method.

For relatively stiff optimal control problems (such as the Apollo reentry problems) the collocation method, which is equivalent to an implicit Runge-Kutta integration scheme, can be very efficient, as a result of the fact that the integration step size can be varied very easily. This requires a mechanism, not present in the implementation yet, which selects the grid (integration step sizes) automatically.

Another improvement in the implementation of the method may be achieved by using quasi-Newton updates for the Hessian of the Lagrangian. When these updates are used, it is no longer necessary to supply the second derivatives of the problem functions. This will simplify the use of the program at the cost of the rate of convergence, which in general will no longer be quadratic, but superlinear.

Appendix A : A method for the solution of finite-dimensional quadratic programming problems.

In this appendix we shall review a method for the solution of the following quadratic programming problem (cf. Powell (1974), Gill et al. (1981)) :

Problem (FDEIQP) :

$$\underset{d}{Minimize} \quad c^T d + \tfrac{1}{2} d^T M d, \tag{A1}$$

$$subject\ to : A_1 d = b_1, \tag{A2}$$

$$A_2 d \leqslant b_2, \tag{A3}$$

where : c and d are $\bar{n}$-vectors,
M is a symmetric $\bar{n} \times \bar{n}$ matrix,
A_1 and A_2 are resp. $\bar{m}_1 \times \bar{n}$ and $\bar{m}_2 \times \bar{n}$ matrices,
b_1 and b_2 are resp. $\bar{m}_1$ and $\bar{m}_2$ vectors.

We shall assume that problem (FDEIQP) has a solution for which the regularity constant (cf. Chapter 2) may be set nonzero. The optimality conditions of Kuhn-Tucker (cf. Gill et al. (1981)) then imply that there exist multipliers $\hat{\lambda}_1 \in \mathbb{R}^{\bar{m}_1}$ and $\hat{\lambda}_2 \in \mathbb{R}^{\bar{m}_2}$, that satisfy

$$M\hat{d} + A_1^T \hat{\lambda}_1 + A_2^T \hat{\lambda}_2 = -c, \tag{A4}$$

$$\hat{\lambda}_{2j}(a_{2j}\hat{d} - b_{2j}) = 0 \qquad j = 1,...\bar{m}_2, \tag{A5}$$

$$\hat{\lambda}_{2j} \geqslant 0 \qquad j = 1,...\bar{m}_2. \tag{A6}$$

In addition, the second order necessary condition for optimality are

$$y^T M y \geqslant 0 \quad for\ all \quad y \in \{d : A_1 d = 0 \wedge A_{2j} d = 0\ for\ all\ j \in I_A\}, \tag{A7}$$

with :

$$I_A := \{j : A_{2j}\hat{d} = b_{2j} \wedge \hat{\lambda}_{2j} > 0\}, \tag{A8}$$

i.e. the Hessian matrix M must be positive semi-definite on the tangent subspace of the 'active' constraints at $\hat{d}$. The second order sufficiency conditions have a similar form with $\geqslant$ replaced by $>$, i.e. M must be positive definite on the tangent subspace of the active constraints at $\hat{d}$.

The method we shall discuss is basically an iterative minimization of the objective function

$$\bar{f}(d) := c^T d + \tfrac{1}{2} d^T M d, \tag{A9}$$

over the *set of feasible points*,

$$H := \{d : A_1 d = b_1 \wedge A_2 d \leqslant b_2\}. \tag{A10}$$

This means that a sequence (d^i) is constructed for which

$$\bar{f}(d^{i+1}) \leqslant \bar{f}(d^i) \quad for\ all\ i = 0,1,... \tag{A11}$$

and

$$d^i \in H \qquad \text{for all } i = 0,1,.... \tag{A12}$$

The method assumes that a feasible initial point d^0 is given, which is used as a first element of the sequence.

In each iteration of the method, a key role is played by the socalled *working set*. This set consists of the constraints (A2) and a subset of the constraints (A3) which must be satisfied as equalities. The working set is an estimate for the set of active constraints in the solution point.

Essentially one iteration consists of three steps :

1) Calculation of a direction of search Δd^i.

2) Calculation of a step size α_i.

3) Updating the working set.

The direction of search is calculated such that the objective function is minimized with respect to the constraints in the working set, i.e. a solution of

Problem (FDEQP) :

$$\underset{\Delta d}{Minimize} \quad c^T(d^i+\Delta d) + \tfrac{1}{2}(d^i+\Delta d)^T M(d^i+\Delta d), \tag{A13}$$

$$subject\ to : A_1(d^i+\Delta d) = b_1, \tag{A14}$$

$$\bar{A}_2(d^i+\Delta d) = \bar{b}_2, \tag{A15}$$

where (A15) denotes the subset of constraints (A3), which are in the working set.

Because of the fact that for constraints in the working set equality holds, this problem is equivalent to :

$$\underset{\Delta d}{Minimize} \quad (c^T + d^{iT}M)\Delta d + \tfrac{1}{2}\Delta d^T M \Delta d, \tag{A16}$$

$$subject\ to : A_1 \Delta d = 0, \tag{A17}$$

$$\bar{A}_2 \Delta d = 0. \tag{A18}$$

If M is positive definite on the subspace

$$H^i := \{d : A_1 d = 0 \wedge \bar{A}_2 d = 0\}, \tag{A19}$$

then the problem (A16) - (A18) will have a unique solution and hence the direction of search Δd^i is uniquely determined. If M is only positive semi-definite on H^i, the problem does not have a unique solution. In this case the direction Δd^i is chosen to be the negative gradient of $\bar{f}$, i.e. the vector $c+Md^i$, projected on H^i. When the matrix M is indefinite on H^i then a solution to problem (FDEQP) does not exist, because along any *direction of negative curvature on* H^i, i.e. any Δd that satisfies

$$\Delta d^T M \Delta d < 0, \tag{A20}$$

and

$$\Delta d \in H^i, \tag{A21}$$

the value of the objective function is unbounded from below. When however, problem (FDEIQP) has a solution, then along any direction of negative curvature of M of H^i, an

inequality constraint, which is not in the working set, must become active. Hence, in the case that M is indefinite, any direction of negative curvature is a suitable choice for Δd^i.

Once a (nonzero) direction of search is calculated, a step size α_i must be determined.

In the case that M is positive definite on H^i, the step size α_i is taken so that $\bar{f}$ is minimized along Δd^i on H^i, i.e.

$$\alpha_i := \min_j \{1, -\frac{A_{2j} d^i - b_{2j}}{A_{2j} \Delta d^i} \wedge A_{2j} \Delta d^i > 0\}. \tag{A22}$$

A similar choice is made in the case that M is only positive semi-definite on H^i :

$$\alpha_i := \min_j \{-\frac{(c^T + d^{iT} M)\Delta d^i}{\Delta d^{iT} M \Delta d^i}, -\frac{A_{2j} d^i - b_{2j}}{A_{2j} \Delta d^i} \wedge A_{2j} \Delta d^i > 0\}. \tag{A23}$$

If M is indefinite on H^i, the step size α_i is taken as

$$\alpha_i := \min_j \{-\frac{A_{2j} d^i - b_{2j}}{A_{2j} \Delta d^i} \wedge A_{2j} \Delta d^i > 0\}. \tag{A24}$$

The third step of an iteration consists of updating the working set.

A constraint is only added to the working set when it restricts the step size α_i. We note that if the matrix of constraints

$$A = \begin{pmatrix} A_1 \\ \overline{A_2} \end{pmatrix},$$

was of full row rank before the constraint was added, then it will also be of full row rank after a constraint is added. For if this constraint was linearly dependent of some constraints already in the working set, then the constraint would not have restricted the step size α_i.

If the direction of search Δd^i is zero, then the minimum in the current subspace is achieved and hence no further progress can be made with the current working set. The subspace may be enlarged by deleting constraints with negative Lagrange multipliers from the working set. If only one such constraint is deleted, then the direction of search Δd^i, computed as the solution of problem (FDEQP), will be directed into the feasible region (cf. Powell (1974)).

When the direction of search Δd^i becomes zero and there are no constraints with a negative multiplier, then d^i is a solution of problem (FDEIQP).

The method described above may be summarized as the algorithm below :

Algorithm A1 :

(0) $d^0 \in H^0$ *given.*
$i := 0$.

(i) *Test for convergence.*
Terminate if the minimum in the subspace H^i *is achieved and if the Lagrange multipliers have correct sign.*

(ii) *Calculate a direction of search* Δd^i.

(iii) *If* $\|\Delta d^i\| = 0$ *then goto (vii).*

(iv) *Calculate a step size* α_i *and set*

$$d^{i+1} := d^i + \alpha_i \Delta d^i .$$

(v) *If the step size* α_i *was restricted by one or more constraints, add one of these constraints to the working set.*

(vi) $i := i+1$
goto (ii).

(vii) *Delete a constraint with a negative Lagrange multiplier from the working set.*
goto (ii).

Appendix B : Transformation of state constraints.

Consider problem (SCOCP) with a scalar state constraint

$$T(x(t),t) \leqslant 0 \quad \textit{for all} \quad 0 \leqslant t \leqslant T, \tag{B1}$$

for which condition (5.1.2.4) is not satisfied. This means that the functions T^j defined as :

$$T^j := \begin{cases} T(x,t) & j=0 \\ T_x^{j-1}(x,t)f(x,u,t) + T_t^{j-1}(x,t) & j=1,...p \end{cases} \tag{B2}$$

do not satisfy the condition :

$$T_{xx}^j(x,t) = 0 \qquad \textit{for all} \quad j=0,1,.....p-1. \tag{B3}$$

The transformation requires the introduction of p additional state variables, denoted y_j ($j=1,..p$), that satisfy the differential equations :

$$\dot{y}_j = y_{j+1} \qquad j=1,...p-1 \tag{B4}$$

$$\dot{y}_p = T^p(x,u,t) \tag{B5}$$

with initial conditions :

$$y_j(0) = T^{j-1}(x(0),0) \quad j=1,...p. \tag{B6}$$

For a trajectory (x,y,u) that satisfies (3.1.2), (3.1.3), (B4), (B5) and (B6), we have

$$y_j(t) = T^{j-1}(x(t),t) \quad 0 \leqslant t \leqslant T. \tag{B7}$$

The state constraint (B1) is now replaced by :

$$S_2(x(t),y(t),t) := y_1(t) \leqslant 0 \qquad 0 \leqslant t \leqslant T, \tag{B8}$$

which makes no difference for the solution of the orginal problem (SCOCP). However, one may easily verify that for the transformed problem we have

$$S_2^j = \begin{cases} y_{j+1} & j=0,1,...p-1 \\ T^p(x,u,t) & j=p \end{cases} \tag{B9}$$

and hence condition (5.1.2.4) is satisfied for the transformed problem.

Appendix C : Results on the reduction of the working set.

During the execution of Algorithm 5.8, which determines a solution of problem (EIQP/SCOCP/Δ), the direction of search can become zero. In this case it is possible that further progress towards a solution can be achieved by a suitable reduction of the working set. In this appendix we shall investigate this reduction of the working set.

We note that when the direction of search becomes zero, then the current estimate of the solution of problem (EIQP/SCOCP/Δ), (d_x^i, d_u^i), is the solution to problem (EQP/SCOCP) with the current working set.

The working set of iteration i will be denoted as :

$$W^i := W_1^i \times W_2^i \times \ldots\ldots \times W_{k_1+k_2}^i. \tag{C1}$$

Reducing the working set in iteration i yields $W^i \subset W^{i-1}$. The direction of search $(\Delta d_x^i, \Delta d_u^i)$, in iteration i will be determined from the solution of problem (EQP/SCOCP) with the working set W^i. Because all equality constraints of problem (EIQP/SCOCP/Δ) will also hold for solutions to problem (EQP/SCOCP), the direction of search satisfies :

$$\Delta \dot{d}_x^i = f_x[t]\Delta d_x^i + f_u[t]\Delta d_u^i \qquad a.e.\ 0 \leqslant t \leqslant T, \tag{C2}$$

$$D_x[0]\Delta d_x^i(0) = 0, \tag{C3}$$

$$E_x[T]\Delta d_x^i(T) = 0. \tag{C4}$$

A requirement for the choice of the working set W^i is that a step size α_i can be determined so that (d_x^{i+1}, d_u^{i+1}) is at least Δ-feasible. This requires that the direction of search must satisfy :

$$S_{1lx}[\bar{t}_r^1]\Delta d_x^i(\bar{t}_r^1) + S_{1lu}[\bar{t}_r^1]\Delta d_u^i(\bar{t}_r^1) \leqslant 0 \qquad \text{for all} \quad \bar{t}_r^1 \in W_l^{i-1} \quad l = 1,..k_1, \tag{C5}$$

$$S_{2lx}[\bar{t}_r^2]\Delta d_x^i(\bar{t}_r^2) \leqslant 0 \qquad \text{for all} \quad \bar{t}_r^2 \in W_{k_1+l}^{i-1} \quad l = 1,..k_2, \tag{C6}$$

i.e. the direction of search $(\Delta d_x^i, \Delta d_u^i)$ must be feasible for the grid points which were in the working set in the previous iteration. Obviously, this requirement is satisfied for all time points which remain in the working set in iteration i, because for these time points (C5) and (C6) will hold as equalities. The choice of the working set W^i is governed by the fact that (C5) and (C6) must also hold for time points which are deleted from the working set in iteration $i-1$. In view of this choice we shall first prove Lemma C1.

To simplify notation, the superscript i-1 is omitted for the Lagrange multipliers, which are used in the sequel.

Without loss of generality we shall assume that the working set W_l^{i-1}, $(l = 1,...k_1+k_2)$ consists of one boundary interval $[t_1^l, t_2^l]$ and, in addition, that the working sets $W_{k_1+l}^{i-1}$, $(l = 1,...k_2)$ contain one contact point t_3^l, $(l = 1,...k_2)$.

Lemma C1 : *Suppose the solutions of problem (EQP/SCOCP) with the working sets W^{i-1} and W^i are unique. Let (d_x^i, d_u^i) be the solution of problem (EQP/SCOCP) with working set W^{i-1} and the multipliers $(\bar{\lambda}, \bar{\eta}, \bar{\sigma}, \bar{\mu}, \bar{\beta}_{t1}^k, \bar{\upsilon}_{t1})$ satisfy the conditions of Theorem 5.5. Suppose that the multipliers $\bar{\eta}_l$, $(l = 1,...k_1)$ are continuous on the intervals (t_1^l, t_2^l), $(l = 1,...k_1)$ and that the multipliers $\bar{\eta}_{k_1+l}$, $(l = 1,..k_2)$ are p_l-times differentiable on the intervals $(t_1^{k_1+l}, t_2^{k_1+l})$, $(l = 1,..k_2)$.*

Appendix C

Let

$$\Delta d_x^i := \bar{d}_x - d_x^i,$$

$$\Delta d_u^i := \bar{d}_u - d_u^i,$$

where $(\bar{d}_x, \bar{d}_u)$ *is the solution of problem (EQP/SCOCP) with working set* $W^i \subset W^{i-1}$.

If W^i *is obtained from* W^{i-1} *and if an interval* $(t_1, t_2) \subset W_l^{i-1}$, $(1 \leqslant l \leqslant k_1)$ *with* $t_1 < t_2$ *is eliminated from the working set, i.e. if*

$$W^i := W_1^{i-1} \times \ldots \times W_l^{i-1} \setminus (t_1, t_2) \times \ldots W_{k_1}^{i-1} \times \ldots \times W_{k_1+k_2}^{i-1},$$

then

$$\int_{t_1}^{t_2} \bar{\eta}_l(t)(S_{1lx}[t]\Delta d_x^i(t) + S_{1lu}[t]\Delta d_u^i(t))\, dt > 0. \tag{C7}$$

If W^i *is obtained from* W^{i-1} *and if an interval* $[t_1^{k_1+l}, \tilde{t}_1^{k_1+l}) \subset W_{k_1+l}^{i-1}$, $(1 \leqslant l \leqslant k_2)$ *with* $t_1^{k_1+l} < \tilde{t}_1^{k_1+l} < t_2^{k_1+l}$ *is eliminated from the working set, then*

$$\int_{t_1^{k_1+l}}^{\tilde{t}_1^{k_1+l}} \bar{\eta}_{0l}(t) S_{2lx}[t]\Delta d_x^i(t)\, dt + \sum_{j=1}^{p_l} \bar{\nu}_{l1}^{j-1} S_{2lx}^{j-1}[t_1^{k_1+l}]\Delta d_x^i(t_1^{k_1+l}) > 0 \tag{C8}$$

where :

$$\bar{\eta}_{0l}(t) := (-1)^{p_l} \frac{d^{p_l}\bar{\eta}_{k_1+l}(t)}{dt^{p_l}} \quad \text{for all } t_1^{k_1+l} \leqslant t \leqslant t_2^{k_1+l}. \tag{C9}$$

$$\bar{\nu}_{l1}^{j-1} := \bar{\beta}_{l1}^{j} + (-1)^{p_l-j} \frac{d^{p_l-j+1}\bar{\eta}_{k_1+l}}{dt^{p_l-j+1}}(t_1^{k_1+l}+) \quad j = 1,2..p_l. \tag{C10}$$

If W^i *is obtained from* W^{i-1} *and if an interval* $(\tilde{t}_2^{k_1+l}, t_2^{k_1+l}] \subset W_{k_1+l}^{i-1}$, $(1 \leqslant l \leqslant k_2)$ *with* $t_1^{k_1+l} < \tilde{t}_2^{k_1+l} < t_2^{k_1+l}$ *is eliminated from the working set, then*

$$\int_{\tilde{t}_2^{k_1+l}}^{t_2^{k_1+l}} \bar{\eta}_{0l}(t) S_{2lx}[t]\Delta d_x^i(t)\, dt + \sum_{j=1}^{p_l} \bar{\nu}_{l2}^{j-1} S_{2lx}^{j-1}[t_2^{k_1+l}]\Delta d_x^i(t_2^{k_1+l}) > 0 \tag{C11}$$

where : $\bar{\eta}_{0l}(t)$ *is defined by (C9) and*

$$\bar{\nu}_{l2}^{j-1} := -(-1)^{p_l-j} \frac{d^{p_l-j+1}\bar{\eta}_{k_1+l}}{dt^{p_l-j+1}}(t_2^{k_1+l}-) \quad j = 1,2..p_l. \tag{C12}$$

If W^i *is obtained from* W^{i-1} *and if an interval* $(t_1,t_2)\subset W^{i-1}_{k_1+l}$, $(1\leqslant l\leqslant k_2)$ *with* $t_1^{k_1+l}<t_1<t_2<t_2^{k_1+l}$ *is eliminated from the working set, then*

$$\int_{t_1}^{t_2}\bar{\eta}_{0l}(t)S_{2lx}[t]\Delta d_x^i(t)\,dt\ >\ 0 \tag{C13}$$

where : $\bar{\eta}_{0l}(t)$ *is defined by (C9).*

If W^i *is obtained from* W^{i-1} *and if a contact point* $t_3^l\in W^{i-1}_{k_1+l}$, $(1\leqslant l\leqslant k_2)$ *is eliminated from the working set, then*

$$\bar{v}_{l1}S_{2lx}[t_3^l]\Delta d_x^i(t_3^l)\ >\ 0. \tag{C14}$$

Proof : As a notation for the objective function of problem (EQP/SCOCP) we shall use $\bar{f}(d_x,d_u)$. Because (d_x^i,d_u^i) is a solution to problem (EQP/SCOCP) with working set W^{i-1} and multipliers $(\bar{\lambda},\bar{\eta},\bar{\sigma},\bar{\mu},\bar{\beta}^j_{l1},\bar{v}_{l1})$, we have

$$\bar{f}'(d_x^i,d_u^i)(\Delta d_x,\Delta d_u)-\int_0^T\bar{\lambda}^T(\Delta\dot{d}_x-f_x[t]\Delta d_x-f_u[t]\Delta d_u)\,dt+\bar{\sigma}^TD_x[0]\Delta d_x(0)+$$

$$\bar{\mu}^TE_x[T]\Delta d_x(T)+\sum_{l=1}^{k_1}\int_{t_1^l}^{t_2^l}\bar{\eta}_l(t)(S_{1lx}[t]\Delta d_x+S_{1lu}[t]\Delta d_u)\,dt+$$

$$\sum_{l=1}^{k_2}\left\{\int_{t_1^{k_1+l}}^{t_2^{k_1+l}}\bar{\eta}_{k_1+l}(t)(S^{p_l}_{2lx}[t]\Delta d_x+S^{p_l}_{2lu}[t]\Delta d_u)\,dt+\right.$$

$$\left.\sum_{j=1}^{p_l}\bar{\beta}^j_{l1}S^j_{2lx}{}^{-1}[t_1^{k_1+l}]\Delta d_x(t_1^{k_1+l})+\bar{v}_{l1}S_{2lx}[t_3^l]\Delta d_x(t_3^l)\right\}=0,$$

$$\text{for all}\quad \Delta d_x\in W_{1,\infty}[0,T]^r,\ \Delta d_u\in L_\infty[0,T]^m. \tag{C15}$$

Because the solutions of problem (EQP/SCOCP) with the working sets W^{i-1} and W^i are supposed to be unique and the working set is reduced, we have

$$\bar{f}(\bar{d}_x,\bar{d}_u)<\bar{f}(d_x^i,d_u^i), \tag{C16}$$

and hence, ($\bar{f}$ is convex),

$$\bar{f}'(d_x^i,d_u^i)(\Delta d_x^i,\Delta d_u^i)<0. \tag{C17}$$

Because for the direction of search $(\Delta d_x^i,\Delta d_u^i)$, equations (C2), (C3) and (C4) hold, (C15) yields :

$$\sum_{l=1}^{k_1}\int_{t_1^l}^{t_2^l}\bar{\eta}_l(t)(S_{1lx}[t]\Delta d_x^i+S_{1lu}[t]\Delta d_u^i)\,dt+$$

$$\sum_{l=1}^{k_2}\Bigg\{\int_{t_1^{k_1+l}}^{t_2^{k_1+l}}\bar{\eta}_{k_1+l}(t)(S_{2lx}^{P_l}[t]\Delta d_x^i+S_{2lu}^{P_l}[t]\Delta d_u^i)\,dt\;+$$

$$\sum_{j=1}^{P_l}\bar{\beta}_{l1}^{j}S_{2lx}^{j-1}[t_1^{k_1+l}]\Delta d_x^i(t_1^{k_1+l})+\bar{v}_{l1}S_{2lx}[t_3^{l}]\Delta d_x^i(t_3^{l})\Bigg\}$$

$$=-\bar{f}'(d_x^i,d_u^i)(\Delta d_x^i,\Delta d_u^i)>0 \qquad (C18)$$

For all time points which remain in the working set, equality will hold. Therefore (C7) and (C14) follow directly from (C18).
Equations (C8), (C11) and (C13) follow indirectly from (C18) using

$$\int_{t_1^{k_1+l}}^{t_2^{k_1+l}}\bar{\eta}_{k_1+l}(t)(S_{2lx}^{P_l}[t]\Delta d_x^i+S_{2lu}^{P_l}[t]\Delta d_u^i)\,dt+\sum_{j=1}^{P_l}\bar{\beta}_{l1}^{j}S_{2lx}^{j-1}[t_1^{k_1+l}]\Delta d_x^i(t_1^{k_1+l})>0. \qquad (C19)$$

Similar to the integration by parts performed in Section 5.1.1 on (5.1.1.33) this yields :

$$\int_{t_1^{k_1+l}}^{t_2^{k_1+l}}\bar{\eta}_{0l}(t)S_{2lx}[t]\Delta d_x^i\,dt\;+$$

$$\sum_{j=1}^{P_l}\left\{\bar{v}_{l1}^{j-1}S_{2lx}^{j-1}[t_1^{k_1+l}]\Delta d_x^i(t_1^{k_1+l})+\bar{v}_{l2}^{j-1}S_{2lx}^{j-1}[t_2^{k_1+l}]\Delta d_x^i(t_2^{k_1+l})\right\}>0, \qquad (C20)$$

which implies (C8), (C11) and (C13).
□

The results contained in Lemma C1 are used to develope the active set strategy for the case that the working set is to be reduced. We shall consider a number of different cases in the lemmas below.

Lemma C2 : *Assume the hypotheses of Lemma C1 hold,* $l\in\{1,\ldots k_1\}$, $r\in\{1,\ldots\bar{p}_1\}$,

$$\bar{\eta}_l(\bar{t}_r^1)<0, \qquad (C21)$$

and

$$(\bar{t}_{r-1}^1,\bar{t}_{r+1}^1)\subset W_l^{i-1}.$$

If W^i *is obtained from* W^{i-1} *and if the interval* $(\bar{t}_{r-1}^1,\bar{t}_{r+1}^1)$ *is eliminated from the working set and* Δd_u^i *is continuous on the interval* $(\bar{t}_{r-1}^1,\bar{t}_{r+1}^1)$, *then there exist a* $\delta>0$, *such that for all* $0<t_{r+1}^1-t_{r-1}^1\leqslant\delta$.

$$S_{1lx}[\bar{t}_r^1]\Delta d_x^i(\bar{t}_r^1)+S_{1lu}[\bar{t}_r^1]\Delta d_u^i(\bar{t}_r^1)<0. \qquad (C22)$$

Proof : (C7) gives with $t_1=\bar{t}_{r-1}^1$ and $t_2=\bar{t}_{r+1}^1$,

$$\int_{\bar{t}_{r-1}^1}^{\bar{t}_{r+1}^1}\bar{\eta}_l(t)(S_{1lx}[t]\Delta d_x^i+S_{1lu}[t]\Delta d_u^i)\,dt=(\bar{t}_{r+1}^1-\bar{t}_{r-1}^1)\bar{\eta}_l(\bar{t}_r^1)(S_{1lx}[\bar{t}_r^1]\Delta d_x^i(\bar{t}_r^1)+$$

$$S_{1lu}[\bar{t}_r^1]\Delta d_u^i(\bar{t}_r^1)) + o(\bar{t}_{r+1}^1 - \bar{t}_{r-1}^1) < 0, \tag{C23}$$

and hence if $\bar{t}_{r+1}^1 - \bar{t}_{r-1}^1$ is 'sufficiently small' this yields

$$\bar{\eta}_l(\bar{t}_r^1)(S_{1lx}[\bar{t}_r^1]\Delta d_x^i(\bar{t}_r^i) + S_{1lu}[\bar{t}_r^1]\Delta d_u^i(\bar{t}_r^1)) < 0, \tag{C24}$$

which in view of (C21) is equivalent to (C22).

□

We note that in case $\bar{t}_r^1$ is an entry- respectively exit point, an analogous result can be derived under the hypothesis that Δd_u^i is continuous on $(\bar{t}_r^1, \bar{t}_{r+1}^1)$ resp. $(\bar{t}_{r-1}^1, \bar{t}_r^1)$.

Lemma C3 : *Assume the hypotheses of Lemma C1 hold,* $l \in \{1,..k_2\}$, $p_l = 1$ *and* $r \in \{0,....\bar{p}_2\}$.

(i) *Suppose*

$$\bar{\nu}_{l1}^0 < 0, \tag{C25}$$

and

$$\bar{t}_r^2 = t_1^l.$$

If W^i *is obtained from* W^{i-1} *and if the interval* $[\bar{t}_r^2, \bar{t}_{r+1}^2)$ *is eliminated from the working set, then there exists a* $\delta > 0$, *such that for all* $0 < \bar{t}_{r+1}^2 - \bar{t}_r^2 \leqslant \delta$,

$$S_{2lx}[\bar{t}_r^2]\Delta d_x^i(\bar{t}_r^2) < 0. \tag{C26}$$

(ii) *Suppose*

$$\bar{\nu}_{l2}^0 < 0, \tag{C27}$$

and

$$\bar{t}_r^2 = t_2^l.$$

If W^i *is obtained from* W^{i-1} *and if the interval* $(\bar{t}_{r-1}^2, \bar{t}_r^2]$ *is eliminated from the working set, then there exists a* $\delta > 0$, *such that for all* $0 < \bar{t}_r^2 - \bar{t}_{r-1}^2 \leqslant \delta$,

$$S_{2lx}[\bar{t}_r^2]\Delta d_x^i(\bar{t}_r^2) < 0. \tag{C28}$$

Proof : (C8) is used to prove (C26) in the following way

$$\int_{\bar{t}_r^2}^{\bar{t}_{r+1}^2} \bar{\eta}_{0l}(t) S_{2lx}[t]\Delta d_x^i(t)\, dt + \bar{\nu}_{l1}^0 S_{2lx}^{-1}[\bar{t}_r^2]\Delta d_x^i(\bar{t}_r^2) > 0. \tag{C29}$$

This gives

$$\left(\bar{\nu}_{l1}^0 + \tfrac{1}{2}\bar{\eta}_{0l}(\bar{t}_r^2)(\bar{t}_{r+1}^2 - \bar{t}_r^2)\right) S_{2lx}[\bar{t}_r^2]\Delta d_x^i(\bar{t}_r^2) +$$
$$\tfrac{1}{2}\bar{\eta}_{0l}(\bar{t}_{r+1}^2)(\bar{t}_{r+1}^2 - \bar{t}_r^2) S_{2lx}[\bar{t}_{r+1}^2]\Delta d_x^i(\bar{t}_{r+1}^2) + o(\bar{t}_{r+1}^2 - \bar{t}_r^2) > 0. \tag{C30}$$

Because the time point $\bar{t}_{r+1}^2$ is not removed from the working set, the second term is zero, and hence for 'sufficiently small' $\bar{t}_{r+1}^2 - \bar{t}_r^2$ we have

$$\left(\bar{\nu}_{l1}^0 + \tfrac{1}{2}\bar{\eta}_{0l}(\bar{t}_r^2)(\bar{t}_{r+1}^2 - \bar{t}_r^2)\right) S_{2lx}[\bar{t}_r^2]\Delta d_x^i(\bar{t}_r^2) > 0. \tag{C31}$$

Also for 'sufficiently small' $\bar{t}_{r+1}^2 - \bar{t}_r^2$, condition (C25) yields

$$\bar{\nu}_{l1}^{0} + \tfrac{1}{2}\bar{\eta}_{0l}(\bar{t}_r^2)(\bar{t}_{r+1}^2 - \bar{t}_r^2) < 0, \tag{C32}$$

and hence (C31) implies (C26).
(C28) follows similar from (C11).
□

Lemma C4 : *Assume the hypotheses of Lemma C1 hold,* $l \in \{1,..k_2\}$, $p_l = 2$ *and* $r \in \{0,1,...\bar{p}_2\}$.

(i) *Suppose, in addition, that* $\bar{t}_r^2 = t_1^l$,

$$\bar{\nu}_{l1}^{0} - \frac{\bar{\nu}_{l1}^{1}}{\bar{t}_{r+1}^2 - \bar{t}_r^2} < 0, \tag{C33}$$

If W^i *is obtained from* W^{i-1} *and if the interval* $[\bar{t}_r^2, \bar{t}_{r+1}^2)$ *is eliminated from the working set, then there is a* $\delta > 0$, *such that for all* $0 < \bar{t}_{r+1}^2 - \bar{t}_r^2 \leqslant \delta$,

$$S_{2lx}[\bar{t}_r^2]\Delta d_x^i(\bar{t}_r^2) < 0. \tag{C34}$$

(ii) *Suppose, in addition, that* $\bar{t}_r^2 = t_1^l$,

$$\bar{\nu}_{l1}^{0} - \frac{\bar{\nu}_{l1}^{1}}{\bar{t}_{r+1}^2 - \bar{t}_r^2} > 0, \tag{C35}$$

$$\bar{\nu}_{l1}^{1} < 0, \tag{C36}$$

$$[\bar{t}_r^2, \bar{t}_{r+2}^2] \subset W_{k_1+l}^{i-1}. \tag{C37}$$

If W^i *is obtained from* W^{i-1} *and if the interval* $(\bar{t}_r^2, \bar{t}_{r+2}^2)$ *is eliminated from the working set, then there is a* $\delta > 0$, *such that for all* $0 < \bar{t}_{r+2}^2 - \bar{t}_r^2 \leqslant \delta$,

$$S_{2lx}[\bar{t}_{r+1}^2]\Delta d_x^i(\bar{t}_{r+1}^2) < 0. \tag{C38}$$

(iii) *Suppose, in addition, that* $\bar{t}_r^2 = t_2^l$,

$$\bar{\nu}_{l2}^{0} + \frac{\bar{\nu}_{l2}^{1}}{\bar{t}_r^2 - \bar{t}_{r-1}^2} < 0, \tag{C39}$$

If W^i *is obtained from* W^{i-1} *and if the interval* $[\bar{t}_{r-1}^2, \bar{t}_r^2)$ *is eliminated from the working set, then there is a* $\delta > 0$, *such that for all* $0 < \bar{t}_r^2 - \bar{t}_{r-1}^2 \leqslant \delta$,

$$S_{2lx}[\bar{t}_r^2]\Delta d_x^i(\bar{t}_r^2) < 0. \tag{C40}$$

(iv) *Suppose, in addition, that* $\bar{t}_r^2 = t_2^l$,

$$\bar{\nu}_{l2}^{0} + \frac{\bar{\nu}_{l2}^{1}}{\bar{t}_r^2 - \bar{t}_{r-1}^2} > 0, \tag{C41}$$

$$\bar{\nu}_{l2}^{1} > 0, \tag{C42}$$

$$[\bar{t}_{r-2}^2, \bar{t}_r^2] \subset W_{k_1+l}^{i-1}. \tag{C43}$$

If W^i *is obtained from* W^{i-1} *and if the interval* $(\bar{t}_{r-2}^2, \bar{t}_r^2)$ *is eliminated from the working set, then there is a* $\delta > 0$, *such that for all* $0 < \bar{t}_r^2 - \bar{t}_{r-2}^2 \leqslant \delta$,

$$S_{2lx}[\bar{t}_{r-1}^2]\Delta d_x^i(\bar{t}_{r-1}^2) < 0. \tag{C44}$$

Proof : We shall first prove part (i), which follows from (C8). For the case $p_l = 2$, this yields :

$$\int_{\bar{t}_r^2}^{\bar{t}_{r+1}^2} \bar{\eta}_{0l}(t) S_{2lx}[t]\Delta d_x^i(t)\, dt + \bar{\nu}_{l1}^0 S_{2lx}[\bar{t}_r^2]\Delta d_x^i(\bar{t}_r^2) + \bar{\nu}_{l1}^1 S_{2lx}^1[\bar{t}_r^2]\Delta d_x^i(\bar{t}_r^2) > 0. \quad (C45)$$

The last term of (C45) is now considered as :

$$S_{2l}^1[\bar{t}_r^2] + S_{2lx}^1[\bar{t}_r^2]d_x^i(\bar{t}_r^2) = \frac{d}{dt}\left[S_{2l}[t] + S_{2lx}[t]d_x^i(t)\right]_{t=\bar{t}_r^2} \quad (C46)$$

$$S_{2l}^1[\bar{t}_r^2] + S_{2lx}^1[\bar{t}_r^2]\bar{d}_x(\bar{t}_r^2) = \frac{d}{dt}\left[S_{2l}[t] + S_{2lx}[t]\bar{d}_x(t)\right]_{t=\bar{t}_r^2} \quad (C47)$$

((C46) and (C47) are true because d_x^i and $\bar{d}_x$ both satisfy the linear differential system of problem (EQP/SCOCP)). And hence,

$$S_{2lx}^1[\bar{t}_r^2]\Delta d_x^i(\bar{t}_r^2) = \frac{d}{dt}\left[S_{2lx}[t]\Delta d_x^i(t)\right]_{t=\bar{t}_r^2} \quad (C48)$$

An approximation of (C48) is :

$$S_{2lx}^1[\bar{t}_r^2]\Delta d_x^i(\bar{t}_r^2) = \frac{S_{2lx}[\bar{t}_{r+1}^2]\Delta d_x^i(\bar{t}_{r+1}^2) - S_{2lx}[\bar{t}_r^2]\Delta d_x^i(\bar{t}_r^2)}{\bar{t}_{r+1}^2 - \bar{t}_r^2} + o(\bar{t}_{r+1}^2 - \bar{t}_r^2), \quad (C49)$$

which becomes :

$$S_{2lx}^1[\bar{t}_r^2]\Delta d_x^i(\bar{t}_r^2) = -\frac{S_{2lx}[\bar{t}_r^2]\Delta d_x^i(\bar{t}_r^2)}{\bar{t}_{r+1}^2 - \bar{t}_r^2} + o(\bar{t}_{r+1}^2 - \bar{t}_r^2), \quad (C50)$$

because $\bar{t}_{r+1}^2$ remains in the working set.

The remaining terms of (C45) are treated similar as in the proof of Lemma C3, part (i), this gives :

$$\left[\bar{\nu}_{l1}^0 - \frac{\bar{\nu}_{l2}^1}{\bar{t}_{r+1}^2 - \bar{t}_r^2} + \tfrac{1}{2}(\bar{t}_{r+1}^2 - \bar{t}_r^2)\bar{\eta}_{0l}(\bar{t}_r^2)\right] S_{2lx}[\bar{t}_r^2]\Delta d_x^i(\bar{t}_r^2) + o(\bar{t}_{r+1}^2 - \bar{t}_r^2) > 0. \quad (C51)$$

For 'sufficiently small' $\bar{t}_{r+1}^2 - \bar{t}_r^2$ we have

$$\bar{\nu}_{l1}^0 - \frac{\bar{\nu}_{l2}^1}{\bar{t}_{r+1}^2 - \bar{t}_r^2} + \tfrac{1}{2}(\bar{t}_{r+1}^2 - \bar{t}_r^2)\bar{\eta}_{0l}(\bar{t}_r^2) < 0, \quad (C52)$$

whenever (C33) holds. This yields (C34).

To prove part (ii), we consider (C45) with $\bar{t}_{r+1}^2$ replaced by $\bar{t}_{r+2}^2$. Because the time point $\bar{t}_r^2$ will remain in the working set as a contact point we have

$$S_{2lx}[\bar{t}_r^2]\Delta d_x^i(\bar{t}_r^2) = 0. \quad (C53)$$

Therefore (C49) becomes :

$$S_{2lx}^1[\bar{t}_r^2]\Delta d_x^i(\bar{t}_r^2) = \frac{S_{2lx}[\bar{t}_{r+1}^2]\Delta d_x^i(\bar{t}_{r+1}^2)}{\bar{t}_{r+1}^2 - \bar{t}_r^2} + o(\bar{t}_{r+1}^2 - \bar{t}_r^2), \quad (C54)$$

Now (C45) with $\bar{t}_{r+1}^2$ replaced by $\bar{t}_{r+2}^2$ gives :

$$\left|\bar{v}_{l1}^{1} + (\bar{t}_{r+2}^{2} - \bar{t}_{r}^{2})\bar{\eta}_{0l}(\bar{t}_{r+1}^{2})\right| S_{2lx}[\bar{t}_{r+1}^{2}]\Delta d_{x}^{i}(\bar{t}_{r+1}^{2}) + o(\bar{t}_{r+2}^{2} - \bar{t}_{r}^{2}) > 0. \qquad (C55)$$

For 'sufficiently small' $\bar{t}_{r+2}^{2} - \bar{t}_{r}^{2}$ we have

$$\bar{v}_{l1}^{1} + (\bar{t}_{r+2}^{2} - \bar{t}_{r}^{2})\bar{\eta}_{0l}(\bar{t}_{r+1}^{2}) < 0, \qquad (C56)$$

whenever (C36) holds. This yields (C38).

The proofs of parts (iii) and (iv) are omitted because they are straightforward modifications of the proofs of parts (i) and (ii), based on (C11).

□

Lemma C5 : *Assume the hypotheses of Lemma C1 hold,* $l \in \{1,...k_2\}$ *and* $r \in \{1,...\bar{p}_2 - 1\}$. *Suppose in addition that*

$$\bar{\eta}_{0l}(\bar{t}_{r}^{2}) < 0, \qquad (C57)$$

and

$$[\bar{t}_{r-1}^{2}, \bar{t}_{r+1}^{2}] \subset W_{k_1+l}^{i-1}. \qquad (C58)$$

If W^i *is obtained from* W^{i-1} *and if the interval* $(\bar{t}_{r-1}^{2}, \bar{t}_{r+1}^{2})$ *is eliminated from the working set, then there is a* $\delta > 0$, *such that for all* $0 < \bar{t}_{r+1}^{2} - \bar{t}_{r-1}^{2} \leq \delta$,

$$S_{2lx}[\bar{t}_{r}^{2}]\Delta d_{x}^{i}(\bar{t}_{r}^{2}) < 0. \qquad (C59)$$

A proof of Lemma C5 is omitted because it is a direct analogue to the proof of Lemma C2, based on expression (C13).

Lemma C6 : *Assume the hypotheses of Lemma C1 hold,* $l \in \{1,...k_2\}$ *and* $r \in \{0,1,...\bar{p}_2\}$. *Suppose in addition that*

$$\bar{v}_{l1} < 0. \qquad (C60)$$

If W^i *is obtained from* W^{i-1} *and if the time point* t_3^l *is eliminated from the working set, then*

$$S_{2lx}[t_3^l]\Delta d_{x}^{i}(t_3^l) > 0. \qquad (C61)$$

A proof of Lemma C6 is omitted because it follows almost immediate from Lemma C1.

Appendix D : LQ-factorization of the matrix of constraint normals C.

D1 : Structure of the matrix of constraint normals C.

D2 : LQ-factorization of a block banded system using Householder transformations.

D3 : LQ-factorization of the matrix C after modifications in the working set.

This appendix deals with the LQ-factorization of the matrix of constraint normals C (cf. (6.1.2.11)), which is an important issue in the application of the Null space method for the solution of the collocation scheme. The standard approach for dense matrices is to compute the LQ-factorization of an $\bar{m} \times \bar{n}$ matrix by means of Householder transformations. This requires approximately $\bar{m}^2(\bar{n} - \bar{m}/3)$ flops, if $\bar{m} \leq \bar{n}$ (cf. Golub et al. (1983), p.148). In the present case $\bar{m}$ and $\bar{n}$ are 'large' ($\bar{m}, \bar{n} > 100$) which makes the standard approach not feasible. Fortunately, the matrix C is a block banded system for which an LQ-factorization algorithm can be used which preserves its sparsity. In Appendix D1 the structure of the matrix C is considered in more detail. The computation of the LQ-factorization of a block banded system using Householder transformations is thereafter discussed in Appendix D2. For the solution of problem (EIQP/SCOCP/Δ) via Algorithm 5.8, it is necessary to solve a series of problems (EQP/SCOCP), each with a slightly modified working set. It is possible to obtain the LQ-factorization of the modified matrix C in this situation using the LQ-factorization of the matrix C belonging to the previous working set. This is discussed in Appendix D3.

Appendix D1 : Structure of the matrix of constraint normals C.

The matrix C defined by (6.1.2.11) - (6.1.2.13) turns out to have the following structure :

$$C = \left| \begin{array}{cccccc} \bar{C}_1 & & & & & \\ a_1 & \bar{C}_2 & & & & \\ & a_2 & \bar{C}_3 & & & \\ & & a_3 & \ddots & & \\ & & & & \bar{C}_{l_c-1} & \\ & & & & a_{l_c-1} & \bar{C}_{l_c} \\ & & & & & a_{l_c} \end{array} \right| \qquad (D1.1)$$

$$n_1 \quad n_2 \quad n_3 \qquad\qquad n_{l_c-1} \quad n_{l_c}$$

where the matrices $\bar{C}_j$ are $m_j \times n_j$ matrices and $a_j (\geq 0)$ denotes the number of rows of block $\bar{C}_j$ which have no overlap with the rows of block $\bar{C}_{j+1}$. For the last block $\bar{C}_{l_c}$ we define $a_{l_c} := m_{l_c}$. Because C is an $\bar{m} \times \bar{n}$ matrix :

$$\bar{m} = \sum_{j=1}^{l_c} a_j , \qquad (D1.2)$$

$$\bar{n} = \sum_{j=1}^{l_c} n_j . \qquad (D1.3)$$

There are various alternatives for the actual choice of the submatrices $\bar{C}_r$. One possible choice is to set $\bar{C}_r = C_r$ for all r. However, as revealed by the Definition (6.1.2.12) the submatrices C_r still contain a number of trivial elements. One alternative is to split the blocks C_r $(r = 0,1,...p-1)$ into two submatrices $\bar{C}_{2r+1}$ and $\bar{C}_{2r+2}$, where the matrix $\bar{C}_{2r+1}$ contains the first n columns of the block C_r and $\bar{C}_{2r+2}$ the remaining $l(m+n)$ columns. A second alternative is to splitt depending on the upperpart of the last $l(m+n)$ columns C_r into two or more submatrices. For simplicity this road was not followed in the actual implementation. The submatrices are chosen as :

$$\bar{C}_{2r+1} := \begin{bmatrix} K_r \\ N_x[t_r] \\ 0 \\ \cdot \\ \cdot \\ 0 \\ I \\ I \\ \cdot \\ \cdot \\ I \\ I \end{bmatrix} \qquad r = 0,1,.....,p-1, \qquad (D1.4)$$

$$m_{2r+1} := \begin{cases} c + (l+1)n + \sum_{i=1}^{l} \bar{k}(\tau_i) & r = 0 \\ c_r + (l+2)n + \sum_{i=1}^{l} \bar{k}(\tau_{lr+i}) & r = 1,2,.....,p-1 \end{cases} \qquad (D1.5)$$

(The matrices $N_x[t_r]$ are $c_r \times n$ matrices.)

$$n_{2r+1} := n \qquad r = 0,1,.....,p-1, \qquad (D1.6)$$

$$\bar{C}_{2r+2} := \begin{vmatrix} R_x^p[\tau_{lr+1}] & R_u^p[\tau_{lr+1}] & 0 & \cdots & 0 \\ 0 & 0 & \cdot & \cdots & \cdot \\ \cdot & \cdot & \cdot & \cdots & \cdot \\ 0 & 0 & \cdot & \cdots & R_u^p[\tau_{lr+l}] \\ G_{11r} & H_{11r} & G_{12r} & \cdots & H_{1lr} \\ G_{21r} & \cdot & \cdot & \cdots & \cdot \\ \cdot & \cdot & \cdot & \cdots & \cdot \\ \cdot & \cdot & \cdot & \cdots & \cdot \\ G_{l1r} & H_{l1r} & \cdot & \cdots & H_{llr} \\ \bar{G}_{1r} & \bar{H}_{1r} & \cdot & \cdots & \bar{H}_{lr} \end{vmatrix} \qquad r = 0,1,\ldots,p-1, \qquad (D1.7)$$

$$m_{2r+2} := (l+1)n + \sum_{i=1}^{l} \bar{k}(\tau_{lr+i}) \qquad r = 0,1,\ldots,p-1, \qquad (D1.8)$$

$$n_{2r+2} := l(m+n) \qquad r = 0,1,\ldots,p-1, \qquad (D1.9)$$

$$\bar{C}_{2p+1} := \begin{vmatrix} -I \\ E_x[T] \end{vmatrix}, \qquad (D1.10)$$

$$m_{2p+1} := n+q, \qquad (D1.11)$$

$$n_{2p+1} := n \qquad (D1.12)$$

The total number of submatrices $\bar{C}_j$ is

$$l_c := 2p+1. \qquad (D1.13)$$

The numbers a_j are :

$$a_{2r+1} := \begin{cases} c & r = 0 \\ n + c_r & r = 1,2,\ldots p-1 \end{cases} \qquad (D1.14)$$

$$a_{2r+2} := ln + \sum_{i=1}^{l} \bar{k}(\tau_{lr+i}), \qquad (D1.15)$$

$$a_{2p+1} := n + q, \qquad (D1.16)$$

If the matrix C is stored in the same way as dense matrices are stored, then the storage would require $\bar{m} \cdot \bar{n}$ locations. Because the matrix C is large and sparse, i.e. $\sum m_j \cdot n_j \ll \bar{m} \cdot \bar{n}$, this would be rather inefficient. In view of the fact that the LQ-factorization exploits the block structure of the matrix C an obvious choice is to store the matrices $\bar{C}_j$ as dense matrices.

Appendix D2 : LQ-factorization of a block banded system using Householder transformations.

In this appendix the LQ-factorization of a block banded system, i.e. the matrix C, will be considered. Thereto to the notations and terminology of Appendix D1 are adopted.

For the sake of completeness we shall first recapitulate the Householder transformation which is used to zero a number of elements of an $\tilde{n}$-vector v (e.g. cf. Golub et al. (1983) or Lawson et al. (1974)).

Essentially a Householder transformation applied to an $\tilde{n}$-vector is an $\tilde{n} \times \tilde{n}$ orthogonal matrix of the form :

$$Q = I_{\tilde{n}} + b^{-1} u u^T, \tag{D2.1}$$

where u is an $\tilde{n}$-vector and $b = -\|u\|^2/2$.

$$v = \begin{vmatrix} v_1 \\ \cdot \\ \cdot \\ v_{p-1} \\ v_p \\ v_{p+1} \\ \cdot \\ \cdot \\ v_{l_1-1} \\ v_{l_1} \\ \cdot \\ \cdot \\ v_{l_2} \\ v_{l_2+1} \\ \cdot \\ \cdot \\ v_{\tilde{n}} \end{vmatrix} \qquad Q \cdot v = \begin{vmatrix} v_1 \\ \cdot \\ \cdot \\ v_{p-1} \\ \bar{v}_p \\ v_{p+1} \\ \cdot \\ \cdot \\ v_{l_1-1} \\ 0 \\ \cdot \\ \cdot \\ 0 \\ v_{l_2+1} \\ \cdot \\ \cdot \\ v_{\tilde{n}} \end{vmatrix} \tag{D2.2}$$

The effect of the matrix Q in transforming the vector v, is depicted by (D2.2) and can be described by means of three nonnegative integers p, l_1 and l_2 (with $p < l_1 \leqslant l_2$) as follows :

1) If $p > 1$, then the components $v_1, \ldots, v_{p-1}$ are to be left unchanged.

2) Component v_p is permitted to change and is called the *pivot element*.

3) If $p < l_1 - 1$, then components $v_{p+1}, \ldots, v_{l_1-1}$ are to be left unchanged.

4) If $l_1 \leqslant l_2$, then the components $v_{l_1}, \ldots, v_{l_2}$ are to be zeroed.

5) If $l_2 < \tilde{n}$, then components $v_{l_2+1}, \ldots, v_{\tilde{n}}$ are to be left unchanged.

The components of the vector u and the factor b, necessary to compute the Householder matrix $Q(p, l_1, l_2)$, which has the above mentioned properties follow from the algorithm below (cf. Lawson et al. (1974)) :

Algorithm D1 (p,l_1,l_2,v,b,u,$\tilde{n}$)

(i) $s := -sign(v_p)\left|v_p^2 + \sum_{i=l_1}^{l_2} v_i^2\right|^{\frac{1}{2}}$

(ii) $u_i := 0 \qquad i = 1,\ldots,p-1$

(iii) $u_p := v_p - s$

(iv) $u_i := 0 \qquad i = p+1,\ldots,l_1-1$

(v) $u_i := v_i \qquad i = l_1,\ldots,l_2$

(vi) $u_i := 0 \qquad i = l_2+1,\ldots,\tilde{n}$

(vii) $b := su_p$

(viii)

$$Q(p,l_1,l_2) := \begin{cases} I_{\tilde{n}} + b^{-1}uu^T & \text{if } b \neq 0 \\ I_{\tilde{n}} & \text{if } b = 0 \end{cases}$$

In most cases it suffices when matrix-vector products of the form $Q \cdot v$ can be computed. We note that because Q is symmetric we have $(Q \cdot v)^T = v^T \cdot Q$. Using the vector u and the factor b as computed by Algorithm D1, the multiplication $Q \cdot v$ can efficiently make use of the special structure of the matrix Q, as follows :

$$Q \cdot v = v + b^{-1}uu^T v = v + tu, \qquad (D2.3)$$

with :

$$t = (u^T v)/b. \qquad (D2.4)$$

Because matrix-matrix products of the form $Q \cdot A$ and $A \cdot Q$ consist of a number of matrix-vector products, this type of multiplication allows a similar use of the structure of the matrix Q.

As a first step towards the LQ-factorization of the matrix C we will consider the LQ-factorization of the block banded system (D1.1) using the standard procedure for dense matrices, which may be described as follows :

Algorithm D2

$C^0 := C$

For $j := 1$ *to* $\bar{m}$

do

Calculate a Householder transformation $Q_j(j,j+1,\bar{n})$ *that zeroes the elements* $(j,j+1),\ldots,(j,\bar{n})$ *of the matrix* $C^j \cdot Q_j(j,j+1,\bar{n})$.

Calculate $C^{j+1} := C^j \cdot Q_j(j,j+1,\bar{n})$.

od

In order to give a simple description of the inefficiency of Algorithm D2 for the LQ-factorization of the matrix C, we consider the following slightly different form of a banded system, which is also denoted as the matrix C (strictly speaking it is a special case

of the matrix (D1.1), where the submatrices $\tilde{D}_j$ contain trivial elements).

$$C = \left|\begin{array}{cccccccc} \tilde{C}_1 & \tilde{D}_1 & & & & & & \tilde{m} \\ & \tilde{C}_2 & \tilde{D}_2 & & & & & \tilde{m} \\ & & \tilde{C}_3 & \ddots & & & & \tilde{m} \\ & & & \ddots & \ddots & & & \tilde{m} \\ & & & & \ddots & \ddots & & \tilde{m} \\ & & & & & \tilde{D}_{l_c-1} & & \tilde{m} \\ & & & & & \tilde{C}_{l_c} & & \tilde{m} \\ \tilde{n} & \tilde{n} & \tilde{n} & & & \tilde{n} & & \end{array}\right| \qquad (D2.5)$$

The matrices $\tilde{C}_j$ and $\tilde{D}_j$ are $\tilde{m} \times \tilde{n}$ matrices, with $\tilde{m} \leqslant \tilde{n}$.

Lemma D3 : *If Algorithm D2 is used to triangularize the block banded system (D2.5), then the matrix $C^{i\tilde{m}}$, i.e. the matrix C^j after i times $\tilde{m}$ orthogonalization steps, $(1 \leqslant i < l_c)$, has the following form*

$$C^{i\tilde{m}} = \left|\begin{array}{ccccccccc} L_1 & & & & & & & & \tilde{m} \\ F_1 & L_2 & & & & & & & \tilde{m} \\ & F_2 & \ddots & & & & & & \tilde{m} \\ & & \ddots & L_i & & & & & \tilde{m} \\ & & & F_i & G_i & \bar{C}_{i+1}^{i\tilde{m}} & \tilde{D}_{i+1} & & \tilde{m} \\ & & & & & & \tilde{C}_{i+2} & \ddots & \tilde{m} \\ & & & & & & & \ddots & \\ & & & & & & & \tilde{D}_{l_c-1} & \tilde{m} \\ & & & & & & & \tilde{C}_{l_c} & \tilde{m} \\ \tilde{m} & \tilde{m} & \ddots & \tilde{m} & i(\tilde{n}-\tilde{m}) & \tilde{n} & \tilde{n} & \ddots & \tilde{n} \end{array}\right| \qquad (D2.6)$$

where the submatrices L_j are $\tilde{m} \times \tilde{m}$ lowertriangular matrices $(j = 1,\ldots,i)$, the submatrices F_j $(j = 1,\ldots,i)$ are $\tilde{m} \times \tilde{m}$ matrices and the matrix G_i is an $\tilde{m} \times i(\tilde{n}-\tilde{m})$ matrix.

Proof : The proof is given by induction. Therefore the case $i = 1$ is considered first.

The kth row of the matrix C^j is denoted by c_k^j. The rows of the matrix C^1 satisfy :

$$c_k^1 := c_k^0 + u^1(u^{1T} c_k^0 / b), \qquad (D2.7)$$

where the vector u^1 is calculated by Algorithm D1 and thus satisfies :

$$u_l^1 = 0 \qquad \text{for all } l = 2\tilde{n}+1,\ldots,\bar{n}. \tag{D2.8}$$

Because

$$c_{kl}^0 = 0 \qquad \text{for all } k = 2\tilde{m}+1,\ldots\bar{m} \text{ and } l = 1,\ldots,2\tilde{n}, \tag{D2.9}$$

we obtain

$$c_k^1 = c_k^0 \qquad \text{for all } k = 2\tilde{m}+1,\ldots,\bar{m}. \tag{D2.10}$$

As a consequence of (D2.8), the elements of the columns $2\tilde{n}+1,\ldots,\bar{n}$ will remain unchanged.

Thus 'fill-in' is generated in the rows $\tilde{m}+1,\ldots,2\tilde{m}$ and columns $1,\ldots,\tilde{n}$. The matrices $\tilde{D}_2,\ldots,\tilde{D}_{l_c-1}$ and $\tilde{C}_3,\ldots,\tilde{C}_{l_c}$ remain unchanged.

The orthogonalization steps for $l = 2,\ldots,\tilde{m}$ are essentially the same as this first step, because the block structure of C^1 is almost the same as the structure of C^0.
After $\tilde{m}$ steps we have :

$$\begin{bmatrix} \tilde{C}_1 & \tilde{D}_1 \\ 0 & \tilde{C}_2 \end{bmatrix} \rightarrow \begin{bmatrix} L_1 & 0 & 0 \\ F_1 & G_1 & \bar{C}_2^{\tilde{m}} \end{bmatrix}$$

i.e. the matrices F_1 and G_1 represent the 'fill-in' in the rows $\tilde{m}+1,\ldots,2\tilde{m}$ and columns $1,\ldots\tilde{n}$. The dimension of the matrix G_1 is $\tilde{m}\times(\tilde{n}-\tilde{m})$.

To prove the induction step $i \rightarrow i+1$ we use the following result :

> "The first $i\tilde{m}$ rows and columns of $C^{i\tilde{m}}$ and $C^{(i+1)\tilde{m}}$ are identical." (cf. Tewarson (1973)).

Because $i \leqslant l_c - 1$ it suffices to consider the triangularization of

$$E = \left|\begin{array}{ccccccc} G_i & \bar{C}_{i+1}^{i\tilde{m}} & \tilde{D}_{i+1} & & & & \tilde{m} \\ & & \tilde{C}_{i+2} & \ddots & & & \tilde{m} \\ & & & \ddots & \ddots & & \\ & & & & & \tilde{D}_{l_c-1} & \tilde{m} \\ & & & & & \tilde{C}_{l_c} & \tilde{m} \\ i(\tilde{n}-\tilde{m}) & \tilde{n} & \tilde{n} & \ddots & \ddots & \tilde{n} & \end{array}\right| \tag{D2.11}$$

The approach is now essentially the same as before. In the first step the vector u satisfies :

$$u_k = 0 \qquad k > i(\tilde{n}-\tilde{m})+2\tilde{n}. \tag{D2.12}$$

Because only the first $2\tilde{m}$ rows of E contain nonzero entries in the columns $1,\ldots,i(\tilde{n}-\tilde{m})+2\tilde{n}$, fill-in will only be generated in rows $\tilde{m}+1,\ldots,2\tilde{m}$ and columns $1,\ldots,i(\tilde{n}-\tilde{m})+2\tilde{m}$. Observing that this proces is essentially the same for the steps $j = 2,\ldots,\tilde{m}$, we obtain the desired result. We note that during these steps the total amount of fill-in has increased with $\tilde{m}(\tilde{n}-\tilde{m})$.

□

The result of Lemma D3 indicates that during the factorization proces of C, fill-in is generated in a way that, if $\tilde{m} < \tilde{n}$, large nonzero submatrices are generated. Fortunately it is

possible to modify the procedure such that this problem is circumvented. This modification was invented by Reid (1967) and makes use of the block structure of the matrix C as depicted by (D1.1).

The essence of the approach is that instead of zeroing all elements of a row with one Householder transformation, the elements of a row are zeroed by several Householder transformations. Each of these Householder transformations is constructed so that it zeroes all elements on one row of one specific block and leaves the elements of all other blocks unaffected. For the statement of the Algorithm D3 we need the following terminology. Suppose that the nonzero elements of the jth row of the matrix C are in the submatrices $\bar{C}_i$ and $\bar{C}_{i+1}$ and that the column indices, relative to the matrix C, of the first and the last column of the submatrix $\bar{C}_i$ are respectively i_1 and i_2. The column index of the last column of the matrix $\bar{C}_{i+1}$ is denoted by i_3. Algorithm D2 is modified into :

Algorithm D3

$C^0 := C$

For $j := 1$ *to* $\bar{m}$

do

Calculate a Householder transformation $Q_j^1(j,i_1,i_2)$ *that zeroes the elements* $(j,i_1),\ldots,(j,i_2)$ *of the matrix* $C^j \cdot Q_j^1(j,i_1,i_2)$.

Calculate a Householder transformation $Q_j^2(j,i_2+1,i_3)$ *that zeroes the elements* $(j,i_2+1),\ldots,(j,i_3)$ *of the matrix* $C^j \cdot Q_j^1(j,i_1,i_2) \cdot Q_j^2(j,i_2+1,i_3)$.

Calculate $C^{j+1} := C^j \cdot Q_j^1(j,i_1,i_2) \cdot Q_j^2(j,i_2+1,i_3)$.

od

Reffering to the proof of Lemma D3, we observe that in Algorithm D3 the vector u for the Householder transformations is chosen so that during this proces only fill-in is generated in the pivotal column. The triagularization of the matrix C follows essentially the same pattern as in Lemma D3, with the matrix G_i containing only zeroes.

This approach has the following two advantages if $\tilde{m} < \tilde{n}$:

1) There is a considerable saving in flops.
   In the terminology of Lemma D3, using the standard approach the elements of the submatrix G_i must all be zeroed (cf. Reid (1967)).
2) Except for the pivot elements u_p, the nontrivial elements of the vectors u can be stored by overwriting the entries of the matrix C, similar to the standard procedure with Householder triangularization of dense matrices. This is possible because the matrix G_i contains only trivial elements and hence requires no storage.

In the actual implementation of the LQ-factorization the matrix L is formed explicitly. This matrix can be stored efficiently by taking the sparse block structure into account. A simple analysis reveals that, except in very special cases, the matrix Q is a dense matrix. Because of this nonsparsity the matrix Q is not formed explicitly, but it is stored in factored form, i.e. the vectors u defining the Householder matrices Q_j^1 and Q_j^2 are stored. Hence the storage of the Householder factors requires the same amount of storage as the storage of the matrix C.

Appendix D3 : LQ-factorization of the matrix C after modifications in the working set.

The solution of problem (EIQP/SCOCP/Δ) requires, in general, the solution of several problems (EQP/SCOCP) with a slightly modified working set. A numerical approximation to the solution of problem (EQP/SCOCP) is obtained as the solution of the collocation scheme. A modification of the working set of problem (EQP/SCOCP) translates into modifications in the matrix of constraint normals C.
We mention the following possible modifications and their influences on the matrix C, that follow immediate from Section 6.1.2 :

Modification of the working set of problem (EQP/SCOCP)	Modification of the matrix C
A mixed control state constraint changes from inactive to active at a collocation point	A row is added to C
A state constraint becomes a contact point at a grid point	A row is added to C
A boundary arc of a state constraint is expanded with one grid interval	l rows are added to C
A state constraint has a contact point which changes into a boundary arc of one grid interval	$l+(p_j-1)$ rows are added to C (p_j is the order of the state constraint)

In the table above the modifications are all constraints which change from inactive to active. A similar table can be made up for the reverse case, i.e. constraints which change from active to inactive. The resulting modification of the matrix C is in this case that rows are deleted from the matrix C. We note that modifications of the working set of a state constraint may result in a modification of the matrix C of more than one row.

In linearly constrained optimization it is common practice to make use of the previous factorization of the matrix of constraint normals, with the calculation of the factorization of the modified matrix of constraint normals. We do not intend to give a survey on methods for the calculation of these updated LQ-factorizations, for this we refer to Gill et al. (1974a), Golub et al. (1983, p.437) and Lawson et al. (1974, p.174 and p.208). Most of these techniques focus on calculating an update for the matrix L. The matrix Q is considered to be either explicitly formed, or to be discarded completely, immediately after the factorization.

In the present case however, the matrix Q, which can only be stored in factored form, playes a key role in the Null space method. Because it is our desire to preserve the sparsity properties of the factored form of the matrix Q, a suitable way of updating the factorization is to 'restart' the LQ-factorization algorithm at a suitable point. We shall outline the method first without making explicit reference to the sparsity of the matrix C.

Let

$$C^{old} = [L^{old}\ 0]\left[Q^{old}\right]^T \tag{D3.1}$$

and

$$C^{old} = \begin{bmatrix} C_1^{old} \\ C_2^{old} \end{bmatrix} \qquad C^{new} = \begin{bmatrix} C_1^{old} \\ C_2^{new} \end{bmatrix} \tag{D3.2}$$

where C_1^{old} is an $\bar{m}_1 \times \bar{n}$ matrix. The LQ-factorization of the matrix C^{old} is known and the rows $\bar{m}_1, \ldots, \bar{m}$ of the matrix C are modified.

The LQ-factorization (D3.1) is now rewritten as :

$$\begin{bmatrix} C_1^{old} \\ C_2^{old} \end{bmatrix} \cdot Q_1^{old} \cdot Q_2^{old} = \begin{bmatrix} L_1^{old} & 0 \\ L_2^{old} & 0 \end{bmatrix} \tag{D3.3}$$

where the matrix Q_1^{old} is the product of the Householder transforms which were used to obtain L_1^{old}, i.e.

$$C_1^{old} Q_1^{old} = L_1^{old}. \tag{D3.4}$$

Now consider

$$C^{new} Q_1^{old} = \begin{bmatrix} C_1^{old} \\ C_2^{new} \end{bmatrix} Q_1^{old} = \begin{bmatrix} L_1^{old} & 0 \\ C_2^{new} Q_1^{old} & \end{bmatrix} \tag{D3.5}$$

Once the matrix $C_2^{new} Q_1^{old}$ is calculated, the LQ-factorization proces can be restarted with the triangularization of row $\bar{m}_1+1$. We note that this method is essentially the 'removal part' of method 2 of Lawson et al. (1974) (p.177-178).

Now consider this method for updating the LQ-factorization of the block banded system (D1.1). In the implementation of the method, a copy of the matrix C is preserved. When the working set is modified, this copy is modified first. The LQ-factorization of the previous matrix C is thereafter updated using this modified matrix.

For the calculation of the product $C_2^{new} Q_1^{old}$, we consider the actual block structure of the matrix C which follows from Appendix D1.

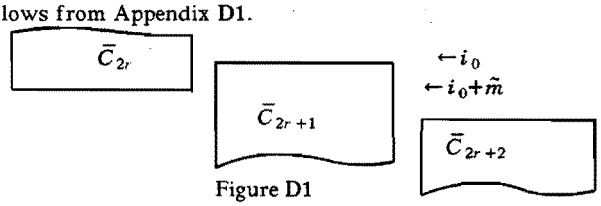


Figure D1

The blocks $\bar{C}_{2r+1}$ and $\bar{C}_{2r+2}$ contain the coefficients of the linear equations, corresponding to the constraints on the grid interval $[t_r, t_{r+1})$.

If the factorization proces is to be restarted at row $i_0+\tilde{m}$ as depicted above, the calculation of the matrix $C_2^{new} Q_1^{old}$ involves only Householder transformations used in the previous factorization proces for the triangularization of rows $i_0, i_0+1, \ldots, i_0+\tilde{m}-1$. When the factorization proces would be restarted at another point, this would involve also Householder transformations from other blocks. (Note : the row of the blocks $\bar{C}_{2r}$ and $\bar{C}_{2r+2}$ never overlap.) Because this strategy allows a simple implementation, this strategy was adopted for implementation.

Appendix E : Computational details.

E1 : Calculation of the Lagrange multipliers for the active set strategy.

E2 : Approximation of the Lagrange multipliers of problem (EIQP/SCOCP).

E3 : Calculation of the matrices M_2, M_3 and M_4.

E4 : Strategy in case of rank deficiency of the matrix of constraint normals.

E5 : Automatic adjustment of the penalty constant of the merit function.

E6 : Computation of the merit function.

E7 : Miscellaneous details.

The Appendices E1 - E7 deal with a number of computational details of rather specialized nature. In Appendix E1 the computation of the Lagrange multipliers, required for the active set strategy of Algorithm 5.8, is discussed. The computation of the Lagrange multipliers, which are used for the computation of the merit function, is discussed in Appendix E2. The matrices M_2, M_3 and M_4 (cf. (4.2.1.12) - (4.2.1.14)) can be computed in two different ways, this is discussed in Appendix E3. In Appendix E4 the case of rank deficiency of the matrix of constraint normals, which may arise during the execution of Algorithm 5.8, is considered. A procedure for the automatic adjustment of the penalty constant of the merit function is given in Appendix E5 and the computation of the merit function is discussed in Appendix E6. Appendix E7 deals with some details related to the implementation of the method.

Appendix E1 : Calculation of the Lagrange multipliers for the active set strategy.

For the solution of problem (EIQP/SCOCP/Δ), more specifically for the active set strategy (cf. Section 5.2), the Lagrange multipliers $(\bar{\eta}_{0k}, \bar{\nu}_{k1}^{j-1}, \bar{\nu}_{k2}^{j-1})$ are required. These multipliers are related to the multipliers $\bar{\eta}_{k_1+k}$ and $\bar{\beta}_{kl}^{j}$, which are obtained via the solution of the linear multipoint boundary value problem, by (5.2.23), (5.2.26) and (5.2.27), i.e.

$$\bar{\eta}_{0k}(t) := (-1)^{p_k} \frac{d^{p_k} \bar{\eta}_{k_1+k}(t)}{dt^{p_k}} \quad \text{for all} \quad \bar{t}_1^{k_1+k} + \leqslant t \leqslant \bar{t}_2^{k_1+k} - \quad k = 1,\ldots,k_2, \quad (E1.1)$$

$$\bar{\nu}_{kl}^{j-1} := \bar{\beta}_{kl}^{j} + (-1)^{p_k-j} \frac{d^{p_k-j+1}\bar{\eta}_{k_1+k}}{dt^{p_k-j+1}}(\bar{t}_1^{k_1+k}+) \quad j = 1,\ldots,p_k \quad k = 1,\ldots,k_2, \quad (E1.2)$$

$$\bar{\nu}_{k2}^{j-1} := -(-1)^{p_k-j} \frac{d^{p_k-j+1}\bar{\eta}_{k_1+k}}{dt^{p_k-j+1}}(\bar{t}_2^{k_1+k}-) \quad j = 1,\ldots,p_k \quad k = 1,\ldots,k_2, \quad (E1.3)$$

where p_k is the order of the state constraint S_{2k}. (As in Appendix C, it is assumed that the working set S_{2k} has only one boundary arc $[\bar{t}_1^{k_1+k}, \bar{t}_2^{k_1+k}]$.)

The collocation method yields a numerical approximation to the multipliers $\bar{\eta}_{k_1+k}$ at the collocation points τ_{lr+i}. Because a boundary interval contains at least one grid interval $[t_r, t_{r+1}]$ and each grid interval contains l collocation points, there are at least l values $\bar{\eta}_{k_1+k}$ available.

To obtain a numerical approximation to the multipliers $(\bar{\eta}_{0k}, \bar{\nu}_{k1}^{j-1}, \bar{\nu}_{k2}^{j-1})$ from (E1.1) - (E1.3), a numerical approximation to the time function $\bar{\eta}_{k_1+k}(t)$ is required on the entire interval $[\bar{t}_1^{k_1+k}, \bar{t}_2^{k_1+k}]$.

One possible approach is to approximate the function $\bar{\eta}_{k_1+k}(t)$ on the grid intervals with an $(l-1)$th order polynomial. However, this approximation will in general be discontinuous at the grid points t_r. It is reasonable to expect that $\bar{\eta}_{k_1+k}(t)$ is a C^{p_k}-function on $(\bar{t}_1^{k_1+k}, \bar{t}_2^{k_1+k})$, i.e. $\bar{\eta}_{k_1+k}(t)$ is continuous at the time points $t_r \in (\bar{t}_1^{k_1+k}, \bar{t}_2^{k_1+k})$.

Therefore a more logical choice is to consider an interpolation of $\bar{\eta}_{k_1+k}(t)$ over the entire interval $(\bar{t}_1^{k_1+k}, \bar{t}_2^{k_1+k})$. In the implementation $\bar{\eta}_{k_1+k}(t)$ is approximated using a cubic spline (cf. de Boor, (1978)) over the entire boundary interval. This interpolation technique is suitable for dealing with the cases $p_k = 1$ and $p_k = 2$, because a cubic spline has continuous first and second derivatives. For cases with $p_k > 2$, a higher order spline interpolation should be used, because in general, the third derivative of a cubic spline has discontinuities.

Appendix E2 : Approximation of the Lagrange multipliers of problem (EIQP/SCOCP).

In this Appendix we shall consider the calculation of approximations to the Lagrange multipliers of problem (EIQP/SCOCP), as they are required for the calculation of the merit function.

First consider the exact solution of problem (EIQP/SCOCP), which is also a special case of problem (EQP/SCOCP). Using the multipliers defined by (E1.1) - (E1.3), the Lagrange multipliers corresponding to the state constraints of problem (EIQP/SCOCP) satisfy :

$$\tilde{\eta}_k(t) = \bar{\eta}_{0k}(t) \quad \text{for all} \quad \bar{t}_1^{k_1+k} \leqslant t \leqslant \bar{t}_2^{k_1+k} \quad k = 1,\ldots,k_2, \tag{E2.1}$$

$$\tilde{\nu}_{k1} = \bar{\nu}_{k1}^0 \quad k = 1,\ldots,k_2, \tag{E2.2}$$

$$\tilde{\nu}_{k2} = \bar{\nu}_{k2}^0 \quad k = 1,\ldots,k_2. \tag{E2.3}$$

For this solution the multipliers $(\bar{\nu}_{k1}^{j-1}, \bar{\nu}_{k2}^{j-1})$ $(j = 2,\ldots p_k)$ must satisfy (cf. (3.3.6.2) - (3.3.6.6)) :

$$\bar{\nu}_{k1}^{j-1} = 0 \quad j = 2,\ldots,p_k \quad k = 1,\ldots,k_2, \tag{E2.4}$$

$$\bar{\nu}_{k2}^{j-1} = 0 \quad j = 2,\ldots,p_k \quad k = 1,\ldots,k_2, \tag{E2.5}$$

Instead of solving problem (EIQP/SCOCP) exactly, the solution of problem (EIQP/SCOCP) is approximated, by using the solution of problem (EIQP/SCOCP/Δ). Based on (E2.1) - (E2.3) we use the multipliers $(\bar{\eta}_{0k}, \bar{\nu}_{k1}^0, \bar{\nu}_{k2}^0)$ as approximations to the Lagrange multipliers corresponding to the state constraints of problem (EIQP/SCOCP). Thus it is neglected that (E2.4) and (E2.5) may not hold exactly.

We now consider the adjoint variable of problem (EIQP/SCOCP). Similar to the approach followed above we first consider the exact solution of problem (SCOCP). In this case the adjoint variable $\bar{\lambda}$, which is obtained as a solution to the linear multipoint boundary value problem of Section 5.1.3, satisfies the conditions of Theorem 3.16 for $i = p$. The adjoint variable which satisfies the conditions of Theorem 3.16 for $i = 0$ may thus be obtained as (cf. (3.3.6.2) - (3.3.6.6)) :

$$\tilde{\lambda}(t) = \bar{\lambda}(t) + \sum_{k=1}^{k_2} \sum_{j=1}^{p_k} (-1)^{p_k - j} \frac{d^{p_k - j} \bar{\eta}_{k_1 + k}(t)}{dt^{p_k - j}} S^{j}_{2kx}{}^{-1}[t]^T \quad 0 \leqslant t \leqslant T. \tag{E2.6}$$

It is this multiplier that is used for the calculation of the merit function.

The multipliers $\tilde{\eta}_1$, $\tilde{\sigma}$ and $\tilde{\mu}$ corresponding to the solution of problem (EIQP/SCOCP) are approximated by the multipliers $\bar{\eta}_1$, $\bar{\sigma}$ and $\bar{\mu}$, which are directly obtained as the solution of the linear multipoint boundary value problem.

Appendix E3 : Calculation of the matrices M_2, M_3 and M_4.

In this appendix the calculation of the Hessian of the Lagrangian, more specifically of the matrices M_2, M_3 and M_4 is considered.

We recall that the matrices M_2, M_3 and M_4 are defined by (4.2.1.12) - (4.2.1.14) as : †

$$M_2[t] := f_{0xx}[t] + \lambda(t) * f_{xx}[t] + \eta_1(t) * S_{1xx}[t] \quad 0 \leqslant t \leqslant T. \tag{E3.1}$$

$$M_3[t] := f_{0xu}[t] + \lambda(t) * f_{xu}[t] + \eta_1(t) * S_{1xu}[t] \quad 0 \leqslant t \leqslant T. \tag{E3.2}$$

$$M_4[t] := f_{0uu}[t] + \lambda(t) * f_{uu}[t] + \eta_1(t) * S_{1uu}[t] \quad 0 \leqslant t \leqslant T. \tag{E3.3}$$

We note that in the definition of the matrix M_2 use was made of the assumption done in Chapter 5 :

$$S^{j}_{2kxx}[t] = 0 \quad \textit{for all} \quad j = 0,1,\ldots,p_k - 1 \quad k = 1,\ldots,k_2. \tag{E3.4}$$

The multiplier λ is the multiplier whose calculation was discussed in Appendix E2 and is computed by (E2.6).

The following lemma shows that the matrices M_2, M_3 and M_4 can also be calculated using multipliers $(\bar{\lambda}, \bar{\eta})$.

Lemma E1 : *If*

$$S^{k}_{2jxx}[t] = 0 \quad \textit{for all} \quad k = 0,1,\ldots,p_j - 1 \quad j = 1,\ldots,k_2, \tag{E3.5}$$

and

$$\lambda(t) = \bar{\lambda}(t) + \sum_{k=1}^{k_2} \sum_{j=1}^{p_k} (-1)^{p_k - j} \frac{d^{p_k - j} \bar{\eta}_{k_1 + k}(t)}{dt^{p_k - j}} S^{j}_{2kx}{}^{-1}[t]^T \quad 0 \leqslant t \leqslant T, \tag{E3.6}$$

then

$$M_2[t] = f_{0xx}[t] + \bar{\lambda}(t) * f_{xx}[t] + \bar{\eta}_1(t) * \tilde{S}^{p}_{xx}[t] \quad 0 \leqslant t \leqslant T, \tag{E3.7}$$

$$M_3[t] = f_{0xu}[t] + \bar{\lambda}(t) * f_{xu}[t] + \bar{\eta}_1(t) * \tilde{S}^{p}_{xu}[t] \quad 0 \leqslant t \leqslant T, \tag{E3.8}$$

$$M_2[t] = f_{0xu}[t] + \bar{\lambda}(t) * f_{xu}[t] + \bar{\eta}_1(t) * \tilde{S}^{p}_{xu}[t] \quad 0 \leqslant t \leqslant T, \tag{E3.9}$$

where $\tilde{S}^{p}$ is defined by (3.3.5.11).

Proof : To prove (E3.7) - (E3.9) we have to show that

† For the sake of brevity the iteration index i was omitted for the multipliers.

$$\lambda(t)*f_{xx}[t] = \bar{\lambda}(t)*f_{xx}[t] + \bar{\eta}_2(t)*S^p_{2xx}[t] \qquad 0 \leqslant t \leqslant T, \tag{E3.10}$$

$$\lambda(t)*f_{xu}[t] = \bar{\lambda}(t)*f_{xu}[t] + \bar{\eta}_2(t)*S^p_{2xu}[t] \qquad 0 \leqslant t \leqslant T, \tag{E3.11}$$

$$\lambda(t)*f_{xu}[t] = \bar{\lambda}(t)*f_{xu}[t] + \bar{\eta}_2(t)*S^p_{2xu}[t] \qquad 0 \leqslant t \leqslant T, \tag{E3.12}$$

where $\bar{\eta}_2$ denotes the last k_2 components of the vector $\bar{\eta}$ and S^p_{2xx}, S^p_{2xu} and S^p_{2uu} denote the Hessian of the last k_2 components of the vector $\tilde{S}^p$.

Considering (E3.10), using (E3.6) we obtain :

$$\lambda(t)*f_{xx}[t] = \bar{\lambda}(t)*f_{xx}[t] + \sum_{k=1}^{k_2}\sum_{j=1}^{p_k}(-1)^{p_k-j}\frac{d^{p_k-j}\,\bar{\eta}_{k_1+k}(t)}{dt^{p_k-j}}S^{j-1}_{2kx}[t]^T f_{xx}[t] \qquad 0 \leqslant t \leqslant T, \tag{E3.13}$$

From Section 3.3.5 we recall the definition of S^j :

$$S^j_{2k} := S_{2kt} + S^{j-1}_{2kx} f \qquad j = 1,\ldots,p_k, \tag{E3.14}$$

and hence

$$S^j_{2kx} = S^{j-1}_{2ktx} + S^{j-1}_{2kxx} f + S^{j-1}_{2kx} f_x \qquad j = 1,\ldots,p_k. \tag{E3.15}$$

Using (E3.5) this becomes :

$$S^j_{2kx} = S^{j-1}_{2ktx} + S^{j-1}_{2kx} f_x \qquad j = 1,\ldots,p_k. \tag{E3.16}$$

and hence

$$S^j_{2kxx} = S^{j-1}_{2ktxx} + S^{j-1}_{2kxx} f_x + S^{j-1}_{2kx} f_{xx} \qquad j = 1,\ldots,p_k. \tag{E3.17}$$

Using (E3.5) once more we obtain :

$$S^{j-1}_{2kx} f_{xx} = \begin{cases} 0 & j = 1,\ldots,p_k - 1 \\ S^{p_k}_{2kxx} & j = p_k \end{cases} \tag{E3.18}$$

Substitution of (E3.18) in (E3.13) yields (E3.10).

The proof of (E3.11) and (E3.12) follows similar lines.

□

Lemma E1 shows that there are two alternatives for the calculation of the matrices M_2, M_3 and M_4. Now consider the case that the step size α_i in Algorithm 4.4 equals one. In this case $\lambda^i = \tilde{\lambda}^{i-1}$, i.e. the current estimate of the multiplier λ is the multiplier $\tilde{\lambda}$ of the previous iteration (the adjoint variable corresponding to the solution of problem (EIQP/SCOCP/Δ) in the previous iteration). This adjoint variable is obtained from the multipliers $\bar{\lambda}$ and $\bar{\eta}$ which were obtained as the solution of the linear multipoint boundary value problem, via relation (E3.6). (cf. Appendices E1 and E2). It is well known that in general, the numerical differentiation of $\bar{\eta}_2$ yields relatively large truncation errors in λ. Therefore the actual calculation of the matrices M_2, M_3 and M_4 is done using (E3.7), (E3.8) and (E3.9) with $\bar{\lambda}$ and $\bar{\eta}_2$ corresponding to the solution of the last linear multipoint boundary value problem. When the step size α_i not equals one $\bar{\lambda}$ and $\bar{\eta}_2$ are modified in a way similar to all other multipliers in Algorithm 4.4.

Appendix E4 : Strategy in case of rank deficiency of the matrix of constraint normals C.

In Algorithm 5.8 it was assumed that throughout the solution process of problem (EIQP/SCOCP/Δ), the matrix of constraint normals C has full row rank. However, in practice it turns out that this assumption may not always be satisfied.

We shall first analyse this phenomenon from the point of view of finite-dimensional quadratic programming. In this case, the constraints, which restrict the step size, are added to the working set one by one, therefore the matrix of constraint normals will never become rank deficient. Considering the addition of constraints to the working set in Algorithm 5.8, we observe that (in case of state constraints with order $\geqslant 1$), more than one row can be added to the matrix C at the same time (cf. Appendix D3).

An alternative point of view follows from the consideration of working sets for problem (EQP/SCOCP). It is not difficult to establish examples for which a solution does not exist. Consider the following example :

$$\underset{d_x, d_u}{Minimize} \quad \frac{1}{2} \int_0^T d_u^2 \, dt \,, \tag{E4.1}$$

$$subject\ to : \dot{d}_x(t) = d_u(t) \quad 0 \leqslant t \leqslant T, \tag{E4.2}$$

$$d_x(0) = 0 \tag{E4.3}$$

$$d_x(t) = d_{xmax} \quad 0 < t_1 \leqslant t \leqslant t_2 < T, \tag{E4.4}$$

$$d_u(t) = d_{umax} \quad 0 \leqslant t \leqslant t_3. \tag{E4.5}$$

If $t_3 < t_1$, then problem (E4.1) - (E4.5) has a solution and if $t_3 \geqslant t_1$, then (E4.1) - (E4.5) may fail to have a solution. In the latter case the matrix of constraint normals will be rank deficient.

We now turn to the consideration of possible remedies for the case that rank deficiency is encountered.

A remedy suggested by Han (1981) in the context of finite-dimensional quadratic programming† is to use a least squares interpretation of the constraints. At first sight this seems a suitable alternative, because we have already an LQ-factorization available for the matrix of constraint normals (cf. Appendix D2). A complete orthogonal decomposition can be obtained by premultiplication with orthogonal matrices which zero the linear dependent rows.

However, when there are state constraints of order $\geqslant 1$ present, the solution procedure relies entirely on the transformation of state equality constraints into interior point constraints and mixed control state constraints. This transformation is based on the fact that (d_x, d_u) satisfies the linear differential system of problem (EQP/SCOCP). If the solution of the collocation scheme would be obtained using a least squares interpretation of the matrix of constraint normals, then this transformation would no longer be valid, because (d_x, d_u) will no longer satisfy the linear equations which were obtained via collocation on the

† With the method described by Han (1981) also more than one constraint can be added to the working set at one time.

linear differential equations. Hence for problems with state constraints of order $\geqslant 1$, this remedy fails.

Therefore the following, heuristic, strategy is followed. When rank deficiency is encountered, a kind of restoration phase is started, which calculates a feasible point with a matrix of constraint normals of full row rank. This restoration phase follows essentially the same strategy as the phase 1 of the Algorithm 5.8 as outlined in Section 5.2. For the sake of brevity, we shall not go into the details of this phase. From the new point, obtained from the restoration phase, the Algorithm 5.8 is restarted.

We note that with this strategy cycling is possible to occur, i.e. Algorithm 5.8 may return to the same situation. Therefore a check on cycling is made whenever a constraint is to be deleted from the working set, i.e. using a unique code for all possible working sets, it is verified whether the current working set is equivalent to any of the previous working sets.

Appendix E5 : Automatic adjustment of the penalty constant of the merit function.

The merit function (cf. (4.3.8)) is used in the first phase of Algorithm 4.4. The penalty constant ρ is, in first instance, supposed to be specified in advance and for a 'sufficiently high' value of ρ the direction of seach obtained as the solution of problem (EIQP/SCOCP/Δ) will be a direction of descent of the merit function.

Essentially, the role of the penalty constant ρ is to balance a decrease of the objective function versus violation of the constraints. Taking a very large value for ρ would therefore have the effect of placing large penalties on constraint violation and making the merit function relatively insensitive to decreasing the objective function. This makes a procedure for the automatic adjustment of the penalty constant attractive, for is such a procedure is available, it is possible to start with a relatively low value of ρ. The procedure will then increase the value of ρ automatically to a 'sufficiently high' value.

The procedure is essentially based on the result contained in the lemma below.

Lemma E2: *Let the merit function be defined by (4.3.8) and let the problem functions satisfy the assumptions of problem (SCOCP). For any direction of search* $(d_x, d_u, \tilde{\lambda}-\lambda, \tilde{\eta}_1-\eta_1, \tilde{\xi}-\xi, \tilde{\sigma}-\sigma, \tilde{\mu}-\mu)$ *for which* (d_x, d_u) *is a solution to problem (EIQP/SCOCP/*Δ*) with Lagrange multipliers* $(\tilde{\lambda}, \tilde{\eta}_1, \tilde{\xi}, \tilde{\sigma}, \tilde{\mu})$ *that satisfy*

$$\tilde{\eta}_{1k}(t) \geqslant 0 \quad a.e.\ \ 0 \leqslant t \leqslant T \qquad k = 1,\ldots,k_1. \tag{E5.1}$$

$$\tilde{\xi}_k(t) \ \textit{is nondecreasing on } [0,T] \quad k = 1,\ldots,k_2. \tag{E5.2}$$

let

$$\tilde{M}(d_x, d_u) := d_x(0)^T M_1 d_x(0) + \int_0^T (d_x^T\ d_u^T) \begin{bmatrix} M_2 & M_3 \\ M_3^T & M_4 \end{bmatrix} \begin{bmatrix} d_x \\ d_u \end{bmatrix} dt +$$

$$d_x(T)^T M_5 d_x(T), \tag{E5.3}$$

$$\|(d_x, d_u)\|_2^2 := \|d_x(0)\|^2 + \int_0^T (\|d_x(t)\|^2 + \|d_u(t)\|^2)\, dt + \|d_x(T)\|^2, \tag{E5.4}$$

$$\|(\tilde{\lambda}-\lambda, \tilde{\eta}_1-\eta_1, \tilde{\xi}-\xi, \tilde{\sigma}-\sigma, \tilde{\mu}-\mu)\|_2^2 := \|\tilde{\sigma}-\sigma\|^2 + \|\tilde{\mu}-\mu\|^2 + \sum_j \|\tilde{\nu}_j - \nu_j\|^2 +$$

$$\int_0^T (\|\tilde{\lambda}(t)-\lambda(t)\|^2 + \|\tilde{\eta}_1(t)-\eta_1(t)\|^2 + \|\tilde{\eta}_2(t)-\eta_2(t)\|^2)\, dt. \tag{E5.5}$$

If there are a $\delta > 0$ *and an* $\epsilon > 0$, *such that*

$$\tilde{M}(d_x, d_u) \geqslant \delta \|(d_x, d_u)\|_2^2, \tag{E5.6}$$

and

$$\|(d_x, d_u)\|_2^2 \geqslant \epsilon \|(\tilde{\lambda}-\lambda, \tilde{\eta}_1-\eta_1, \tilde{\xi}-\xi, \tilde{\sigma}-\sigma, \tilde{\mu}-\mu)\|_2^2, \tag{E5.7}$$

then, for all $\rho > 0$

$$-M'\{0\}(d_x, d_u, \tilde{\lambda}-\lambda, \tilde{\eta}_1-\eta_1, \tilde{\xi}-\xi, \tilde{\sigma}-\sigma, \tilde{\mu}-\mu) \geqslant \tfrac{1}{2}\delta \|(d_x, d_u)\|_2^2 +$$

$$\left[\frac{\delta\epsilon}{2} - \frac{1}{\rho}\right] \|(\tilde{\lambda}-\lambda, \tilde{\eta}_1-\eta_1, \tilde{\xi}-\xi, \tilde{\sigma}-\sigma, \tilde{\mu}-\mu)\|_2^2. \tag{E5.8}$$

The proof of this lemma is a rather lengthy derivation and follows similar lines as the proof of part b of Theorem 4.2 of Schittkowski (1981). We note that in the proof use is made of the conditions (E5.1) and (E5.2).

Now we shall consider the existence of a number $\delta > 0$, as mentioned in the hypotheses of Lemma E2. Because a solution of problem (EIQP/SCOCP/Δ) is also a solution of problem (EQP/SCOCP), the second order sufficient optimality condition of part (ii) of Theorem 2.16 may be expressed for problem (EQP/SCOCP) at this point. This sufficient optimality conditoin assumes the existence of a $\bar{\delta} > 0$, such that

$$L''(d_x, d_u, \tilde{\lambda}, \tilde{\eta}_1, \tilde{\xi}, \tilde{\sigma}, \tilde{\mu})(\delta x, \delta u)(\delta x, \delta u) = \tilde{M}(\delta x, \delta u) \geqslant \bar{\delta} \|(\delta x, \delta u)\|_2^2, \tag{E5.9}$$

for all $(\delta x, \delta u)$ satisfying

$$\delta\dot{x} = f_x[t]\delta x + f_u[t]\delta u \qquad a.e.\ \ 0 \leqslant t \leqslant T, \tag{E5.10}$$

$$D_x[0]\delta x(0) = 0, \tag{E5.11}$$

$$E_x[T]\delta x(T) = 0, \tag{E5.12}$$

$$R_x[t]\delta x + R_u[t]\delta u = 0 \qquad a.e.\ \ 0 \leqslant t \leqslant T. \tag{E5.13}$$

Condition (E5.9) is equivalent to (E5.6) and hence the first part of the hypotheses of Lemma E2 hold, whenever the second order sufficiency condition of Theorem 2.16 holds for problem (EQP/SCOCP) and (d_x, d_u) satisfy the homogeneous constraints (E5.10) - (E5.13). Because (d_x, d_u) satisfy the inhomogeneous relations (4.2.1.22) - (4.2.1.25), the hypotheses of Lemma E2 may fail to hold, even when the second order optimality conditions hold for the solution of problem (EQP/SCOCP). However, this situation is only likely to occur 'far from the solution', i.e. when the inhomogeneous terms in the relations (4.2.1.22) - (4.2.1.25) are relatively large. Considering the second part of the hypotheses of Lemma E2, we notice that an $\epsilon > 0$ exists, whenever $\|(d_x, d_u)\|_2^2 \neq 0$.

The adjustment of the penalty constant is primarily based on expression (E5.8), i.e. if $\delta > 0$ and $\epsilon > 0$ both exist, then the penalty constant is increased, such that

$$\left| \frac{\delta\epsilon}{2} - \frac{1}{\rho} \right| > 0. \tag{E5.14}$$

In the case that (E5.9) cannot be satisfied for any $\delta > 0$, it is likely that the inhomogeneous terms in (4.2.1.22) - (4.2.1.25) are relatively large. In this case the direction of search will still be a direction of descent of the merit function, for a 'sufficiently high' value of ρ, because (d_x, d_u) will be a direction of descent of the penalty term of the merit function.

The penalty constant in iteration i of Algorithm 4.4, denoted ρ_i is adjusted using the algorithm below. This adjustment takes place between steps (iv) and (v) of Algorithm 4.4.

Algorithm E3

Given $x^i, u^i, \lambda^i, \eta_1^i, \sigma^i, \mu^i$ *and* $d_x^i, d_u^i, \tilde{\lambda}^i, \tilde{\eta}_1^i, \tilde{\sigma}^i, \tilde{\mu}^i$ *and* ρ_{i-1}.

If $\tilde{M}(d_x^i, d_u^i) > 0$ *then*

$\delta_i := \tilde{M}(d_x^i, d_u^i) / \|(d_x^i, d_u^i)\|_2^2$

$\epsilon_i := \|(d_x^i, d_u^i)\|_2^2 / \|(\tilde{\lambda}^i - \lambda^i, \tilde{\eta}_1 - \eta_1, \tilde{\xi}^i - \xi^i, \tilde{\sigma}^i, \sigma^i, \tilde{\mu}^i - \mu^i)\|_2^2$

$\rho_i := \rho_{i-1}$

while $\rho_i < 2/(\delta_i \epsilon_i)$

do

$\rho_i := 10 \cdot \rho_i$

od

else

$\rho_i := \rho_{i-1}$

while (d_x, d_u) *is no direction of descent*

do

$\rho_i := 10 \cdot \rho_i$

od

Fi

Appendix E6 : Computation of the merit function.

The computation of the merit function (4.3.8) is based on the quadrature rules discussed in Section 6.1.1, i.e.

$$\int_0^T \phi(t)\,dt \sim \sum_{r=0}^{p-1} h_r \sum_{i=1}^{l} \bar{\omega}_i \phi(\tau_{lr+i}), \tag{E6.1}$$

where $h_r := t_{r+1} - t_r$.

A consideration of the terms of the merit function that involve the mixed control state constraints S_1 yields that (E6.1) gives a suitable approximation for the penalty term :

$$\sum_{k=1}^{k_1} \int_0^T (\eta_{1k} \bar{S}_{1k}(x,u,\eta_{1k} t;\rho) + \tfrac{1}{2}\rho \bar{S}_{1k}(x,u,\eta_{1k} t;\rho)^2)\,dt. \tag{E6.2}$$

Because the constraints S_{1k} are taken active and inactive per collocation point.

Similarily, consider the term :

$$\sum_{k=1}^{k_2} \int_0^T (\eta_{2k} \bar{S}_{2k}(x,\eta_{2k} t;\rho) + \tfrac{1}{2}\rho \bar{S}_{2k}(x,\eta_{2k} t;\rho)^2)\,dt. \tag{E6.3}$$

Because the constraints S_{2k} are taken active and inactive per grid interval, formula (E6.1) is not suitable for the calculation of this term. For this would lead to penalizing constraints at collocation points where the constraint is not active. Therefore (E6.3) is approximated using a trapeziodal quadrature rule, i.e.

$$\int_0^T \phi(t)\,dt \sim \sum_{r=0}^{p-1} \tfrac{1}{2} h_r (\phi(t_r) + \phi(t_{r+1})). \tag{E6.4}$$

The merit function (4.3.8) is thus computed using the quadrature formula (E6.1) for all terms but (E6.3), which is computed by means of the quadrature formula (E6.4).

Appendix E7 : Miscellaneous details.

In this appendix we shall discuss some details regarding the implementation of the method.

Restoration phase

Before the first stage of Algorithm 4.4 is started, a restoration phase is executed. This restoration phase is essentially the same as the one used in the sequential gradient-restoration method of Miele (1980). The restoration phase is used in order to obtain an approximately feasible point and starts at an initial point, which is specified in advance.

The direction of search in the restoration phase is determined as the solution of a linear-quadratic optimal control problem which is similar to problem (EIQP/SCOCP/Δ). More specifically, the constraints of this problem are the same as those of problem (EIQP/SCOCP/Δ), but the objective function (4.2.1.5) is replaced by :

$$d_x(0)^T d_x(0) + \int_0^T [d_x(t)^T \; d_u(t)^T] \begin{bmatrix} d_x(t) \\ d_u(t) \end{bmatrix} dt \; + d_x(T)^T d_x(T). \qquad (E7.1)$$

As a merit function the penalty part of (4.3.8) is used, i.e.

$$P(x,u,\eta_1,\xi;\rho) := \tfrac{1}{2} \left\{ \int_0^T (\|\dot{x} - f(x,u,t)\|^2 + \sum_{l=1}^{k_1} \bar{S}_{1l}(x,u,\eta_{1l},t;\rho)^2 + \right.$$

$$\left. \sum_{l=1}^{k_2} \bar{S}_{2l}(x,\eta_{2l},t;\rho)^2) \, dt \; + \sum_j \sum_{l=1}^{k_2} \bar{S}_{2l}(x,\nu_{jl},t_j;\rho)^2 + \right.$$

$$\left. \|D(x(0))\|^2 + \|E(x(T),T)\|^2 \right\}, \qquad (E7.2)$$

with :

$$\bar{S}_{1l}(x,u,\eta_{1l},t;\rho) := max\,\{S_{1l}(x,u,t)\,, -\eta_{1l}/\rho\}, \qquad (E7.3)$$

$$\bar{S}_{2l}(x,\eta_{2l},t;\rho) := max\,\{S_{2l}(x,t)\,, -\eta_{2l}/\rho\}, \qquad (E7.4)$$

The restoration phase is terminated once the norm of the direction of search is below a specified quantity.

Implementation of the line minimization.

The approximate line minimization outlined in Section 4.3 was implemented with $\beta = \frac{1}{2}$ and $\epsilon = \frac{1}{4}$. In addition to the condition (4.3.12) which must be satisfied for the step size $\alpha = \beta^k$, the penalty term (E7.2) must satisfy :

$$P\{\alpha\} \leqslant max \left\{ P^*, P\{0\} \right\} \qquad (E7.5)$$

where $\{\alpha\}$ was used to replace $(x^i + \alpha d_x^i,\, u^i + \alpha d_u^i,\, \eta_1^i + \alpha(\bar{\eta}_1^i - \eta_1^i),\, \xi^i + \alpha(\bar{\xi}^i - \xi^i))$.

Obviously, condition (E7.5) ascertains that 'away from the solution', i.e. at points where $P^* < P\{0\}$, the penalty term in the merit function is not increased.

Non-convergence of Algorithm 5.8

Non-convergence of the solution procedure of problem (EIQP/SCOCP/Δ) is possible to occur as a result of the following conditions :

1) Problem (EIQP/SCOCP/Δ) has no bounded solution.
2) The constraints of problem (EIQP/SCOCP/Δ) are inconsistent (no feasible point).
3) The maximum number of iterations in Algorithm 5.8 exceeded.
4) Cycling detected (cf. Appendix E4).
5) The maximum number of grid modifications exceeded.
5) Rank deficiency of the matrix of constraint normals was encountered too many times (cf. Appendix E4).

In each of these cases, Algorithm 5.8 is terminated. The last estimate for the solution of problem (EIQP/SCOCP/Δ) which was used in Algorithm 5.8, is used as a direction of search in Algorithm 4.4. After the determination of the step size α, Algorithm 4.4 is continued at step (i), i.e. an initialization step is executed which determines first order estimates for the Lagrange multipliers at the new point.

Appendix F : Numerical results.

This appendix contains a number of tables, with the convergence histories that correspond to the numerical solution process of some of the problems discussed in Chapter 7. The Tables F1 - F8 contain :

Table	Convergence history of
F1	unconstrained glider problem.
F2	glider problem with acceleration constraint, $n_{max} = 4$.
F3	glider problem with velocity constraint, $v_{max} = 50$.
F4	glider problem with altitude constraint, $y_{min} = -30$.
F5	unconstrained reentry problem.
F6	reentry problem with acceleration constraint, $n_{max} = 6$.
F7	reentry problem with altitude constraint, $\xi_{max} = 0.0090$.
F8	servo problem with $V_{max,1} = 1.5$, $A_{max,1} = 3$, $c = 1$.

On top of each table the number of gridpoints (p) and the order of the polynomials (l) are given. In most cases the convergence table consists of three parts. The first part shows the convergence behaviour of the method in the restoration phase (cf. Appendix E7). The second part of the convergence table shows the convergence behaviour in the first stage of the method. The last part of the table shows the convergence behaviour in the second stage. The columns of the convergence table contain the following entities :

IT	Iteration number
T	Type of iteration (R = Restoration step, I = Initialization step, G = Gradient step, N = Newton step)
‖D2‖	Norm of direction of search
OBJECTIVE	Value of objective function
MERIT FUNCTION	Value of merit function
LAGRANGIAN	Value of Lagrangian part of merit function
PCRIT	Value of penalty part of merit function (excl. penalty constant)
RHOP	Penalty constant
IQP	Number of iteration steps used for the solution of problem (EIQP/SCOCP/Δ)
IG	Number of grid modifications
IR	Number of times that rank deficiency of the matrix C was encounted
QPZ	Number of linear conjugate gradient steps done during the solution of problem (EIQP/SCOCP/Δ)
DN	Dimension of Null space of matrix C after solution of problem (EIQP/SCOCP/Δ)
DR	Dimension of Range space of matrix C^T after solution of problem (EIQP/SCOCP/Δ)
C	Termination condition of Algorithm 5.8 (* = Subproblem unbounded from below)

Below the convergence table the solutions obtained for the state and control vectors are given at the time points $t = 0$, $t = 0.1$,..., $t = 1$ and the active set is listed.

NUMBER OF GRIDPOINTS = 50
ORDER OF POLYNOMIALS = 2

IT	T	ALPHA	||D2||	OBJECTIVE	MERIT FUNCTION	LAGRANGIAN	PCRIT	RHOP	IQP	IG	IR	QPZ	DN	DR C
0	R	0.10D+01	0.42D+02		0.6870964963790+00		0.14D+04	0.10D-02	0	0	0	1	98	304
1	R	0.10D+01	0.12D+01		0.299617907164D-02		0.60D+01	0.10D-02	0	0	0	0	98	304
2	R	0.10D+01	0.70D+00		0.721943883111D-04		0.14D+00	0.10D-02	0	0	0	0	98	304
	END OF RESTORATION PHASE													
3	I			0.143825974177D+02	0.144793154540D+02	0.144790519615D+02	0.53D+00	0.10D-02	4	0	0	4	97	305
3	N	0.25D+00	0.54D+02	0.104205654109D+02	0.111015203610D+02	0.106509887755D+02	0.90D+02	0.10D-01	3	0	0	33	98	304
4	N	0.50D+00	0.27D+02	0.680912177509D+01	0.896346291391D+01	0.789460581076D+01	0.21D+03	0.10D-01	4	0	0	67	97	305
5	N	0.10D+01	0.98D+01	0.621244662912D+01	0.747159816745D+01	0.732175010675D+01	0.30D+02	0.10D-01	6	0	0	101	94	308
6	N	0.10D+01	0.12D+01	0.729339016149D+01	0.730254023375D+01	0.730245125311D+01	0.18D-01	0.10D-01	9	0	0	98	98	304
7	N	0.10D+01	0.12D+00	0.730229160901D+01	0.730234083536D+01	0.730234083099D+01	0.87D-06	0.10D-01	0	0	0	14	98	304
8	N	0.10D+01	0.96D-03	0.730234079927D+01	0.730234081446D+01	0.730234081446D+01	0.56D-14	0.10D+02	0	0	0	13	98	304
9	N	0.10D+01	0.54D-06	0.730234081446D+01	0.730234081446D+01	0.730234081446D+01	0.11D-25	0.10D+02	0	0	0	10	98	304
10	N	0.10D+01	0.54D-12	0.730234081446D+01	0.730234081446D+01	0.730234081446D+01	0.72D-23	0.10D+02	0	0	0	3	98	304

STATE VECTOR X

0.00D+00	0.41631000000000000D+02	-0.13440000000000000D+01
0.10D+00	0.41597485717678650+02	-0.30052056688434710+01
0.20D+00	0.41575926174248960+02	-0.11747221017823930+02
0.30D+00	0.40935170775596830+02	-0.23878396266362710+02
0.40D+00	0.45156848858009780+02	-0.28653765920180150+02
0.50D+00	0.58494283314891400+02	0.31295409816813520+01
0.60D+00	0.46331089334338800+02	0.32854598466572750+02
0.70D+00	0.42263589273871400+02	0.26506648055090310+02
0.80D+00	0.42509915570026740+02	0.13057527091732550+02
0.90D+00	0.41964079593929320+02	0.27317633850049000+01
0.10D+01	0.41631000000000000+02	-0.13440000000000000+01

CONTROL VECTOR U

0.00D+00	0.39373635528376660+00
0.10D+00	0.17714230349708260+00
0.20D+00	0.40915883740529250-02
0.30D+00	0.14650004801144000-01
0.40D+00	0.34224763282605460+00
0.50D+00	0.11678946452784960+01
0.60D+00	0.30834163653544290+00
0.70D+00	-0.12972590354201150-01
0.80D+00	-0.30587235118181970-01
0.90D+00	0.12011310659707420+00
0.10D+01	0.31136774150673810+00

CONVERGENCE HISTORY OF THE UNCONSTRAINED GLIDER PROBLEM.
TABLE F1

```
NUMBER OF GRIDPOINTS =     50
ORDER OF POLYNOMIALS =      2

IT T     ALPHA      ||D2||          OBJECTIVE       MERIT FUNCTION          LAGRANGIAN          PCRIT       RHOP IQP IG IR  QPZ  DN  DR C

 0 R 0.10D+01  0.42D+02                          0.6870964963790+00                          0.14D+04  0.10D-02   0  0  0    1  98  304
 1 R 0.10D+01  0.12D+01                          0.2996179071640-02                          0.60D+01  0.10D-02   0  0  0    0  98  304
 2 R 0.10D+01  0.70D+00                          0.7219438831110-04                          0.14D+00  0.10D-02   0  0  0    0  98  304

   END OF RESTORATION PHASE

 3 I                     0.1438259741770+02  0.1434139239180+02  0.1434114285330+02  0.50D+00  0.10D-02   6  0  0    6  95  307
 3 N 0.50D+00  0.46D+02  0.8299863796960+01  0.9150291518740+01  0.8770832242850+01  0.76D+03  0.10D-02  25  0  0  301  82  320
 4 N 0.10D+01  0.11D+02  0.6630396294320+01  0.7915319286700+01  0.7906126246680+01  0.18D+02  0.10D-02   6  0  0   85  82  320
 5 N 0.10D+01  0.11D+01  0.7896452548060+01  0.7897012056590+01  0.7897008530020+01  0.71D-02  0.10D-02   3  0  0   29  83  319
 6 N 0.10D+01  0.18D-01  0.7897005238020+01  0.7897004247890+01  0.7897004247890+01  0.82D-09  0.10D-02   0  0  0   11  83  319
 7 N 0.10D+01  0.29D-04  0.7897004247740+01  0.7897004247760+01  0.7897004247760+01  0.49D-20  0.10D+05   0  0  0   10  83  319
 8 N 0.10D+01  0.55D-06  0.7897004247760+01  0.7897004247760+01  0.7897004247760+01  0.74D-26  0.10D+05   0  0  0    8  83  319
 9 N 0.10D+01  0.31D-12  0.7897004247760+01  0.7897004247760+01  0.7897004247760+01  0.99D-26  0.10D+05   0  0  0    3  83  319

   START OF SECOND STAGE

**** grid update ( add ) ***** AT  0.5700000000000+00

 9 N 0.10D+01  0.97D-01  0.7897001787250+01                                          0.80D+01              0  0  0   12  84  326
10 N 0.10D+01  0.34D-02  0.7895970107850+01                                          0.15D-02              0  0  0   11  84  326
11 N 0.10D+01  0.85D-06  0.7896497588050+01                                          0.30D-03              0  0  0    9  84  326
12 N 0.10D+01  0.92D-12  0.7896497732920+01                                          0.30D-03              0  0  0    3  84  326

***** grid update (shift) ***** FROM  0.4200000000000+00 TO  0.4206278866100+00
***** grid update (shift) ***** FROM  0.5700000000000+00 TO  0.5743602496810+00

13 N 0.10D+01  0.14D+01  0.7956842442990+01                                          0.38D+01              0  0  0   11  84  326
14 N 0.10D+01  0.67D-02  0.7902386325890+01                                          0.30D-01              0  0  0   11  84  326
15 N 0.10D+01  0.25D-04  0.7896700612480+01                                          0.42D-06              0  0  0    9  84  326
16 N 0.10D+01  0.48D-09  0.7896702976230+01                                          0.16D-15              0  0  0    6  84  326
17 N 0.10D+01  0.23D-12  0.7896702976260+01                                          0.46D-23              0  0  0    3  84  326

***** grid update (shift) ***** FROM  0.5743602496810+00 TO  0.5743768876420+00

18 N 0.10D+01  0.50D-02  0.7896843289170+01                                          0.58D-04              0  0  0   10  84  326
19 N 0.10D+01  0.93D-07  0.7896703075590+01                                          0.46D-11              0  0  0    8  84  326
20 N 0.10D+01  0.32D-12  0.7896702979810+01                                          0.63D-23              0  0  0    2  84  326
```

STATE VECTOR X

0.00D+00	0.41631000000000000D+02	-0.13440000000000000D+01
0.10D+00	0.41385083812926420+02	-0.11420137080391330+01
0.20D+00	0.41334865808473080+02	-0.86908916922637670+01
0.30D+00	0.41092758373160510+02	-0.19921315849930750+02
0.40D+00	0.45721388741279020+02	-0.22560436389996460+02
0.50D+00	0.54840827816728680+02	0.25152024163291380+01
0.60D+00	0.46938810784342320+02	0.26553230147743390+02
0.70D+00	0.42564323111832580+02	0.22238691474826290+02
0.80D+00	0.42349233224754130+02	0.97953337853129540+01
0.90D+00	0.41754036540423450+02	0.72486679391794680+00
0.10D+01	0.41631000000000000+02	-0.13440000000000010+01

CONTROL VECTOR U

0.00D+00	0.45952142933663440+00
0.10D+00	0.22227234522123430+00
0.20D+00	0.33701691187359360-01
0.30D+00	0.41076867200336780-01
0.40D+00	0.46506181753092780+00
0.50D+00	0.75981698602144170+00
0.60D+00	0.41654886408183760+00
0.70D+00	0.96766405642136750-02
0.80D+00	-0.37074093784465460-02
0.90D+00	0.16560614510069980+00
0.10D+01	0.38345202065984840+00

JUNCTION AND CONTACT POINTS OF CONSTRAINT S1

1 0.42062788661OD+00 0.5743768876420+00

CONVERGENCE HISTORY GLIDER PROBLEM WITH CONSTRAINT ON THE ACCELERATION (NMAX = 4).
TABLE F2

NUMBER OF GRIDPOINTS = 20
ORDER OF POLYNOMIALS = 2

IT	T	ALPHA	||D2||	OBJECTIVE	MERIT FUNCTION	LAGRANGIAN	PCRIT	RHOP	IQP	IG	IR	QPZ	DN	DR	C
0	R	0.10D+01	0.18D+01		0.103208813859D-03		0.21D+00	0.10D-02	0	0	0	1	38	185	
1	R	0.10D+01	0.12D+00		0.368905902594D-07		0.74D-04	0.10D-02	0	0	0	0	38	185	
		END OF RESTORATION PHASE													
2	I			0.144775542397D+02	0.144808470847D+02	0.144808470478D+02	0.74D-04	0.10D-02	0	0	0	2	38	185	
2	N	0.25D+00	0.30D+02	0.119002080119D+02	0.120902437400D+02	0.116264669920D+02	0.93D+01	0.10D+00	4	0	0	52	37	186	
3	I			0.119002080119D+02	0.167232107387D+02	0.121100686393D+02	0.92D+01	0.10D+01	3	0	0	4	38	185	
3	G	0.49D+00	0.21D+02	0.116749279205D+02	0.136292406930D+02	0.121743918284D+02	0.29D+01	0.10D+01	3	0	0	4	38	185	
4	N	0.13D+00	0.18D+02	0.109418058727D+02	0.128680936114D+02	0.112983989075D+02	0.31D+01	0.10D+01	4	0	0	50	37	186	
5	N	0.13D+00	0.16D+02	0.103639175203D+02	0.122022177802D+02	0.106143276905D+02	0.32D+01	0.10D+01	0	0	0	13	37	186	
6	N	0.13D+00	0.14D+02	0.991305712073D+01	0.116063531255D+02	0.100854009754D+02	0.30D+01	0.10D+01	0	0	0	13	37	186	
7	N	0.25D+00	0.12D+02	0.921486584679D+01	0.111455538238D+02	0.933719613129D+01	0.36D+01	0.10D+01	0	0	0	13	37	186	
8	N	0.25D+00	0.88D+01	0.883479264600D+01	0.104591977880D+02	0.892180269189D+01	0.31D+01	0.10D+01	0	0	0	13	37	186	
9	N	0.50D+00	0.63D+01	0.842690962061D+01	0.979766594718D+01	0.853628366368D+01	0.25D+01	0.10D+01	0	0	0	13	37	186	
10	N	0.10D+01	0.28D+01	0.832405892929D+01	0.860301243173D+01	0.841263711532D+01	0.38D+00	0.10D+01	0	0	0	14	37	186	
11	N	0.10D+01	0.11D+00	0.841216267664D+01	0.841242961178D+01	0.841242857206D+01	0.21D-05	0.10D+01	0	0	0	13	37	186	
12	N	0.10D+01	0.56D-03	0.841242856181D+01	0.841242856655D+01	0.841242856655D+01	0.13D-14	0.10D+01	0	0	0	12	37	186	
13	N	0.10D+01	0.18D-06	0.841242856655D+01	0.841242856655D+01	0.841242856655D+01	0.16D-26	0.10D+01	0	0	0	9	37	186	
14	N	0.10D+01	0.28D-12	0.841242856655D+01	0.841242856655D+01	0.841242856655D+01	0.14D-26	0.10D+01	0	0	0	3	37	186	
		START OF SECOND STAGE													
***** grid update (shift) ***** FROM 0.500000000000D+00 TO 0.485088720768D+00															
14	N	0.10D+01	0.75D+01	0.678880791401D+01			0.42D+03		0	0	0	14	37	186	
15	N	0.10D+01	0.94D+00	0.909577654701D+01			0.72D+02		0	0	0	14	37	186	
16	N	0.10D+01	0.72D-01	0.842910728639D+01			0.49D+00		0	0	0	13	37	186	
17	N	0.10D+01	0.45D-03	0.842809132001D+01			0.27D-04		0	0	0	12	37	186	
18	N	0.10D+01	0.30D-06	0.842813047592D+01			0.82D-11		0	0	0	10	37	186	
19	N	0.10D+01	0.52D-12	0.842813047777D+01			0.29D-22		0	0	0	3	37	186	
***** grid update (shift) ***** FROM 0.485088720768D+00 TO 0.487669500463D+00															
20	N	0.10D+01	0.12D+01	0.870682727352D+01			0.96D+01		0	0	0	14	37	186	
21	N	0.10D+01	0.52D-01	0.843408630933D+01			0.78D-01		0	0	0	13	37	186	
22	N	0.10D+01	0.71D-04	0.842880065219D+01			0.27D-06		0	0	0	11	37	186	
23	N	0.10D+01	0.15D-06	0.842882313388D+01			0.20D-11		0	0	0	10	37	186	
24	N	0.10D+01	0.40D-12	0.842882313391D+01			0.18D-22		0	0	0	2	37	186	
***** grid update (shift) ***** FROM 0.487669500463D+00 TO 0.487244467986D+00															
25	N	0.10D+01	0.20D+00	0.838280985809D+01			0.23D+00		0	0	0	13	37	186	
26	N	0.10D+01	0.13D-02	0.842900429780D+01			0.50D-04		0	0	0	12	37	186	
27	N	0.10D+01	0.20D-06	0.842881229582D+01			0.59D-11		0	0	0	10	37	186	
28	N	0.10D+01	0.30D-12	0.842881230860D+01			0.73D-23		0	0	0	3	37	186	
***** grid update (shift) ***** FROM 0.487244467986D+00 TO 0.487134176851D+00															
29	N	0.10D+01	0.52D-01	0.841684031656D+01			0.16D-01		0	0	0	13	37	186	
30	N	0.10D+01	0.89D-04	0.842881541080D+01			0.23D-06		0	0	0	11	37	186	
31	N	0.10D+01	0.22D-06	0.842880289111D+01			0.34D-11		0	0	0	10	37	186	
32	N	0.10D+01	0.37D-12	0.842880289117D+01			0.12D-22		0	0	0	3	37	186	

```
***** grid update (shift) ***** FROM  0.487134176851D+00 TO  0.487134113437D+00

33 N 0.10D+01  0.30D-04  0.842879600627D+01                                  0.53D-08      0  0  0   11  37  186
34 N 0.10D+01  0.30D-10  0.842880288498D+01                                  0.26D-19      0  0  0    6  37  186

***** grid update (shift) ***** FROM  0.487134113437D+00 TO  0.487134113430D+00

35 N 0.10D+01  0.36D-08  0.842880288415D+01                                  0.76D-16      0  0  0    7  37  186
36 N 0.10D+01  0.45D-12  0.842880288497D+01                                  0.88D-23      0  0  0    3  37  186

***** grid update (shift) ***** FROM  0.487134113430D+00 TO  0.487134113437D+00

37 N 0.10D+01  0.36D-08  0.842880288580D+01                                  0.76D-16      0  0  0    7  37  186
38 N 0.10D+01  0.80D-13  0.842880288497D+01                                  0.48D-23      0  0  0    2  37  186

STATE VECTOR X

 0.00D+00   0.41631000000000000D+02  -0.13440000000000000D+01   0.6939251408212397D+00
 0.10D+00   0.3938282222521462D+02    0.6221287126345293D+01   0.6372825035890981D+00
 0.20D+00   0.3823175405055955D+02   -0.2096908037443302D+00   0.5847459210487001D+00
 0.30D+00   0.3782863617521061D+02   -0.1283755808574389D+02   0.6383233739561128D+00
 0.40D+00   0.4039528192101349D+02   -0.1994128597039742D+02   0.8617720823723663D+00
 0.50D+00   0.4986519229566438D+02    0.3461637051773330D+01   0.9971266513597516D+00
 0.60D+00   0.4198135692220334D+02    0.2385012696896370D+02   0.8801816776143176D+00
 0.70D+00   0.3971473185298190D+02    0.1512874265394532D+02   0.7224379131285582D+00
 0.80D+00   0.3962848301165066D+02    0.1134940333864124D+01   0.6294384421832522D+00
 0.90D+00   0.3979704252155648D+02   -0.7094857393703671D+01   0.6521101506785117D+00
 0.10D+01   0.41631000000000000D+02  -0.13440000000000000D+01   0.6939083297780013D+00

CONTROL VECTOR U

 0.00D+00   0.8534829439296494D+00
 0.10D+00   0.3377503924106235D+00
 0.20D+00   0.5274575341010654D-01
 0.30D+00   0.2203386075692628D-02
 0.40D+00   0.3422733011699218D+00
 0.50D+00   0.1060851828943368D+01
 0.60D+00   0.2611311933894094D+00
 0.70D+00  -0.4357221191521222D-01
 0.80D+00  -0.6051488438168196D-02
 0.90D+00   0.2659840485399376D+00
 0.10D+01   0.8309644274755009D+00

JUNCTION AND CONTACT POINTS OF CONSTRAINT S2

     1  0.487134113437D+00
```

CONVERGENCE HISTORY GLIDER PROBLEM WITH CONSTRAINT ON THE VELOCITY (VMAX = 50).
TABLE F3

```
NUMBER OF GRIDPOINTS =     20
ORDER OF POLYNOMIALS =      2

IT T    ALPHA     ||D2||          OBJECTIVE       MERIT FUNCTION          LAGRANGIAN       PCRIT       RHOP IQP IG IR  QPZ  DN   DR C

 0 R 0.10D+01  0.18D+01                        0.103221077676D-03                        0.21D+00  0.10D-02   0  0  0    1  38  246
 1 R 0.10D+01  0.12D+00                        0.369082543981D-07                        0.74D-04  0.10D-02   0  0  0    0  38  246
 2 R 0.10D+01  0.96D-03                        0.175704572894D-15                        0.35D-12  0.10D-02   0  0  0    0  38  246

   END OF RESTORATION PHASE

 3 I                   0.144806176120D+02  0.144806177947D+02  0.144806177947D+02  0.35D-12  0.10D-02   0  0  0    8  38  246
 3 N 0.25D+00  0.29D+02  0.118380473193D+02  0.121549188372D+02  0.116880004706D+02  0.93D+01  0.10D+00   4  0  0   51  37  247
 4 N 0.25D+00  0.20D+02  0.103414940836D+02  0.109534360172D+02  0.102148014216D+02  0.15D+02  0.10D+00   0  0  0   12  37  247
 5 N 0.50D+00  0.14D+02  0.871415831881D+01  0.102765642012D+02  0.891939729313D+01  0.27D+02  0.10D+00   0  0  0   13  37  247
 6 N 0.10D+01  0.54D+01  0.833348936962D+01  0.886043318846D+01  0.863681531054D+01  0.45D+01  0.10D+00   0  0  0   13  37  247
 7 N 0.10D+01  0.48D+00  0.862905774699D+01  0.863217279998D+01  0.863213318140D+01  0.79D-03  0.10D+00   0  0  0   14  37  247
 8 N 0.10D+01  0.22D-01  0.863212433256D+01  0.863212799952D+01  0.863212799944D+01  0.16D-08  0.10D+00   0  0  0   13  37  247
 9 N 0.10D+01  0.46D-04  0.863212799933D+01  0.863212799935D+01  0.863212799935D+01  0.46D-19  0.10D+04   0  0  0   11  37  247

   START OF SECOND STAGE

***** grid update (shift) ***** FROM  0.500000000000D+00 TO  0.493118684429D+00
***** norm grid shift *****  0.688131557149D-02

 9 N 0.10D+01  0.31D+01  0.795195082838D+01                                        0.54D+02                 0  0  0   14  37  247
10 N 0.10D+01  0.27D+00  0.870607851151D+01                                        0.25D+01                 0  0  0   13  37  247
11 N 0.10D+01  0.20D-02  0.863497067307D+01                                        0.33D-03                 0  0  0   12  37  247
12 N 0.10D+01  0.76D-06  0.863540285897D+01                                        0.48D-10                 0  0  0   10  37  247

***** grid update (shift) ***** FROM  0.493118684429D+00 TO  0.494575846176D+00
***** norm grid shift *****  0.145716174730D-02

13 N 0.10D+01  0.68D+00  0.877828161713D+01                                        0.27D+01                 0  0  0   13  37  247
14 N 0.10D+01  0.15D-01  0.863708670844D+01                                        0.67D-02                 0  0  0   13  37  247
15 N 0.10D+01  0.60D-05  0.863554464118D+01                                        0.18D-08                 0  0  0   11  37  247

***** grid update (shift) ***** FROM  0.494575846176D+00 TO  0.494584519993D+00
***** norm grid shift *****  0.867381724852D-05

16 N 0.10D+01  0.40D-02  0.863639369118D+01                                        0.91D-04                 0  0  0   12  37  247
17 N 0.10D+01  0.51D-06  0.863554600656D+01                                        0.78D-11                 0  0  0    9  37  247

***** grid update (shift) ***** FROM  0.494584519993D+00 TO  0.494584530490D+00
***** norm grid shift *****  0.104972722667D-07

18 N 0.10D+01  0.49D-05  0.863554696793D+01                                        0.13D-09                 0  0  0   10  37  247

***** grid update (shift) ***** FROM  0.494584530490D+00 TO  0.494584530491D+00
***** norm grid shift *****  0.114049047983D-11

19 N 0.10D+01  0.53D-09  0.863554594209D+01                                        0.16D-17                 0  0  0    7  37  247
20 N 0.10D+01  0.16D-11  0.863554594198D+01                                        0.16D-21                 0  0  0    5  37  247
21 N 0.10D+01  0.56D-12  0.863554594198D+01                                        0.12D-22                 0  0  0    3  37  247
```

STATE VECTOR X

0.00D+00	0.41631000000000000D+02	-0.13440000000000000D+01	0.00000000000000000D+00	-0.3228363479138142D-01
0.10D+00	0.38657949046431280D+02	0.79072972498779340+01	0.13034017808650950-01	0.20454523786497490+00
0.20D+00	0.37161659168485070+02	0.13820234505374810+01	0.27698123249609490-01	0.37188983840283880-01
0.30D+00	0.36731526599822330+02	-0.11786477340179630+02	0.14015524200512150-01	-0.32088228469775130+00
0.40D+00	0.39123606992961190+02	-0.19695476421473340+02	-0.31212481550428810-01	-0.50342486587773510+00
0.50D+00	0.48978533293962340+02	0.20380028811447610+01	-0.59876553036208440-01	0.40320000357153110-01
0.60D+00	0.41565323218354480+02	0.23020909194857500+02	-0.24148107210011110-01	0.55367468456512980+00
0.70D+00	0.39335840383141830+02	0.14247758639831350+02	0.25248468035869690-01	0.36202975525331580+00
0.80D+00	0.39213011137952620+02	0.22133872839223420+00	0.43276325848638840-01	0.54662328848859790-02
0.90D+00	0.39509475680780670+02	-0.78766745216457270+01	0.31433571213539320-01	-0.19953992663761930+00
0.10D+01	0.41631000000000000+02	-0.13440000000000000+01	0.15438067486265110-01	-0.32461590947101880-01

CONTROL VECTOR U

0.00D+00	0.95666780772138500+00
0.10D+00	0.35900666223407760+00
0.20D+00	0.53557511168770100-01
0.30D+00	-0.51000887834434010-02
0.40D+00	0.33403623717339680+00
0.50D+00	0.11162882035116480+01
0.60D+00	0.26469543283400470+00
0.70D+00	-0.43219952315119620-01
0.80D+00	-0.19848884498798830-02
0.90D+00	0.27787440659358040+00
0.10D+01	0.87628323877568030+00

JUNCTION AND CONTACT POINTS OF CONSTRAINT S2

1 0.4945845300491D+00

CONVERGENCE HISTORY GLIDER PROBLEM WITH ALTITUDE CONSTRAINT (YMIN = -30).
TABLE F4

NUMBER OF GRIDPOINTS = 50
ORDER OF POLYNOMIALS = 3

IT	T	ALPHA	||D2||	OBJECTIVE	MERIT FUNCTION	LAGRANGIAN	PCRIT	RHOP	IQP	IG	IR	QPZ	DN	DR	C
0	R	0.10D+01	0.88D+00		0.294927849306D-03		0.59D+00	0.10D-02	0	0	0	0	148	1007	
1	R	0.10D+01	0.35D+00		0.114216471618D-05		0.23D-02	0.10D-02	0	0	0	0	148	1007	
2	R	0.10D+01	0.68D-01		0.114081807660D-08		0.23D-05	0.10D-02	0	0	0	0	148	1007	
3	R	0.10D+01	0.26D-02		0.381570131351D-14		0.76D-11	0.10D-02	0	0	0	0	148	1007	
4	R	0.10D+01	0.43D-05		0.276488924475D-26		0.55D-22	0.10D-02	0	0	0	2	148	1007	
	END OF RESTORATION PHASE														
5	I			0.1680700273250-01	0.1680700273250-01	0.1680700273250-01	0.55D-22	0.10D-02	0	0	0	4	148	1007	
5	N	0.10D+01	0.71D-01	0.1680504581500-01	0.1680504675220-01	0.1680504675220-01	0.44D-09	0.10D-02	0	0	0	6	148	1007	*
6	I			0.1680504581500-01	0.1680504683570-01	0.1680504683550-01	0.44D-09	0.10D-02	0	0	0	1	148	1007	
6	N	0.10D+01	0.71D-01	0.1680343093350-01	0.1680343086610-01	0.1680343086610-01	0.29D-09	0.10D-02	0	0	0	6	148	1007	*
7	I			0.1680343093350-01	0.1680343090450-01	0.1680343090430-01	0.29D-09	0.10D-02	0	0	0	1	148	1007	
7	N	0.10D+01	0.71D-01	0.1680174721120-01	0.1680174736770-01	0.1680174736760-01	0.30D-09	0.10D-02	0	0	0	6	148	1007	*
8	I			0.1680174721120-01	0.1680174743600-01	0.1680174743600-01	0.30D-09	0.10D-02	0	0	0	1	148	1007	
8	N	0.10D+01	0.71D-01	0.1680003175270-01	0.1680003203260-01	0.1680003203240-01	0.28D-09	0.10D-02	0	0	0	6	148	1007	*
9	I			0.1680003175270-01	0.1680003212300-01	0.1680003212300-01	0.28D-09	0.10D-02	0	0	0	1	148	1007	
9	N	0.10D+01	0.71D-01	0.1679825613770-01	0.1679825666000-01	0.1679825662980-01	0.28D-09	0.10D-02	0	0	0	6	148	1007	*
10	I			0.1679825613770-01	0.1679825674980-01	0.1679825674970-01	0.28D-09	0.10D-02	0	0	0	1	148	1007	
10	N	0.10D+01	0.71D-01	0.1679639931950-01	0.1679640009890-01	0.1679640009870-01	0.27D-09	0.10D-02	0	0	0	6	148	1007	*
11	I			0.1679639931950-01	0.1679640025500-01	0.1679640025490-01	0.27D-09	0.10D-02	0	0	0	1	148	1007	
11	N	0.10D+01	0.71D-01	0.1679443037510-01	0.1679443155980-01	0.1679443155970-01	0.26D-09	0.10D-02	0	0	0	6	148	1007	*
12	I			0.1679443037510-01	0.1679443176400-01	0.1679443176380-01	0.26D-09	0.10D-02	0	0	0	1	148	1007	
12	N	0.10D+01	0.71D-01	0.1679230908490-01	0.1679231084390-01	0.1679231084380-01	0.26D-09	0.10D-02	0	0	0	6	148	1007	*
13	I			0.1679230908490-01	0.1679231111420-01	0.1679231111410-01	0.26D-09	0.10D-02	0	0	0	1	148	1007	
13	N	0.10D+01	0.71D-01	0.1678998299740-01	0.1678998558000-01	0.1678998557980-01	0.26D-09	0.10D-02	0	0	0	6	148	1007	*
14	I			0.1678998299740-01	0.1678998594440-01	0.1678998594430-01	0.26D-09	0.10D-02	0	0	0	1	148	1007	
14	N	0.10D+01	0.71D-01	0.1678737943090-01	0.1678738323120-01	0.1678738323110-01	0.27D-09	0.10D-02	0	0	0	6	148	1007	*
15	I			0.1678737943090-01	0.1678738373570-01	0.1678738373560-01	0.27D-09	0.10D-02	0	0	0	1	148	1007	
15	N	0.10D+01	0.71D-01	0.1678439701820-01	0.1678440267720-01	0.1678440267710-01	0.29D-09	0.10D-02	0	0	0	6	148	1007	*
16	I			0.1678439701820-01	0.1678440339970-01	0.1678440339950-01	0.29D-09	0.10D-02	0	0	0	1	148	1007	
16	N	0.10D+01	0.71D-01	0.1678087904150-01	0.1678088772300-01	0.1678088772280-01	0.32D-09	0.10D-02	0	0	0	6	148	1007	*
17	I			0.1678087904150-01	0.1678088680960-01	0.1678088880950-01	0.32D-09	0.10D-02	0	0	0	1	148	1007	
17	N	0.10D+01	0.71D-01	0.1677656271010-01	0.1677657677150-01	0.1677657677130-01	0.38D-09	0.10D-02	0	0	0	6	148	1007	*
18	I			0.1677656271010-01	0.1677657853200-01	0.1677657853180-01	0.38D-09	0.10D-02	0	0	0	1	148	1007	
18	N	0.10D+01	0.71D-01	0.1677095819600-01	0.1677098319790-01	0.1677098317960-01	0.54D-09	0.10D-02	0	0	0	6	148	1007	*
19	I			0.1677095819600-01	0.1677098638820-01	0.1677098638790-01	0.54D-09	0.10D-02	0	0	0	1	148	1007	
19	N	0.10D+01	0.71D-01	0.1676303010150-01	0.1676308193130-01	0.1676308193080-01	0.10D-08	0.10D-02	0	0	0	6	148	1007	*
20	I			0.1676303010150-01	0.1676308904600-01	0.1676308904550-01	0.10D-08	0.10D-02	0	0	0	1	148	1007	
20	N	0.10D+01	0.71D-01	0.1675060016020-01	0.1675073127010-01	0.1675073126840-01	0.34D-08	0.10D-02	0	0	0	6	148	1007	*
21	I			0.1675060016020-01	0.1675075205740-01	0.1675075205570-01	0.34D-08	0.10D-02	0	0	0	1	148	1007	
21	N	0.10D+01	0.10D+02	0.1643956133650-01	0.1655961239780-01	0.1650889926860-01	0.10D+00	0.10D-02	0	0	0	13	148	1007	
22	N	0.10D+01	0.14D+01	0.1651281962650-01	0.1651281164900-01	0.1651270003970-01	0.22D-03	0.10D-02	0	0	0	3	148	1007	*
23	I			0.1651281962650-01	0.1653961957680-01	0.1653950796750-01	0.22D-03	0.10D-02	0	0	0	1	148	1007	
23	N	0.10D+01	0.66D-01	0.1650994020600-01	0.1651011432100-01	0.1651011430790-01	0.26D-07	0.10D-02	0	0	0	4	148	1007	*
24	I			0.1650994020600-01	0.1651012333450-01	0.1651012332140-01	0.26D-07	0.10D-02	0	0	0	1	148	1007	
24	N	0.50D+00	0.15D+02	0.1650472494420-01	0.1649588155250-01	0.1649051256120-01	0.11D-01	0.10D-02	0	0	0	14	148	1007	
25	N	0.10D+01	0.33D+00	0.1647515426180-01	0.1648238324270-01	0.1648220900210-01	0.35D-03	0.10D-02	0	0	0	12	148	1007	
26	N	0.10D+01	0.92D+00	0.1648256999370-01	0.1648213982750-01	0.1648212897160-01	0.22D-05	0.10D-01	0	0	0	11	148	1007	
27	N	0.10D+01	0.40D-01	0.1648212918030-01	0.1648212846920-01	0.1648212846300-01	0.12D-10	0.10D+01	0	0	0	10	148	1007	
28	N	0.10D+01	0.31D-03	0.1648212846300-01	0.1648212846290-01	0.1648212846290-01	0.28D-19	0.10D+01	0	0	0	9	148	1007	
29	N	0.10D+01	0.14D-05	0.1648212846290-01	0.1648212846290-01	0.1648212846290-01	0.20D-23	0.10D+01	0	0	0	116	148	1007	
30	N	0.10D+01	0.30D-10	0.1648212846290-01	0.1648212846290-01	0.1648212846290-01	0.24D-23	0.10D+01	0	0	0	13	148	1007	

CONVERGENCE HISTORY UNCONSTRAINED REENTRY PROBLEM.
TABLE F5

```
NUMBER OF GRIDPOINTS = 50
ORDER OF POLYNOMIALS =  3

IT T    ALPHA      ||D2||          OBJECTIVE        MERIT FUNCTION          LAGRANGIAN        PCRIT      RHOP  IQP IG IR  QPZ   DN   DR C

 0 R 0.10D+01  0.35D+00                      0.230490100287D+00                          0.46D+01  0.10D+00   19  0  0   18  132  822
 1 R 0.10D+01  0.21D+00                      0.139621723486D-02                          0.28D-01  0.10D+00    4  0  0    3  130  824
 2 R 0.10D+01  0.20D-01                      0.104062945419D-06                          0.21D-05  0.10D+00    0  0  0    0  130  824
 3 R 0.10D+01  0.29D-03                      0.136149530964D-14                          0.27D-13  0.10D+00    0  0  0    0  130  824

   END OF RESTORATION PHASE

 4 I                  -0.266086386893D+00 -0.266086384951D+00 -0.266086384951D+00  0.27D-13  0.10D+00    3  0  0    6  131  823
 4 N 0.50D+00  0.18D+02 -0.267615472782D+00 -0.267059148662D+00 -0.267074737774D+00  0.31D-03  0.10D+00   12  0  0  123  132  822
 5 N 0.50D+00  0.11D+02 -0.267649125884D+00 -0.267304417048D+00 -0.267319901627D+00  0.31D-03  0.10D+00    4  0  0   57  131  823
 6 N 0.10D+01  0.71D+01 -0.267605880887D+00 -0.267509849939D+00 -0.267533998480D+00  0.48D-03  0.10D+00    5  0  0   30  133  821
 7 N 0.10D+01  0.39D+00 -0.267585916311D+00 -0.267569743186D+00 -0.267569832199D+00  0.18D-05  0.10D+00    0  0  0    7  133  821
 8 N 0.10D+01  0.55D-01 -0.267569838053D+00 -0.267569832486D+00 -0.267569832487D+00  0.30D-10  0.10D+00    0  0  0    7  133  821
 9 N 0.10D+01  0.11D-02 -0.267569832490D+00 -0.267569832490D+00 -0.267569832490D+00  0.65D-17  0.10D+00    0  0  0    6  133  821
10 N 0.10D+01  0.30D-04 -0.267569832489D+00 -0.267569832489D+00 -0.267569832489D+00  0.46D-23  0.10D+00    0  0  0   22  133  821
11 N 0.10D+01  0.12D-09 -0.267569832489D+00 -0.267569832489D+00 -0.267569832489D+00  0.47D-24  0.10D+00    0  0  0   23  133  821

   START OF SECOND STAGE

***** grid update ( add ) ***** AT  0.186127016654D+00
***** grid update ( add ) ***** AT  0.293872983346D+00

11 N 0.10D+01  0.13D+00 -0.267569832489D+00                                          0.16D-01             0  0  0   24  137  855
12 N 0.10D+01  0.22D+00 -0.267570788745D+00                                          0.32D-01             0  0  0    6  137  855
13 N 0.10D+01  0.36D-02 -0.267564582654D+00                                          0.25D-01             0  0  0    6  137  855
14 N 0.10D+01  0.11D-03 -0.267564510994D+00                                          0.25D-01             0  0  0   18  137  855
15 N 0.10D+01  0.65D-08 -0.267564510999D+00                                          0.25D-01             0  0  0   23  137  855

***** grid update (shift) ***** FROM  0.186127016654D+00 TO  0.187332209502D+00
***** grid update (shift) ***** FROM  0.293872983346D+00 TO  0.296460810204D+00
***** norm grid shift *****  0.160035654764D-01

16 N 0.10D+01  0.64D+01 -0.267564510999D+00                                          0.35D+01             0  0  0   26  137  855
17 N 0.10D+01  0.65D+01 -0.268979693064D+00                                          0.19D+00             0  0  0    7  137  855
18 N 0.10D+01  0.33D+00 -0.267595382251D+00                                          0.18D-03             0  0  0    7  137  855
19 N 0.10D+01  0.15D+00 -0.267569962998D+00                                          0.14D-05             0  0  0    6  137  855
20 N 0.10D+01  0.25D-03 -0.267564251622D+00                                          0.18D-08             0  0  0    6  137  855
21 N 0.10D+01  0.11D-03 -0.267564238001D+00                                          0.71D-12             0  0  0   23  137  855
22 N 0.10D+01  0.38D-08 -0.267564237999D+00                                          0.15D-20             0  0  0   23  137  855

***** norm grid shift *****  0.112314974881D-01

23 N 0.10D+01  0.35D+01 -0.267564237999D+00                                          0.36D-01             0  0  0   26  137  855
24 N 0.10D+01  0.74D+00 -0.267798269793D+00                                          0.16D-03             0  0  0    7  137  855
25 N 0.10D+01  0.81D-02 -0.267545328274D+00                                          0.82D-02             0  0  0    6  137  855
26 N 0.10D+01  0.39D-04 -0.267542695211D+00                                          0.82D-02             0  0  0    9  137  855
27 N 0.10D+01  0.13D-08 -0.267542693262D+00                                          0.82D-02             0  0  0   21  137  855

***** grid update (shift) ***** FROM  0.192307692308D+00 TO  0.187153814906D+00
***** grid update (shift) ***** FROM  0.307692307692D+00 TO  0.296348995870D+00
***** norm grid shift *****  0.113433118224D-01
```

```
28 N 0.10D+01  0.26D+01 -0.267542693262D+00                                   0.78D+00          0  0  0   26  137  855
29 N 0.10D+01  0.17D+00 -0.267634083131D+00                                   0.37D-02          0  0  0    7  137  855
30 N 0.10D+01  0.85D-03 -0.267565097284D+00                                   0.62D-07          0  0  0    6  137  855
31 N 0.10D+01  0.17D-04 -0.267564287211D+00                                   0.42D-14          0  0  0   22  137  855
32 N 0.10D+01  0.21D-08 -0.267564287200D+00                                   0.24D-21          0  0  0   23  137  855

***** grid update (shift) ***** FROM  0.296348995870D+00 TO  0.296499483312D+00
***** norm grid shift *****  0.150487441797D-03

33 N 0.10D+01  0.17D-02 -0.267564287200D+00                                   0.45D-06          0  0  0   24  137  855
34 N 0.10D+01  0.14D-03 -0.267564303986D+00                                   0.23D-10          0  0  0   22  137  855
35 N 0.10D+01  0.58D-08 -0.267564286929D+00                                   0.32D-20          0  0  0   23  137  855

STATE VECTOR X

 0.00D+00   0.36000000000000000D+00  -0.14137160000000000D+00   0.19138700000000000D-01   0.2468144433183481D+03
 0.10D+00   0.36073043704043470D+00  -0.12054449452247640D+00   0.13574026217138440D-01   0.2468144433183481D+03
 0.20D+00   0.34520156298675320D+00  -0.72488963949854490D-01   0.92247752924823410D-02   0.2468144433183481D+03
 0.30D+00   0.30647142301436540D+00   0.25395432371517270D-01   0.83117109394970640D-02   0.2468144433183481D+03
 0.40D+00   0.28906539576445560D+00   0.34284362289183190D-01   0.96245627187062740D-02   0.2468144433183481D+03
 0.50D+00   0.28171722435947170D+00   0.21477290445846920D-01   0.10544212872402380D-01   0.2468144433183481D+03
 0.60D+00   0.27733664798989150D+00   0.14230617922994630D-01   0.11122933729213140D-01   0.2468144433183481D+03
 0.70D+00   0.27423535574811070D+00   0.96454800409498860D-02   0.11507146886641250D-01   0.2468144433183481D+03
 0.80D+00   0.27176105635094460D+00   0.62114418950739540D-02   0.11760936453776660D-01   0.2468144433183481D+03
 0.90D+00   0.26960201524155520D+00   0.31644729827747750D-02   0.11910539914177720D-01   0.2468144433183481D+03
 0.10D+01   0.26756428692914770D+00  -0.15949087766316640D-16   0.11961700000000000D-01   0.2468144433183481D+03

CONTROL VECTOR U

 0.00D+00    0.1355386076373254D+01
 0.10D+00    0.1383897959963399D+01
 0.20D+00    0.9717697751874525D+00
 0.30D+00    0.4101397228499013D+00
 0.40D+00   -0.4013762177606460D+00
 0.50D+00   -0.6284552456295689D+00
 0.60D+00   -0.7385460022357578D+00
 0.70D+00   -0.8161212760206942D+00
 0.80D+00   -0.8796202407893045D+00
 0.90D+00   -0.9344270766911634D+00
 0.10D+01   -0.9827879432329048D+00

 JUNCTION AND CONTACT POINTS OF CONSTRAINT S1

      1  0.187153814906D+00  0.296499483312D+00
```

CONVERGENCE HISTORY REENTRY PROBLEM WITH ACCELERATION CONSTRAINT (NMAX = 6).
TABLE F6

```
NUMBER OF GRIDPOINTS = 25
ORDER OF POLYNOMIALS =  3
```

IT	T	ALPHA	||D2||	OBJECTIVE	MERIT FUNCTION	LAGRANGIAN	PCRIT	RHOP	IQP	IG	IR	QPZ	DN	DR	C
0	R	0.10D+01	0.12D+01		0.155420617729D+00		0.31D+00	0.10D+01	4	0	0	2	71	711	
1	R	0.10D+01	0.43D+00		0.745799264260D-02		0.15D-01	0.10D+01	3	0	0	1	72	710	
		END OF RESTORATION PHASE													
2	I			0.171751947484D-01	0.294036982010D-01	0.219457055584D-01	0.15D-01	0.10D+01	4	0	0	3	71	711	
2	N	0.50D+00	0.28D+01	0.169597796622D-01	0.224786684619D-01	0.181733685245D-01	0.86D-02	0.10D+01	0	0	0	2	71	711	*
3	I			0.169597796622D-01	0.224932708088D-01	0.181918466830D-01	0.86D-02	0.10D+01	3	0	0	2	72	710	
3	N	0.10D+01	0.58D+00	0.166037685121D-01	0.173939275829D-01	0.166483914159D-01	0.15D-02	0.10D+01	0	0	0	2	72	710	*
4	I			0.166037685121D-01	0.176467876914D-01	0.169012515244D-01	0.15D-02	0.10D+01	4	0	0	3	71	711	
4	N	0.25D+00	0.16D+01	0.166429024123D-01	0.600704713217D-01	0.554136218893D-02	0.11D-02	0.10D+03	0	0	0	2	71	711	*
5	I			0.166429024123D-01	0.6570382265740-01	0.168075463666D-01	0.98D-03	0.10D+03	0	0	0	1	71	711	
5	N	0.50D+00	0.55D+00	0.166932522511D-01	0.318487010994D-01	0.146835950931D-01	0.34D-03	0.10D+03	0	0	0	2	71	711	*
6	I			0.166932522511D-01	0.328875380472D-01	0.167572246500D-01	0.32D-03	0.10D+03	0	0	0	1	71	711	
6	N	0.10D+01	0.43D+00	0.167070959760D-01	0.207413279508D-01	0.167122315207D-01	0.81D-04	0.10D+03	0	0	0	2	71	711	*
7	I			0.167070959760D-01	0.207485302717D-01	0.167194338416D-01	0.81D-04	0.10D+03	0	0	0	1	71	711	
7	N	0.13D+00	0.16D+01	0.166737562070D-01	0.200709335110D-01	0.166851752090D-01	0.68D-04	0.10D+03	0	0	0	13	71	711	
8	N	0.13D+00	0.11D+01	0.166469337234D-01	0.196015381263D-01	0.166575129701D-01	0.59D-04	0.10D+03	0	0	0	12	71	711	
9	N	0.13D+00	0.72D+00	0.166252110713D-01	0.191788345987D-01	0.166350175322D-01	0.51D-04	0.10D+03	0	0	0	13	71	711	
10	N	0.13D+00	0.50D+00	0.166075909261D-01	0.187805686671D-01	0.166166749103D-01	0.43D-04	0.10D+03	0	0	0	13	71	711	
11	N	0.25D+00	0.40D+00	0.165792526830D-01	0.184566497167D-01	0.165885118941D-01	0.37D-04	0.10D+03	0	0	0	13	71	711	
12	N	0.25D+00	0.95D+00	0.165609410521D-01	0.179539599217D-01	0.165696367846D-01	0.28D-04	0.10D+03	0	0	0	12	71	711	
13	N	0.50D+00	0.13D+01	0.165375504157D-01	0.175044797942D-01	0.165471185088D-01	0.19D-04	0.10D+03	0	0	0	12	71	711	
14	N	0.50D+00	0.15D+01	0.165304139627D-01	0.170721989033D-01	0.165374080721D-01	0.11D-04	0.10D+03	0	0	0	12	71	711	
15	N	0.50D+00	0.10D+01	0.165292879354D-01	0.168357928508D-01	0.165336085530D-01	0.60D-05	0.10D+03	0	0	0	11	71	711	
16	N	0.50D+00	0.57D+00	0.165298111643D-01	0.166729259270D-01	0.165322542664D-01	0.28D-05	0.10D+03	0	0	0	11	71	711	
17	N	0.50D+00	0.25D+00	0.165304670927D-01	0.166234863771D-01	0.165317830221D-01	0.18D-05	0.10D+03	0	0	0	11	71	711	
18	N	0.13D+00	0.13D+00	0.165305773600D-01	0.166095512292D-01	0.165317284987D-01	0.16D-05	0.10D+03	0	0	0	11	71	711	
19	N	0.13D+00	0.13D+00	0.165306765898D-01	0.165985211864D-01	0.165316837030D-01	0.13D-05	0.10D+03	0	0	0	11	71	711	
20	N	0.13D+00	0.13D+00	0.165307657106D-01	0.165895554588D-01	0.165316469018D-01	0.12D-05	0.10D+03	0	0	0	11	71	711	
21	N	0.13D+00	0.12D+00	0.165308456545D-01	0.165820276936D-01	0.165316167180D-01	0.10D-05	0.10D+03	0	0	0	11	71	711	
22	N	0.13D+00	0.11D+00	0.165309173099D-01	0.165755023139D-01	0.165315920359D-01	0.88D-06	0.10D+03	0	0	0	11	71	711	
23	N	0.13D+00	0.11D+00	0.165309814967D-01	0.165697031202D-01	0.165315719329D-01	0.76D-06	0.10D+03	0	0	0	11	71	711	
24	N	0.13D+00	0.98D-01	0.165310389564D-01	0.165644748199D-01	0.165315556348D-01	0.66D-06	0.10D+03	0	0	0	11	71	711	
25	N	0.13D+00	0.91D-01	0.165310903541D-01	0.165597430313D-01	0.165315424857D-01	0.56D-06	0.10D+03	0	0	0	11	71	711	
26	N	0.13D+00	0.92D-01	0.165311362855D-01	0.165554789358D-01	0.165315319288D-01	0.48D-06	0.10D+03	0	0	0	11	71	711	
27	N	0.25D+00	0.91D-01	0.165312187120D-01	0.165519403171D-01	0.165315161422D-01	0.41D-06	0.10D+03	0	0	0	11	71	711	
28	N	0.25D+00	0.84D-01	0.165312830071D-01	0.165472051158D-01	0.165315064438D-01	0.31D-06	0.10D+03	0	0	0	11	71	711	
29	N	0.25D+00	0.74D-01	0.165313328384D-01	0.165426338075D-01	0.165315006004D-01	0.22D-06	0.10D+03	0	0	0	11	71	711	
30	N	0.50D+00	0.62D-01	0.165314101292D-01	0.165381906818D-01	0.165314946245D-01	0.13D-06	0.10D+03	0	0	0	11	71	711	
31	N	0.10D+01	0.37D-01	0.165314916967D-01	0.165327796676D-01	0.165314923405D-01	0.26D-07	0.10D+03	0	0	0	11	71	711	
32	N	0.10D+01	0.28D-02	0.165314923409D-01	0.165314923405D-01	0.165314923387D-01	0.35D-13	0.10D+03	0	0	0	10	71	711	
33	N	0.10D+01	0.18D-04	0.165314923387D-01	0.165314923387D-01	0.165314923387D-01	0.93D-22	0.10D+03	0	0	0	52	71	711	
34	N	0.10D+01	0.43D-06	0.165314923387D-01	0.165314923387D-01	0.165314923387D-01	0.53D-24	0.10D+03	0	0	0	57	71	711	
35	N	0.10D+01	0.29D-08	0.165314923387D-01	0.165314923387D-01	0.165314923387D-01	0.69D-24	0.10D+03	0	0	0	55	71	711	
		START OF SECOND STAGE													
35	N	0.10D+01	0.10D+01	0.168978787647D-01			0.41D+01		0	0	0	71	71	711	
36	N	0.10D+01	0.39D-01	0.165489950175D-01			0.22D-01		0	0	0	11	71	711	
37	N	0.10D+01	0.11D-01	0.165324128795D-01			0.56D-05		0	0	0	10	71	711	
38	N	0.10D+01	0.11D-03	0.165325628257D-01			0.19D-08		0	0	0	9	71	711	
39	N	0.10D+01	0.17D-05	0.165325630588D-01			0.43D-12		0	0	0	60	71	711	
40	N	0.10D+01	0.42D-07	0.165325630588D-01			0.23D-15		0	0	0	56	71	711	
41	N	0.10D+01	0.57D-09	0.165325630588D-01			0.45D-19		0	0	0	36	71	711	

```
***** grid update (shift) ***** FROM  0.480000000000D+00 TO  0.491385569084D+00
***** norm grid shift *****  0.113855690835D-01
```

```
***** grid update (shift) ***** FROM  0.491385569084D+00 TO  0.489150483855D+00
***** norm grid shift *****  0.223508522835D-02

42 N 0.10D+01  0.16D+00  0.164621549868D-01                                    0.14D+00          0  0  0   70   71  711
43 N 0.10D+01  0.26D-02  0.165332833258D-01                                    0.24D-04          0  0  0   10   71  711
44 N 0.10D+01  0.13D-03  0.165328301374D-01                                    0.25D-08          0  0  0   53   71  711
45 N 0.10D+01  0.15D-05  0.165328308810D-01                                    0.32D-12          0  0  0   57   71  711
46 N 0.10D+01  0.34D-07  0.165328308810D-01                                    0.15D-15          0  0  0   56   71  711
47 N 0.10D+01  0.49D-09  0.165328308810D-01                                    0.32D-19          0  0  0   34   71  711

***** grid update (shift) ***** FROM  0.489150483855D+00 TO  0.489063678541D+00
***** norm grid shift *****  0.868053139880D-04

48 N 0.10D+01  0.48D-02  0.165328324938D-01                                    0.22D-05          0  0  0   66   71  711
49 N 0.10D+01  0.93D-05  0.165328310422D-01                                    0.51D-10          0  0  0   42   71  711
50 N 0.10D+01  0.91D-06  0.165328310448D-01                                    0.99D-13          0  0  0   57   71  711
51 N 0.10D+01  0.79D-08  0.165328310448D-01                                    0.93D-17          0  0  0   54   71  711

***** grid update (shift) ***** FROM  0.489063678541D+00 TO  0.489063295172D+00
***** norm grid shift *****  0.383369111928D-06

52 N 0.10D+01  0.21D-04  0.165328310518D-01                                    0.43D-10          0  0  0   60   71  711
53 N 0.10D+01  0.75D-07  0.165328310448D-01                                    0.13D-14          0  0  0   57   71  711
54 N 0.10D+01  0.40D-08  0.165328310448D-01                                    0.19D-17          0  0  0   52   71  711

***** grid update (shift) ***** FROM  0.489063295172D+00 TO  0.489063295276D+00
***** norm grid shift *****  0.103781337690D-09

55 N 0.10D+01  0.57D-08  0.165328310448D-01                                    0.32D-17          0  0  0   55   71  711
56 N 0.10D+01  0.68D-11  0.165328310448D-01                                    0.49D-22          0  0  0    7   71  711

STATE VECTOR X

 0.00D+00   0.3500000000000000D+00  -0.1003565390000000D+00   0.1913533000000000D-01   0.0000000000000000D+00
 0.10D+00   0.3505971412569669D+00  -0.7051712407574255D-01   0.1341250228804272D-01   0.1377000174809896D+02
 0.20D+00   0.3288086737473640D+00  -0.6503938494791618D-01   0.9317052269744551D-02   0.2742167482523260D+02
 0.30D+00   0.1014158621488773D+00   0.7774065694550526D-01   0.6205216794870917D-02   0.3616909655510078D+02
 0.40D+00   0.7706606438663080D-01   0.1072624552951307D+00   0.8269896450573234D-02   0.3947367069705462D+02
 0.50D+00   0.7280891544382313D-01  -0.1453681561637122D-01   0.8972702430172123D-02   0.4243483618055976D+02
 0.60D+00   0.6900126343033364D-01  -0.1361776257195076D+00   0.7918246470287154D-02   0.4525116923553719D+02
 0.70D+00   0.5450059933805496D-01  -0.1524475752641374D+00   0.5983600487191049D-02   0.4773721985888177D+02
 0.80D+00   0.3305169431337365D-01  -0.7752587609383452D-01   0.5080715955315987D-02   0.4944332628182864D+02
 0.90D+00   0.2065950277239199D-01  -0.1891933773653124D+00   0.4510689460658851D-02   0.5048544997086090+02
 0.10D+01   0.1239929000003645D-01  -0.4579246880002607D+00   0.3602263999999817D-02   0.5110198000000011D+02

CONTROL VECTOR U

 0.00D+00   0.1086573457026461D+01
 0.10D+00   0.1103713787427391D+01
 0.20D+00   0.8490149228278056D+00
 0.30D+00  -0.1444883622039660D+01
 0.40D+00  -0.2261954226972686D+01
 0.50D+00  -0.1792741583017244D+01
 0.60D+00  -0.1706474611448757D+01
 0.70D+00  -0.1864068939875445D+01
 0.80D+00  -0.2114784085705738D+01
 0.90D+00  -0.2005928817041350D+01
 0.10D+01  -0.2144879112843496D+01

JUNCTION AND CONTACT POINTS OF CONSTRAINT S2

      1  0.489063295276D+00
```

CONVERGENCE HISTORY REENTRY PROBLEM WITH ALTITUDE CONSTRAINT (XIMAX = 0.009).
TABLE F7

```
NUMBER OF GRIDPOINTS =  40
ORDER OF POLYNOMIALS =   2

IT T    ALPHA      ||D2||          OBJECTIVE       MERIT FUNCTION          LAGRANGIAN        PCRIT      RHOP      IQP IG IR  QPZ  ON  DR C

 0 R 0.10D+01  0.38D-01                          0.423749634978D-06                        0.85D-06  0.10D+01   10  0  0   35  31  192
 1 R 0.10D+01  0.10D-03                          0.860764355028D-15                        0.17D-14  0.10D+01    3  0  0    5  32  191

   END OF RESTORATION PHASE

 2 I                     0.257464885267D+01  0.253452336817D+01  0.253268768270D+01  0.37D-02  0.10D+01   17  0  0   47  27  196
 2 N 0.10D+01  0.76D+00  0.240646367287D+01  0.243547619323D+01  0.242832025815D+01  0.14D-01  0.10D+01   11  0  0   56  19  204
 3 N 0.10D+01  0.83D-01  0.243197932063D+01  0.242384072250D+01  0.241556317715D+01  0.17D-01  0.10D+01    0  0  0    5  19  204
 4 N 0.10D+01  0.36D-02  0.243209228269D+01  0.243126897128D+01  0.243044554424D+01  0.16D-03  0.10D+02    0  0  0    5  19  204
 5 N 0.10D+01  0.28D-05  0.243209239780D+01  0.243126895963D+01  0.243044552146D+01  0.16D-03  0.10D+02    0  0  0    4  19  204
 6 N 0.10D+01  0.11D-11  0.243209239780D+01  0.243126895963D+01  0.243044552146D+01  0.16D-03  0.10D+02    0  0  0    2  19  204

   START OF SECOND STAGE

***** grid update (shift) ***** FROM  0.250000000000D+00 TO  0.269542610095D+00
***** grid update (shift) ***** FROM  0.350000000000D+00 TO  0.337375215392D+00
***** grid update (shift) ***** FROM  0.500000000000D+00 TO  0.488368704743D+00
***** grid update (shift) ***** FROM  0.700000000000D+00 TO  0.671469951075D+00
***** grid update (shift) ***** FROM  0.850000000000D+00 TO  0.847879561436D+00
***** norm grid shift *****  0.2853004892530-01

 6 N 0.10D+01  0.16D+00  0.243607370726D+01                                        0.13D-01            0  0  0    6  19  204
 7 N 0.10D+01  0.12D-01  0.243224121481D+01                                        0.16D-03            0  0  0    5  19  204
 8 N 0.10D+01  0.15D-04  0.243186710059D+01                                        0.94D-10            0  0  0    5  19  204
 9 N 0.10D+01  0.22D-10  0.243186711916D+01                                        0.28D-21            0  0  0    3  19  204

***** grid update (shift) ***** FROM  0.269542610095D+00 TO  0.267767976600D+00
***** grid update (shift) ***** FROM  0.337375215392D+00 TO  0.339291301013D+00
***** grid update (shift) ***** FROM  0.488368704743D+00 TO  0.488359798384D+00
***** grid update (shift) ***** FROM  0.671469951075D+00 TO  0.671598656342D+00
***** grid update (shift) ***** FROM  0.847879561436D+00 TO  0.847884340240D+00
***** norm grid shift *****  0.191608562131D-02

10 N 0.10D+01  0.83D-02  0.243180258606D+01                                        0.35D-04            0  0  0    6  19  204
11 N 0.10D+01  0.18D-03  0.243187311710D+01                                        0.35D-06            0  0  0    5  19  204
12 N 0.10D+01  0.26D-08  0.243186863595D+01                                        0.32D-06            0  0  0    3  19  204

***** grid update (shift) ***** FROM  0.267767976600D+00 TO  0.267765607362D+00
***** grid update (shift) ***** FROM  0.339291301013D+00 TO  0.339246535579D+00
***** grid update (shift) ***** FROM  0.488359798384D+00 TO  0.488902688767D+00
***** grid update (shift) ***** FROM  0.671598656342D+00 TO  0.675015506602D+00
***** grid update (shift) ***** FROM  0.847884340240D+00 TO  0.848735963857D+00
***** norm grid shift *****  0.341685026011D-02

13 N 0.10D+01  0.17D-01  0.243155215611D+01                                        0.68D-04            0  0  0    5  19  204
14 N 0.10D+01  0.14D-04  0.243186926823D+01                                        0.31D-06            0  0  0    4  19  204
15 N 0.10D+01  0.30D-10  0.243186856455D+01                                        0.31D-06            0  0  0    3  19  204

***** grid update (shift) ***** FROM  0.267765607362D+00 TO  0.267768829349D+00
***** grid update (shift) ***** FROM  0.339246535579D+00 TO  0.337755039497D+00
***** grid update (shift) ***** FROM  0.488902688767D+00 TO  0.487975431092D+00
***** grid update (shift) ***** FROM  0.675015506602D+00 TO  0.674652623919D+00
***** grid update (shift) ***** FROM  0.848735963857D+00 TO  0.849825271221D+00
***** norm grid shift *****  0.149149608191D-02
```

```
16 N 0.10D+01  0.66D-02  0.243165498287D+01                                  0.25D-04        0  0  0    6   19  204
17 N 0.10D+01  0.45D-03  0.243187048146D+01                                  0.12D-06        0  0  0    5   19  204
18 N 0.10D+01  0.59D-07  0.243186762724D+01                                  0.12D-10        0  0  0    4   19  204
19 N 0.10D+01  0.39D-13  0.243186762802D+01                                  0.12D-10        0  0  0    1   19  204

***** grid update (shift) ***** FROM  0.267768829349D+00 TO  0.267763478421D+00
***** grid update (shift) ***** FROM  0.337755039497D+00 TO  0.339212480285D+00
***** grid update (shift) ***** FROM  0.487975431092D+00 TO  0.487974770600D+00
***** grid update (shift) ***** FROM  0.674652623919D+00 TO  0.674641711930D+00
***** norm grid shift *****  0.145744078831D-02

20 N 0.10D+01  0.62D-02  0.243191873289D+01                                  0.14D-04        0  0  0    5   19  204
21 N 0.10D+01  0.11D-03  0.243187094870D+01                                  0.30D-06        0  0  0    5   19  204
22 N 0.10D+01  0.84D-09  0.243186868011D+01                                  0.29D-06        0  0  0    3   19  204

***** grid update (shift) ***** FROM  0.267763478421D+00 TO  0.267760037732D+00
***** grid update (shift) ***** FROM  0.339212480285D+00 TO  0.339179917700D+00
***** grid update (shift) ***** FROM  0.487974770600D+00 TO  0.487975464986D+00
***** grid update (shift) ***** FROM  0.674641711930D+00 TO  0.674650298865D+00
***** norm grid shift *****  0.325625853181D-04

23 N 0.10D+01  0.14D-03  0.243186678349D+01                                  0.28D-06        0  0  0    5   19  204
24 N 0.10D+01  0.53D-07  0.243186865606D+01                                  0.27D-06        0  0  0    4   19  204
25 N 0.10D+01  0.28D-13  0.243186865496D+01                                  0.27D-06        0  0  0    1   19  204

***** grid update (shift) ***** FROM  0.267760037732D+00 TO  0.267753973036D+00
***** grid update (shift) ***** FROM  0.339179917700D+00 TO  0.337784246027D+00
***** grid update (shift) ***** FROM  0.487975464986D+00 TO  0.487938384022D+00
***** grid update (shift) ***** FROM  0.674650298865D+00 TO  0.674649658421D+00
***** norm grid shift *****  0.139567167265D-02

26 N 0.10D+01  0.62D-02  0.243181699085D+01                                  0.15D-04        0  0  0    5   19  204
27 N 0.10D+01  0.46D-03  0.243186982313D+01                                  0.12D-06        0  0  0    5   19  204
28 N 0.10D+01  0.60D-07  0.243186765281D+01                                  0.13D-10        0  0  0    4   19  204
29 N 0.10D+01  0.90D-14  0.243186765359D+01                                  0.13D-10        0  0  0    1   19  204

***** grid update (shift) ***** FROM  0.267753973036D+00 TO  0.267777831606D+00
***** grid update (shift) ***** FROM  0.337784246027D+00 TO  0.337777359951D+00
***** grid update (shift) ***** FROM  0.674649658421D+00 TO  0.674650760094D+00
***** norm grid shift *****  0.238585699480D-04

30 N 0.10D+01  0.11D-03  0.243186907750D+01                                  0.26D-08        0  0  0    5   19  204
31 N 0.10D+01  0.24D-08  0.243186764825D+01                                  0.82D-11        0  0  0    3   19  204

***** grid update (shift) ***** FROM  0.267777831606D+00 TO  0.267746442002D+00
***** grid update (shift) ***** FROM  0.337777359951D+00 TO  0.337777319552D+00
***** grid update (shift) ***** FROM  0.674650760094D+00 TO  0.674642262767D+00
***** norm grid shift *****  0.313896042167D-04

32 N 0.10D+01  0.15D-03  0.243186582324D+01                                  0.42D-08        0  0  0    5   19  204
33 N 0.10D+01  0.30D-08  0.243186764945D+01                                  0.46D-14        0  0  0    3   19  204

***** grid update (shift) ***** FROM  0.267746442002D+00 TO  0.267746410354D+00
***** grid update (shift) ***** FROM  0.337777319552D+00 TO  0.337777359951D+00
***** grid update (shift) ***** FROM  0.674642262767D+00 TO  0.674642057250D+00
***** norm grid shift *****  0.205517203630D-06

34 N 0.10D+01  0.10D-05  0.243186765907D+01                                  0.25D-12        0  0  0    4   19  204
35 N 0.10D+01  0.79D-13  0.243186764932D+01                                  0.15D-15        0  0  0    1   19  204
```

```
CONSTRAINT S1  ACTIVE AT COLLOCATION POINTS

     1       35       40

CONSTRAINT S2  ACTIVE AT BREAK POINTS

     1        5        7
     1       10       14

JUNCTION AND CONTACT POINTS OF CONSTRAINT S1

     1  0.849825271221D+00  0.100000000000D+01

JUNCTION AND CONTACT POINTS OF CONSTRAINT S2

     1  0.267746410354D+00  0.337777359951D+00
     1  0.487938364022D+00  0.674642057250D+00
```

CONVERGENCE HISTORY SERVO PROBLEM (VMAX,1 = 1.5, AMAX,1 = 3, C=1).
TABLE F8

References.

Bals J. (1983) *Ein Sequentieller Gradienten-Restorations Algoritmus zur Lösung Optimaler Steuerungsproblemen*. M.Sc. Thesis, Technische Universität Munchen.

Bazaraa M.S. and C.M. Shetty (1976) *Foundations of Optimization*. Springer New York.

Bertsekas D.P. (1982) *Constrained Optimization and Lagrange Multiplier Methods*. Academic Press, New York.

Bobrow J.E., S. Dubowsky and J.S. Gibson (1985) *Time optimal control of robotic manipulators along specified paths*. The International Journal of Robotics Research, Vol.4, No.3.

Bock H.G. (1983) *Numerische Behandlung von Zustandsbeschränkten und Chebychev-Steuerungsproblemen*. Syllabus of the course 'Optimierungsverfahren' of the Carl Cranz Geselschaft, Manuskript 9, Oberphaffenhofen FRG.

de Boor C. and B. Swartz (1973) *Collocation at Gaussian Points*. SIAM Journal on Numerical Analysis, Vol. 10, No. 4.

de Boor C. (1978) *A Practical Guide to Splines*. Series in Applied Mathematical Sciences, Vol. 27, Spinger Verlag, New York.

Bryson A.E., W.F. Denham and S.E. Dreyfus (1963a) *Optimal programming problems with inequality constraints. I : Necessary conditions for extremal solutions*. AIAA Journal, Vol. 1.

Bryson A.E., W.F. Denham and S.E. Dreyfus (1963b) *Optimal programming problems with inequality constraints. II : Solution by steepest ascent*. AIAA Journal, Vol. 2.

Bryson A.E. and Y.C. Ho (1975) *Applied Optimal Control : optimization, estimation and control (revised printing)*. Hemisphere Publishing Corporation, Washington.

Bryson A.E. and A. Weinreb (1985) *Minimum-time control of a two-link robot arm*. Paper presented at the 5th workshop on Control Applications of Nonlinear Programming and Optimization, Capri, Italy.

Bulirsch R. (1983) *Die Mehrzielmethode zur Numerische Lösung von Nichtlinearen Randwertproblemen und Aufgaben der Optimalen Steuerung*. Syllabus of the course 'Optimierungsverfahren' of the Carl Cranz Geselschaft, Oberphaffenhofen FRG.

Dickmans E.D. and K.H. Well (1975) *Approximate solution of optimal control problems using third order Hermite polynomial functions*. Proc. IFIP-TC 7, VI Techn. Conf. on Optimization Techniques, held at Novosibirsk, 1974. Proceedings Springer New York.

Dunford N. and J.T. Schwarz (1958) *Linear Operators, Part I*. Interscience, New York.

Edge E.R. and W.F. Powers (1976) *Function-space quasi-Newton algorithms for optimal control problems with bounded controls and singular arcs*. Journal of Optimization Theory and Applications, Vol. 20, No. 4, p. 455-479.

Falb P.L. and M. Athans (1966) *Optimal Control, an Introduction to the Theory and Its Applications*. McGraw-Hill Book Company, New York.

Falb P.L. and de Jong J.L. (1969) *Some Successive Approximation Methods in Control and Oscillation Theory*. Academic Press, New York.

Fletcher R. (1981) *Practical Methods of Optimization, Vol. 2 Constrained Optimization*. John Wiley & Sons, New York.

Fletcher R. (1983) *Penalty functions*. Mathematical Programming, the State of the Art, eds. A. Bachem, M. Grötschel and B. Korte, p. 87-144, Springer-Verlag, New York.

Geerts A.H.W. (1985) *Optimality conditions for solutions of singular and state constrained optimal control problems in economics*. M.Sc. Thesis, Eindhoven University of Technology, The Netherlands.

Gill P.E., G.H. Golub, W. Murray and M.A. Saunders (1974a) *Methods for modifying matrix factorizations*. Mathematics of Computation, Vol. 28, No. 126, p. 505-535.

Gill P.E. and W. Murray (1974b) *Newton-type methods for linearly constrained optimization*. Numerical Methods for Constrained Optimization, eds. P.E. Gill and W. Murray, Academic Press, New York, p. 29-66.

Gill P.E., W. Murray, M.H. Wright (1981) *Practical Optimization*. Academic Press, New York.

Gill P.E., W. Murray, M.A. Saunders and M.H. Wright (1984) *Model building and practical implementation aspects in nonlinear programming*. Paper presented at the NATO Advanced Study Institute on "Computational Mathematical Programming", Bad Windsheim, FRG, july 23 - august 2, 1984.

Gillessen (1974) *Optimale Steuerung bei einer Beschränkung in Phasenraum und deren Numerische Berechnung unter Verwendung der Mehrzielmethode*. M.Sc. Thesis, Technische Universität Koln.

Girsanov I.V. (1972) *Lectures on Mathematical Theory of Extremum Problems*. Springer-Verlag, New York.

Golub G. and C. van Loan (1983) *Matrix Computations*. The Johns Hopkins University Press, Baltimore, Maryland, USA.

Gomez L.A. (1985) *Optimal control of a robot system*. Philips International Institute Report No. 1055.

Hamilton W.E. (1972) *On nonexistence of boundary arcs in control problems with bounded state variables*. IEEE Transactions on Automatic Control, AC-17, p. 338-343.

Han S.P. (1976) *Superlinearly convergent variable metric algorithms for general nonlinear programming problems*. Mathematical Programming 11, p. 263-282.

Han S.P. (1981) *Solving quadratic programs by an exact penalty function*. Nonlinear Programming 4, eds. Mangasarian et al., Academic Press, New York.

Hermes H. and J.P. LaSalle (1969) *Functional Analysis and Time Optimal Control*. Academic Press, New York.

Hestenes M.R. (1966) *Calculus of Variations and Optimal Control*. John Wiley & Sons, New York.

Hestenes M.R. (1975) *Optimization Theory, the Finite Dimensional Case*. John Wiley & Sons, New York.

Hiltmann P. (1983) *Numerische Behandlung Optimaler Steuerprozesse mit Zustandbeschränkungen mittels der Mehrzielmethode*. M.Sc. Thesis, Technische Universität Munchen, FRG.

Jacobson D.H. and M.M. Lele (1969) *A transformation technique for optimal control problems with a state variable constraint*. IEEE Trans. on Automatic Control, Vol.14/5.

Jacobson D.H., M.M. Lele and J.L. Speyer (1971) *New necessary conditions of optimality for control problems with state-variable inequality constraints.* Journal of Mathematical Analysis and Applications 35, p. 255-284.

Jong de J.L. (1985) *Instationary dolphin flight : the optimal energy exchange between a sailplane and vertical currents in the atmosphere.* Optimal Control Applications & Methods, Vol. 6, p. 113-124.

Jong de J.L. and K.C.P. Machielsen (1985) *On the application of sequential quadratic programming to state-constrained optimal control problems.* Paper presented at the Fifth IFAC workshop on Control Applications of Nonlinear Programming and Optimization, Capri, Italy.

Kantorovich L.V. and G.P. Akilov (1982) *Functional Analysis (second edition).* Pergamon Press, New York.

Keller H.B. (1969) *Accurate difference methods for linear ordinary differential systems subject to linear constraints.* SIAM Journal of Numerical Analysis, Vol. 6, No. 1.

Kirsch A., W. Warth and J. Werner (1978) *Notwendige Optimalitätsbedingungen und ihre Anwendung.* Lecture Notes in Economics and Mathematical Systems, No. 152, Springer Verlag, New York

Köhler M. (1980) *Pointwise maximum principle for convex optimal control problems with mixed control - phase variable inequality constraints.* Journal of Optimization Theory and Applications, Vol. 30, No. 2.

Kolmogorov A.N. and S.V. Fomin (1961) *Elements of the Theory of Functional Analysis, Vol. 1 : Metric and Normed Spaces, Vol. 2 : Measure, the Lebesgue Integral, Hilbert Space.* Graylock Press, Albany N.Y., USA.

Koster M.P. (1973) *Vibrations of CAM mechanisms and their consequences on the design.* Ph.D. Thesis, Eindhoven University of Technology, The Netherlands.

Kraft D. (1980) *Fortran-Programme zur Numerische Lösung Optimale Steuerungsprobleme.* DFVLR, Institut der Flugsysteme, Mitt. 80-03, Oberpfaffenhofen, FRG.

Kraft D. (1984) *On converting optimal control problems into nonlinear programming problems.* Paper presented at the NATO Advanced Study Institute on "Computational Mathematical Programming", Bad Windsheim, FRG, july 23 - august 2, 1984.

Kreindler E. (1982) *Additional necessary conditions for optimal control problems with state-variable inequality constraints.* Journal of Optimization Theory and Applications, Vol. 38, No. 2.

Kurcyusz S. (1976) *On the existence and nonexistence of Lagrange multipliers in Banach spaces.* Journal of Optimization Theory and Applications, Vol. 20, No. 1, p. 81-110.

Lawson C.L. and H.J. Hanson (1974) *Solving Least Squares Problems.* Englewood Cliffs, Prentice-Hall.

Loon van P. (1982) *A Dynamic Theory of the Firm : Production, Finance and Investment.* PhD Thesis, Tilburg University, The Netherlands.

Lorentz J. (1985) *Numerical solution of the minimum-time flight of a glider through a thermal by use of multiple shooting methods.* Optimal Control Applications & Methods, Vol. 6, p. 125-140.

References

Luenberger D.G. (1969) *Optimization by Vector Space Methods*. John Wiley & Sons, New York.

Machielsen K.C.P. (1983) *Some aspects of the dynamic behaviour of an assembly robot*. M.Sc. Thesis, Eindhoven University of Technology, The Netherlands.

Machielsen K.C.P. (1984) *Contour calculation and optimization procedure for 3D winding machine*. Philips CFT Technical Note, TN 046/84E.

Mangasarian O.L. (1969) *Nonlinear Programming*. McGraw-Hill, New York.

Maurer H. and U. Heidemann (1974) *Optimale Steuerprozesse mit Zustandsbeschränkungen*. Proceedings of the conference on Optimization and Optimal Control, Oberwolfach, Nov. 17-23, 1974. Lecture Notes in Mathematics 477, Springer Verlag, 1975.

Maurer H. and W. Gillessen (1975) *Application of multiple shooting to the numerical solution of optimal control problems with bounded state variables*. Computing 15, p. 105-126.

Maurer H. (1976) *Optimale Steuerprozesse mit Zustandsbeschränkungen*. Habilitationsschrift, Mathematisches Institut der Universität Wurzburg, FRG.

Maurer H. (1977) *On optimal control problems with bounded state variables and control appearing linearly*. SIAM Journal on Control and Optimization, Vol. 15, No. 3, p. 345-362.

Maurer H. (1979) *On the minimum principle for optimal control problems with state constraints*. Schriftenreihe des Rechenzentrums der Universität Münster, FRG, Report No. 41.

Maurer H. and Zowe J. (1979) *First and second order necessary and sufficient optimality conditions for infinite-dimensional programming problems*. Mathematical Programming 16, p. 98-110.

Maurer H. (1981) *First and second order necessary and sufficient optimality conditions in mathematical programming and optimal control*. Mathematical Programming Study 14, p. 163-177, North-Holland Publishing Company.

Miele A. (1968) *Method of particular solutions for linear two-point boundary value problems*. Journal of Optimization Theory and Applications, Vol. 2, No. 4

Miele A. (1975) *Recent advances in gradient algorithms for optimal control problems*. Journal of Optimization Theory and Applications, Vol. 17, No. 5/6

Miele A. and A.K. Wu (1980) *Sequential conjugate gradient-restoration algorithm for optimal control problems with nondifferential constraints and general boundary conditions. Part 1 : Theory*. Optimal Control Applications and Methods, Vol. 1, p. 69-88.

Miele A., Y.M. Kuo and E.M. Coker (1982) *Modified quasilinearization algorithm for optimal control problems with nondifferential constraints and general boundary conditions. Part 1 : Theory*. Aero-Astronautics Report No. 161, Rice University Houston, Texas, USA.

Nash S.G. (1983) *Truncated-Newton methods for large-scale function minimization*. Proceedings of the IFAC workshop on Applications of Nonlinear Programming to Optimization and Control, Palo Alto, California, USA.

Nash S.G. (1984) *Newton-type minimization via the Lanczos method*. SIAM Journal on Numerical Analysis, Vol. 21, No. 4, p. 770-788.

Nash S.G. (1985) *Preconditioning of truncated-Newton methods.* SIAM Journal on Scientific Statistical Computing, Vol. 6, No. 3, p. 599-616.

Neustadt L.W. (1969) *A general theory of extremals.* Journal of Computer and System Sciences, 3, p. 57-92.

Newman W.S. and N. Hogan (1986) *Time optimal control of balanced manipulators.* Paper to be published in the International Journal of Robotics Research.

Norris D.O. (1971) *A generalized Lagrange multiplier rule for equality constraints in normed linear spaces.* SIAM Journal of Control, Vol. 9, No. 4, p. 561-567.

Norris D.O. (1973) *Nonlinear programming applied to state constrained optimization problems.* Journal of Mathematical Analysis and Applications, Vol. 43, p. 261-272.

Oberle H.J. (1977) *Numerische Behandlung Singulärer Steuerungen mit der Mehrzielmethode am Beispiel der Klimatisierung von Sonnenhausern.* Dissertation, Institut für Mathematik, Technische Universität Munchen, FRG.

Oberle H.J. (1983) *Numerische Behandlung Optimaler Steuerungen von Heizung und Kühlung für ein Realistisches Sonnenhausmodel.* Habilitationsschrift, Institut für Mathematik, Technische Universität Munchen, FRG.

Paul R.P. (1981) *Robot Manipulators : mathematics, programming and control.* MIT-Press.

Pesch H.J. (1985) Habilitationsschrift, Technische Universität Munchen, FRG.

Pontryagin L.S., V.G. Boltyanskii, R.V. Gamkrelidze and E.F. Mishchenko (1962) *The Mathematical Theory of Optimal Processes.* John Wiley & Sons, New York.

Porter W.A. (1966) *Foundations of Systems Engineering.* The Macmillan Company, New York.

Pourciau B.H. (1983) *Multiplier rules and the separation of convex sets.* Journal of Optimization Theory and Applications, Vol. 40, No. 3, p. 321-331.

Pourciau B.H. (1980) *Modern multiplier rules.* American Mathematical Monthly, Vol. 87, p. 433-452.

Powell M.D.J. (1974) *Introduction to constrained optimization.* Numerical Methods for Constrained Optimization, eds. P.E. Gill and W. Murray, Academic Press, New York, p. 1-28.

Powell M.D.J. (1978) *A fast algorithm for nonlinearly constrained optimization calculations.* Numerical Analysis. Proceedings of the Biennial Conference held at Dundee, june 1977, ed. G.A. Watson, Lecture Notes in Mathematics, Vol. 630, Springer, New York.

Powell M.D.J. (1983) *Variable metric methods for constrained optimization.* Mathematical Programming, the State of the Art, eds. A. Bachem, M. Grötschel and B. Korte, p. 288-311, Springer-Verlag, New York.

Reid (1967) *A note on the least squares solution of a band system of linear equations by Householder reductions.* Computing Journal 10, p. 188-189.

Royden H.L. (1963) *Real Analysis.* The Macmillan Company, New York.

Rudin W. (1976) *Principles of Mathematical Analysis (third edition).* McGraw-Hill Kogakusha, Ltd. Tokyo.

Russak B. (1970a) *On problems with bounded state variables.* Journal of Optimization Theory and Applications, Vol. 5, No. 2.

Russak B. (1970b) *On general problems with bounded state variables.* Journal of Optimization Theory and Applications, Vol. 6, No. 6.

Schittkowski K. (1980) *Nonlinear Programming Codes. Information, Tests, Performance.* Lecture Notes in Economics and Mathematical Systems, No. 183, Springer-Verlag, New York.

Schittkowski K. (1981) *The nonlinear programming method of Wilson, Han and Powell with an augmented Lagrangian type line search function, Part 1 : Convergence Analysis.* Numer. Math. 38, p. 83-114.

Shin K.G. and N.D. McKay (1985) *Minimum-time control of a robotic manipulator with geometric path constraints.* IEEE Transactions on Automatic Control, AC-30(6).

Souren F.J.M. (1984) *On the numerical solution of optimal control problems; Modifications and test results of SGRA.* Philips CFT Technical Note 50/84E, The Netherlands.

Souren F.J.M. (1986) *Analytical and numerical aspects of sequential quadratic programming applied to state constrained optimal control problems.* M.Sc. Thesis, Eindhoven University of Technology, The Netherlands.

Stoer J. and R. Bulirsch (1980) *Introduction to Numerical Analysis.* Springer Verlag, New York.

Stoer J. (1984) *Foundations of recursive quadratic programming methods for solving nonlinear programs.* Paper presented at the NATO advanced study institute on "Computational Mathematical Programming", Bad Windsheim, FRG, july 23 - august 2, 1984.

Tapia R.A. (1974a) *A stable approach to Newton's method for general mathematical programming problems in* $\mathbb{R}^n$. SIAM Journal on Numerical Analysis, Vol. 11, No. 5.

Tapia R.A. (1974b) *Newton's method for optimization problems with equality constraints.* Journal of Optimization Theory and Applications, Vol. 14, No. 5.

Tapia R.A. (1977) *Diagonalized multiplier methods and quasi-Newton methods for constrained optimization.* Journal of Optimization Theory and Applications, Vol. 22, No. 2.

Tapia R.A. (1978) *Quasi-Newton methods for equality constrained optimization : equivalence of existing methods and a new implementation.* Nonlinear Programming 3, eds. O.L. Mangasarian, R.R. Meyer and S.M. Robinson, p. 125-164, Academic Press, New York.

Tewarson R.P. (1973) *Sparse Matrices.* Academic Press, New York.

Varaiya P.P. (1967) *Nonlinear programming in Banach space.* SIAM Journal of Applied Mathematics, Vol. 15, No. 2, p. 284-293.

Weiss R. (1974) *The application of implicit Runge-Kutta and collocation methods to boundary value problems.* Mathematics of Computation, Vol. 28, No. 126, p. 449-464.

Well K.H. (1983) *Übungen zu den Optimale Steuerungen.* Syllabus of the course 'Optimierungsverfahren' of the Carl Cranz Geselschaft, Oberphaffenhofen FRG.

Zowe J. (1978) *A remark on a regularity assumption for the mathematical programming problem in Banach spaces.* Journal of Optimization Theory and Applications, Vol. 25, No. 3, p. 375-381.

Zowe J. (1980) *The Slater condition in infinite-dimensional vector spaces.* American Mathematical Monthly, Vol. 87, p. 475-476.

Notations and symbols.

Throughout the thesis the following notations are used :

$\hat{}$	A variable with a hat ($\hat{}$) denotes either a solution of an optimization problem or a Lagrange multiplier corresponding to the solution of an optimization problem.
$\mathrm{N}(\tilde{h})$	Null space of the operator $\tilde{h}$.
$\mathrm{R}(\tilde{h})$	Range space of the operator $\tilde{h}$.
X^*	Dual space of the Banach space X.
x^*	Element of the dual space of the Banach space X.
$<x^*, x>$	Result of the linear functional $x^* \in X^*$ acting on $x \in X$.
$x^* x$	Same as $<x^*, x>$.
$\tilde{g}^{-1}(B)$	When $\tilde{g}$ is an operator and B a set, then $\tilde{g}^{-1}(B)$ denotes the set $\{x \in X : \tilde{g}(x) \in B\}$.
K^*	If K is a set, then K^* denotes the dual cone (cf. Definition 2.3).
S^*	If S is an operator then S^* denotes the adjoint operator (cf. Definition 2.4).
$\mathrm{S}(\tilde{u}, \epsilon)$	Neighborhood of the vector $\tilde{u}$.
$\delta J(u; \delta u)$	Fréchet differential of the operator J at u with variation δu.
$J'(u)$	Fréchet derivative of the operator J at u.
$J''(u)$	Second Fréchet derivative of the operator J at u.
$x(\cdot)$	$\bigcup_{t \in [0,T]} (x(t))$.
a.e.	almost everywhere.
ess sup	essential supremum.
$[t]$	Replaces argument lists with $\hat{x}(t)$, $\hat{u}(t)$, $\hat{\lambda}(t)$, etc. in Chapter 3 and argument lists with $x^i(t)$, $u^i(t)$, $\lambda^i(t)$, etc. in Chapters 4, 5 and 6.
$\bar{a} * M$	Denotes the tensor product of a vector $\bar{a}$ with a block matrix M.
f_x, f_u	Denote partial derivatives of the function $f(x, u, t)$ with respect to x and u.
A^{-1}	Inverse of matrix A.
A^+	Pseudo-inverse of matrix A.
A^T	Transpose of matrix A.
$\kappa(A)$	Condition number of matrix A, i.e. $\|A\| \cdot \|A^{-1}\|$. The 2-norm is used for matrix norms.

Spaces

$\boldsymbol{R}$	Space of real numbers.
$\boldsymbol{R}^n$	Euclidian space of n-vectors.
$C[0,T]$	Space of continuous functions on $[0,T]$.
$L_\infty[0,T]$	Space of measurable and essentially bounded function on $[0,T]$.
$L_\infty(W_l)$	Space of measurable and essentially bounded function on the closed set W_l.
$W_{1,\infty}[0,T]$	Space of absolutely continuous functions on $[0,T]$ with measurable and essentially bounded time derivatives.
$NBV[0,T]$	Normalized space of functions on $[0,T]$ of bounded variation.

Symbols used in terms of nonlinear programming in Banach spaces.
(Chapters 1, 2 and 4)

$A(M,\bar{u})$	Cone of admissible directions to M at $\bar{u}$, (cf. Definition 2.5).
$C(M,\bar{u})$	Conical hull of $M-\{\bar{u}\}$, (cf. Definition 2.7).
$K(K,l^*)$	Set of points for second order optimality conditions (cf. (2.3.10)).
$L(S,K,\bar{u})$	Linearizing cone of $S^{-1}(K)$ at $\bar{u}$, (cf. Definition 2.8).
$T(M,\bar{u})$	Sequential tangent cone of M at $\bar{u}$, (cf. Definition 2.6).
J	Objective functional of problems (P_0) and (P_1).
K	Cone defining constraints in problem (P_1).
L	Banach space used in the definition of problem (P_1).
l^*	Lagrange multiplier of problem (P_1).
$L(u,l^*)$	Lagrangian of problem (P_1).
M	Constraint set in problem (P_1).
S	Constraint operator in problem (P_1).
S_0	Constraint set in problem (P_0).
U	Banach space used in the definition of problems (P_0) and (P_1).
u	Variable in optimization problems (P_0) and (P_1).
A	Constraint set in problem (EIP).
B	Cone defining constraints in problem (EIP).
$\tilde{f}$	Objective functional of problem (EIP).
$\tilde{g}$	Inequality constraint operator of problem (EIP).
$\tilde{h}$	Equality constraint operator of problem (EIP).
X	Banach space used in the definition of problem (EIP).
x	Variable in optimization problem (EIP).
Y	Banach space used in the definition of problem (EIP).
Z	Banach space used in the definition of problem (EIP).
ρ	Regularity constant (cf. Theorem 2.10).
y^*	Lagrange multiplier of problem (EIP) (corresponding to $\tilde{g}$).
z^*	Lagrange multiplier of problem (EIP) (corresponding to $\tilde{h}$).
$L(x,y^*,z^*)$	Lagrangian of problem (EIP).
$M\{\alpha\}$	Merit function dependent of step size (cf. Section 4.1.2).
G	Mapping used to imitate an inner product in Banach space.
x_i	Current estimate for the solution in Algorithm 4.1.
y_i^*	Current estimate for Lagrange multiplier $\hat{y}^*$ in Algorithm 4.1.
z_i^*	Current estimate for Lagrange multiplier $\hat{z}^*$ in Algorithm 4.1.

Symbols used in terms of optimal control.
(Chapters 1, 3, 4 and 5)

t	Time variable.
T	Final time.
$x(t)$	State variable.
$u(t)$	Control variable.
$f_0(x,u,t)$	Problem function of problem (SCOCP) (objective function).
$g_0(x,T)$	Problem function of problem (SCOCP) (objective function).
$h_0(x)$	Problem function of problem (SCOCP) (objective function).
$f(x,u,t)$	Problem function of problem (SCOCP) (differential system).
$D(x)$	Problem function of problem (SCOCP) (initial point constraints).
$E(x,T)$	Problem function of problem (SCOCP) (terminal point constraints).
$S_1(x,u,t)$	Problem function of problem (SCOCP) (mixed control state constraints).
$S_2(x,t)$	Problem function of problem (SCOCP) (state constraints).
U	Constraint set in problem (SCOCP) (control constraints).
n	Dimension of state vector x.
m	Dimension of control vector u.
c	Dimension of vector function D.
q	Dimension of vector function E.
k_1	Dimension of vector function S_1.
k_2	Dimension of vector function S_2.
λ	Lagrange multipliers corresponding to the differential system also called adjoint variable.
σ	Lagrange multipliers corresponding to the initial point constraints D.
μ	Lagrange multipliers corresponding to the terminal point constraints E.
η_1	Lagrange multipliers corresponding to the mixed control state constraints S_1.
ξ	Lagrange multipliers corresponding to state constraints S_2.
η_2	Time derivative of the Lagrange multiplier ξ.
ν_j	Discontinuity of the Lagrange multiplier ξ at time point t_j.
$H(x,u,\rho,\lambda,t)$	Hamiltonian (cf. (3.3.3.1)).
S_{2i}^j	Functions defined by (3.3.5.7) - (3.3.5.8), that have the interpretation of time derivatives of the state constraint S_{2i}.
p_i	Order of the state constraint S_{2i} (cf. (3.3.5.9)).
$\tilde{S}$	Vector function of state constraints (cf. (3.3.5.10)).
$\tilde{S}^p$	Vector function of mixed control state constraints (cf. (3.3.5.11)).
$\bar{H}^i(x,u,\hat{\rho},\hat{\lambda}^i,\hat{\eta}^i,t)$	Augmented Hamiltonian (cf. (3.3.6.1)).
$\hat{\lambda}^i$	Adjoint variable in alternative formulation of optimality conditions.
$\hat{\eta}^i$	Multiplier in alternative formulation of optimality conditions.
$\hat{\beta}^i$	Multiplier in alternative formulation of optimality conditions.

$x^i(t)$	Current estimate for the state variable in Algorithm 4.4.
$u^i(t)$	Current estimate for the control variable in Algorithm 4.4.
$\lambda^i(t)$	Current estimate for the adjoint variable in Algorithm 4.4.
$\eta_1^i(t)$	Current estimate for the multiplier η_1 in Algorithm 4.4.
$\xi^i(t)$	Current estimate for the multiplier ξ in Algorithm 4.4.
$\eta_2^i(t)$	Current estimate for the multiplier η_2 in Algorithm 4.4.
ν_j^i	Current estimate for the multiplier ν_j in Algorithm 4.4.
σ^i	Current estimate for the multiplier σ in Algorithm 4.4.
μ^i	Current estimate for the multiplier μ in Algorithm 4.4.
M_1	Matrix in definition of subproblems (cf. (4.2.1.11)).
M_2	Matrix in definition of subproblems (cf. (4.2.1.12)).
M_3	Matrix in definition of subproblems (cf. (4.2.1.13)).
M_4	Matrix in definition of subproblems (cf. (4.2.1.14)).
M_5	Matrix in definition of subproblems (cf. (4.2.1.15)).
M_6	Matrix in definition of subproblems (cf. (4.2.1.16)).
W_l	Working set of state constraint $\tilde{S}_l$.
$R[t]$	Vector function of state equality constraints (cf. (4.2.1.19)).
$R^p[t]$	Vector function of mixed control state equality constraints (cf. (5.1.2.14)).
$I(t)$	Index set of active constraints at time point t.
$\bar{k}(t)$	Number of constraints in the set $I(t)$.
m_l^b	Number of boundary intervals of working set W_l.
m_l^c	Number of contact points of working set W_l.
$[t_{2j-1}^l, t_{2j}^l]$	j-th boundary interval in working set W_l.
$t_{2m_l^b+j}^l$	j-th contact point in working set W_l.
Δ^1	Grid for the junction and contact points of the mixed control state constraints (cf. (4.2.2.1)).
Δ^2	Grid for the junction and contact points of the state constraints (cf. (4.2.2.1)).
Δ	$\Delta^1 \times \Delta^2$.
$\bar{p}_j$	Number of points of the grid Δ^j.
t_i^j	Time point i of grid Δ^j.
J_B^{1l}	Set of boundary points of constraint S_{1l} (cf. Definition 4.3).
J_B^{2l}	Set of boundary points of constraint S_{2l} (cf. Definition 4.3).
$M(\cdots)$	Merit function (cf. (4.3.8)).
$\bar{\eta}_{0l}(t)$	Multiplier for active set strategy (cf. (5.2.23)).
$\bar{\nu}_{l1}^k$	Multiplier for active set strategy (cf. (5.2.26)).
$\bar{\nu}_{l2}^k$	Multiplier for active set strategy (cf. (5.2.27)).

Symbols used in the numerical implementation.
(Chapter 6)

l	Order of polynomials on grid intervals.
p	Number of grid intervals.
ρ_i	Collocation points relative to the interval [0.1].
t_r	Grid point ($r = 0,1,\ldots,p$).
τ_{lr+i}	Collocation point i on the interval $[t_r, t_{r+1}]$.
h_r	Size of grid interval.
ω_{jk}	Weight in quadrature formula (6.1.1.22).
$\bar{\omega}_k$	Weight in quadrature formula (6.1.1.24).
$\tilde{t}_j$	Junction or contact point.
χ	Lagrange multiplier associated with interior point constraints (6.1.2.8).
η_I	Lagrange multiplier associated with mixed control state constraints (4.2.1.25).
d_x^r	Numerical approximation to $d_x(t_r)$.
$d_x^{r,i}$	Numerical approximation to $d_x(\tau_{lr+i})$.
$d_u^{r,i}$	Numerical approximation to $d_u(\tau_{lr+i})$.
$\lambda_x^{r,+}$	Numerical approximation to $\lambda_x(t_r+)$.
$\lambda_x^{r,-}$	Numerical approximation to $\lambda_x(t_r-)$.
$\lambda_x^{r,i}$	Numerical approximation to $\lambda_x(\tau_{lr+i})$.
$\lambda_x^{0,+}$	Numerical approximation to $\lambda_x(0)$.
$\lambda_x^{p,-}$	Numerical approximation to $\lambda_x(T)$.
$\eta_I^{r,i}$	Numerical approximation to $\eta_I(\tau_{lr+i})$.
$\zeta_{r,k}$	Transformed adjoint variable (cf. (6.1.2.26)).
θ_{lr+i}	Transformed multiplier $\eta_I^{r,i}$ (cf. (6.1.2.27)).
M	Matrix in objective function of quadratic programming problem (cf. (6.1.2.34)).
C	Matrix of constraint normals in quadratic programming problem (cf. (6.1.2.35)).
c	Vector in objective function of quadratic programming problem (cf. (6.1.2.34)).
d	Variable in quadratic programming problem.
b	Inhomogeneous part of constraints (cf. (6.1.2.35)).
ζ	Lagrange multiplier of quadratic programming problem (6.1.2.34) - (6.1.2.35).
$\bar{n}$	Dimension of vector d.
$\bar{m}$	Number of constraints, i.e. row dimension of the matrix C.
d_R	Range space part of vector d, i.e. $Cd_R = b$.
d_N	Null space part of vector d, i.e. $Cd_N = 0$.
Y	$\bar{n} \times \bar{m}$ matrix whose columns are a base for the range space of the matrix C^T.
Z	$\bar{n} \times (\bar{n} - \bar{m})$ matrix whose columns are a base for the null space of the matrix C.
L	Lower-triangular matrix in LQ-factorization of the matrix C.
Q	Orthogonal matrix in LQ-factorization of the matrix C.
$I_{\bar{m}}$	$\bar{m} \times \bar{m}$ identy matrix.
e_j	j-th columns of the identy matrix.
D_1, D_2	Scaling matrices.

Samenvatting

Het doel van dit proefschrift is een beschrijving te geven van een nieuwe methode voor het numeriek oplossen van optimale besturingsproblemen met toestandsbeperkingen.

Allereerst worden de optimaliseringsproblemen geintroduceerd en beschouwd in een abstrakte formulering. Het voordeel van zo'n abstrakte benadering is dat voorwaarden voor optimaliteit, die voor oplossingen van de optimaliseringsproblemen moeten gelden, afgeleid kunnen worden voor de abstrakte formulering. Dit houdt in dat men zich in eerste instantie niet hoeft te bekommeren om de details van de probleem specificatie. Voor de abstrakt geformuleerde optimaliseringsproblemen worden een aantal min of meer standaard resultaten uit de literatuur herhaald.

Omdat toestandsbeperkte optimale besturingsproblemen geidentificeerd kunnen worden als speciale gevallen van de abstrakte optimaliseringsproblemen, kunnen de optimaliteitsvoorwaarden voor de abstrakte problemen direct hierop worden toegepast. In de formulering van de optimale besturingsproblemen gaan de optimaliteitsvoorwaarden voor de abstrakte problemen over in het bekende minimum principe.

De methode die wordt voorgesteld voor de numerieke oplossing van de optimale besturingsproblemen, wordt eerst gepresenteerd in een abstrakte formulering. De methode is een analogie met de methode van het sequentieel kwadratisch programmeren, hetgeen een bekende methode is voor het oplossen van eindig dimensionale niet-lineaire programmeringsproblemen. Dit houdt in dat de methode een iteratieve 'descent' methode is, waarbij de zoekrichting bepaald wordt door het oplossen van een subprobleem met een kwadratische objektfunktie en lineaire beperkingen. Een stapgrootte wordt bepaald door het minimaliseren van een exakte penalty funktie. De toepassing van de (abstrakte) methode voor toestandsbeperkte optimale besturingsproblemen wordt gecompliceerd door het feit dat de subproblemen niet eenvoudig opgelost kunnen worden, als de struktuur van de oplossing niet bekend is. Daarom is een modificatie van de subproblemen noodzakelijk. Als gevolg van deze modificatie zal de methode, in het algemeen, niet convergeren naar een oplossing van het besturingsprobleem, maar naar een punt dichtbij een oplossing. Daarom is een tweede stap nodig die, uitgaande van de struktuur bepaald in de eerst stap, de oplossing exakt bepaald.

De numerieke implementatie van de methode komt in essentie neer op het numeriek oplossen van een lineair meerpunts randwaarde probleem. Hiervoor zijn in principe verschillende methoden geschikt, echter de collocatie methode die hier gekozen is heeft enige belangrijke voordelen ten opzichte van andere mogelijke methoden. Het stelsel van lineaire vergelijkingen dat resulteert uit de collocatie methode kan efficient opgelost worden met behulp van sparse matrix technieken.

Enige numerieke resultaten van het programma voor enkele praktische problemen zijn samengevat. Omdat numerieke resultaten voor twee van deze problemen tevens in de literatuur vermeld zijn, is een vergelijking met andere methoden mogelijk.

Uiteindelijk wordt de relatie gegeven tussen de voorgestelde methode en enige uit de literatuur bekende methoden.

Curriculum vitae

De schrijver van dit proefschrift werd op 23 mei 1956 te Vlaardingen geboren. Van 1968 tot 1973 volgde hij M.A.V.O. te Den Haag. Daarna bezocht hij tot 1978 de R.K. H.T.S. Rijswijk, alwaar hij onderwijs volgde aan de afdeling der elektrotechniek.

Na het behalen van het HTS diploma trad hij in dienst bij de Nederlandse Philips bedrijven B.V. te Eindhoven als ontwerper bij de elektrische bedrijfsmechanisatie professionele componenten en materialen van de produkt divisie Elcoma. Daarnaast begon hij in dat zelfde jaar met de studie voor elektrotechnisch ingenieur aan de technische universiteit te Eindhoven. In het kader van deze studie verrichtte hij, onder verantwoording van Prof. Dr. Ir. P. Eijkhoff, van mei 1982 tot mei 1983 een afstudeeronderzoek op het Natuurkundig Laboratorium van Philips in de groep measurement and control, onder leiding van Ir. A.F. Verkruissen.

Na het behalen van de graad van elektrotechnisch ingenieur (met lof), trad hij op 1 juni 1983, in dienst van het Centrum voor Fabricage Technieken, als wetenschappelijk medewerker in de groep machine and process control van het vakgebied signal processing, hetgeen een onderdeel is van het CAM centre. Gedurende de afgelopen jaren heeft hij zich in het kader van zijn werk bezig gehouden met optimale baansturing van vrij programmeerbare mechanismen. De door hem ontwikkelde methode voor het numeriek oplossen van optimale besturingsproblemen met toestandsbeperkingen is het onderwerp van dit proefschrift. Sedert 1 september 1986 is hij groepleider van de groep machine and process control.

Sinds november 1984 is hij tevens verbonden als docent aan de avond H.T.S. van het I.H.B.O. te Eindhoven, alwaar hij onderwijs geeft in de meet - en regeltechniek aan de eindexamen klas van de onderafdeling werktuigbouwkunde.

STELLINGEN

I

De door Craven afgeleide noodzakelijke voorwaarden voor oplossingen van optimale besturingsproblemen met toestandsbeperkingen zijn niet correct, als gevolg van een foutief aangenomen representatie van de Lagrange multiplicatoren die geassocieerd zijn met de toestandsbeperkingen.

Craven B.D. (1978) *Mathematical Programming and Control Theory*. Chapman and Hall mathematics series, London.

II

De numerieke testvoorbeelden gebruikt door Miele en Wu zijn zinvol om de correctheid te testen van een implementatie van een methode voor het numeriek oplossen van een optimaal besturingsprobleem. Daarentegen zijn ze te eenvoudig om een uitspraak te rechtvaardigen met betrekking tot de geschiktheid van de methode voor het oplossen van algemenere optimale besturingsproblemen.

Miele A. and A.K. Wu (1980) *Sequential conjugate gradient-restoration algorithm for optimal control problems with nondifferential constraints and general boundary conditions. Part 2 : examples*. Optimal Control Applications and Methods, Vol. 1.

III

De noodzakelijke voorwaarden voor optimaliteit van Russak zijn in wezen een andere formulering van de noodzakelijke voorwaarden voor optimaliteit van Jacobson, Lele en Speyer. De voorwaarden van Russak zijn daarom niet superieur, zoals opgemerkt wordt door van Loon.

Jacobson D.H., M.M. Lele, and J.L. Speyer (1971) *New necessary conditions of optimality for control problems with state-variable inequality constraints*. Journal of Mathematical Analysis and Applications 35, p. 255-284.

Loon P. van (1982) *A dynamic theory of the firm : production, finance and investment*. PhD thesis, Tilburg University, The Netherlands, p. 142.

Russak B. (1970) *On general problems with bounded state variables*. Journal of Optimization Theory and Applications, Vol. 6, No. 6.

IV

Uit het bewijs van stelling 4.2. deel b van Schittkowski, blijkt dat de 'augmented Lagrangian' ook gebruikt kan worden als 'merit' funktie in een SQP-methode, voor het oplossen van niet-lineaire programmeringsproblemen met ongelijkheidsbeperkingen, waarbij de zoekrichting gevonden wordt als de oplossing van een kwadratisch programmeringsprobleem met alleen gelijkheidsbeperkingen.

Schittkowski K. (1981) *The nonlinear programming method of Wilson, Han and Powell with an augmented Lagrangian type line search function. Part 1 : convergence analysis*. Numer. Math. 38, p. 83-114.

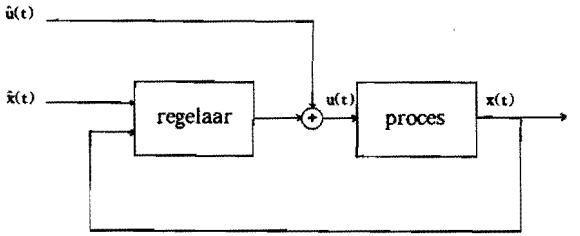
V

De conditie die door Kurcyusz de "Kuhn-Tucker" conditie wordt genoemd is in feite de 'constraint qualification' van Abadie (zie Bazaraa e.a. (1976)). Aangezien de Abadie en de Kuhn-Tucker 'constraint qualifications' niet equivalent zijn, is de door Kurcyusz gebruikte benaming verwarrend.

Bazaraa M.S. and C.M. Shetty (1976) *Foundations of Optimization*. Springer-Verlag, New York.

Kurcuysz S. (1976) *On the existence and nonexistence of Lagrange multipliers in Banach spaces*. Journal of Optimization Theory and Applications, Vol. 20, No. 1, p. 81-110.

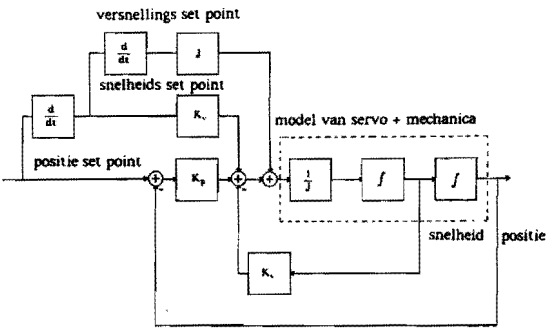
Voor de praktische toepassing van de theorie over de optimale 'open loop' besturing van dynamische systemen op geregelde processen, is het van belang dat ingezien wordt dat de besturing van een geregeld proces in veel gevallen gesplitst kan worden in een 'open loop' besturingsprobleem en een (quasi-)stationair regelprobleem.



figuur 1

Een en ander is weergegeven in bovenstaande figuur. De besturing $\hat{u}$ en de daarbij behorende toestand $\hat{x}$ worden bepaald als de oplossing van een 'open loop' optimaal besturingsprobleem, d.w.z. de terugkoppeling wordt genegeerd. Vervolgens wordt de regelaar ontworpen op basis van een lineair model van het proces, dat verkregen wordt door lineariseren van het proces model langs de trajectorie $(\hat{x}(t), \hat{u}(t))$.

Wanneer bij de besturing van een electro-mechanisch servosysteem uitgegaan wordt van het tweede-orde-model zoals beschreven door Bouwens, dan kan een aan figuur 1 equivalent besturingsschema verkregen worden door voorwaartse koppeling van de gewenste snelheid en versnelling.



figuur 2

Bij de implementatie van bovenstaand schema kan de differentiatie van de set-point-functie $\hat{y}(t)$ veelal analytisch geschieden, omdat $\hat{y}(t)$ in de meeste gevallen een vooraf bekende analytische struktuur heeft.

Bouwens H.B. (1984) *Servo design procedure*. Philips CFT report 15/83.

VII

Bij de optimale besturing van een digitaal geregeld proces is het vaak zinvol om de optimale besturing te berekenen op basis van een tijdcontinu proces-model. Immers de keuze van de bemonstertijd voor de (tijddiscrete) regeling van het proces geschiedt op basis van de gewenste eigenschappen van het geregelde proces (bemonstertijd meestal zo klein mogelijk), terwijl de keuze van de tijddiscretisatie bij het uitrekenen van de optimale besturing geschiedt op basis van numerieke argumenten (integratiestap meestal zo groot mogelijk en eventueel variabel).

VIII

De veronderstelling van een tweede-orde-model voor het dynamisch gedrag van een electro-mechanisch servo-systeem correspondeert bij een electrisch aangedreven robot met de veronderstelling van starre lichamen voor de armdelen en oneindig stijve transmissies tussen armdelen en aandrijvingen. Voor een goede regeling van een robot is het in het algemeen noodzakelijk de interactie tussen de vrijheidsgraden te compenseren volgens het zogenaamde 'inverse plant' principe.

Machielsen K.C.P. (1983) *Some aspects of the dynamic behaviour of an assembly robot.* M.Sc. thesis. Eindhoven University of Technology. The Netherlands.

K.C.P. Machielsen, 31 maart 1987.