2002 IEEE International Symposium on Computer
Aided Control System Design Proceedings
September 18-20, 2002 • Glasgow, Scotland, U.K.

Designing model predictive controllers with prioritised constraints and objectives

Eric C. Kerrigan and Jan M. Maciejowski

Control Group, Department of Engineering, University of Cambridge
Trumpington Street, Cambridge CB2 1PZ, United Kingdom
Tel: +44-1223-339706/2732, Fax: +44-1223-332662
eck21@eng.cam.ac.uk, jmm@eng.cam.ac.uk
http://www-control.eng.cam.ac.uk/

Abstract

This paper shows how a class of objective functions can be incorporated into a prioritised, multi-objective optimisation problem, for which a solution can be obtained by solving a sequence of single-objective, constrained, convex programming problems. The objective functions considered in this paper typically arise in Model Predictive Control (MPC) of constrained, linear systems. The framework presented in this paper can be used to design a flexible, multi-objective MPC controller that takes priorities into account during the on-line computation of the control input.

Keywords: Multi-objective optimization, receding horizon control, lexicographic programming, prioritized objectives, constraint handling, convex programming.

1 Introduction

When designing controllers, it is very difficult to express all the objectives as a single cost function. Assigning a high weight to the control of an output is not really the same as assigning it a high priority. For example, by using a single cost one cannot express objectives such as "keep output 1 near its set-point only if output 2 can be kept at its set-point". Also, sometimes a disturbance or fault occurs that makes it impossible to satisfy all control objectives simultaneously and an arbitrary trade-off is not desirable, since some control actions are preferable to others. As a result, the control objectives have to be changed to accommodate this new information. For example, it might not be possible to keep both output 1 and output 2 within given constraints. A control action that guarantees that the constraints on output 1 will be satisfied might be preferred over a control action that satisfies the constraints on output 2. A multi-objective optimisation formulation makes it possible to resolve conflicting control objectives in a systematic fashion [4, Sects. 10.1–10.2]. The paper aims to describe a multi-objective framework that gives a means for optimally handling such a problem.

This paper will consider the case where a number of objectives with various priorities are given. The multi-objective problem is solved via a hierarchy of single-objective optimisation problems; the most important optimisation problem is solved first and the solution to this problem is then used to impose additional constraints on the second optimisation, etc. Section 2 gives a brief description of this procedure, which is well-known within the academic [5, 7, 8, 10] and industrial [4, App. A] control community.

Section 3 defines the type of objective functions that will be considered in this paper. The objectives considered typically arise in the control of constrained, linear systems and most of the objectives defined in [2, 3, 5, 7, 8, 9, 10] can be defined in terms of the functions given in this section. As such, this paper can be interpreted as a slight generalisation of these previous approaches to prioritised, multi-objective controller design.

Section 4 shows, for each type of objective function considered in Section 3, how each single-valued optimisation problem needs to be constrained in order to guarantee that the solution of the final single-valued optimisation problem is optimal with respect to the original, prioritised, multi-objective optimisation problem. One of the points made in Section 4 is that, provided certain convexity conditions are satisfied, the multi-objective optimisation problem can be solved using convex programming techniques. Finally, Section 5 shows, via an example, how the multi-objective framework discussed in the previous sections can be used to design a prioritised, model predictive control (MPC) scheme.

2 Prioritised, Multi-objective Optimisation

Consider the multi-objective optimisation problem

$$\mathcal{P}: \quad \min_{\theta \in \Theta} f(\theta), \tag{1}$$

where the admissible set of decision variables $\Theta \subseteq \mathbb{R}^d$, the vector-valued objective function $f : \Theta \to \mathbb{R}^p$,

$$f(\theta) := [f_1(\theta), f_2(\theta), \dots, f_p(\theta)]^T \tag{2}$$

and the scalar-valued objective functions $f_i : \Theta \to \mathbb{R}$ for all $i \in \{1, \dots, p\}$. It is assumed that all the $f_i(\cdot)$ attain their minima inside $\Theta \neq \emptyset$. The rest of this section is concerned with reviewing different definitions for what is meant by a minimiser of problem $\mathcal{P}$. An initial, naive guess at a definition is the following:

Definition 1 *A given* $\theta^* \in \Theta$ *is a* minimiser *and* $f(\theta^*)$ *is the* minimum *of problem* $\mathcal{P}$ *iff* $f(\theta^*) \leq f(\theta)$ *for all* $\theta \in \Theta$.

Note that if a minimiser exists, then the minimum is guaranteed to be unique. Unfortunately, in contrast to optimisation problems with a single objective function, a minimiser to a multi-objective optimisation problem (as defined above) is not guaranteed to exist, hence one is forced to compromise. A commonly-used alternative definition for a minimiser is the following:

Definition 2 *A given* $\theta^* \in \Theta$ *is an* efficient *or* Pareto-optimal *minimiser and* $f(\theta^*)$ *is a* Pareto-optimal *minimum of* $\mathcal{P}$ *iff there does not exist a* $\theta \in \Theta$ *and an* i *such that* $f(\theta) \leq f(\theta^*)$ *and* $f_i(\theta) < f_i(\theta^*)$.

In other words, a minimiser is Pareto-optimal if and only if an objective f_i can be reduced only at the expense of increasing at least one of the other objectives.

Under the assumptions above, a Pareto-optimal minimiser is guaranteed to exist. However, the minimiser and minimum are not necessarily unique. As a consequence, unless a suitably-defined *order* [6, Chap. 1] over the decision variables or objective functions is given, it is not easy to determine which Pareto-optimal solutions are preferred over others. Fortunately, in many applications it is more important to minimise certain objectives and a hierarchy of the objectives can be formulated. This hierarchy defines an order on the objective functions and makes it possible to give a definition for a minimum of problem $\mathcal{P}$ that results in less ambiguity than the standard Pareto-optimal definition given above.

As such, it is assumed that the given objective functions can be ranked according to a hierarchy of p distinct priority levels such that the minimisation of $f_1(\theta)$ assumes the highest priority and the minimisation of $f_2(\theta)$ the second highest priority, etc. The problem now becomes that of finding the set of minimisers of problem $\mathcal{P}$ that take into account the relative importance of the individual objective functions. The following definition, adapted from [1, Sect. 3.3.1] and [6, Sect. 2.3.2], will be used to define such a minimiser:

Definition 3 *A given* $\theta^* \in \Theta$ *is a* lexicographic minimiser *and* $f^* := f(\theta^*)$ *is the* lexicographic minimum *of problem* $\mathcal{P}$ *iff there does not exist a* $\theta \in \Theta$ *and an* $i^* := \min_i\{i \mid f_i(\theta) \neq f_i(\theta^*)\}$ *such that* $f_{i^*}(\theta) < f_{i^*}(\theta^*)$.

An interpretation of the above definition is that a minimiser is a lexicographic minimiser if and only if an objective f_i can be reduced only at the expense of increasing at least one of the higher-prioritised objectives $\{f_1(\cdot), \ldots, f_{i-1}(\cdot)\}$. Hence, a lexicographic minimiser is a special type of Pareto-optimal minimiser that takes into account the order of the objectives.

Fact 1 *A lexicographic minimiser exists and the lexicographic minimum of problem* $\mathcal{P}$ *is unique.*

The above observation, especially the fact that the lexicographic minimum is unique, helps to make it less ambiguous in determining whether a given $\theta \in \Theta$ is optimal in some sense, than if the original Pareto-optimal definition was used.

A standard method for finding a lexicographic minimiser of problem $\mathcal{P}$ is to solve a hierarchical sequence of single-objective, constrained optimisation problems:

Fact 2 $f^* := [f_1^*, f_2^*, \ldots, f_p^*]^T$ *is the lexicographic minimum of problem* $\mathcal{P}$ *if and only if*

$$f_1^* = \min_{\theta \in \Theta} f_1(\theta) \tag{3}$$

and for all $i \in \{2, \ldots, p\}$,

$$f_i^* = \min_{\theta \in \Theta} \{f_i(\theta) \,|\, f_j(\theta) \leq f_j^*, j = 1, \ldots, i-1\}. \tag{4}$$

θ^* *is a lexicographic minimiser of problem* $\mathcal{P}$ *if and only if*

$$\theta^* \in \{\theta \in \Theta \,|\, f_j(\theta) \leq f_j^*, j = 1, \ldots, p\}. \tag{5}$$

Remark 1 *Note that* (5) *is equivalent to*

$$\theta^* \in \arg\min_{\theta \in \Theta} \{f_p(\theta) \,|\, f_j(\theta) \leq f_j^*, j = 1, \ldots, p-1\}. \tag{6}$$

Remark 2 *To improve numerical conditioning, the constraints in* (4) *are often relaxed a little, e.g.* (4) *can be replaced by*

$$f_i^* = \min_{\theta \in \Theta} \{f_i(\theta) \,|\, f_j(\theta) \leq f_j^* + \varepsilon, j = 1, \ldots, i-1\}, \tag{7}$$

where $\varepsilon > 0$ *is a small tolerance.*

As can be seen in (3) and (4), the most important objectives are minimised before continuing to minimise the lower-prioritised objectives. The constraints in (4) ensure that the higher-prioritised objectives are equal to their optimal values.

As mentioned before, a lexicographic minimiser of problem $\mathcal{P}$ is not guaranteed to be unique. However, one can derive a sufficient condition for guaranteeing that a lexicographic minimiser of problem $\mathcal{P}$ is unique:

Fact 3 *If* $f_i(\cdot)$ *is strictly convex, then the lexicographic minimiser* θ^* *of problem* $\mathcal{P}$ *is unique. If* f_1 *is strictly convex, then*

$$\theta^* = \arg\min_{\theta \in \Theta} f_1(\theta). \tag{8}$$

If $f_i(\cdot)$ *is strictly convex and* $i \in \{2, \ldots, p\}$, *then*

$$\theta^* = \arg\min_{\theta \in \Theta} \{f_i(\theta) \,|\, f_j(\theta) \leq f_j^*, j = 1, \ldots, i-1\}. \tag{9}$$

This result implies that if the i'th objective function is strictly convex, then there is no point in solving (4) for $\{f_{i+1}^*, \ldots, f_p^*\}$, since the unique lexicographic minimiser is given by (9). In order not to waste computational effort, it is therefore sensible to ensure that the problem $\mathcal{P}$ has been defined such that the only strictly convex function is $f_p(\cdot)$.

3 Objective Functions to be Considered

Many objectives that typically arise in the optimal control of constrained systems can be described in terms of a prioritised, multi-objective optimisation problem, such as problem $\mathcal{P}$. This section suggests a few types of functions that can be used to define an appropriate set of objective functions $\{f_1(\cdot), \ldots, f_p(\cdot)\}$. The functions that will be considered are based on control objectives that typically arise in the design of model predictive controllers for constrained, linear systems [4]. It is possible, however, to extend the discussion in this paper to a number of functions not considered here.

Let H and R be positive, semi-definite matrices and h and r be vectors of suitable dimensions. Let $g : \Theta \to \mathbb{R}^q$ and $g(\theta) := [g_1(\theta), g_2(\theta), \ldots, g_q(\theta)]^T$, where each $g_i : \Theta \to \mathbb{R}$. The function $g(\cdot)$ defines a set of constraints on θ in the sense that it is desired that θ satisfy $g(\theta) \leq 0$. Also, let

$$g_i^+(\theta) := \max\{0, g_i(\theta)\} \qquad (10)$$

represent the amount of violation of the i'th constraint and $g^+(\theta) := [g_1^+(\theta), \ldots, g_q^+(\theta)]^T$.

The function $V(\cdot)$ is defined below in terms of weights on the decision variable θ. The functions $L(\cdot)$, $S(\cdot)$ and $D(\cdot)$ are defined below in terms of the constraint violations $g^+(\theta)$.

Standard cost function: Let

$$V(\theta|H,h) := \theta^T H\theta + h^T\theta. \qquad (11)$$

Equivalently, one could have defined

$$V(\theta|H,h) := \|H^{\frac{1}{2}}\theta\|_2^2 + \|\text{diag}(h)\theta\|_1. \qquad (12)$$

Note that $V(\cdot|H,h)$ is quadratic and convex if $H \neq 0$ and strictly convex if H is positive definite. The cost function is linear and convex (but not strictly convex) if $H = 0$.

Size of largest constraint violation: Let

$$L(\theta|g) := \max\{0, g_1(\theta), g_2(\theta), \ldots, g_q(\theta)\}. \qquad (13)$$

Note that $L(\theta|g) = \|g^+(\theta)\|_\infty$ and $g(\theta) \leq 0 \Leftrightarrow L(\theta|g) = 0$.

Weighted sum of constraint violations: Let

$$S(\theta|g,R,r) := g^+(\theta)^T Rg^+(\theta) + r^T g^+(\theta). \qquad (14)$$

Equivalently, one could have defined

$$S(\theta|g,R,r) := \|R^{\frac{1}{2}} g^+(\theta)\|_2^2 + \|\text{diag}(r)g^+(\theta)\|_1. \qquad (15)$$

If R is positive definite or $r > 0$, then $g(\theta) \leq 0 \Leftrightarrow S(\theta|g,R,r) = 0$. This equivalence is not guaranteed if R is not positive definite and r is not strictly positive.

Remark 3 *Note that, in general, $S(\cdot|g,R,r)$ and $L(\cdot|g)$ are not strictly convex.*

Largest element in index set of violated constraints: Let

$$D(\theta|g) := \begin{cases} 0 & \text{if } g(\theta) \leq 0 \\ \max_i\{i|g_i(\theta) > 0\} & \text{otherwise} \end{cases} \qquad (16)$$

An application of this function is as follows: Suppose that $g_1(\cdot)$ defines a constraint which applies at step 1 of a prediction horizon (in MPC, say), $g_2(\cdot)$ applies at step 2, ..., and $g_q(\cdot)$ applies at the last step q. Then $D(\theta|g)$ indicates the duration, into the prediction horizon, of constraint relaxations. The minimisation of $D(\theta|g)$ can be interpreted as solving a separate, prioritised, multi-objective problem where the satisfaction of $g_{i+1}(\cdot)$ assumes a higher priority than the satisfaction of $g_i(\cdot)$; $g_i(\cdot)$ will not be satisfied until $g_{i+1}(\cdot)$ has been satisfied.

Remark 4 *A function related to $D(\cdot)$ is*

$$E(\theta|g) := \begin{cases} 0 & \text{if } g(\theta) \leq 0 \\ 1+q-\min_i\{i|g_i(\theta) > 0\} & \text{otherwise} \end{cases} \qquad (17)$$

Recalling the time horizon interpretation, $E(\theta|g)$ is proportional to the number of constraints between the first violated constraint and the last constraint. Since $E(\cdot)$ arises less frequently in applications as a cost to minimise and can be defined in terms of $D(\cdot)$ after reversing the order of the constraints, it will not be considered in this paper as a separate function.

Obviously, one can define many other useful functions that can capture a larger set of control objectives. For example,

$$V(\theta|H,h) + S(\theta|g,R,r) + sL(\theta|g), \qquad (18)$$

where s is a non-negative scalar, is an objective function that typically arises in many model predictive control schemes with so-called "soft constraints" [4, 7]. As another example, one could define an objective in terms of

$$\max\{D(\theta|g^1), \ldots, D(\theta|g^t)\}, \qquad (19)$$

where each $g^i : \Theta \to \mathbb{R}^q$. However, in order to keep things simple, only objective functions defined in terms of (11), (13), (14) and (16) will be studied in the following sections.

Example 1 *In the framework presented in this paper, a QP in the form*

$$\min_{\theta \in \mathbb{R}^d} \{\theta^T H\theta + h^T\theta \,|A\theta \le b\}, \tag{20}$$

where $A \in \mathbb{R}^{q\times d}$ and $b \in \mathbb{R}^q$, can be interpreted as a prioritised, multi-objective optimisation problem where the satisfaction of the constraint $A\theta \le b$ is more important than the minimisation of the objective function $\theta^T H\theta + h^T\theta$. In other words, the above QP is equivalent to the prioritised, multi-objective problem

$$\text{lex}\min_{\theta \in \mathbb{R}^d} [f_1(\theta), f_2(\theta)]^T, \tag{21}$$

where "lex" denotes that the lexicographic minimum is sought, the objective functions are defined as

$$f_1(\theta) := L(\theta|g), \quad f_2(\theta) := V(\theta|H,h) \tag{22}$$

and the constraint function is defined as

$$g(\theta) := A\theta - b. \tag{23}$$

4 Main Results

This section will show how, provided all the constraint functions and Θ are convex, and the individual objective functions are suitably defined in terms of $V(\cdot)$, $D(\cdot)$, $L(\cdot)$ and $S(\cdot)$ as in the previous section, one can compute the lexicographic minimum of problem $\mathcal{P}$ iteratively via (3) and (4) using convex programming solvers, without having to resort to solving a (possibly highly inefficient) mixed-integer programming problem as in [2, 3, 8].

In the discussion to follow, it will assumed that

$$F_1 := \Theta \times \{0\}. \tag{24}$$

Obviously, (3) is equivalent to

$$f_1^* = \min_{\theta,\nu} \{f_1(\theta)\,|(\theta,\nu) \in F_1\}. \tag{25}$$

Similarly, it will be assumed that an $F_i \subseteq \theta \times \mathbb{R}^{s_i}$ has been given such that

$$f_i^* = \min_{\theta,\nu} \{f_i(\theta)\,|(\theta,\nu) \in F_i\}. \tag{26}$$

The necessity to introduce the above, seemingly unnecessary, notation and slack variables, arises when one or more of the objective functions $\{f_1(\cdot),\ldots,f_p(\cdot)\}$ of problem $\mathcal{P}$ have been defined in terms of (14). The assumption regarding F_i helps to keep the notation to a minimum, while still accounting for slack variables that might have arisen from solving for $\{f_1^*,\ldots,f_{i-1}^*\}$.

In the next few results, it will be shown how one can define $F_{i+1} \subseteq \Theta \times \mathbb{R}^{s_{i+1}}$, where $s_{i+1} \ge s_i$, in terms of F_i, f_i^* and additional slack vectors that might have arisen from solving for f_i^*, such that

$$f_{i+1}^* = \min_{\theta,w} \{f_{i+1}(\theta)\,|(\theta,w) \in F_{i+1}\}. \tag{27}$$

Fact 4 (Weighted sum of constraint violations) *If $f_i(\cdot)$ has the same structure as* (14), *i.e. $f_i(\theta) := S(\theta|g,R,r)$, and F_i is such that* (26) *holds, then*

$$f_i^* = \min_{\theta,\nu,z} \{z^T Rz + r^T z\,|(\theta,\nu) \in F_i, g(\theta) \le z, z \ge 0\}. \tag{28}$$

If $i < p$ and

$$F_{i+1} := \{(\theta,\nu,z) \in F_i \times \mathbb{R}^q \mid z^T Rz + r^T z \le f_i^*, \\ g(\theta) \le z, z \ge 0\}, \tag{29}$$

then $f_{i+1}^* = \min_{\theta,w} \{f_{i+1}(\theta)\,|(\theta,w) \in F_{i+1}\}$, where $w := (\nu,z)$.

Note that a vector-valued slack vector z is needed to compute f_i^* in (28) when $f_i(\cdot)$ has the same structure as (14). This slack vector needs to be included in $\{F_{i+1},\ldots,F_p\}$ when computing $\{f_{i+1}^*,\ldots,f_p^*\}$.

Fact 5 (Size of largest constraint violation) *If $f_i(\cdot)$ has the same structure as* (13), *i.e. $f_i(\theta) := L(\theta|g)$, and F_i is such that* (26) *holds, then*

$$f_i^* = \min_{\theta,\nu,z} \{z\,|(\theta,\nu) \in F_i, z \ge 0, g_j(\theta) \le z, j = 1,\ldots,q\}. \tag{30}$$

If $i < p$ and

$$F_{i+1} := \{(\theta,\nu) \in F_i\,|g_j(\theta) \le f_i^*, j = 1,\ldots,q\}, \tag{31}$$

then $f_{i+1}^* = \min_{\theta,\nu} \{f_{i+1}(\theta)\,|(\theta,\nu) \in F_{i+1}\}$.

Contrary to the case when $f_i(\cdot)$ has the same structure as (14), even though a scalar-valued slack vector z is needed to compute f_i^* in (30), it is not necessary to include this slack variable in $\{F_{i+1},\ldots,F_p\}$ when computing $\{f_{i+1}^*,\ldots,f_p^*\}$.

Fact 6 (Standard cost function) *If $f_i(\cdot)$ has the same structure as* (11), *i.e. $f_i(\theta) := V(\theta|H,h)$, and F_i is such that* (26) *holds, then*

$$f_i^* = \min_{\theta,\nu} \{\theta^T H\theta + h^T\theta\,|(\theta,\nu) \in F_i\} \tag{32}$$

If H is positive definite then the lexicographic minimiser θ^* of problem $\mathcal{P}$ is unique and is given by

$$(\theta^*,\nu^*) = \arg\min_{\theta,\nu} \{\theta^T H\theta + h^T\theta\,|(\theta,\nu) \in F_i\}. \tag{33}$$

If H is not positive definite, $i < p$ and

$$F_{i+1} := \{(\theta,\nu) \in F_i\,|\theta^T H\theta + h^T\theta \le f_i^*\}, \tag{34}$$

then $f_{i+1}^* = \min_{\theta,\nu} \{f_{i+1}(\theta)\,|(\theta,\nu) \in F_{i+1}\}$.

Recalling the comment at the end of Section 2, the above uniqueness observation implies that it is sensible to ensure that $f_p(\cdot)$ is the only objective function defined in terms of (11) with a positive definite Hessian.

Fact 7 (Largest element in index set of violations) *If $f_i(\cdot)$ has the same structure as* (16), *i.e.* $f_i(\theta) := D(\theta|g)$, *$F_i$ is such that* (26) *holds and*

$$0 < \min_{\theta,\nu,z} \{ z \,|\, (\theta,\nu) \in F_i, z \geq 0, g_q(\theta) \leq z \}, \qquad (35)$$

then $f_i^* = q$. If (35) *does not hold, then*

$$f_i^* = \min_{j \in \{1,\ldots,q-1\}} \{ j-1 \,|\, C_j(g,F_i) = 0 \}, \qquad (36)$$

where

$$C_j(g,F_i) := \min_{\theta,\nu,z} \{ z \,|\, (\theta,\nu) \in F_i, z \geq 0, g_j(\theta) \leq z, \; g_k(\theta) \leq 0, k = j+1,\ldots,q \}. \qquad (37)$$

If $f_i^* = q$, then let

$$F_{i+1} := F_i, \qquad (38)$$

otherwise let

$$F_{i+1} := \{ (\theta,\nu) \in F_i \,|\, g_k(\theta) \leq 0, k = f_i^* + 1,\ldots,q \}. \qquad (39)$$

If $i < p$, then $f_{i+1}^* = \min_{\theta,\nu} \{ f_{i+1}(\theta) \,|\, (\theta,\nu) \in F_{i+1} \}$.

Note that (35) is equivalent to

$$0 < \min_{(\theta,\nu) \in F_i} L(\theta|g_q). \qquad (40)$$

and that

$$C_j(g,F_i) = 0 \Leftrightarrow \min_{(\theta,\nu) \in F_i} L(\theta|g^j) = 0, \qquad (41)$$

where $g^j(\theta) := [g_j(\theta),\ldots,g_q(\theta)]^T$.

To summarise, an algorithm for computing a lexicographic minimiser of problem $\mathcal{P}$, is as follows:

Inputs: $f(\cdot) := [f_1(\cdot),\ldots,f_p(\cdot)]^T$ and Θ.
Outputs: $\theta^* \in \arg \operatorname{lex} \min_{\theta \in \Theta} f(\theta)$.

1: Let $F_1 := \Theta \times \{0\}$ and set $i \leftarrow 1$.
2: **while** $i < p$ **do**
3: Based on the structure of $f_i(\cdot)$, use one of the results given in this section to compute $f_i^* = \min_{\theta,\nu} \{ f_i(\theta) \,|\, (\theta,\nu) \in F_i \}$ and F_{i+1}.
4: Set $i \leftarrow i+1$.
5: **end while**
6: Based on the structure of $f_p(\cdot)$, use one of the results given in this section to compute $\theta^* \in \arg\min_{\theta,\nu} \{ f_p(\theta) \,|\, (\theta,\nu) \in F_p \}$.

Observe that if Θ and the constraint functions are defined by linear and/or convex, quadratic functions, steps 3 and 6 can be implemented by solving a number of linearly- and/or quadratically-constrained LPs and QPs.

The following example illustrates the use of the above results and notation.

Example 2 *Let* $f_1(\theta) := S(\theta|g,0,r)$, $f_2(\theta) := L(\theta|g)$ *and* $f_3(\theta) := S(\theta|g,R,0)$. *For simplicity of notation*[1], *let* $F_1 := \Theta$ *and hence* $f_1^* = \min_{\theta,\nu} \{ r^T \nu \,|\, \theta \in F_1, g(\theta) \leq \nu, \nu \geq 0 \}$. *If* $F_2 := \{ (\theta,\nu) \in F_1 \times \mathbb{R}^q \,|\, r^T \nu \leq f_1^*, g(\theta) \leq \nu, \nu \geq 0 \}$ *then* $f_2^* = \min_{\theta,\nu,w} \{ w \,|\, (\theta,\nu) \in F_2, w \geq 0, g_j(\theta) \leq w, j = 1,\ldots,q \}$. *Defining* $F_3 := \{ (\theta,\nu) \in F_2 \,|\, g_j(\theta) \leq f_2^*, j = 1,\ldots,q \}$ *it follows that* $f_3^* = \min_{\theta,\nu,w} \{ w^T R w \,|\, (\theta,\nu) \in F_3, g(\theta) \leq w, w \geq 0 \}$.

5 Application to Model Predictive Control

This section demonstrates, via an example, how the multi-objective framework developed in this paper can be applied to Model Predictive Control (MPC) of constrained, linear systems [4].

A brief overview of the idea behind MPC is as follows: Given an estimate of the current state x and a linear model of the plant

$$\begin{aligned} x_{k+1} &= Ax_k + Bu_k, \quad x_0 = x \\ y_k &= Cx_k \end{aligned} \qquad (42)$$

where $u_k \in \mathbb{R}^m$, $x_k \in \mathbb{R}^n$ and $y_k \in \mathbb{R}^l$, the objective in MPC is to compute on-line an open-loop sequence of N control inputs

$$\theta := [u_0^T, u_1^T, \ldots, u_{N-1}^T]^T \qquad (43)$$

that guarantees that constraints on the input and output of the plant, e.g.

$$\underline{u} \leq u_k \leq \overline{u}, \quad \underline{y_k} \leq y_k \leq \overline{y_k}, \qquad (44)$$

are satisfied over some finite time horizon, while minimising the deviation of the input and output trajectory from a desired reference trajectory. The first control input in the optimal open-loop control sequence is implemented, the state at the next time instant is estimated from new output data and a new open-loop control sequence is computed.

Typically, the constraints on the input represent physical constraints that cannot be violated (also known as "hard" constraints) and constraints on the output represent performance constraints that may be violated (also known as "soft" constraints). As such, it is sensible to choose

$$\Theta := \{ \theta \in \mathbb{R}^{mN} \,|\, \underline{u} \leq u_k \leq \overline{u}, k = 0,\ldots,N-1 \}. \qquad (45)$$

[1] To be consistent with (24) one should have $F_1 := \Theta \times \{0\}$, in which case $f_1^* = \min_{\theta,\nu,w} \{ r^T w \,|\, (\theta,\nu) \in F_1, g(\theta) \leq w, w \geq 0 \}$.

To illustrate how one can design a multi-objective MPC controller using the framework presented in this paper, let $l = 2$ and the prediction horizon be equal to the control horizon N. Given the current state x and a sequence of control inputs θ, one can easily show that

$$y_k(\theta,x) = C\left(A^k x + \sum_{j=0}^{k-1} A^j B u_{k-1-j}\right) \tag{46}$$

or, equivalently,

$$y_k(\theta,x) = C\left(A^k x + \begin{bmatrix} A^{k-1}B & \dots & B & 0 & \dots & 0 \end{bmatrix} \theta\right). \tag{47}$$

Let $y^i(\theta,x) := [y_1^i(\theta,x),\dots,y_N^i(\theta,x)]^T$ be the resulting predicted sequence of values for output i. Finally, let

$$g_k^i(\theta,x) := \begin{bmatrix} 1 \\ -1 \end{bmatrix} y_k^i(\theta,x) - \begin{bmatrix} \overline{y_k}^i \\ -\underline{y_k}^i \end{bmatrix} \tag{48}$$

and $g^i(\theta,x) := [g_1^i(\theta,x),\dots,g_N^i(\theta,x)]^T$ define the constraints on output i over the whole prediction horizon.

The control objectives to minimise, in order of importance, are as follows:

1: Duration of constraint relaxations for Output 1.
2: Duration of constraint relaxations for Output 2.
3: Size of the largest constraint violation of Output 1.
4: ℓ_1 norm of constraint violations of Output 2.
5: Quadratic norm of deviations of the outputs from 0.
6: Quadratic norm of deviations of the inputs from 0.

Minimising the above control objectives can be achieved by solving for the lexicographic minimum of problem $\mathcal{P}$ with the objective functions[2]:

1: $f_1(\theta,x) := D(\theta|g^1(\cdot,x))$
2: $f_2(\theta,x) := D(\theta|g^2(\cdot,x))$
3: $f_3(\theta,x) := L(\theta|g^1(\cdot,x))$
4: $f_4(\theta,x) := S(\theta|g^2(\cdot,x),0,[1,\dots,1]^T)$
5: $f_5(\theta,x) := V(\theta|H,h(x))$
6: $f_6(\theta,x) := V(\theta|I,0)$

where the H and $h(x)$ are found by a standard process [4, Chap. 3] of substituting (47) into $\sum_{k=1}^{N} \|y_k(\theta,x)\|_2^2$, rearranging and ignoring terms that are not dependent on θ.

In other words, given the current state x, a multi-objective MPC controller that aims to minimise the above prioritised objectives, only implements the first part $u_0^*(x)$ of $\theta^*(x)$, which is given by

$$\theta^*(x) = \arg \operatorname{lex} \min_{\theta\in\Theta} [f_1(\theta,x),\dots,f_6(\theta,x)]^T. \tag{49}$$

Remark 5 *In the above example, $f_6(\cdot,x)$ is strictly convex, hence the lexicographic minimiser $\theta^*(x)$ is unique.*

[2]Note that the objective functions are dependent on the current state x.

6 Concluding Remarks

This paper presented a multi-objective framework that allows one to design an MPC controller that can handle a large class of prioritised objectives and constraints in an optimal fashion. Under certain convexity assumptions, it is possible to compute the MPC control action by solving a sequence of convex programs, without having to resort to a mixed-integer approach, as in [2, 3, 8].

The framework presented in this paper is more general and flexible than the parametric programming approach presented in [9, 10]. The internal model, objectives and their relative priorities can be changed on-line without the need for redesigning the controller off-line. However, this increase in flexibility also demands an increase in the amount of on-line computational power that is required. It would be interesting to see whether a parametric programming equivalent of the framework in this paper can be developed, in order to decrease the amount of on-line computational power required.

References

[1] R.L. Keeney and H. Raiffa. *Decisions with Multiple Objectives.* Cambridge University Press, UK, 1993.

[2] E.C. Kerrigan. *Robust Constraint Satisfaction: Invariant Sets and Predictive Control.* PhD Thesis, University of Cambridge, UK, 2000. Thesis and associated Matlab Invariant Set Toolbox available for download at `http://www-control.eng.cam.ac.uk/eck21/`.

[3] E.C. Kerrigan, A. Bemporad, D. Mignone, M. Morari and J.M. Maciejowski. Multi-objective prioritisation and reconfiguration for the control of constrained hybrid systems. In: *Proceedings of the 2000 American Control Conference*, Chicago IL, USA, 2000.

[4] J.M. Maciejowski. *Predictive Control with Constraints.* Prentice Hall, UK, 2001.

[5] T.A. Meadowcroft, G. Stephanopoulos and C. Brosilow. The modular multivariable controller: I: Steady-state properties. *AIChE Journal* **38:8** 1254–1278, 1992.

[6] Y. Sawaragi, H. Nakayama and T. Tanino. *Theory of Multiobjective Optimization.* Academic Press Inc., USA, 1985.

[7] P.O.M. Scokaert and J.B. Rawlings. Feasibility Issues in Linear Model Predictive Control. *AIChE Journal* **45:8** 1649–1659, 1999.

[8] M.L. Tyler and M. Morari. Propositional logic in control and monitoring problems. *Automatica* **35:4** 565–582, 1999.

[9] J. Vada, O. Slupphaug and T.A. Johansen. Optimal prioritized infeasibility handling in model predictive control: parametric preemptive multiobjective linear programming approach. *Journal of Optimization Theory & Applications* **109:2**, 385–413, 2001.

[10] J. Vada, O. Slupphaug, T.A. Johansen and B.A. Foss. Linear MPC with optimal prioritized infeasibility handling: application, computational issues and stability. *Automatica* **37:11** 1835–1843, 2001.