

# Gain-Scheduled Controller Design: An Analytic Framework Directly Incorporating Non-Equilibrium Plant Dynamics

D.J.Leith  
W.E.Leithead

Department of Electronic & Electrical Engineering,  
University of Strathclyde,  
GLASGOW G1 1QE, U.K.

## Abstract

In this paper, a velocity-based linearisation framework is employed to develop a novel rigorous approach to gain-scheduling design. The proposed approach enables knowledge concerning the plant dynamics at non-equilibrium operating points to be incorporated directly into the controller design. Since the velocity-based linearisation framework supports the analysis of the transient response, performance considerations can be accommodated. The approach retains continuity with linear methods, which is central to the existing conventional gain-scheduling methodology, and, since a single type of linearisation is employed throughout, the design procedure is both straightforward and conceptually appealing.

## 1. Introduction

Whilst nonlinear dynamic systems are widespread, the analysis and design of such systems remains relatively difficult. In contrast, techniques for the analysis and design of linear time-invariant systems are rather better developed even though systems with genuinely linear time-invariant dynamics do not, in reality, exist. It is, therefore, attractive to adopt a divide and conquer philosophy whereby the design of a nonlinear system is decomposed into the design of a family of linear time-invariant systems. This type of strategy forms the basis of one of the most widely, and successfully, applied techniques for the design of nonlinear controllers; namely, gain-scheduling.

Gain-scheduled controllers are linked by the design approach employed, whereby a nonlinear controller is constructed by interpolating, in some manner, between the members of a family of linear time-invariant controllers. In the conventional, and most common, gain-scheduling design approach (see, for example, Astrom & Wittenmark 1989, Hyde & Glover 1993), each linear controller is typically associated with a specific equilibrium operating point of the plant and is designed to ensure that, locally to the equilibrium operating point, the performance requirements are met. (The existence of a family of equilibrium operating points, which spans the envelope of plant operation, is central to most gain-scheduling arrangements and it is not sufficient to restrict consideration to a single, isolated, equilibrium operating point). By employing a first-order series expansion approximation which, locally to the equilibrium operating point, has similar stability properties to the plant, linear techniques may be applied to this local design task. However, the requirement is usually for a controller which functions well not only when operating in the vicinity of a single equilibrium point but also during transitions between equilibrium operating points and periods of sustained non-equilibrium operation. Conventionally, this requirement is addressed by employing extensive simulation studies to iteratively refine the gain-scheduled design, but this quickly becomes extremely time-consuming and inefficient for any but the simplest nonlinear plants. There is, therefore, a considerable incentive to directly incorporate, into the analytical part of the design procedure, knowledge of the plant dynamics during transitions between equilibrium operating points and during sustained non-equilibrium operation.

In Leith & Leithead (1997b,c), a framework is proposed for the analysis of gain-scheduled and nonlinear systems which associates a family of velocity-based linearisations with a nonlinear system. Each operating point of the nonlinear system, including operating points far from equilibrium, has an associated member of the velocity-based linearisation family which describes the dynamic characteristics in the vicinity of that operating point. Hence, in contrast to the conventional series expansion linearisation about an equilibrium operating point, the velocity-based linearisation family indicates the plant dynamics not only in the vicinity of a single equilibrium operating point but also during transitions between equilibrium operating points and when operating far from equilibrium. The purpose of this paper is to investigate the direct exploitation of information concerning the plant dynamics at non-equilibrium points in gain-scheduling design by employing velocity-based

linearisation families. The paper is organised as follows. In section 2, the existing gain-scheduling design approaches are reviewed and, in section 3, the velocity-based linearisation families are described. A framework employing velocity-based linearisations for the analysis and design of gain-scheduled controllers is developed in section 4 and specialised to the class of plants satisfying an extended local linear equivalence condition in section 5. The conclusions are summarised in section 6.

## 2. Conventional gain-scheduling design

Consider the nonlinear plant with dynamics,

$$\dot{\mathbf{x}} = \mathbf{F}(\mathbf{x}, \mathbf{r}), \quad \mathbf{y} = \mathbf{G}(\mathbf{x}, \mathbf{r}) \quad (1)$$

where  $\mathbf{F}(\cdot, \cdot)$  and  $\mathbf{G}(\cdot, \cdot)$  are continuous with Lipschitz continuous first derivatives,  $\mathbf{r} \in \mathfrak{R}^m$  denotes the input to the plant,  $\mathbf{y} \in \mathfrak{R}^p$  the output and  $\mathbf{x} \in \mathfrak{R}^n$  the state. When necessary, assume

$[\nabla_{\mathbf{r}}\mathbf{F} \quad \nabla_{\mathbf{x}}\mathbf{F}\nabla_{\mathbf{r}}\mathbf{F} \quad \dots \quad (\nabla_{\mathbf{x}}\mathbf{F})^{n-1}\nabla_{\mathbf{r}}\mathbf{F}]$  has rank  $n \quad \forall \mathbf{x}, \mathbf{r}$ . The set of equilibrium operating points of the nonlinear plant, (1), consists of those points,  $(\mathbf{x}_o, \mathbf{y}_o, \mathbf{r}_o)$ , for which

$$\mathbf{F}(\mathbf{x}_o, \mathbf{r}_o) = \mathbf{0}, \quad \mathbf{y}_o = \mathbf{G}(\mathbf{x}_o, \mathbf{r}_o) \quad (2)$$

Let  $\Phi: \mathfrak{R}^n \times \mathfrak{R}^m$  denote the space consisting of the union of the state,  $\mathbf{x}$ , with the input,  $\mathbf{r}$ . The set of equilibrium operating points of the nonlinear plant, (1), forms a locus of points,  $(\mathbf{x}_o, \mathbf{r}_o)$ , in  $\Phi$  and the response of the plant to a general time-varying input,  $\mathbf{r}(t)$ , is depicted by a trajectory in  $\Phi$ .

The gain-scheduled design approach constructs a nonlinear controller, with certain required dynamic properties, by interpolating, in some sense, between the members of an appropriate family of linear time-invariant controllers. The conventional gain-scheduling design approach may be applied directly to a broad range of nonlinear plants and the design procedure typically involves the following steps (see, for example, Astrom & Wittenmark 1989 section 9.5, Shamma & Athans 1990, Hyde & Glover 1993, Leith & Leithead 1996).

1. The equilibrium operating points of the plant are parameterised by an appropriate quantity,  $\boldsymbol{\rho}$ , which may involve the plant input, output and/or state.
2. The plant dynamics, (1), are approximated, locally to a specific equilibrium operating point,  $(\mathbf{x}_o, \mathbf{r}_o, \mathbf{y}_o)$ , at which  $\boldsymbol{\rho}$  equals  $\boldsymbol{\rho}_o$ , by the series expansion linearisation,

$$\delta \dot{\hat{\mathbf{x}}} = \nabla_{\mathbf{x}}\mathbf{F}(\mathbf{x}_o(\boldsymbol{\rho}_o), \mathbf{r}_o(\boldsymbol{\rho}_o))\delta \hat{\mathbf{x}} + \nabla_{\mathbf{r}}\mathbf{F}(\mathbf{x}_o(\boldsymbol{\rho}_o), \mathbf{r}_o(\boldsymbol{\rho}_o))\delta \mathbf{r} \quad (3)$$

$$\delta \hat{\mathbf{y}} = \nabla_{\mathbf{x}}\mathbf{G}(\mathbf{x}_o(\boldsymbol{\rho}_o), \mathbf{r}_o(\boldsymbol{\rho}_o))\delta \hat{\mathbf{x}} + \nabla_{\mathbf{r}}\mathbf{G}(\mathbf{x}_o(\boldsymbol{\rho}_o), \mathbf{r}_o(\boldsymbol{\rho}_o))\delta \mathbf{r} \quad (4)$$

$$\delta \mathbf{r} = \mathbf{r} - \mathbf{r}_o(\boldsymbol{\rho}_o), \quad \hat{\mathbf{y}} = \delta \hat{\mathbf{y}} + \mathbf{y}_o(\boldsymbol{\rho}_o), \delta \hat{\mathbf{x}} = \hat{\mathbf{x}} - \mathbf{x}_o(\boldsymbol{\rho}_o) \quad (5)$$

3. For a suitable controller input,  $\mathbf{e}$ , with equilibrium value,  $\mathbf{e}_o(\boldsymbol{\rho}_o)$ , a linear time-invariant controller is designed,

$$\delta \dot{\mathbf{z}} = \mathbf{A}(\boldsymbol{\rho}_o)\delta \mathbf{z} + \mathbf{B}(\boldsymbol{\rho}_o)\delta \mathbf{e} \quad (6)$$

$$\delta \mathbf{r} = \mathbf{C}(\boldsymbol{\rho}_o)\delta \mathbf{z} + \mathbf{D}(\boldsymbol{\rho}_o)\delta \mathbf{e} \quad (7)$$

$$\delta \mathbf{e} = \mathbf{e} - \mathbf{e}_o(\boldsymbol{\rho}_o), \quad \mathbf{r} = \delta \mathbf{r} + \mathbf{r}_o(\boldsymbol{\rho}_o) \quad (8)$$

which ensures appropriate closed-loop performance when employed with the plant linearisation,

(3)-(5). It should be noted that  $\boldsymbol{\rho}_o$  is assumed to be constant when designing this linear controller.

4. Repeat steps 2 and 3 as required for a family of equilibrium operating points, ensuring that the linear controller designs have compatible structures; for example, when a smoothly gain-scheduled controller is required, the linear controller designs are selected to permit smooth interpolation, in some appropriate manner, between the designs. A family of linear time-invariant controllers is obtained corresponding to the family of equilibrium operating points; both the controller family and the equilibrium operating points are parameterised by  $\boldsymbol{\rho}$ .
5. Implement the controller input and output transformations, (8). Typically, the controller input,  $\mathbf{e} = \mathbf{y} - \mathbf{y}_{\text{ref}}$ , is zero in equilibrium (that is,  $\mathbf{e}_o(\boldsymbol{\rho}_o) = \mathbf{0}$  and  $\delta \mathbf{e} = \mathbf{e}$ ) and either the plant exhibits pure integral action, so that  $\mathbf{r}_o(\boldsymbol{\rho})$  is identically zero, or each linear controller contains integral action which implicitly generates  $\mathbf{r}_o(\boldsymbol{\rho})$  through the action of the feedback loop (see, for example, Astrom & Wittenmark 1989 section 9.5, Shamma & Athans 1990, Hyde & Glover 1993, Leith & Leithead 1996). Alternatively, the controller output transformation may be implemented by explicitly calculating the equilibrium controller output as a function of  $\boldsymbol{\rho}$  (see, for example, Rugh 1991, Shamma & Athans 1990). However, the latter approach may involve rather complex calculations which are sensitive to modelling errors and, consequently, seems to be largely of theoretical interest (Hyde & Glover 1991).
6. Substitute  $\boldsymbol{\rho}$  (or some related quantity) for  $\boldsymbol{\rho}_o$  in the family of local linear controllers, (6)-(8), to obtain a nonlinear controller. It is noted that the scheduling variable need not be continuous; for

example, it may be piece-wise constant, corresponding to switching between the members of the family of local linear controllers. Typically, the selection of an appropriate scheduling variable is based on physical insight (Astrom & Wittenmark 1989).

The gain-scheduling design process is frequently iterative, with the controller revised in the light of subsequent analysis until a satisfactory design is achieved.

The effectiveness of the gain-scheduled design approach depends on the dynamic characteristics of the nonlinear system, composed of the nonlinear plant and the nonlinear gain-scheduled controller, being related to those of the members of an associated family of linear systems, composed of the plant linearisations and corresponding local linear controllers. The existing results relating the dynamic characteristics of a nonlinear system to those of an associated family of linear systems is reviewed in Leith & Leithead (1997b,c) and summarised below. Series expansion linearisation theory is well established but is strictly confined to the dynamic analysis, locally to a single trajectory or equilibrium operating point, of smooth nonlinear systems. When the family of equilibrium operating points can be parameterised by the input to the nonlinear system (as distinct from the scheduling variable), frozen-input techniques cater for the analysis of smooth nonlinear systems relative to a family of equilibrium operating points and relate the stability of a nonlinear system to the stability of a family of frozen-input *nonlinear* systems. A slow variation requirement is necessary which seems to be inherent to this type of analysis, implicitly restricting the class of allowable inputs and initial conditions; that is, implicitly restricting the trajectories to remain sufficiently close to the equilibrium operating points. In order to relate the stability of the nonlinear system to the properties of a family of *linear* time-invariant systems, a further explicit restriction on the allowable trajectories is necessary to ensure they remain sufficiently close to the equilibrium operating points that series expansion linearisations are valid. This latter restriction is not *a priori* necessary yet may be very strong since the neighbourhoods within which the series expansion linearisations are valid may, in general, be excessively small. The utility of frozen-input theory is, thus, somewhat diminished since it may imply a high degree of unnecessary conservativeness. Series expansion linearisation theory and frozen-input theory consider only the stability properties of the nonlinear system and provide little direct insight into other dynamic properties, such as the transient response. When the scheduling is not continuous, few techniques, other than extensive simulation testing, appear to be available for analysing the dynamic behaviour of the controlled system.

Although frozen-input theory can support the analysis of a nonlinear gain-scheduled control system, it provides little insight into the controller design procedure since the *frozen-input* representation of the controlled system is quite distinct from the mixed series-expansion/frozen-scheduling variable representation employed in step 3 of the design procedure. *Series expansion* linearisations of the plant are employed but the corresponding local controller designs are *frozen-scheduling variable* linearisations of the resulting nonlinear controller. In contrast, when analysing the dynamic behaviour of the controller locally to a single equilibrium operating point, the *series expansion* linearisation is employed instead of the frozen-scheduling variable linearisation. Furthermore, since the scheduling variable,  $\rho$ , is varying in the nonlinear controller but constant in the local designs, the nonlinear controller need not have the designed dynamics, locally to an equilibrium operating point. Moreover, the analysis of the controlled system in the vicinity of the family of equilibrium operating points does not reduce to either the series expansion analysis (see, for example Shamma 1988 p110) or the mixed series-expansion/frozen-scheduling variable analysis employed in the design procedure.

The analysis of conventional gain-scheduling design by means of existing results, relating the dynamic characteristics of a nonlinear system to those of an associated family of linear systems, is, therefore, rather complex and inefficient. The use of a variety of different local approximations obscures insight and is surely unnecessary. In addition, the analysis is confined to stability properties and does not directly extend to other dynamic characteristics such as the transient response. Hence, the existing theory does not provide an adequate framework to support the analysis and design of gain-scheduled controllers.

### 3. Velocity-based linearisation families

An alternative approach, not discussed in section 2, to the analysis of a nonlinear system by relating its dynamic characteristics to those of an associated family of linear systems, is developed in Leith & Leithead (1997b). Consider, the behaviour of the nonlinear system, (1), when there are no restrictions on the class of allowable inputs and initial conditions. The solutions to (1) may trace trajectories anywhere in  $\Phi$  and are not confined to the vicinity of either a single equilibrium operating point or the

locus of equilibrium operating points. Suppose that the nonlinear system is evolving along a trajectory,  $(\mathbf{x}(t), \mathbf{r}(t))$ , in  $\Phi$  and at time,  $t_1$ , the trajectory has reached the point,  $(\mathbf{x}_1, \mathbf{r}_1)$ . It is emphasised that the point,  $(\mathbf{x}_1, \mathbf{r}_1)$ , need not be an equilibrium operating point and, indeed, may lie far from the locus of equilibrium operating points. From Taylor series expansion theory, the subsequent behaviour of the nonlinear system, (1), can be approximated, locally to  $(\mathbf{x}_1, \mathbf{r}_1)$ , by the first order representation,

$$\delta \dot{\hat{\mathbf{x}}} = \mathbf{F}(\mathbf{x}_1, \mathbf{r}_1) + \nabla_{\mathbf{x}}\mathbf{F}(\mathbf{x}_1, \mathbf{r}_1) \delta \hat{\mathbf{x}} + \nabla_{\mathbf{r}}\mathbf{F}(\mathbf{x}_1, \mathbf{r}_1) \delta \mathbf{r} \quad (9)$$

$$\delta \hat{\mathbf{y}} = \nabla_{\mathbf{x}}\mathbf{G}(\mathbf{x}_1, \mathbf{r}_1) \delta \hat{\mathbf{x}} + \nabla_{\mathbf{r}}\mathbf{G}(\mathbf{x}_1, \mathbf{r}_1) \delta \mathbf{r} \quad (10)$$

$$\delta \mathbf{r} = \mathbf{r} - \mathbf{r}_1, \quad \hat{\mathbf{y}} = \mathbf{y}_1 + \delta \hat{\mathbf{y}}, \quad \hat{\mathbf{x}} = \delta \hat{\mathbf{x}} + \mathbf{x}_1, \quad \dot{\hat{\mathbf{x}}} = \delta \dot{\hat{\mathbf{x}}} \quad (11)$$

provided  $\mathbf{x}_1 + \delta \hat{\mathbf{x}} \subseteq N_{\mathbf{x}}$  and  $\mathbf{r}_1 + \delta \mathbf{r} \subseteq N_{\mathbf{r}}$ , where the neighbourhoods,  $N_{\mathbf{x}}$  and  $N_{\mathbf{r}}$ , of, respectively,  $\mathbf{x}_1$  and  $\mathbf{r}_1$  are sufficiently small. When (9)-(11) and (1) have the same initial conditions,  $(\mathbf{x}_1, \mathbf{r}_1)$ , the solution to (9)-(11) is, initially, tangential to the solution of (1) and, indeed, locally to time  $t_1$ , provides a first-order approximation to  $\dot{\mathbf{x}}(t)$  and a second-order approximation to  $\mathbf{x}(t)$  (Leith & Leithead 1997b,c).

The solution to the first-order series expansion, (9)-(11), provides a valid approximation only while the solution,  $\mathbf{x}(t)$ , to the nonlinear system remains in the vicinity the operating point,  $(\mathbf{x}_1, \mathbf{r}_1)$ . However, the solution,  $\mathbf{x}(t)$ , to the nonlinear system need not stay in the vicinity of a single operating point. Consider, therefore, the approximation to  $\mathbf{x}(t)$  over a time interval,  $[t_1, t_2]$ , obtained by partitioning the interval into a number of short sub-intervals. Over each sub-interval, the approximate solution is the solution to the first-order series expansion relative to the operating point reached at the initial time for the sub-interval (with the initial conditions chosen to ensure continuity of the approximate solution). The number of local solutions employed is dependent on the duration of the sub-intervals, but the local solutions are now accurate to second order; that is, the approximation error is proportional to the duration of the sub-interval cubed. Hence, as the number of sub-intervals increases, the approximation error associated with each rapidly decreases and the overall approximation error reduces. Indeed, the overall approximation error tends to zero as the number of sub-intervals becomes unbounded (Leith & Leithead 1997b,c). Hence, the family of first-order series expansions, with members defined by (9)-(11), can provide an accurate approximation to the solution of the nonlinear system. Moreover, this approximation property holds throughout  $\Phi$  and is not confined to the vicinity of a single equilibrium operating point or even of the locus of equilibrium operating points.

The state, input and output transformations, (11), depend on the operating point relative to which the series expansion is carried out. Combining (9) and (10) with the local input, output and state transformations, (11), each member, (9)-(11), of the family of first-order representations may be reformulated as,

$$\dot{\hat{\mathbf{x}}} = \{ \mathbf{F}(\mathbf{x}_1, \mathbf{r}_1) - \nabla_{\mathbf{x}}\mathbf{F}(\mathbf{x}_1, \mathbf{r}_1) \mathbf{x}_1 - \nabla_{\mathbf{r}}\mathbf{F}(\mathbf{x}_1, \mathbf{r}_1) \mathbf{r}_1 \} + \nabla_{\mathbf{x}}\mathbf{F}(\mathbf{x}_1, \mathbf{r}_1) \hat{\mathbf{x}} + \nabla_{\mathbf{r}}\mathbf{F}(\mathbf{x}_1, \mathbf{r}_1) \mathbf{r} \quad (12)$$

$$\hat{\mathbf{y}} = \{ \mathbf{G}(\mathbf{x}_1, \mathbf{r}_1) - \nabla_{\mathbf{x}}\mathbf{G}(\mathbf{x}_1, \mathbf{r}_1) \mathbf{x}_1 + \nabla_{\mathbf{r}}\mathbf{G}(\mathbf{x}_1, \mathbf{r}_1) \mathbf{r}_1 \} + \nabla_{\mathbf{x}}\mathbf{G}(\mathbf{x}_1, \mathbf{r}_1) \hat{\mathbf{x}} + \nabla_{\mathbf{r}}\mathbf{G}(\mathbf{x}_1, \mathbf{r}_1) \mathbf{r} \quad (13)$$

The state, input and output is now the same at every point in  $\Phi$ , but the dynamics, (12)-(13), are nonlinear. By differentiating, (12)-(13) may be reformulated in the equivalent velocity-based form,

$$\dot{\hat{\mathbf{x}}} = \hat{\mathbf{w}}, \quad \dot{\hat{\mathbf{w}}} = \nabla_{\mathbf{x}}\mathbf{F}(\mathbf{x}_1, \mathbf{r}_1) \hat{\mathbf{w}} + \nabla_{\mathbf{r}}\mathbf{F}(\mathbf{x}_1, \mathbf{r}_1) \dot{\mathbf{r}}, \quad \dot{\hat{\mathbf{y}}} = \nabla_{\mathbf{x}}\mathbf{G}(\mathbf{x}_1, \mathbf{r}_1) \hat{\mathbf{w}} + \nabla_{\mathbf{r}}\mathbf{G}(\mathbf{x}_1, \mathbf{r}_1) \dot{\mathbf{r}} \quad (14)$$

With appropriate initial conditions, namely,

$$\hat{\mathbf{x}}(t_1) = \mathbf{x}(t_1), \quad \hat{\mathbf{w}}(t_1) = \dot{\hat{\mathbf{x}}}(t_1) = \dot{\mathbf{x}}(t_1) = \mathbf{F}(\mathbf{x}_1, \mathbf{r}_1), \quad \hat{\mathbf{y}}(t_1) = \mathbf{y}(t_1) = \mathbf{G}(\mathbf{x}_1, \mathbf{r}_1) \quad (15)$$

the transformed system is dynamically equivalent to the original system. However, in contrast to (12)-(13), the transformed system, (14), is linear. There exists a velocity-based linearisation, (14), for every point in  $\Phi$ . Hence, a velocity-based linearisation family, with members defined by (14), can be associated with the nonlinear system, (1).

The relationship between the nonlinear system and its velocity-based linearisation family is direct. Differentiating (1), an alternative representation of the nonlinear system is

$$\dot{\mathbf{x}} = \mathbf{w}, \quad \dot{\mathbf{w}} = \nabla_{\mathbf{x}}\mathbf{F}(\mathbf{x}, \mathbf{r})\mathbf{w} + \nabla_{\mathbf{r}}\mathbf{F}(\mathbf{x}, \mathbf{r})\dot{\mathbf{r}}, \quad \dot{\mathbf{y}} = \nabla_{\mathbf{x}}\mathbf{G}(\mathbf{x}, \mathbf{r})\mathbf{w} + \nabla_{\mathbf{r}}\mathbf{G}(\mathbf{x}, \mathbf{r})\dot{\mathbf{r}} \quad (16)$$

Dynamically, (16), with appropriate initial conditions corresponding to (15), and (1) are equivalent. Evidently, the velocity-based linearisation, (14), is simply the frozen form of (16) at the operating point,  $(\mathbf{x}_1, \mathbf{r}_1)$ . (When  $\mathbf{w} = \mathbf{F}(\mathbf{x}, \mathbf{r})$ ,  $\mathbf{y} = \mathbf{G}(\mathbf{x}, \mathbf{r})$  is invertible at every operating point,  $(\mathbf{x}, \mathbf{r})$ , in an appropriate neighbourhood enclosing the locus of equilibrium operating points, so that  $\mathbf{x}$  may be expressed as a function of  $\mathbf{w}$ ,  $\mathbf{r}$  and  $\mathbf{y}$ , then the transformation relating (16) to (1) is, in fact, algebraic). The solutions to the members of the family of velocity-based linearisations, (14), can be pieced together to approximate the solution to the nonlinear system, (16). In this case, the  $\hat{\mathbf{x}}(t)$  are still second-order approximations to the  $\mathbf{x}(t)$  but the  $\hat{\mathbf{w}}(t)$  are first-order approximations to the  $\mathbf{w}(t)$ . However, it is straightforward to show that the piece-wise approximation converges to the exact solution (Leith & Leithead 1997b,c).

In contrast to the previous approaches discussed in section 2, the velocity-based linearisation analysis has several advantages. There exists a linearisation of the nonlinear system at every operating point and not just the equilibrium operating points. Stability conditions are derived for nonlinear systems which avoid the restrictions, to trajectories lying within an unnecessarily, perhaps excessively, small neighbourhood about the locus of equilibrium operating points, inherent to previous approaches based on frozen-input theory (Leith & Leithead 1997b,c). A restriction on the allowable class of inputs and initial conditions is still required. However, in contrast to previous results, it is emphasised that this restriction is purely a consequence of the slow variation requirement and, in this sense, is as weak as possible. Indeed, for systems where there is no restriction on the rate of variation, the analysis is *global* in nature. Hence, the stability conditions derived using the velocity-based linearisations are inherently much less conservative than those obtained previously. The stability analysis is also extended to include nonlinear systems with non-smooth dynamics, such as gain-scheduled controllers which switch between local controllers rather than employing smooth interpolation (Leith & Leithead 1997b,c). Furthermore, the velocity-based linearisation analysis is not confined to stability. Since the members of the family of velocity-based linearisations can be pieced together to approximate the solution to a nonlinear system, the transient behaviour of the nonlinear system can also be investigated. This approximation is not confined to the vicinity of the equilibrium operating points but is valid throughout the operating envelope, including during transitions between equilibrium operating points and at operating points which are far from equilibrium. Consequently, the velocity-based linearisation theory has considerable potential for supporting the design and analysis of gain-scheduled controllers.

#### 4. Gain-scheduled design using velocity-based linearisation families

The requirement is to directly exploit the advantages of velocity-based linearisations; particularly, to avoid the restriction to operation in the vicinity of the equilibrium operating points which is inherent in existing gain-scheduling design approaches. Since the velocity-based linearisation family associated with a nonlinear plant describes the dynamic behaviour at every operating point, not just equilibrium operating points, it clearly has the potential to meet this requirement.

Consider the nonlinear plant, (1), and the nonlinear controller

$$\dot{\mathbf{x}}_c = \mathbf{F}_c(\mathbf{x}_c, \mathbf{r}_c), \quad \mathbf{y}_c = \mathbf{G}_c(\mathbf{x}_c, \mathbf{r}_c) \quad (17)$$

where  $\mathbf{r}_c \in \mathfrak{R}^{m_c}$  denotes the input to the controller,  $\mathbf{y}_c \in \mathfrak{R}^{p_c}$  the output and  $\mathbf{x}_c \in \mathfrak{R}^{n_c}$  the state. Since the requirement is to design a feedback controller, it is assumed without loss of generality that the input vector,  $\mathbf{r}$ , to the plant includes the output,  $\mathbf{y}_c$ , of the controller and the input vector,  $\mathbf{r}_c$ , to the controller includes the output,  $\mathbf{y}$ , of the plant. Let  $\mathbf{r}_p$  denote the vector consisting of the elements of  $\mathbf{r}$  which are not elements of  $\mathbf{y}_c$ , and let  $\mathbf{r}_c$  denote the vector consisting of the elements of  $\mathbf{r}_c$  which are not elements of  $\mathbf{y}$ . In addition, it is assumed that the inverse plant mapping from  $\mathbf{F}(\mathbf{x}, \mathbf{r})$  to  $(\mathbf{x}, \mathbf{r})$  is bounded; that is,  $\mathbf{x}$  is bounded when  $\mathbf{F}(\mathbf{x}, \mathbf{r})$  and  $\mathbf{r}$  are bounded. By differentiating, the plant may be reformulated in velocity-based form as

$$\dot{\mathbf{x}} = \mathbf{w}, \quad \dot{\mathbf{w}} = \mathbf{A}(\boldsymbol{\rho})\mathbf{w} + \begin{bmatrix} \mathbf{B}(\boldsymbol{\rho}) & \mathbf{B}_{r_p}(\boldsymbol{\rho}) \end{bmatrix} \begin{bmatrix} \dot{\mathbf{y}}_c \\ \dot{\mathbf{r}}_{r_p} \end{bmatrix}, \quad \dot{\mathbf{y}} = \mathbf{C}(\boldsymbol{\rho})\mathbf{w} + \begin{bmatrix} \mathbf{D}(\boldsymbol{\rho}) & \mathbf{D}_{r_p}(\boldsymbol{\rho}) \end{bmatrix} \begin{bmatrix} \dot{\mathbf{y}}_c \\ \dot{\mathbf{r}}_{r_p} \end{bmatrix} \quad (18)$$

and the controller may be reformulated in velocity-based form as

$$\dot{\mathbf{x}}_c = \mathbf{w}_c, \quad \dot{\mathbf{w}}_c = \mathbf{A}_c(\boldsymbol{\rho}_c)\mathbf{w}_c + \begin{bmatrix} \mathbf{B}_c(\boldsymbol{\rho}_c) & \mathbf{B}_{r_c}(\boldsymbol{\rho}_c) \end{bmatrix} \begin{bmatrix} \dot{\mathbf{y}} \\ \dot{\mathbf{r}}_{r_c} \end{bmatrix}, \quad \dot{\mathbf{y}}_c = \mathbf{C}_c(\boldsymbol{\rho}_c)\mathbf{w}_c + \begin{bmatrix} \mathbf{D}_c(\boldsymbol{\rho}_c) & \mathbf{D}_{r_c}(\boldsymbol{\rho}_c) \end{bmatrix} \begin{bmatrix} \dot{\mathbf{y}} \\ \dot{\mathbf{r}}_{r_c} \end{bmatrix} \quad (19)$$

where

$$\begin{aligned} \mathbf{A}(\boldsymbol{\rho}) &= \nabla_{\mathbf{x}} \mathbf{F}(\mathbf{x}, \mathbf{r}), & \mathbf{B}_{r_p}(\boldsymbol{\rho}) &= \nabla_{\mathbf{r}_p} \mathbf{F}(\mathbf{x}, \mathbf{r}), & \mathbf{B}(\boldsymbol{\rho}) &= \nabla_{\mathbf{y}_c} \mathbf{F}(\mathbf{x}, \mathbf{r}) \\ \mathbf{C}(\boldsymbol{\rho}) &= \nabla_{\mathbf{x}} \mathbf{G}(\mathbf{x}, \mathbf{r}), & \mathbf{D}_{r_p}(\boldsymbol{\rho}) &= \nabla_{\mathbf{r}_p} \mathbf{G}(\mathbf{x}, \mathbf{r}), & \mathbf{D}(\boldsymbol{\rho}) &= \nabla_{\mathbf{y}_c} \mathbf{G}(\mathbf{x}, \mathbf{r}) \\ \mathbf{A}_c(\boldsymbol{\rho}_c) &= \nabla_{\mathbf{x}_c} \mathbf{F}_c(\mathbf{x}_c, \mathbf{r}_c), & \mathbf{B}_{r_c}(\boldsymbol{\rho}_c) &= \nabla_{\mathbf{r}_c} \mathbf{F}_c(\mathbf{x}_c, \mathbf{r}_c), & \mathbf{B}_c(\boldsymbol{\rho}_c) &= \nabla_{\mathbf{y}} \mathbf{F}_c(\mathbf{x}_c, \mathbf{r}_c) \\ \mathbf{C}_c(\boldsymbol{\rho}_c) &= \nabla_{\mathbf{x}_c} \mathbf{G}_c(\mathbf{x}_c, \mathbf{r}_c), & \mathbf{D}_{r_c}(\boldsymbol{\rho}_c) &= \nabla_{\mathbf{r}_c} \mathbf{G}_c(\mathbf{x}_c, \mathbf{r}_c), & \mathbf{D}_c(\boldsymbol{\rho}_c) &= \nabla_{\mathbf{y}} \mathbf{G}_c(\mathbf{x}_c, \mathbf{r}_c) \end{aligned} \quad (20)$$

and  $\boldsymbol{\rho}(\mathbf{x}, \mathbf{r})$ ,  $\boldsymbol{\rho}_c(\mathbf{x}_c, \mathbf{r}_c)$  embody the dependence of the dynamics on the states and inputs of the plant and controller, respectively.

The combined closed-loop dynamics are depicted in figure 1a. Assuming  $\mathbf{y} = \mathbf{G}(\mathbf{x}, \mathbf{r})$ ,  $\mathbf{y}_c = \mathbf{G}_c(\mathbf{x}_c, \mathbf{r}_c)$ , with  $\mathbf{r}$  related to  $\mathbf{y}_c$  and  $\mathbf{r}_c$  related to  $\mathbf{y}$  as described above, has a solution,  $\mathbf{y} = \mathbf{H}(\mathbf{x}, \mathbf{x}_c, \mathbf{r}_p, \mathbf{r}_c)$ , the velocity-based form for the closed-loop system may be represented directly in terms of the velocity-

based form of the plant, (18), and the velocity-based form of the controller, (19), as depicted in figure 1b (see Appendix). The velocity-based linearisation families associated with the plant and the controller consist simply of the frozen forms of, respectively, (18) and (19), obtained for constant values of  $\rho$  and  $\rho_c$ . In addition, each member of the velocity-based linearisation family for the closed-loop system may be obtained by enclosing the appropriate members of the plant and controller families in a feedback loop. Given this direct relationship between the velocity-based form of the nonlinear systems and their associated velocity-based linearisation families and the strong correspondence in their dynamic behaviour as discussed in section 3, the velocity-based linearisation families constitute a much more appropriate framework for the analysis and design of gain-scheduled controllers than conventional approaches.

The foregoing analysis suggests the following gain-scheduling design procedure.

1. Determine the velocity-based linearisation family associated with the nonlinear plant dynamics.
2. Based on the plant velocity-based linearisation family, determine the required controller velocity-based linearisation family such that the resulting closed-loop family achieves the performance requirements. Since each member of the plant family is linear, conventional *linear* design methods can be utilised to design each corresponding member of the controller family.
3. Realise a nonlinear controller corresponding to the family of linear controllers designed at step 2. The velocity-based form of the controller can be obtained directly from the family of linear controllers by simply permitting the  $\rho$  to vary with the operating point. Since the velocity-based form of the system, composed of the nonlinear plant, (1), together with the velocity-based form of the controller, (19), is identical to the velocity-based form of the system composed of the velocity-based form of the plant, (18), together with the velocity-based form of the controller, (19), see the Appendix, an alternative to the realisation of figure 1a is that shown in figure 1c with the velocity-based form of the controller. The latter has the advantage of avoiding the need to determine a nonlinear controller, (17), corresponding to the velocity-based form, (19).

This design procedure retains a divide and conquer philosophy and maintains the continuity with linear design methods which is an important feature of the conventional gain-scheduling approach. However, in contrast to the conventional gain-scheduling approach, the resulting nonlinear controller is valid throughout the operating envelope of the plant, not just in the vicinity of the equilibrium operating points. This extension is a direct consequence of employing the velocity-based linearisation framework rather than the conventional series expansion linearisation about an equilibrium operating point.

With regard to step 3 of the design procedure, it should be noted that there are a number of issues which must be considered when determining the nonlinear controller realisation corresponding to the family of linear controllers designed at step 2. In particular, the output,  $\mathbf{y}_c$ , of the controller is an input to the plant and the input,  $\mathbf{r}_c$ , of the controller is an output from the plant. Hence, the value of  $\mathbf{r}_c$  and  $\mathbf{y}_c$  at an equilibrium operating point of the plant is  $(\mathbf{r}_{co}, \mathbf{y}_{co})$  with  $\mathbf{r}_{co}$  dependent on  $\mathbf{y}_{co}$  *via* the plant. However, since  $\mathbf{r}_c$  is the controller input and  $\mathbf{y}_c$  is the controller output,  $(\mathbf{r}_{co}, \mathbf{y}_{co})$  must also be an equilibrium operating point of the controller with  $\mathbf{y}_o$  dependent on  $\mathbf{r}_o$  *via* the controller. Requiring consistency imposes, in general, a strong restriction on the allowable nonlinear controllers. However, this restriction is circumvented by adopting the velocity-based realisation of figure 1c since, in equilibrium, the output of the differentiation term before the controller and the input to the integral term after the controller are both zero. Of course, the presence of a derivative and integral action on the forward path in the velocity-based realisation of figure 1c requires to be treated with some care. However, when the controller contains integral action, the differentiation operator at the input and the pure integrator within the controller may be formally absorbed together<sup>1</sup> so that the input to the controller becomes  $\mathbf{r}_c$  rather than  $\dot{\mathbf{r}}_c$ . The integration of the controller output then explicitly provides the required integral action (Leith & Leithead 1997a). A further issue that must be addressed is the most appropriate manner in which to implement the scheduling variable,  $\rho_c$ . It is straightforward to implement  $\rho_c$  when it is a function of  $\mathbf{y}_c$  and  $\mathbf{r}_c$  alone. When  $\rho_c$  is also a function of  $\mathbf{x}_c$ , a number of approaches can be adopted to obtain an appropriate realisation of the scheduling variable. For example, when  $[\mathbf{F}_c \ \mathbf{G}_c]^T$  is invertible such that  $\mathbf{x}_c$  may be expressed as a function of  $\mathbf{w}_c$ ,  $\mathbf{r}_c$  and  $\mathbf{y}_c$ , then so can  $\rho_c$ . It should be noted that, in these circumstances, the direct formulation, (17), is related to the velocity-based formulation, (19), by an *algebraic* transformation. These issues are discussed in detail in

<sup>1</sup> It is emphasised that this operation is purely formal in nature: no unstable pole-zero cancellation occurs within the implemented controller.

Leith & Leithead (1997a) and the implementation approaches discussed there, whilst developed in the context of the conventional gain-scheduling design approach, may be readily extended to the class of controllers considered here.

### Example 1

Consider the first order plant with dynamics

$$\dot{x} = G(r - 10x), \quad y = x \quad (21)$$

where  $G(s) = \tanh(s) + 0.01s$ . The requirement is to design a controller such that the closed-loop system has a rise time of around 0.3 seconds with less than 25% overshoot in response to demanded step changes in  $y$  of magnitude less than 100 units. At an equilibrium operating point,  $(r_o, x_o, y_o)$ ,

$$G(r_o - 10x_o) = 0 \quad (22)$$

which requires that

$$r_o - 10x_o = 0 \quad (23)$$

Hence, the series expansion linearisation of (21) relative to the equilibrium operating point,  $(r_o, x_o, y_o)$ , is

$$\delta\dot{x} = -10\nabla G(0)\delta x + \nabla G(0)\delta r, \quad \delta y = \delta x \quad (24)$$

$$\delta r = r - r_o, \delta x = x - x_o, y = \delta y + y_o \quad (25)$$

Since the first derivative of the nonlinear function,  $G$ , is

$$\nabla G(s) = 1.01 - \tanh^2 s \quad (26)$$

the series expansion linearisation at an equilibrium operating point may be reformulated as

$$\delta\dot{x} = -10.1\delta x + 1.01\delta r, \quad \delta y = \delta x \quad (27)$$

$$\delta r = r - r_o, \delta x = x - x_o, y = \delta y + y_o \quad (28)$$

Hence, based on the conventional series expansion linearisation at an equilibrium operating point, an appropriate local controller is the PI-type controller

$$\begin{bmatrix} \delta\dot{x}_{c_1} \\ \delta\dot{x}_{c_2} \end{bmatrix} = \begin{bmatrix} -50 & 0 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} \delta x_{c_1} \\ \delta x_{c_2} \end{bmatrix} + \begin{bmatrix} 50 \\ 0 \end{bmatrix} \delta e, \quad \delta r = \begin{bmatrix} K_0 & K_1 \end{bmatrix} \begin{bmatrix} \delta x_{c_1} \\ \delta x_{c_2} \end{bmatrix} \quad (29)$$

$$\delta e = e - e_o, r = \delta r + r_o \quad (30)$$

with  $e = y_{ref} - y$ ,  $K_0 = 3.86$ ,  $K_1 = 100.0$ . The transfer function of the controller, (29), is  $\left(K_0 + \frac{K_1}{s}\right) \frac{50}{s+50}$ .

The Bode plot of the closed-loop transfer function obtained by combining (27) and (29) is depicted in figure 2.

The dynamics, (27), are the same at every equilibrium operating point and so the controller, (29), may be employed at every equilibrium operating point. Owing to the integral action in the controller,  $e_o$  is zero and  $r_o$  is implicitly generated by the feedback. Hence, on the basis of the family of linearisations at the equilibrium operating points, a linear PI-type controller seems to be appropriate; namely,

$$\begin{bmatrix} \dot{x}_{c_1} \\ \dot{x}_{c_2} \end{bmatrix} = \begin{bmatrix} -50 & 0 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} x_{c_1} \\ x_{c_2} \end{bmatrix} + \begin{bmatrix} 50 \\ 0 \end{bmatrix} e, \quad r = \begin{bmatrix} K_0 & K_1 \end{bmatrix} \begin{bmatrix} x_{c_1} \\ x_{c_2} \end{bmatrix} \quad (31)$$

The step response of the closed loop system obtained by combining the nonlinear plant, (21), with the linear controller, (31), is depicted in figure 3. Evidently, the linear controller does not achieve the required performance. Indeed, simulation results indicate that, for step demands greater than approximately 0.3 units, this controller is unable to satisfy the overshoot requirements.

In order to incorporate information about the plant dynamics at non-equilibrium operating points into the controller design, reformulate the nonlinear plant, (21), by differentiating, as

$$\dot{w} = -10\nabla G(r - 10x)w + \nabla G(r - 10x)\dot{r}, \quad \dot{y} = w \quad (32)$$

The velocity-based linearisation family associated with the nonlinear plant, (21), consists of the frozen forms of (32) obtained when  $r$  and  $x$  are constant,

$$\hat{w} = -10\nabla G(r_1 - 10x_1)\hat{w} + \nabla G(r_1 - 10x_1)\dot{r}, \quad \dot{y} = \hat{w} \quad (33)$$

The required velocity-based linearisation family of the controller is determined by using linear methods to design a local controller for each of the members of the plant velocity-based linearisation family. Employing a PI-type controller structure once again, consider the linear controller family

$$\begin{bmatrix} \hat{w}_{c_1} \\ \hat{w}_{c_2} \end{bmatrix} = \begin{bmatrix} -50 & 0 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} \hat{w}_{c_1} \\ \hat{w}_{c_2} \end{bmatrix} + \begin{bmatrix} 50 \\ 0 \end{bmatrix} \dot{e}, \quad \dot{r} = \begin{bmatrix} K_0 & K_1 \end{bmatrix} \begin{bmatrix} \hat{w}_{c_1} \\ \hat{w}_{c_2} \end{bmatrix} \quad (34)$$

where  $K_o = -10.0 + 14.0/\sqrt{G}(r_1 - 10x_1)$  and  $K_I = 100.0/\sqrt{G}(r_1 - 10x_1)$ . At equilibrium operating points, the members of the linear controller family correspond to the controller dynamics, (31), determined previously. However, at non-equilibrium operating points, where  $r_1 - 10x_1$  is non-zero, the gains  $K_o$  and  $K_I$  are now different from their equilibrium values and are designed to compensate for the variation in the dynamics of the members of the plant velocity-based linearisation family. Since the controller contains integral action, a nonlinear controller with the velocity-based linearisation family, (34), can be obtained by directly implementing the velocity form of the controller, see figure 4. The step response of the closed-loop system thereby obtained is depicted in figure 5. It can be seen that the performance requirements are met for the full range of step demands.

## 5. Plants satisfying the extended local linear equivalence condition

Assume that the plant dynamics are of the form

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{r} + \mathbf{f}(\boldsymbol{\rho}), \quad \mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{r} + \mathbf{g}(\boldsymbol{\rho}) \quad (35)$$

where,  $\mathbf{r} \in \mathfrak{R}^m$ ,  $\mathbf{y} \in \mathfrak{R}^p$ ,  $\mathbf{x} \in \mathfrak{R}^n$ ,  $\boldsymbol{\rho}(\mathbf{x}, \mathbf{r}) \in \mathfrak{R}^q$ ,  $\mathbf{A}$ ,  $\mathbf{B}$ ,  $\mathbf{C}$ ,  $\mathbf{D}$  are constant matrices,  $\mathbf{f}(\bullet)$  and  $\mathbf{g}(\bullet)$  are differentiable nonlinear functions and  $\nabla_{\mathbf{x}}\boldsymbol{\rho}$ ,  $\nabla_{\mathbf{r}}\boldsymbol{\rho}$  are functions of  $\boldsymbol{\rho}$  alone. In addition, assume that  $\boldsymbol{\rho}$  minimally parameterises the locus of equilibrium operating points,  $(\mathbf{x}_o, \mathbf{r}_o)$ , of the plant. The variable,  $\boldsymbol{\rho}(\mathbf{x}, \mathbf{r})$ , equals the constant value, say  $\boldsymbol{\rho}_1$ , upon a surface of co-dimension  $q$  in  $\Phi$  and  $\nabla_{\mathbf{x}}\boldsymbol{\rho}$  and  $\nabla_{\mathbf{r}}\boldsymbol{\rho}$  are constant over each surface. Hence, the normal to each surface is identical at every point on the surface and each surface is, therefore, affine. Moreover, to ensure that  $\boldsymbol{\rho}$  is a unique function of  $\mathbf{x}$  and  $\mathbf{r}$ , these surfaces must be parallel for all  $\boldsymbol{\rho}$ . Consequently, it may in fact be assumed, without loss of generality, that  $\nabla_{\mathbf{x}}\boldsymbol{\rho}$  and  $\nabla_{\mathbf{r}}\boldsymbol{\rho}$  are constant.

The velocity-based linearisation family associated with the nonlinear plant, (35), is

$$\begin{aligned} \dot{\hat{\mathbf{x}}} &= \hat{\mathbf{w}} \\ \dot{\hat{\mathbf{w}}} &= (\mathbf{A} + \nabla_{\boldsymbol{\rho}}\mathbf{f}(\boldsymbol{\rho}_1)\nabla_{\mathbf{x}}\boldsymbol{\rho})\hat{\mathbf{w}} + (\mathbf{B} + \nabla_{\boldsymbol{\rho}}\mathbf{f}(\boldsymbol{\rho}_1)\nabla_{\mathbf{r}}\boldsymbol{\rho})\dot{\mathbf{r}} \\ \dot{\hat{\mathbf{y}}} &= (\mathbf{C} + \nabla_{\boldsymbol{\rho}}\mathbf{g}(\boldsymbol{\rho}_1)\nabla_{\mathbf{x}}\boldsymbol{\rho})\hat{\mathbf{w}} + (\mathbf{C} + \nabla_{\boldsymbol{\rho}}\mathbf{g}(\boldsymbol{\rho}_1)\nabla_{\mathbf{r}}\boldsymbol{\rho})\dot{\mathbf{r}} \end{aligned} \quad (36)$$

The members of the family are parameterised by  $\boldsymbol{\rho}$ ; that is, the velocity-based linearisation is the same at operating points lying on a surface of constant  $\boldsymbol{\rho}$ .

The union of the surfaces of constant  $\boldsymbol{\rho}$  covers the entire operating space,  $\Phi$ , and, since  $\boldsymbol{\rho}$  minimally parameterises the locus of equilibrium operating points, each surface of constant  $\boldsymbol{\rho}$  intersects the equilibrium locus at a unique point. Each operating point in  $\Phi$  is, therefore, associated, *via* a surface of constant  $\boldsymbol{\rho}$ , with an equilibrium operating point which has the same velocity-based linearisation. Hence,  $\boldsymbol{\rho}$  may be interpreted as the ‘‘scheduling variable’’ associated with the plant, in the sense that, at any operating point in  $\Phi$ ,  $\boldsymbol{\rho}$  indicates which member of the family of equilibrium linearisations of the plant is valid.

For the class of plants considered here, the velocity-based linearisation of the plant at the equilibrium operating points, when taken together with the scheduling variable,  $\boldsymbol{\rho}$ , completely determines the velocity-based linearisation family associated with the plant. The conventional series expansion linearisation of the nonlinear plant, (35), relative to the equilibrium operating point,  $(\mathbf{x}_o, \mathbf{r}_o, \mathbf{y}_o)$ , at which  $\boldsymbol{\rho}$  equals  $\boldsymbol{\rho}_1$ , is

$$\begin{aligned} \delta\dot{\mathbf{x}} &= (\mathbf{A} + \nabla_{\boldsymbol{\rho}}\mathbf{f}(\boldsymbol{\rho}_1)\nabla_{\mathbf{x}}\boldsymbol{\rho})\delta\mathbf{x} + (\mathbf{B} + \nabla_{\boldsymbol{\rho}}\mathbf{f}(\boldsymbol{\rho}_1)\nabla_{\mathbf{r}}\boldsymbol{\rho})\delta\mathbf{r} \\ \delta\dot{\mathbf{y}} &= (\mathbf{C} + \nabla_{\boldsymbol{\rho}}\mathbf{g}(\boldsymbol{\rho}_1)\nabla_{\mathbf{x}}\boldsymbol{\rho})\delta\mathbf{x} + (\mathbf{C} + \nabla_{\boldsymbol{\rho}}\mathbf{g}(\boldsymbol{\rho}_1)\nabla_{\mathbf{r}}\boldsymbol{\rho})\delta\mathbf{r} \\ \delta\mathbf{x} &= \mathbf{x} - \mathbf{x}_o(\boldsymbol{\rho}_1), \quad \delta\mathbf{r} = \mathbf{r} - \mathbf{r}_o(\boldsymbol{\rho}_1), \quad \mathbf{y} = \delta\mathbf{y} + \mathbf{y}_o(\boldsymbol{\rho}_1) \end{aligned} \quad (37)$$

It can be seen that, whilst the state, input and output differ, the series expansion linearisation at an equilibrium operating point, (37), has the same form as the corresponding velocity-based linearisation, (36), at that equilibrium operating point. Hence, provided some care is taken, it is possible to employ either type of linearisation to characterise the plant dynamics. Recall that the conventional gain-scheduling design approach utilises the series expansion linearisations of the plant at the equilibrium operating points. Clearly, for the class of plants considered here, the potential exists for employing the conventional gain-scheduling design approach to obtain a nonlinear controller which is valid not only in the vicinity of the equilibrium operating points but throughout the operating envelope. In that context, the present framework provides rigorous insight into the role of the choice of controller scheduling variable and realisation: the nature of the velocity-based linearisation family associated with the resulting nonlinear controller and so the controller dynamics, particularly at non-equilibrium



operating points, are influenced by them. Clearly, the controller dynamics need to be compatible with the corresponding plant dynamics at non-equilibrium operating points.

The class of plants considered here consists precisely of those satisfying the extended local linear equivalence condition originally proposed by Leith & Leithead (1996,1997a) in the context of determining appropriate realisations for gain-scheduled controllers. It is natural to select a controller with a nonlinear structure which reflects that of the plant and so it is attractive to require, for the class of plants considered here, that the controller also satisfies the extended local linear equivalence condition. In these circumstances, the gain-scheduling design procedure proposed in section 4 specialises to:

1. Determine the linearisation of the plant at each equilibrium operating point. In addition, determine the scheduling variable,  $\rho$ , associated with the plant; typically, this information might be derived from physical understanding of the plant.
2. Select an appropriate scheduling variable for the controller. An obvious choice is to employ the plant scheduling variable,  $\rho$ , or an estimate thereof. However, an alternative choice might be suggested by other considerations.
3. Design a suitable linear controller for each member of the family of plant equilibrium linearisations. In order to ensure the existence of a corresponding controller satisfying the extended local linear equivalence condition, the variations between the members of the resulting family of linear controllers should be compatible with the choice of scheduling variable. The compatibility requirement is not overly restrictive in general, see Leith & Leithead (1997a).
4. Realise a nonlinear controller, compatible with the choice of scheduling variable and family of linear controllers determined at steps 2 and 3, which satisfies the extended local linear equivalence condition (see Leith & Leithead 1996, 1997a).

#### Example 2

Consider the nonlinear plant, depicted in figure 6, with second-order dynamics

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & -a \\ 1 & -b \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} -a \\ -b \end{bmatrix} r + \begin{bmatrix} A(r) \\ B(r) \end{bmatrix}, \quad y = \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + r \quad (38)$$

where  $A, B$  are differentiable nonlinear functions with  $sA(s) > 0, \nabla B(s) > 0 \forall s \in \mathfrak{R}$ . The nonlinear plant, (38), is of the form, (35), with  $\rho$  equal to  $r$ , and satisfies the extended local linear equivalence condition. The velocity-based linearisation of the nonlinear plant at the general operating point,  $(x_{11}, x_{21}, r_1)$ , is

$$\begin{bmatrix} \dot{\hat{x}}_1 \\ \dot{\hat{x}}_2 \end{bmatrix} = \begin{bmatrix} \hat{w}_1 \\ \hat{w}_2 \end{bmatrix}, \quad \begin{bmatrix} \dot{\hat{w}}_1 \\ \dot{\hat{w}}_2 \end{bmatrix} = \begin{bmatrix} 0 & -a \\ 1 & -b \end{bmatrix} \begin{bmatrix} \hat{w}_1 \\ \hat{w}_2 \end{bmatrix} + \begin{bmatrix} \nabla A(r_1) - a \\ \nabla B(r_1) - b \end{bmatrix} \dot{r}, \quad \dot{y} = \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} \hat{w}_1 \\ \hat{w}_2 \end{bmatrix} + \dot{r} \quad (39)$$

At an equilibrium operating point,  $(x_{10}, x_{20}, r_0)$ ,

$$x_{10} = \frac{b}{a} A(r_0) - B(r_0), x_{20} = \frac{1}{a} A(r_0) - r_0 \quad (40)$$

and so the equilibrium operating points may be parameterised by  $r$ . The velocity-based linearisation at the equilibrium operating point at which  $r$  equals  $r_0$  is

$$\begin{bmatrix} \dot{\hat{x}}_1 \\ \dot{\hat{x}}_2 \end{bmatrix} = \begin{bmatrix} \hat{w}_1 \\ \hat{w}_2 \end{bmatrix}, \quad \begin{bmatrix} \dot{\hat{w}}_1 \\ \dot{\hat{w}}_2 \end{bmatrix} = \begin{bmatrix} 0 & -a \\ 1 & -b \end{bmatrix} \begin{bmatrix} \hat{w}_1 \\ \hat{w}_2 \end{bmatrix} + \begin{bmatrix} \nabla A(r_0) - a \\ \nabla B(r_0) - b \end{bmatrix} \dot{r}, \quad \dot{y} = \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} \hat{w}_1 \\ \hat{w}_2 \end{bmatrix} + \dot{r} \quad (41)$$

Hence, the velocity-based linearisation at the general operating point,  $(x_{11}, x_{21}, r_1)$ , corresponds precisely to the velocity-based linearisation at the equilibrium operating point at which  $r$  equals  $r_1$ . The series expansion linearisation at the equilibrium operating point at which  $r$  equals  $r_0$  is

$$\begin{bmatrix} \delta \dot{x}_1 \\ \delta \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & -a \\ 1 & -b \end{bmatrix} \begin{bmatrix} \delta x_1 \\ \delta x_2 \end{bmatrix} + \begin{bmatrix} \nabla A(r_0) - a \\ \nabla B(r_0) - b \end{bmatrix} \delta r, \quad \delta y = \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} \delta x_1 \\ \delta x_2 \end{bmatrix} + \delta r \quad (42)$$

$$\delta x_1 = x_1 - x_{10}, \delta x_2 = x_2 - x_{20}, \delta r = r - r_0, y = \delta y + y_0 \quad (43)$$

Whilst the state, input and output of (42) and (41) differ, it is clear that (42) has the same form as (41).

Employing the conventional gain-scheduling design approach, a linear controller is designed for each member of the family of series expansion linearisations, (42), at the equilibrium operating points. Consider the family of linear controllers

$$\begin{bmatrix} \delta \dot{x}_{c_1} \\ \delta \dot{x}_{c_2} \\ \delta \dot{x}_{c_3} \end{bmatrix} = \begin{bmatrix} 0 & 0 & -\nabla A(r_0) \\ 1 & 0 & -\nabla B(r_0) \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \delta x_{c_1} \\ \delta x_{c_2} \\ \delta x_{c_3} \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \delta e, \quad \delta r = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \delta x_{c_1} \\ \delta x_{c_2} \\ \delta x_{c_3} \end{bmatrix}, \quad \delta e = \delta y_{ref} - \delta y \quad (44)$$

The closed-loop transfer function obtained by combining (42) and (44) is

$$\frac{1}{s^3 + bs^2 + as + 1} \quad (45)$$

The plant input,  $r$ , is a natural choice of controller scheduling variable since it is also the plant ‘‘scheduling variable’’. Employing one of the approaches proposed in Leith & Leithead (1996, 1997a), an appropriate nonlinear controller realisation is

$$\begin{bmatrix} \dot{w}_{c_1} \\ \dot{w}_{c_2} \\ \dot{w}_{c_3} \end{bmatrix} = \begin{bmatrix} 0 & -\nabla A(r) & 0 \\ 1 & -\nabla B(r) & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} w_{c_1} \\ w_{c_2} \\ w_{c_3} \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} e, \quad r = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} w_{c_1} \\ w_{c_2} \\ w_{c_3} \end{bmatrix}, \quad e = y_{ref} - y \quad (46)$$

It may be shown that the nonlinear controller, (46), satisfies the extended local linear equivalence condition with the scheduling variable,  $r$ , and the family of series expansion linearisations relative to the equilibrium operating points, (44) (Leith & Leithead 1996, 1997a). The form of the controller realisation is clearly directly related to that of the linear family, (44).

In Leith & Leithead (1996, 1997a), controller realisations of the form, (46), (and more general forms of realisation) are derived to minimise the slow variation conditions inherent to the conventional gain-scheduling approach. However, in the more general framework considered here, it is evident that (46) is simply a third-order example, with the nonlinearity purely a function of  $r$ , of the velocity-based controller, (19). Hence, the velocity-based linearisation family of the controller consists of the frozen forms of (46) and the velocity-based linearisation family associated with the closed-loop system is obtained by combining (39) and (46) (or, alternatively, (41) and (46)). The nonlinear controller, (46), although designed by the conventional gain-scheduling approach of using only the plant equilibrium linearisations, is valid throughout the operating envelope, not just in the vicinity of the equilibrium operating points. Indeed, it is straightforward to show that the combined plant and controller dynamics consist of linear dynamics with transfer function, (45), and an exponentially stable, unobservable nonlinear component.

## 6. Conclusions

In this paper, the velocity-based linearisation framework is employed to develop a novel rigorous approach to gain-scheduling design. The approach addresses many of the deficiencies of the conventional gain-scheduling design approach. Whilst retaining continuity with linear methods, which is central to the conventional gain-scheduling methodology, the approach

- Enables knowledge concerning the plant dynamics at *non-equilibrium* operating points to be incorporated directly and rigorously into the controller design. In contrast, the conventional gain-scheduling approach utilises only equilibrium linearisations of the plant and so is inherently restricted to situations where only rather small and/or slowly-varying control demands and disturbances are encountered.
- Enables transient performance requirements, instead of stability alone, to be considered directly during the controller design.
- Encompasses both smooth and non-smooth scheduling within the same analysis and design framework.
- Employs a streamlined analysis and design framework which utilises a *single* type of linearisation throughout. Consequently, in comparison with the conventional gain-scheduling approach, the design procedure proposed here is both straightforward and conceptually appealing.

These benefits stem directly from adoption of the velocity-based linearisation framework for the analysis of nonlinear systems.

## Acknowledgement

D.J.Leith gratefully acknowledges the generous support provided by the Royal Society for the work presented.

## References

- ASTROM, K.J., WITTENMARK, B., 1989, *Adaptive Control*. (Addison-Wesley).
- HYDE, R.A., GLOVER, K., 1991, A Comparison of Different Scheduling Techniques for  $H_\infty$  Controllers. *Proceedings of the Institute of Measurement & Control Symposium on Robust Control*, Cambridge.
- HYDE, R.A., GLOVER, K., 1993, The Application of Scheduled  $H_\infty$  Controllers to a VSTOL Aircraft. *IEEE Transactions on Automatic Control*, **38**, 1021-1039.
- LEITH, D.J., LEITHEAD, W.E., 1996, Appropriate Realisation of Gain Scheduled Controllers with Application to Wind Turbine Regulation. *International Journal of Control*, **65**, 223-248.
- LEITH, D.J., LEITHEAD, W.E., 1997a, Appropriate Realisation of MIMO Gain Scheduled Controllers. *International Journal of Control*, to appear.
- LEITH, D.J., LEITHEAD, W.E., 1997b, Dynamic Analysis of Gain-Scheduled & Nonlinear Systems Using Velocity-Based Linearisation Families. *Proceedings of Workshop on 'Multiple Model Methods in Modelling & Control'*, Trondheim, Norway.
- LEITH, D.J., LEITHEAD, W.E., 1997c, Gain-Scheduled & Nonlinear Systems: Dynamic Analysis by Velocity-Based Linearisation Families. *International Journal of Control*, submitted.
- RUGH, W.J., 1991, Analytical Framework for Gain-Scheduling. *IEEE Control Systems Magazine*, **11**, 79-84.
- SHAMMA, J.S., ATHANS, M., 1990, Analysis of Gain Scheduled Control for Nonlinear Plants. *IEEE Transactions on Automatic Control*, **35**, 898-907.

## Appendix

Consider the nonlinear system with inputs,  $\mathbf{r}$  and  $\mathbf{z}$ ,

$$\dot{\mathbf{x}} = \mathbf{F}(\mathbf{x}, \mathbf{r}, \mathbf{z}), \quad \mathbf{y} = \mathbf{G}(\mathbf{x}, \mathbf{r}, \mathbf{z}) \quad (47)$$

Transforming into velocity-based form, (47) is equivalent to

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{w} \\ \dot{\mathbf{w}} &= \nabla_{\mathbf{x}}\mathbf{F}(\mathbf{x}, \mathbf{r}, \mathbf{z})\mathbf{w} + \nabla_{\mathbf{z}}\mathbf{F}(\mathbf{x}, \mathbf{r}, \mathbf{z})\dot{\mathbf{z}} + \nabla_{\mathbf{r}}\mathbf{F}(\mathbf{x}, \mathbf{r}, \mathbf{z})\dot{\mathbf{r}} \\ \dot{\mathbf{y}} &= \nabla_{\mathbf{x}}\mathbf{G}(\mathbf{x}, \mathbf{r}, \mathbf{z})\mathbf{w} + \nabla_{\mathbf{z}}\mathbf{G}(\mathbf{x}, \mathbf{r}, \mathbf{z})\dot{\mathbf{z}} + \nabla_{\mathbf{r}}\mathbf{G}(\mathbf{x}, \mathbf{r}, \mathbf{z})\dot{\mathbf{r}} \end{aligned} \quad (48)$$

Assuming that

$$\mathbf{y} = \mathbf{G}(\mathbf{x}, \mathbf{r}, \mathbf{y}) \quad (49)$$

has a suitable solution

$$\mathbf{y} = \mathbf{N}(\mathbf{x}, \mathbf{r}) \quad (50)$$

the system, (47), is enclosed in a feedback loop by setting  $\mathbf{z} = \mathbf{y}$ . The resulting closed-loop system is

$$\dot{\mathbf{x}} = \mathbf{M}(\mathbf{x}, \mathbf{r}), \quad \mathbf{y} = \mathbf{N}(\mathbf{x}, \mathbf{r}) \quad (51)$$

with

$$\mathbf{M}(\mathbf{x}, \mathbf{r}) = \mathbf{F}(\mathbf{x}, \mathbf{r}, \mathbf{N}(\mathbf{x}, \mathbf{r})) \quad (52)$$

Transforming into velocity-based form, (51) is equivalent to

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{w} \\ \dot{\mathbf{w}} &= \nabla_{\mathbf{x}}\mathbf{M}(\mathbf{x}, \mathbf{r})\mathbf{w} + \nabla_{\mathbf{r}}\mathbf{M}(\mathbf{x}, \mathbf{r})\dot{\mathbf{r}} \\ \dot{\mathbf{y}} &= \nabla_{\mathbf{x}}\mathbf{N}(\mathbf{x}, \mathbf{r})\mathbf{w} + \nabla_{\mathbf{r}}\mathbf{N}(\mathbf{x}, \mathbf{r})\dot{\mathbf{r}} \end{aligned} \quad (53)$$

Combining (49) and (50)

$$\mathbf{N}(\mathbf{x}, \mathbf{r}) = \mathbf{G}(\mathbf{x}, \mathbf{r}, \mathbf{N}(\mathbf{x}, \mathbf{r})) \quad (54)$$

Hence,

$$\begin{aligned} \nabla_{\mathbf{x}}\mathbf{M}(\mathbf{x}, \mathbf{r}) &= \nabla_{\mathbf{x}}\mathbf{F}(\mathbf{x}, \mathbf{r}, \mathbf{N}(\mathbf{x}, \mathbf{r})) + \nabla_{\mathbf{z}}\mathbf{F}(\mathbf{x}, \mathbf{r}, \mathbf{N}(\mathbf{x}, \mathbf{r}))\nabla_{\mathbf{x}}\mathbf{N}(\mathbf{x}, \mathbf{r}) \\ \nabla_{\mathbf{r}}\mathbf{M}(\mathbf{x}, \mathbf{r}) &= \nabla_{\mathbf{r}}\mathbf{F}(\mathbf{x}, \mathbf{r}, \mathbf{N}(\mathbf{x}, \mathbf{r})) + \nabla_{\mathbf{z}}\mathbf{F}(\mathbf{x}, \mathbf{r}, \mathbf{N}(\mathbf{x}, \mathbf{r}))\nabla_{\mathbf{r}}\mathbf{N}(\mathbf{x}, \mathbf{r}) \\ \nabla_{\mathbf{x}}\mathbf{N}(\mathbf{x}, \mathbf{r}) &= \nabla_{\mathbf{x}}\mathbf{G}(\mathbf{x}, \mathbf{r}, \mathbf{N}(\mathbf{x}, \mathbf{r})) + \nabla_{\mathbf{z}}\mathbf{G}(\mathbf{x}, \mathbf{r}, \mathbf{N}(\mathbf{x}, \mathbf{r}))\nabla_{\mathbf{x}}\mathbf{N}(\mathbf{x}, \mathbf{r}) \\ \nabla_{\mathbf{r}}\mathbf{N}(\mathbf{x}, \mathbf{r}) &= \nabla_{\mathbf{r}}\mathbf{G}(\mathbf{x}, \mathbf{r}, \mathbf{N}(\mathbf{x}, \mathbf{r})) + \nabla_{\mathbf{z}}\mathbf{G}(\mathbf{x}, \mathbf{r}, \mathbf{N}(\mathbf{x}, \mathbf{r}))\nabla_{\mathbf{r}}\mathbf{N}(\mathbf{x}, \mathbf{r}) \end{aligned} \quad (55)$$

and, by substituting (55) into (53), the closed-loop system, (53), can be directly reformulated as

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{w} \\ \dot{\mathbf{w}} &= \nabla_{\mathbf{x}}\mathbf{F}(\mathbf{x}, \mathbf{r}, \mathbf{z})\mathbf{w} + \nabla_{\mathbf{z}}\mathbf{F}(\mathbf{x}, \mathbf{r}, \mathbf{z})\dot{\mathbf{z}} + \nabla_{\mathbf{r}}\mathbf{F}(\mathbf{x}, \mathbf{r}, \mathbf{z})\dot{\mathbf{r}} \\ \dot{\mathbf{y}} &= \nabla_{\mathbf{x}}\mathbf{G}(\mathbf{x}, \mathbf{r}, \mathbf{z})\mathbf{w} + \nabla_{\mathbf{z}}\mathbf{G}(\mathbf{x}, \mathbf{r}, \mathbf{z})\dot{\mathbf{z}} + \nabla_{\mathbf{r}}\mathbf{G}(\mathbf{x}, \mathbf{r}, \mathbf{z})\dot{\mathbf{r}} \\ \mathbf{z} &= \mathbf{y} = \mathbf{N}(\mathbf{x}, \mathbf{r}) \end{aligned} \quad (56)$$

Since  $\mathbf{N}(\mathbf{x}, \mathbf{r})$  satisfies (54), it is clear that (56) is the system obtained when the system, (48), is enclosed in a feedback loop by setting  $\mathbf{z} = \mathbf{y}$ . It follows that the velocity-based form of a closed-loop system is identical to the system obtained by enclosing the velocity-based form of the open-loop system in a feedback loop.

Consider, now, the nonlinear system

$$\dot{\mathbf{x}}_1 = \mathbf{F}_1(\mathbf{x}_1, \mathbf{r}_1, \mathbf{z}_1), \quad \mathbf{y}_1 = \mathbf{G}_1(\mathbf{x}_1, \mathbf{r}_1, \mathbf{z}_1) \quad (57)$$

for which the velocity-based form is

$$\begin{aligned} \dot{\mathbf{x}}_1 &= \mathbf{w}_1 \\ \dot{\mathbf{w}}_1 &= \nabla_{\mathbf{x}_1} \mathbf{F}_1(\mathbf{x}_1, \mathbf{r}_1, \mathbf{z}_1) \mathbf{w}_1 + \nabla_{\mathbf{z}_1} \mathbf{F}_1(\mathbf{x}_1, \mathbf{r}_1, \mathbf{z}_1) \dot{\mathbf{z}}_1 + \nabla_{\mathbf{r}_1} \mathbf{F}_1(\mathbf{x}_1, \mathbf{r}_1, \mathbf{z}_1) \dot{\mathbf{r}}_1 \\ \dot{\mathbf{y}}_1 &= \nabla_{\mathbf{x}_1} \mathbf{G}_1(\mathbf{x}_1, \mathbf{r}_1, \mathbf{z}_1) \mathbf{w}_1 + \nabla_{\mathbf{z}_1} \mathbf{G}_1(\mathbf{x}_1, \mathbf{r}_1, \mathbf{z}_1) \dot{\mathbf{z}}_1 + \nabla_{\mathbf{r}_1} \mathbf{G}_1(\mathbf{x}_1, \mathbf{r}_1, \mathbf{z}_1) \dot{\mathbf{r}}_1 \end{aligned} \quad (58)$$

and the nonlinear system

$$\dot{\mathbf{x}}_2 = \mathbf{F}_2(\mathbf{x}_2, \mathbf{r}_2, \mathbf{z}_2), \quad \mathbf{y}_2 = \mathbf{G}_2(\mathbf{x}_2, \mathbf{r}_2, \mathbf{z}_2) \quad (59)$$

for which the velocity-based form is

$$\begin{aligned} \dot{\mathbf{x}}_2 &= \mathbf{w}_2 \\ \dot{\mathbf{w}}_2 &= \nabla_{\mathbf{x}_2} \mathbf{F}_2(\mathbf{x}_2, \mathbf{r}_2, \mathbf{z}_2) \mathbf{w}_2 + \nabla_{\mathbf{z}_2} \mathbf{F}_2(\mathbf{x}_2, \mathbf{r}_2, \mathbf{z}_2) \dot{\mathbf{z}}_2 + \nabla_{\mathbf{r}_2} \mathbf{F}_2(\mathbf{x}_2, \mathbf{r}_2, \mathbf{z}_2) \dot{\mathbf{r}}_2 \\ \dot{\mathbf{y}}_2 &= \nabla_{\mathbf{x}_2} \mathbf{G}_2(\mathbf{x}_2, \mathbf{r}_2, \mathbf{z}_2) \mathbf{w}_2 + \nabla_{\mathbf{z}_2} \mathbf{G}_2(\mathbf{x}_2, \mathbf{r}_2, \mathbf{z}_2) \dot{\mathbf{z}}_2 + \nabla_{\mathbf{r}_2} \mathbf{G}_2(\mathbf{x}_2, \mathbf{r}_2, \mathbf{z}_2) \dot{\mathbf{r}}_2 \end{aligned} \quad (60)$$

The systems, (57) and (59), are cascaded together by setting  $\mathbf{z}_2 = \mathbf{y}_1$ . The resulting system is

$$\dot{\mathbf{x}} = \mathbf{F}(\mathbf{x}, \mathbf{r}, \mathbf{z}), \quad \mathbf{y} = \mathbf{G}(\mathbf{x}, \mathbf{r}, \mathbf{z}) \quad (61)$$

where

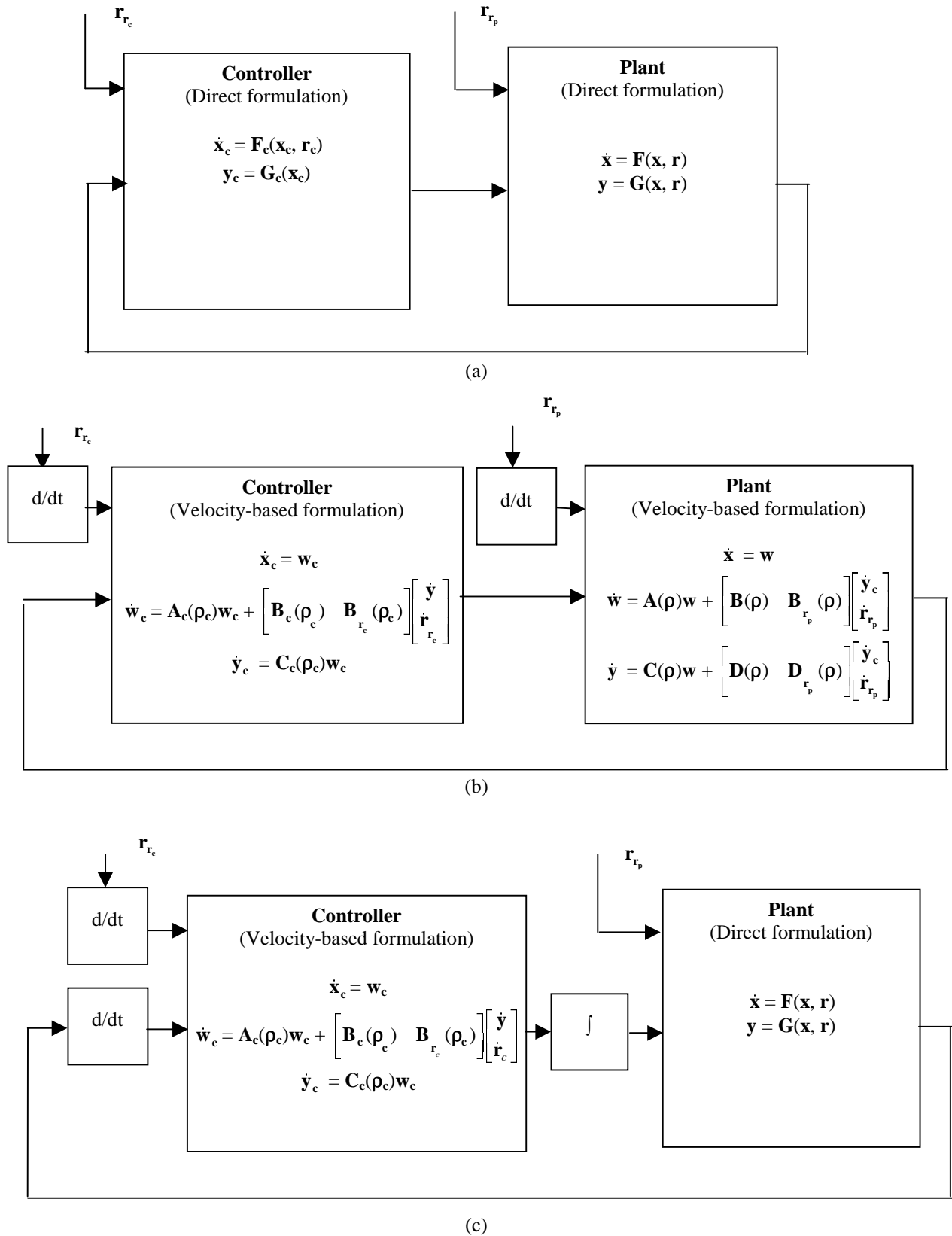
$$\begin{aligned} \mathbf{x} &= \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix}, \mathbf{r} = \begin{bmatrix} \mathbf{r}_1 \\ \mathbf{r}_2 \end{bmatrix}, \mathbf{z} = \mathbf{z}_1, \mathbf{y} = \mathbf{y}_2 \\ \mathbf{F}(\mathbf{x}, \mathbf{r}, \mathbf{z}) &= \begin{bmatrix} \mathbf{F}_1(\mathbf{x}_1, \mathbf{r}_1, \mathbf{z}_1) \\ \mathbf{F}_2(\mathbf{x}_2, \mathbf{r}_2, \mathbf{G}_1(\mathbf{x}_1, \mathbf{r}_1, \mathbf{z}_1)) \end{bmatrix}, \mathbf{G}(\mathbf{x}, \mathbf{r}, \mathbf{z}) = \mathbf{G}_2(\mathbf{x}_2, \mathbf{r}_2, \mathbf{G}_1(\mathbf{x}_1, \mathbf{r}_1, \mathbf{z}_1)) \end{aligned} \quad (62)$$

Transforming into velocity-based form, (61) is equivalent to

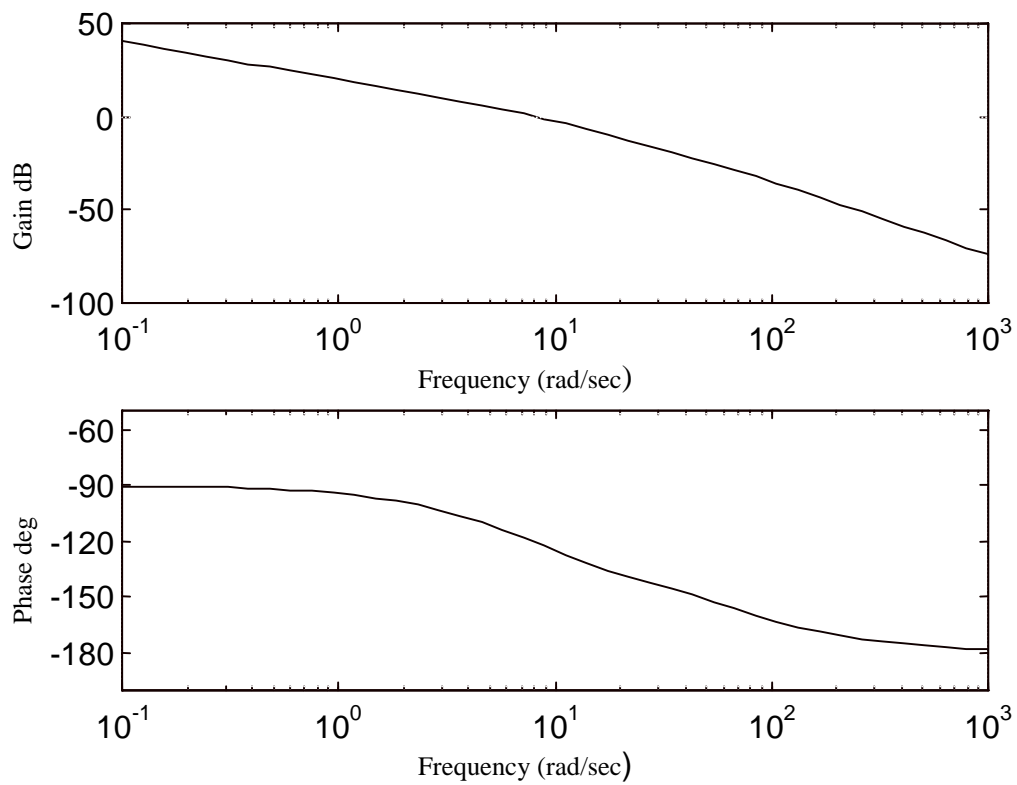
$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{w} = \begin{bmatrix} \mathbf{w}_1 \\ \mathbf{w}_2 \end{bmatrix} \\ \dot{\mathbf{w}} &= \begin{bmatrix} \dot{\mathbf{w}}_1 \\ \dot{\mathbf{w}}_2 \end{bmatrix} = \begin{bmatrix} \nabla_{\mathbf{x}_1} \mathbf{F}_1(\mathbf{x}_1, \mathbf{r}_1, \mathbf{z}_1) & 0 \\ \nabla_{\mathbf{z}_2} \mathbf{F}_2(\mathbf{x}_2, \mathbf{r}_2, \mathbf{z}_2) \nabla_{\mathbf{x}_1} \mathbf{G}_1(\mathbf{x}_1, \mathbf{r}_1, \mathbf{z}_1) & \nabla_{\mathbf{x}_2} \mathbf{F}_2(\mathbf{x}_2, \mathbf{r}_2, \mathbf{z}_2) \end{bmatrix} \begin{bmatrix} \mathbf{w}_1 \\ \mathbf{w}_2 \end{bmatrix} \\ &\quad + \begin{bmatrix} \nabla_{\mathbf{z}_1} \mathbf{F}_1(\mathbf{x}_1, \mathbf{r}_1, \mathbf{z}_1) \\ \nabla_{\mathbf{z}_2} \mathbf{F}_2(\mathbf{x}_2, \mathbf{r}_2, \mathbf{z}_2) \nabla_{\mathbf{z}_1} \mathbf{G}_1(\mathbf{x}_1, \mathbf{r}_1, \mathbf{z}_1) \end{bmatrix} \dot{\mathbf{z}}_1 \\ &\quad + \begin{bmatrix} \nabla_{\mathbf{r}_1} \mathbf{F}_1(\mathbf{x}_1, \mathbf{r}_1, \mathbf{z}_1) & \mathbf{0} \\ \nabla_{\mathbf{z}_2} \mathbf{F}_2(\mathbf{x}_2, \mathbf{r}_2, \mathbf{z}_2) \nabla_{\mathbf{r}_1} \mathbf{G}_1(\mathbf{x}_1, \mathbf{r}_1, \mathbf{z}_1) & \nabla_{\mathbf{r}_2} \mathbf{F}_2(\mathbf{x}_2, \mathbf{r}_2, \mathbf{z}_2) \end{bmatrix} \begin{bmatrix} \dot{\mathbf{r}}_1 \\ \dot{\mathbf{r}}_2 \end{bmatrix} \\ \dot{\mathbf{y}} &= \dot{\mathbf{y}}_2 = \begin{bmatrix} \nabla_{\mathbf{z}_2} \mathbf{G}_2(\mathbf{x}_2, \mathbf{r}_2, \mathbf{z}_2) \nabla_{\mathbf{x}_1} \mathbf{G}_1(\mathbf{x}_1, \mathbf{r}_1, \mathbf{z}_1) & \nabla_{\mathbf{x}_2} \mathbf{G}_2(\mathbf{x}_2, \mathbf{r}_2, \mathbf{z}_2) \end{bmatrix} \begin{bmatrix} \mathbf{w}_1 \\ \mathbf{w}_2 \end{bmatrix} \\ &\quad + \nabla_{\mathbf{z}_2} \mathbf{G}_2(\mathbf{x}_2, \mathbf{r}_2, \mathbf{z}_2) \nabla_{\mathbf{z}_1} \mathbf{G}_1(\mathbf{x}_1, \mathbf{r}_1, \mathbf{z}_1) \dot{\mathbf{z}}_1 \\ &\quad + \begin{bmatrix} \nabla_{\mathbf{z}_2} \mathbf{G}_2(\mathbf{x}_2, \mathbf{r}_2, \mathbf{z}_2) \nabla_{\mathbf{r}_1} \mathbf{G}_1(\mathbf{x}_1, \mathbf{r}_1, \mathbf{z}_1) & \nabla_{\mathbf{r}_2} \mathbf{G}_2(\mathbf{x}_2, \mathbf{r}_2, \mathbf{z}_2) \end{bmatrix} \begin{bmatrix} \dot{\mathbf{r}}_1 \\ \dot{\mathbf{r}}_2 \end{bmatrix} \end{aligned} \quad (63)$$

$$\mathbf{z}_2 = \mathbf{y}_1 = \mathbf{G}_1(\mathbf{x}_1, \mathbf{r}_1, \mathbf{z}_1)$$

Evidently, (63) is just the system obtained when the systems, (58) and (60), are cascaded together. It follows that the velocity-based form of a system consisting of two cascaded sub-systems is identical to the system obtained by cascading together the velocity-based forms of the two sub-systems. Furthermore, except for the output being  $\mathbf{y}_2$  rather than  $\dot{\mathbf{y}}_2$ , both are equivalent to velocity-based system, (58), cascaded together with the nonlinear system, (59).

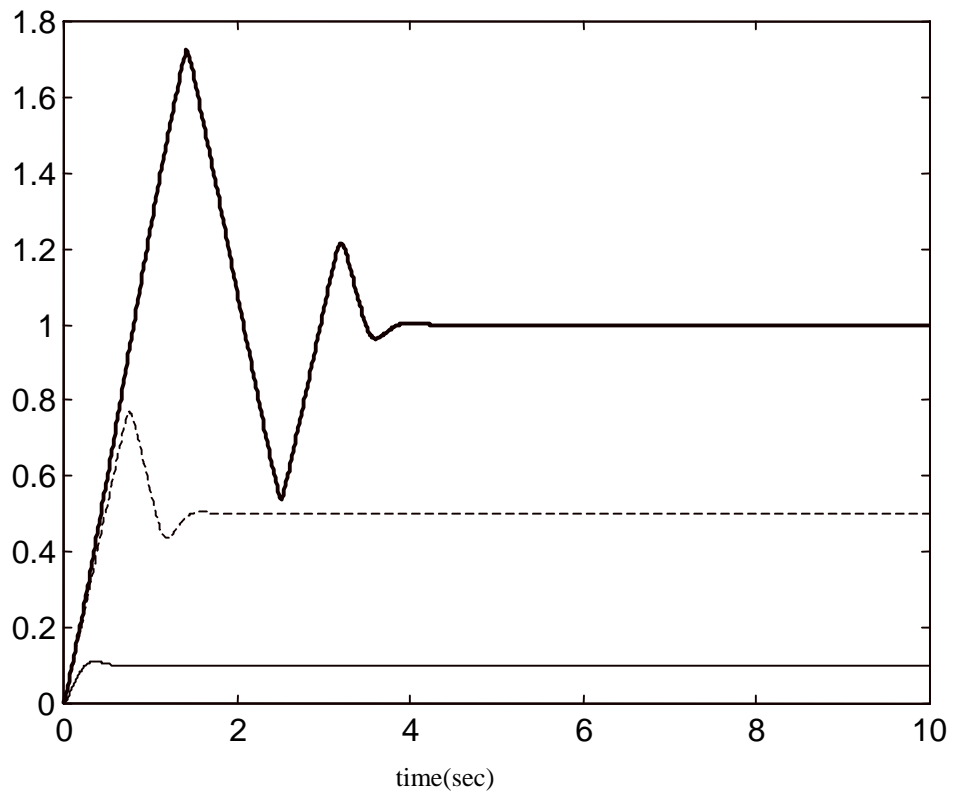


**Figure 1** Different formulations of nonlinear feedback system. Note, the differentiation operators are purely formal in nature.

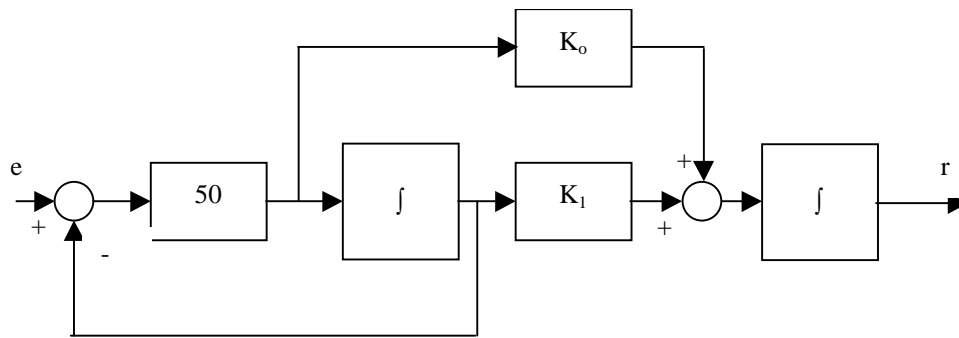


**Figure 2** Bode plot of open-loop transfer function at the equilibrium operating points (Example 1).

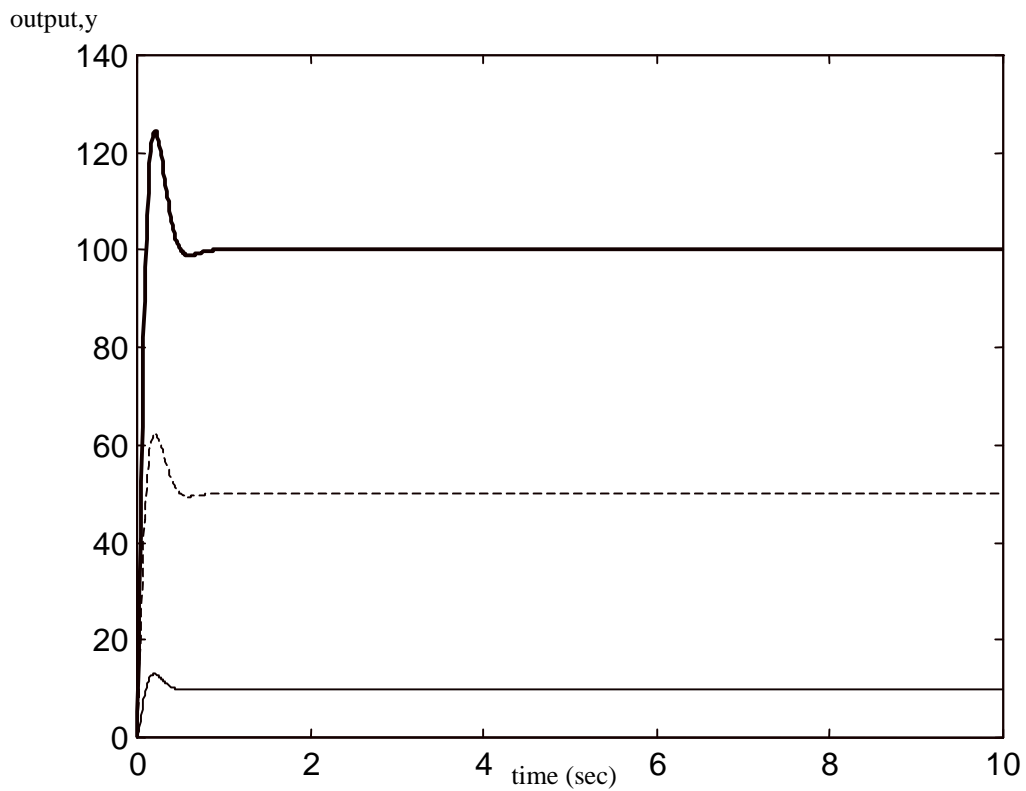
output, y



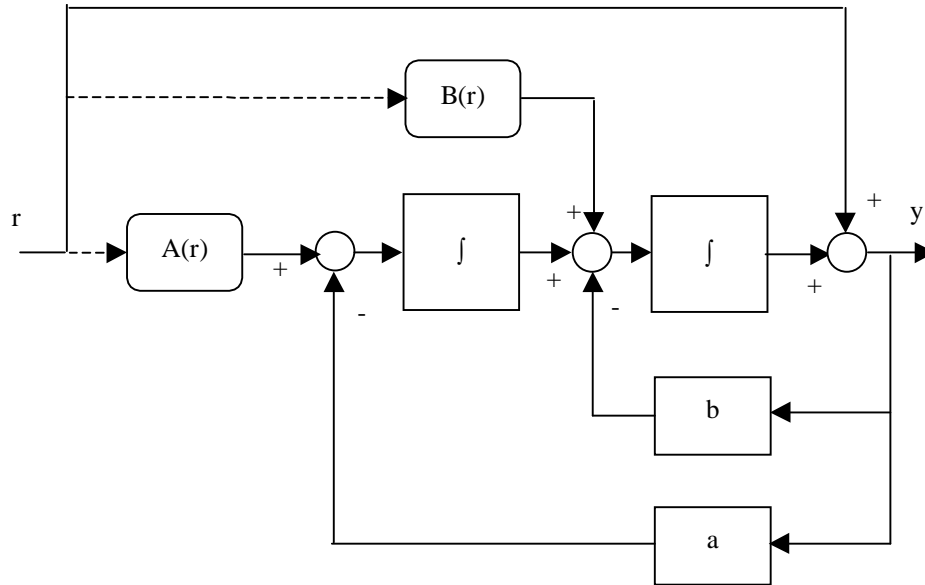
**Figure 3** Responses to steps of magnitudes 0.1, 0.5 and 1.0 with linear controller designed using equilibrium information only (Example 1).



**Figure 4** Nonlinear controller realisation (Example 1)



**Figure 5** Responses to steps of magnitudes 10, 50 and 100 with nonlinear controller designed using off-equilibrium information provided by plant velocity-based linearisation family (Example 1).



**Figure 6** Nonlinear plant considered in Example 2