APPROXIMATE DECOUPLING AND ASYMPTOTIC TRACKING FOR MIMO SYSTEMS

by

D. N. Godbole and S. S. Sastry

Memorandum No. UCB/ERL M93/9

20 January 1993

ELECTRONICS RESEARCH LABORATORY

College of Engineering
University of California, Berkeley
94720

APPROXIMATE DECOUPLING AND ASYMPTOTIC TRACKING FOR MIMO SYSTEMS

by

D. N. Godbole and S. S. Sastry

Memorandum No. UCB/ERL M93/9

20 January 1993

ELECTRONICS RESEARCH LABORATORY

College of Engineering
University of California, Berkeley
94720

APPROXIMATE DECOUPLING AND ASYMPTOTIC TRACKING FOR MIMO SYSTEMS

by

D. N. Godbole and S. S. Sastry

Memorandum No. UCB/ERL M93/9

20 January 1993

Approximate Decoupling and Asymptotic tracking for MIMO systems*

D.N.Godbole and S.S.Sastry

Intelligent Machines and Robotics Laboratory
University of California, Berkeley
Berkeley, CA 94720
510-642-3248
godbole@robotics.berkeley.edu

Abstract

This paper studies approximate input output decoupling of nonlinear MIMO systems, for those systems which exhibit numerical ill posedness or nearly singular behavior in the exact decoupling algorithms. Although the systems considered are regular so that the exact decoupling algorithms are applicable in this case, they require inversion of an ill conditioned matrix, and yield high gain feedback solutions which may result in actuator saturation. The approximate algorithms are numerically robust, and provide solutions which do not cancel far off right half plane zeros. This latter characteristic is especially valuable when some of the far off right half plane zeros are unstable. The algorithms are inspired by and are generalizations of some examples in the flight control literature ([7], [6], [8]).

*Research supported in part by Army Research Office under grant DAAL-91-G-091 and NASA under grant NAG 2-243. The authors would like to acknowledge helpful discussions with M. Di Benedetto, J. Grizzle, J. Hauser, R. Kadiyala and P. Kokotovic

1 Introduction

The nonlinear control toolbox has grown enormously in the last decade. Central to this development is the theory of feedback linearization for nonlinear systems (see [1], [2]). This has provided a solution to a fundamental question in multivariable nonlinear system design: when is it possible to input output decouple a multi-input multi-output (MIMO) nonlinear system? Several algorithms have been proposed in the literature for solving the problem of exact decoupling for nonlinear MIMO systems, see for example [3], [4], [5], [1], [2]. All these algorithms need the determination of the inverse of a so-called *decoupling matrix*. However, practical implementation of these algorithms is difficult when the decoupling matrix is *ill conditioned* or close to singularity, in which case the decoupling of such systems needs excessively large control effort. Further, the algorithm is not numerically robust since it requires the inverse of an ill conditioned matrix.

In this paper, we propose a numerically robust input output decoupling algorithm for invertible nonlinear MIMO systems. Our efforts are motivated, in part, by the use in Hauser, et al [6], the work of Singh [7],[8] of such techniques to aircraft flight control problems and the work of Barbot, et al [9] with applications to models of electric motors and the Belousov Zhabotinsky reaction kinetics. In these examples, the intuition for approximation and the choice of parameters to be approximated is provided by the physics of the problem. This paper attempts to formalize the theory involved in these examples and provide an algorithm for more general MIMO nonlinear systems, whose physical derivation may not be explicitly known by the designer. Anather recent paper by Grizzle and Di Benedetto [10] provides approximate decoupling algorithm for systems which are not decouplable by the exact decoupling algorithms by reason of their not being *regular.* The approximate algorithm in this paper aims at those systems which are *regular* and so decouplable by the exact decoupling algorithms, but the numerics of the decoupling is poorly conditioned.

In addition to the numerical robustness of our approximate algorithm, it also serves another important purpose:
The exact input output decoupling algorithm is essentially a pole-zero cancelling control law. Thus, this law cancels zeros of the open loop system regardless of whether they are in the left half plane or the right half plane and regardless of their magnitude. In particular the input output decoupling control law only works for minimum phase nonlinear systems. For systems with far off zeros, cancellation may not be necessary, since they do not play a large role in the system dynamics and indeed may cause instability when the zeros lie in the right half plane. In either case, cancellation of far off zeros results in high gain controllers. The approximate decoupling algorithm does not cancel the far off zeros of the open loop system, thereby providing reasonable gain, practically implementable solutions. The price to be paid is the replacement of asymptotically exact tracking control laws by approximate tracking control laws. This connection between regular perturbations of nonlinear systems and the far off zeros was first pointed out in [11] for the Single-input Single-output (SISO) case and in [12] for the MIMO case. Systems in which these far off zeros are in the right half plane are called *slightly nonminimum phase* system in [6]. The approximate decoupling controller, in this case, results in a stable closed loop system.

It is possible to develop several different approximate decoupling algorithms starting from the different decoupling algorithms in the literature. The algorithm presented here is based on the Descusse and Moog dynamic decoupling algorithm (see [3],[13]). Section 2 reviews the Descusse-Moog algorithm. In section 3, we state the approximate algorithm and section 4 compares its convergence properties with that of the Descusse-Moog algorithm. In many cases, input output decoupling is a first step in designing a tracking controller. Section 5 examines the effect of the approximate decoupling on the performance of tracking controller. Application of this algorithm to linear systems is presented in section 6 for completeness.

2 Decoupling algorithms for nonlinear systems

Consider the square (i.e. number of inputs is equal to the number of outputs) Multi-input Multi-output (MIMO) nonlinear control system described by

$$
\begin{aligned}
\dot{x} &= f(x) + \sum_{i=1}^{m} g_i(x) u_i \\
y_j &= h_j(x) \quad j = 1, \cdots, m
\end{aligned}
\tag{1}
$$

where $x \in \Re^n$, $f(x), g_1(x), \cdots, g_m(x)$ are smooth vector fields on $\Re^n$ and $h_1(x), \cdots, h_m(x)$ smooth functions on $\Re^n$. For convenience, these equations will be written as

$$
\Sigma_0 : \quad \left\{ \begin{aligned} \dot{x} &= f(x) + g(x) u \\ y &= h(x) \end{aligned} \right.
$$

Throughout the analysis, we will assume that x_0 is an equilibrium point of the autonomous system, that is $f(x_0) = 0$. We will assume (without loss of generality) that $h(x_0) = 0$. All the analysis in this paper will be local and will be valid in a given open neighbourhood U of x_0. We now review some algorithms for decoupling of MIMO nonlinear systems.

We *assume* in what follows that each output y_j has a well defined *relative degree* γ_j. i.e. there exists an integer γ_j such that

$$
L_{g_i} L_f^l h_j(x) \equiv 0 \quad \forall\ l < \gamma_j - 1, \ \forall\ 1 \le i \le m, \ \forall\, x \in U
$$

Collecting these calculations, we have

$$
\begin{aligned}
\begin{bmatrix} y_1^{\gamma_1} \\ y_2^{\gamma_2} \\ \vdots \\ y_m^{\gamma_m} \end{bmatrix}
&= \begin{bmatrix} L_f^{\gamma_1} h_1(x) \\ L_f^{\gamma_2} h_2(x) \\ \vdots \\ L_f^{\gamma_m} h_m(x) \end{bmatrix}
+ \begin{bmatrix} L_{g_1} L_f^{\gamma_1 - 1} h_1(x) & \cdots & L_{g_m} L_f^{\gamma_1 - 1} h_1(x) \\ L_{g_1} L_f^{\gamma_2 - 1} h_2(x) & \cdots & L_{g_m} L_f^{\gamma_2 - 1} h_2(x) \\ \cdots & \cdots & \cdots \\ L_{g_1} L_f^{\gamma_m - 1} h_m(x) & \cdots & L_{g_m} L_f^{\gamma_m - 1} h_m(x) \end{bmatrix} u \\
&:= b(x) + \qquad A(x) u
\end{aligned}
\tag{2}
$$

$A(x)$ is called the decoupling matrix. If $A(x)$ is invertible at every point in U, then the *static state feedback* given by

$$
u = (A(x))^{-1} \left[-b(x) + v \right] \tag{3}
$$

will result in a closed loop system which is decoupled from input v to output y. This decoupled and input-output linearized system is given by

$$\begin{bmatrix} y_1^{\gamma_1} \\ y_2^{\gamma_2} \\ \vdots \\ y_m^{\gamma_m} \end{bmatrix} = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_m \end{bmatrix} \tag{4}$$

If the matrix $A(x)$ is *singular*, we can not use a static state feedback to decouple the nonlinear system ([1]), and we have to search for a dynamic state feedback to achieve input-output decoupling:

2.1 Dynamic Decoupling

If decoupling of the system Σ_0 of (1) can not be achieved by static state feedback, it may still be possible to find a dynamic compensator of the form

$$\Sigma_c : \begin{cases} \dot{z} &= D(x,z) + E(x,z)v \\ u &= F(x,z) + G(x,z)v \end{cases} \tag{5}$$

with $z \in \Re^{n_c}, v \in \Re^m$, such that the closed loop system denoted by Σ_e (for extended system)

$$\Sigma_e : \begin{cases} \dot{x} = f(x) + g(x)F(x,z) + g(x)G(x,z)v \\ \dot{z} = D(x,z) + E(x,z)v \\ y = h(x) \end{cases} \tag{6}$$

is decoupled from v to y. The dynamic feedback which decouples the system Σ_0 of (1) is actually a static state feedback to decouple the extended system Σ_e of (6). There are a number of algorithms in the literature for dynamic decoupling. The approximate decoupling algorithm we will propose is based on the Descusse and Moog algorithm of ([3], [13]), We will review the original algorithm to fix notation.

Descusse and Moog dynamic decoupling algorithm:

Define the extended system at the end of iteration $k-1$ to be Σ_k having x^e as its state and equilibrium point x_0^e. The algorithm will start at $k=0$ with the given system Σ_0 having state $x \in \Re^n$ and x_0 as its equilibrium point. The outputs of the system are unchanged during the course of the algorithm.

Step 1: Compute the relative degrees γ_i^k ($i \in \{1, \cdots, m\}$) for the m outputs of Σ_k. Define the decoupling matrix $A_k(x^e)$ to have its ij th entry given by,

$$a_{ij}^k(x^e) = L_{g_j} L_f^{\gamma_i^k - 1} h_i(x^e)$$

Let r_k be the normal rank of $A_k(x^e)$ in an open neighbourhood of x_0^e. If $r_k = m$, stop.

Step 2: If $r_k < m$, define a square and nonsingular matrix $\hat{G}_k(x^e)$ such that the $(m - r_k)$ last columns of $\hat{A}_k(x^e) := A_k(x^e)\, \hat{G}_k(x^e)$ are identically zero. Moreover this process can be carried out such that there exists r_k rows of which the nonzero elements form an $r_k \times r_k$ nonsingular diagonal matrix.

It is shown in [3] that $\hat{G}_k(x^e)$ always exists and is a smooth function of x^e.

There are r_k columns of $\hat{A}_k(x^e)$ with non-zero elements, out of which q_k columns have two or more non-zero elements. Design a permutation matrix P_k such that the first q_k columns of $\hat{A}_k(x^e)\ P_k$ have two or more elements non-zero, followed by the $(r_k - q_k)$ columns having only one non-zero element and finally the last $(m-r_k)$ columns having all zero elements. Denote $G_k(x^e) = \hat{G}_k(x^e)\ P_k$. Define an intermediate input $\hat{u}$ by:

$$\hat{u} = [G_k(x^e)]^{-1}\ u \tag{7}$$

Step 3: The system Σ_k now is

$$\begin{aligned} \dot{x}^e &= f(x^e) + g(x^e)G_k(x^e)\hat{u} \\ y &= h(x^e) \end{aligned}$$

Add integrators in series with the first q_k inputs. This creates the new input vector $\tilde{u}$ of the form:

$$\tilde{u} = \begin{bmatrix} \dot{\hat{u}}_1 \\ \vdots \\ \dot{\hat{u}}_{q_k} \\ \hat{u}_{q_k+1} \\ \vdots \\ \hat{u}_m \end{bmatrix} \tag{8}$$

Thus the new system after adding these integrators is:

$$\begin{bmatrix} \dot{x}^e \\ \cdots \\ \dot{\hat{u}}_1 \\ \vdots \\ \dot{\hat{u}}_{q_k} \end{bmatrix} = \begin{bmatrix} f(x^e) + \sum_{i=1}^{q_k} \tilde{g}_i(x^e)\hat{u}_i \\ \cdots \\ 0 \\ \vdots \\ 0 \end{bmatrix} + \begin{bmatrix} \sum_{i=q_k+1}^{m} \tilde{g}_i(x^e)\tilde{u}_i \\ \cdots \\ \tilde{u}_1 \\ \vdots \\ \tilde{u}_{q_k} \end{bmatrix} \tag{9}$$
$$y = h(x^e)$$

where $\tilde{g}(x^e) = g(x^e)G_k(x^e)$. Call this system Σ_{k+1}.

Step 4: Go to step 1 and resume the procedure with $k \leftarrow k+1$, new state variables $x^e \leftarrow \{x^e\} \cup \{\hat{u}_i\}_{(i=1,\cdots,q_k)}$ and new input $u \leftarrow \tilde{u}$. Let f, g, h still denote the extended f, g, h for notational simplicity. Let U still denote the open set of interest containing the equilibrium point of the extended system. □

It has been shown by Descusse and Moog that if the system (1) is right invertible and satisfies the accessibility rank condition (cf. [2] page 86) at x_0, then the above algorithm converges in a finite number of steps, L, to an extended system Σ_L which is decouplable by static state feedback.

Note: At the end of i^{th} iteration, the above algorithm adds q_i integrators $(q_i \leq r_i)$ to the system Σ_i. Thus the dimension of x^e increases by q_i.

The *Dynamic Extension algorithm* (cf. [1]) is similar, except it involves nonlinear transformations of the output space instead of the input space as in the Descusse-Moog algorithm.

The computation of the rank of $A_k(x^e)$ will be greatly simplified if $A_k(x^e)$ satisfies a regularity condition (see [14]), which guarantees that the normal rank of $A_k(x^e)$ is the same as the local rank for every $x^e \in U$.

2.2 Normal Form

Let us assume that the Descusse-Moog dynamic decoupling algorithm converges after L steps to a system of the form of (6). Let us denote this extended system by Σ_e. Let (f^e, g^e, h^e) be the triple characterizing Σ_e, $x^e = (x, z) \in \Re^{n+n_c}$ its state, u^e its input and y^e its output. Let $x_0^e = (x_0, z_0)$ be the equilibrium point of interest. This system Σ_e has a well defined vector relative degree $[\gamma_1^e, \cdots, \gamma_m^e]$ at x_0^e. Let $\gamma^e := \sum_{i=1}^m \gamma_i^e$. We can construct a local change of coordinates $\phi(x^e) = (\xi, \eta)$ with $\xi = \mathrm{col}(\xi^i)$, such that $\phi(x_0^e) = 0$, by choosing

$$\begin{aligned} \xi^i &= \mathrm{col}(h_i^e(x^e), L_{f^e}h_i^e(x^e), \cdots, L_{f^e}^{\gamma_i^e-1}h_i^e(x^e)) \\ &:= \mathrm{col}(\xi_1^i, \xi_2^i, \cdots, \xi_{\gamma_i^e}^i) \end{aligned} \tag{10}$$

and remaining $(n + n_c - \gamma^e)$ complementary coordinates η. In these coordinates, Σ_e takes the *normal* form (see [1]):

$$\begin{aligned} \dot{\xi}_1^i &= \xi_2^i \\ &\vdots \\ \dot{\xi}_{\gamma_i^e-1}^i &= \xi_{\gamma_i^e}^i \\ \dot{\xi}_{\gamma_i^e}^i &= b_i^e(\xi, \eta) + \sum_{j=1}^m a_{ij}^e(\xi, \eta) u_j^e \\ \dot{\eta} &= q(\xi, \eta) + P(\xi, \eta) u^e \\ y_i^e &= \xi_1^i \end{aligned} \tag{11}$$

for $i = 1, \cdots, m$, where

$$\begin{aligned} b_i^e(\xi, \eta) &= L_{f^e}^{\gamma_i^e} h_i^e(\phi^{-1}(\xi, \eta)) \quad 1 \le i \le m \\ a_{ij}^e(\xi, \eta) &= L_{g_j^e} L_{f^e}^{\gamma_i^e-1} h_i^e(\phi^{-1}(\xi, \eta)) \; 1 \le i, j \le m \end{aligned}$$

The static state feedback which decouples the system Σ_e is given by:

$$u^e = (A^e(\xi, \eta))^{-1}[-b^e(\xi, \eta) + v] \tag{12}$$

The decoupling state feedback renders the η dynamics unobservable. The zero dynamics of system Σ_e are the dynamics of the η coordinates in the subspace $\xi = 0$ with the decoupling feedback law of (12) (with $v = 0$), i.e.

$$\dot{\eta} = q(0, \eta) - P(0, \eta)[(A^e(0, \eta))^{-1} \; b^e(0, \eta)] \tag{13}$$

For a detailed discussion of the zero dynamics and the transmission zeros of nonlinear systems we refer the reader to [15],([1] Chapter 6), ([2] Chapter 11).

3 Approximate dynamic decoupling algorithm

The difficulty in implementing the decoupling algorithm comes from the ill-conditioning of the decoupling matrix or from a situation in which the decoupling matrix may be non-singular but close to singularity. To be precise, this occurs if the smallest singular value of $A^e(x^e)$ is smaller than a certain prespecified $\epsilon > 0$ uniformly for $x \in U$. In this case, the algorithm calls for the inverse of an ill conditioned matrix. In addition to the fact that the inverse is not numerically robust, it may cause large feedback gains in the controller and also cause the cancellation of far-off zeros (see [12], Section 4). To alleviate these difficulties, we propose the following numerically robustified decoupling algorithm: while the algorithm appears to have numerical considerations in mind, it is, in fact, valuable for the reason that it does not cancel far off *right half plane zeros* and may help control the magnitudes of the control inputs. To state the algorithm, recall a few basic facts and definitions about the *numerical rank* of a matrix:

Definition 1 A matrix $A \in \Re^{n \times n}$ is said to have ϵ *numerical rank* r if

$$\inf \text{rank}\{B : ||B - A|| < \epsilon\} = r$$

The norm in the above definition is the induced norm of the matrix induced by the Euclidean norm.

Thus, the numerical rank of a matrix is the lowest it can drop to in an ϵ neighborhood of the given matrix. In particular, if the matrix has $(n-r)$ of its singular values less than ϵ, then its numerical rank is r.

Approximate dynamic decoupling algorithm:

This algorithm starts at $k = 0$ with the given system Σ_0 having x_0 as its equilibrium point. We are given a threshold $\epsilon > 0$. Let the extended system at the end of iteration $(k-1)$ be denoted by Σ_k having x^e as its state and x_0^e, the corresponding equilibrium point.

Step 1: Compute the relative degrees of the outputs, namely, $\gamma_i^k, i = 1, \ldots, m$, and the decoupling matrix $A_k(x^e)$. Let r_k be the normal rank of $A_k(x^e)$ in U. If $r_k = m$ and if the smallest singular value of $A_k(x^e)$ is greater than the threshold ϵ uniformly on U, stop.

Step 2: If all the nonzero singular values of $A_k(x^e)$ are less than ϵ uniformly on U, approximate $A_k(x^e)$ by a *zero* matrix. Go to step 1 and recalculate the relative degrees γ_i^k with this approximation.

If $r_k = m$, go to step 3, with $\hat{A}_k(x^e) = A_k(x^e)$ and $\hat{G}_k(x^e) = I_{m \times m}$.

Design a smooth, square and nonsingular matrix $\hat{G}_k(x^e)$ such that the last $m - r_k$ columns of $\hat{A}_k(x^e) := A_k(x^e)\, \hat{G}_k(x^e)$ are identically zero.

Step 3: If the smallest *nonzero* singular value of $\hat{A}_k(x^e)$ is greater than the threshold ϵ, go to step 4, with $\tilde{A}_k(x^e) = \hat{A}_k(x^e)$ and $w_k = r_k$.

If the smallest *nonzero* singular value of $\hat{A}_k(x^e)$ is smaller than ϵ, then there exists a positive integer $w_k (< r_k)$ such that the ϵ rank of $\hat{A}_k(x^e)$ is w_k uniformly on U. i.e. $(r_k - w_k)$ *nonzero* singular values of $\hat{A}_k(x^e)$ are less than ϵ uniformly on U.

Design a smooth square nonsingular matrix $\tilde{G}_k(x^e)$ [1] such that

$$\tilde{A}_k(x^e) = \hat{A}_k(x^e)\,\tilde{G}_k(x^e) = [a_1^k(x^e), \cdots, a_{w_k}^k(x^e), \epsilon a_{w_k+1}^k(x^e), \cdots, \epsilon a_{r_k}^k(x^e), 0, \cdots, 0] \tag{14}$$

Approximate the $r_k - w_k$ columns, which are small in norm, by identically zero columns. Go to step 4, with

$$\tilde{A}_k(x^e) = [a_1^k(x^e), \cdots, a_{w_k}^k(x^e), 0, \cdots, 0]$$

Step 4: Out of w_k nonzero columns of $\tilde{A}_k(x^e)$, q_k columns will have two or more nonzero elements. Design a permutation matrix P_k such that the first q_k columns of $\tilde{A}_k(x^e)\ P_k(x^e)$ have two or more non-zero elements, followed by the $(w_k - q_k)$ columns having only one non-zero element and finally the last $(m - w_k)$ identically zero columns.
Denote $G_k(x^e) = \hat{G}_k(x^e)\ \tilde{G}_k(x^e)\ P_k$. Define an intermediate input by:

$$\hat{u} = [G_k(x^e)]^{-1}\, u$$

Step 5: The system Σ_k now is

$$\begin{aligned} \dot{x}^e &= f(x^e) + g(x^e)G_k(x^e)\hat{u} \\ y &= h(x^e) \end{aligned}$$

Add an integrator in series with the first q_k inputs. This creates the new input vector $\tilde{u}$ of the form:

$$\tilde{u} = \begin{bmatrix} \dot{\hat{u}}_1 \\ \vdots \\ \dot{\hat{u}}_{q_k} \\ \hat{u}_{q_{k+1}} \\ \vdots \\ \hat{u}_m \end{bmatrix} = \begin{bmatrix} \tilde{u}_1 \\ \vdots \\ \tilde{u}_{q_k} \\ \hat{u}_{q_{k+1}} \\ \vdots \\ \hat{u}_m \end{bmatrix} \tag{15}$$

Thus the new system after adding these integrators is:

$$\begin{bmatrix} \dot{x}^e \\ \cdots \\ \dot{\hat{u}}_1 \\ \vdots \\ \dot{\hat{u}}_{q_k} \end{bmatrix} = \begin{bmatrix} f(x^e) + \sum_{i=1}^{q_k} \tilde{g}_i(x^e)\hat{u}_i \\ \cdots \\ 0 \\ \vdots \\ 0 \end{bmatrix} + \begin{bmatrix} \sum_{i=q_{k+1}}^{m} \tilde{g}_i(x^e)\tilde{u}_i \\ \cdots \\ \tilde{u}_1 \\ \vdots \\ \tilde{u}_{q_k} \end{bmatrix} \tag{16}$$
$$y = h(x^e)$$

where $\tilde{g}(x^e) = g(x^e)G_k(x^e)$. Call this system Σ_{k+1}.
Step 6: Return to step 1 and resume the procedure with $k \leftarrow k+1$, new state variables $x^e \leftarrow \{x^e\} \cup \{\hat{u}_i\}_{(i=1,\cdots,q_k)}$ and new input $u \leftarrow \tilde{u}$ Let f, g, h still denote the extended f, g, h for

[1]The proof of existence of such a matrix is given in the appendix

notational simplicity. Let U still denote the open set containing the equilibrium point of the extended system. □.

Let us assume that the approximate dynamic decoupling algorithm converges after L steps to a system of the form of (6). Let us denote this extended system by $\tilde{\Sigma}_e$. Let $(\tilde{f}^e, \tilde{g}^e, \tilde{h}^e)$ be the triple characterizing $\tilde{\Sigma}_e$, $\tilde{x}^e = (x, z)$ its state, $\tilde{u}^e$ its input and $\tilde{y}^e$ its output. Let $\tilde{x}_0^e = (x_0, z_0)$ be the equilibrium point of interest. The system $\tilde{\Sigma}_e$ has a well defined vector relative degree $[\tilde{\gamma}_1^e, \cdots, \tilde{\gamma}_i^e]$ at $\tilde{x}_0^e$. Thus we can construct a local diffeomorphism of the form of (10) such that $\tilde{\Sigma}_e$ will be transformed into the normal form $(\tilde{\xi}, \tilde{\eta})$ coordinates given by,

$$\begin{aligned} \tilde{\xi}^i &= \operatorname{col}(\tilde{h}_i^e(x^e), L_{\tilde{f}^e}\tilde{h}_i^e(\tilde{x}^e), \cdots, L_{\tilde{f}^e}^{\tilde{\gamma}_i^e - 1}\tilde{h}_i^e(\tilde{x}^e)) \\ &= \operatorname{col}(\tilde{\xi}_1^i, \cdots, \tilde{\xi}_{\tilde{\gamma}_i^e}^i) \end{aligned} \tag{17}$$

and remaining $(n + \tilde{n}_c - \sum_{i=1}^m \tilde{\gamma}_i^e)$ complementary coordinates $\tilde{\eta}$. In the course of the approximate dynamic decoupling algorithm, we have neglected order ϵ terms at each iteration of the algorithm. Thus the original or *exact* system in the normal form coordinates of $\tilde{\Sigma}_e$ will be:

$$\begin{aligned} \dot{\tilde{\xi}}_1^i &= \tilde{\xi}_2^i \\ &\vdots \\ \dot{\tilde{\xi}}_{\tilde{\gamma}_i^0 - 1}^i &= \tilde{\xi}_{\tilde{\gamma}_i^0}^i \\ \dot{\tilde{\xi}}_{\tilde{\gamma}_i^0}^i &= \tilde{\xi}_{\tilde{\gamma}_i^0 + 1}^i + \epsilon \sum_{j=1}^m (\beta_{\tilde{\gamma}_i^0}^i)_j \tilde{u}_j^e \\ &\vdots \\ \dot{\tilde{\xi}}_{\tilde{\gamma}_i^e - 1}^i &= \tilde{\xi}_{\tilde{\gamma}_i^e}^i + \epsilon \sum_{j=1}^m (\beta_{\tilde{\gamma}_i^e - 1}^i)_j \tilde{u}_j^e \\ \dot{\tilde{\xi}}_{\tilde{\gamma}_i^e}^i &= \tilde{b}_i^e(\tilde{\xi}, \tilde{\eta}) + \sum_{j=1}^m \tilde{a}_{ij}^e(\tilde{\xi}, \tilde{\eta}) \tilde{u}_j^e \\ \dot{\tilde{\eta}} &= q(\tilde{\xi}, \tilde{\eta}) + P(\tilde{\xi}, \tilde{\eta}) \tilde{u}^e \\ \tilde{y}_i^e &= \tilde{\xi}_1^i \end{aligned} \tag{18}$$

for $i = 1, \cdots, m$, where

$$\begin{aligned} \tilde{b}_i^e(\tilde{\xi}, \tilde{\eta}) &= L_{\tilde{f}^e}^{\tilde{\gamma}_i^e} \tilde{h}_i^e(\phi^{-1}(\tilde{\xi}, \tilde{\eta})) \quad 1 \le i \le m \\ \tilde{a}_{ij}^e(\tilde{\xi}, \tilde{\eta}) &= L_{\tilde{g}_j^e} L_{\tilde{f}^e}^{\tilde{\gamma}_i^e - 1} \tilde{h}_i^e(\phi^{-1}(\tilde{\xi}, \tilde{\eta})) \quad 1 \le i, j \le m \end{aligned}$$

Note: If we substitute $\epsilon = 0$ in the above equations (18), then we get the representation of the system $\tilde{\Sigma}_e$ in its normal form coordinates $(\tilde{\xi}, \tilde{\eta})$. The static state feedback which decouples the system $\tilde{\Sigma}_e$ is given by:

$$\tilde{u}^e = (\tilde{A}^e(\tilde{\xi}, \tilde{\eta}))^{-1}[-\tilde{b}^e(\tilde{\xi}, \tilde{\eta}) + v] \tag{19}$$

If we compare the approximate and exact decoupling algorithm, we see that

- If the exact decoupling algorithm converges with a decoupling matrix $A^e(x^e)$ whose smallest singular value is greater than ϵ uniformly on U, then the approximate decoupling algorithm yields the same result.

- If, on the other hand, $A^e(x^e)$ obtained by exact decoupling algorithm has its smallest singular value of the order of ϵ uniformly on U then the decoupling control law (12) will have terms of the order of $\frac{1}{\epsilon}$ which will result in a high gain controller which may not be practically feasible if the actuators have saturation limits.

The approximate decoupling algorithm is of use if we can answer the following two questions:

1. If the given system (1) is right invertible and locally accessible (i.e. the exact decoupling algorithm converges in L steps), then does the approximate decoupling algorithm converge in a finite number of steps?

2. Suppose the answer to the first question is *yes*, then how much error do we introduce in tracking of the reference outputs, by using the approximate decoupling and tracking law instead of the exact one.

These questions will be answered in the following sections.

4 Convergence of approximate decoupling algorithm

The approximate decoupling algorithm is based on the Descusse and Moog dynamic decoupling algorithm. Consequently, the convergence properties of the approximate decoupling algorithm will be compared with that of the Descusse and Moog algorithm. Ideally the approximate decoupling algorithm should preserve the convergence properties of the Descusse and Moog algorithm.

The Descusse and Moog algorithm adds dynamics to the given system Σ_0, in order to get an extended system Σ_L which is decouplable by static state feedback. The following theorem ([3]), relates this notion to the nonsingularity of the decoupling matrix $A_L(x)$.

Theorem 1 *System Σ of the form of (1) is decouplable by static state feedback, iff, the decoupling matrix for Σ is invertible*

The approximate decoupling algorithm converges to an extended system which is robustly decouplable by static state feedback. This notion is defined as follows:

Definition 2 *An $m \times m$ matrix $A(x)$ is ϵ* robustly invertible *in an open set U with respect to a threshold ϵ if the ϵ numerical rank of $A(x)$ is m uniformly on U.*

Definition 3 *A system Σ of the form of (1) is ϵ* robustly decouplable *by static state feedback with respect to a threshold ϵ, if, Σ is decouplable by static state feedback and the decoupling matrix $A(x)$ is robustly invertible with respect to ϵ.*

The following lemma considers the effect of one iteration of approximate decoupling algorithm on a system which is decouplable by using the Descusse and Moog algorithm.

Lemma 1 *Suppose that the Descusse and Moog algorithm converges for a system Σ_k of the form of (1). Apply the approximate decoupling algorithm. After one iteration, we get the extended system Σ_{k+1}. One of the following is true for Σ_{k+1}:*

1. $\gamma_i^{k+1} = \infty$ *for some* i
2. $A_{k+1}(x)$ *is singular and the Descusse and Moog decoupling algorithm does not converge for* Σ_{k+1}
3. $\gamma^{k+1} = n_{\Sigma_{k+1}}$ *and* $A_{k+1}(x)$ *is not robustly invertible.*
4. $A_{k+1}(x)$ *is singular, but the Descusse and Moog algorithm converges for* Σ_{k+1}
5. $A_{k+1}(x)$ *is nonsingular, not robustly invertible and* $\gamma^{k+1} < n_{\Sigma_{k+1}}$
6. $A_{k+1}(x)$ *is robustly invertible*

where $\gamma^{k+1} = \sum_{i=1}^{m} \gamma_i^{k+1}$ *and* $n_{\Sigma_{k+1}}$ = *the dimension of state space of* Σ_{k+1}.

Proof: This is a list of all the cases after application of one step of the approximate decoupling algorithm. It is easy to check that the list exhausts all the possibilities. □.

We will analyze each of the above cases in detail in order to understand why in some cases the approximate decoupling algorithm may not converge.

While applying the Descusse and Moog algorithm, we differentiate each output until at least one input appears on the right hand side. Some of the inputs show up earlier than others making the decoupling matrix singular. Integrators are added in front of these inputs to delay their appearance for at least one more step of differentiation.

In this process, at a particular step, some of the inputs might be weakly connected to the outputs, i.e. the functions multiplying them are smaller than the threshold ϵ uniformly in U. These functions are approximated by zero in the approximate decoupling algorithm. This modification in the original Descusse and Moog algorithm might fail because of the following reasons.

- A particular input might have *almost singular* functions multiplying it at each step of the algorithm.
- At any step, the approximation might make the resulting system noninvertible.

A system which is decouplable by the Descusse-Moog algorithm but which is not decouplable by the approximate algorithm is said to be *ϵ-unnormalized*. Such a system can be normalized by replacing the unnormalized inputs u_i by $\frac{u_i}{\epsilon}$. The classification of various cases in the previous lemma helps detect an unnormalized system in the following manner:

Recall that γ_i^k are the relative degrees of the outputs of Σ_k. We have,

$$\begin{bmatrix} y_1^{\gamma_1^k} \\ \vdots \\ y_m^{\gamma_m^k} \end{bmatrix} = b_k(x) + A_k(x)u$$

rank of $A_k(x) = r_k$. At the end of step 2, $(m - r_k)$ columns of $\hat{A}_k(x)$ are identically zero and there is at least one nonzero element in each row of $\hat{A}_k(x)$.

At the end of step 4, some of the rows of $A_k(x)G_k(x)$ might be identically zero because of the approximation of $(r_k - w_k)$ columns by zero. The number of such rows is less than or equal to $(r_k - w_k)$. Let us denote the set of outputs corresponding to these rows by $\bar{Y}_k$.

$(w_k - q_k)$ columns of $A_k(x)G_k(x)$ have only one nonzero element. By construction, these nonzero elements will be in $(w_k - q_k)$ different rows. Let us denote the $(w_k - q_k)$ outputs corresponding to these rows by $\hat{Y}_k$.
The remaining outputs will be denoted by $\tilde{Y}_k$.

The outputs of Σ_k do not change in the process. Thus for Σ_{k+1}, we have

- $\gamma_i^{k+1} = \gamma_i^k, \forall\, i$ s.t. $y_i \in \hat{Y}_k$
- $\gamma_i^{k+1} = \gamma_i^k + 1, \forall\, i$ s.t. $y_i \in \tilde{Y}_k$
- $\gamma_i^{k+1} \geq \gamma_i^k + 1, \forall\, i$ s.t. $y_i \in \bar{Y}_k$

Case 1: If the only non zero entries in the *ith* row are in the j column, then when the j^{th} column of $A_k(x)G_k(x)$ is approximated by zero, the i^{th} row is made identically zero. y_i is only affected by u_j, and after the approximation, u_j never appears on the right hand side again. This makes $\gamma_i^{k+1} = \infty$. The problem can be solved by normalizing the input u_j to $\frac{u_j}{\epsilon}$.
Case 2: The situation here is that Case 1 recurs, but not immediately at the end of the $k-1$ st step but later in the algorithm. Thus the system Σ_k loses invertibility because of the approximation. Some rows of $A_{k+l}(x)$ will be dependent for all $l > 0$. This case can also be avoided by normalizing the j^{th} input.

In the course of the approximate decoupling algorithm, case 2 goes unnoticed until you reach case 3. Thus the reason for case 3 is in fact the occurrence of case 2 during one of the previous iterations.

If the approximate decoupling algorithm converges to a system Σ_L which can be decoupled by static state feedback, the outputs y_i and their respective derivatives upto the order of $(\gamma_i^L - 1)$ qualify as a partial change of coordinates. In the normal form notation these are the ξ coordinates.

During each iteration of the approximate decoupling algorithm, the state space dimension of Σ_k is extended by q_k whereas at least $(m - w_k + q_k)$ new ξ coordinates are introduced. The difference between the state space dimension of Σ_{k+1} and the dimension of ξ coordinates decreases by $(m - w_k)$ during each iteration of the approximate decoupling algorithm.
Case 3: If we go through $(k+1)^{th}$ iteration, dimension of ξ coordinates will exceed the dimension of the state space of Σ_{k+2}. Thus we can not proceed further.
Cases 4,5 and 6 lead towards the convergence of approximate decoupling algorithm.
Now we are ready to define an unnormalized system:

Definition 4 Suppose Σ_0 satisfies the hypothesis of Descusse and Moog algorithm. Apply the approximate decoupling algorithm, for some $\epsilon > 0$. Let $k \geq 0$ be the smallest integer such that either case 1, 2 or 3 of the previous lemma is true for Σ_{k+1}. Then the systems Σ_0 is said to be *ϵ-unnormalized.*

Theorem 2 *Suppose Σ_0 satisfies the hypothesis of the Descusse and Moog dynamic decoupling algorithm. Apply the approximate decoupling algorithm for a given $\epsilon > 0$. Then one of the following is true*

- *The approximate decoupling algorithm converges in finite steps.*
- *The system Σ_0 is ϵ-unnormalized.*

Proof: During each step of the approximate decoupling algorithm, the difference between the state space dimension and the dimension of ξ coordinates decreases by $(m - w_k)$. Thus if the first three cases of the previous lemma are avoided during each iteration, the algorithm has to converge in a finite number of steps.

The discussion following the previous lemma shows that the first three cases correspond to the underlying system being unnormalized. □

In general, it is not possible, a priori, to find out whether a given system is normalized or not. If the approximate decoupling algorithm does not converge in n steps, then the system is unnormalized, provided it was decouplable by using the exact Descusse and Moog dynamic decoupling algorithm.

4.1 Multiple time scale zero dynamics

Since the zero dynamics of a system does not change by addition of integrators to its input channels ([1] page 389), or by input space transformations or by state feedback, the zero dynamics of Σ_0 is same as that of Σ_L, where Σ_L is the extended system at the end of Descusse and Moog algorithm. The next lemma compares the zero dynamics of $\tilde{\Sigma}_L$ and Σ_L, where $\tilde{\Sigma}_L$ is the extended system at the end of approximate decoupling algorithm. Note that the approximate decoupling algorithm can not converge in fewer steps than the Descusse and Moog algorithm.

Lemma 2 *Suppose Σ_0 is right invertible (cf. [3]) and satisfies the strong accessibility rank condition (cf. [2] page 86) at x_0. Suppose the approximate decoupling algorithm converges for this system in exactly the same number of steps as the Descusse and Moog algorithm. Let Σ_L and $\tilde{\Sigma}_L$ be the system at the end of the Descusse and Moog algorithm and the approximate decoupling algorithms respectively. Then*

$$dim(\eta_{\tilde{\Sigma}_L}) \leq dim(\eta_{\Sigma_L}) - \sum_{i=0}^{L-1}(r_i - w_i)$$

where η denotes the zero dynamics coordinates.

Proof: During each iteration, the difference between state space coordinates and the ξ coordinates decreases by $(m - w_k)$ for approximate decoupling algorithm and by $(m - r_k)$ for the Descusse and Moog decoupling algorithm. After L steps when both the algorithms converge, the zero dynamics dimension is given by the difference between the state space dimension of the extended system and the dimension of ξ coordinates. Thus the normal form of Σ_L will have $\sum_{i=0}^{L-1}(r_i - w_i)$ more η coordinates than that of $\tilde{\Sigma}_L$. Hence the result. □.

Thus, in general, the zero dynamics of the extended system at the end of the approximate decoupling algorithm (i.e. $\tilde{\Sigma}_e$) will have smaller dimension than that of Σ_e. We would like to investigate the relationship between $\eta_{\tilde{\Sigma}_e}$ and η_{Σ_e}.

Recent results in the area of *singularly perturbed zero dynamics of nonlinear systems* (ref. [12], [11]) lead us to the following conclusion: Under some suitable technical hypotheses, the zero dynamics of Σ_e can be decomposed into two or more time scales using singular perturbation theory (cf. [16]). The slow or reduced system is described by the zero dynamics of $\tilde{\Sigma}_e$ and the dynamics which was neglected during the process of approximation constitutes the faster time scale or boundary layer subsystem. In ([12]), the authors have proved the above conclusion for a restricted class of two-input two-output systems. The full details and the technical hypotheses needed to guarantee the existence of singularly perturbed zero dynamics for this general class of systems remain to be worked out. We will conclude this section with the following remarks:

The approximate decoupling algorithm creates an extended system which does not include the *far off* zeros of the original system. Since the static state feedback which achieves decoupling of $\tilde{\Sigma}_L$ is a pole zero cancellation law, we do not cancel the *far off* zeros of Σ_0 in the case of the approximate decoupling algorithm. The cancellation of these far off zeros requires a large control effort resulting in a high gain controller. If these far off zeros are unstable then their cancellation makes the closed loop system unstable. These systems are referred to as *slightly nonminimum phase systems* in [6]. Application of approximate decoupling algorithm to slightly nonminimum phase systems, results in a stable closed loop system.

5 Approximate Asymptotic Tracking

Input output decoupling is closely related to tracking of reference trajectories by the outputs of a MIMO nonlinear system. If the desired trajectories to be tracked fall into a restricted class of functions, say constants or sinusoids with a finite spectrum, then we can use the regulator theory (see [1], Chapter 7). If the class of desired trajectories is more general, for example, functions which are N times continuously differentiable but otherwise arbitrary, then according to [17], the decoupling controller forms an inner loop of the overall tracking controller. If the given system is not robustly decouplable by using the exact decoupling algorithms, then we have to use the approximate decoupling feedback. This section considers the effects of approximate decoupling on the performance and stability of the overall tracking controller.

Let us assume that the approximate decoupling algorithm converges for Σ_0 giving us the approximate extended system $\tilde{\Sigma}_e$. The equations (18) with $\epsilon = 0$ represent $\tilde{\Sigma}_e$ in its normal form $(\tilde{\xi}, \tilde{\eta})$ coordinates. If the objective of the controller is to track the desired reference trajectory $y_d(t) = [y_{d_1}(t), \cdots, y_{d_m}(t)]^T$ which is smooth and bounded with bounded derivatives, we design the control input $\tilde{u}^e$ to be:

$$
\begin{aligned}
\tilde{u}^e &= (\tilde{A}^e(\tilde{\xi},\tilde{\eta}))^{-1}[-b^e(\tilde{\xi},\tilde{\eta}) + v] \\
v &= \begin{bmatrix} v_1 \\ \vdots \\ v_m \end{bmatrix} = \begin{bmatrix} y_{d_1}^{\tilde{\gamma}_1^e} + \sigma^1_{\tilde{\gamma}_1^e-1}(y_{d_1}^{\tilde{\gamma}_1^e-1} - \xi^1_{\tilde{\gamma}_1^e-1}) + \cdots + \sigma^1_0(y_{d_1} - \xi^1_1) \\ \vdots \\ y_{d_m}^{\tilde{\gamma}_m^e} + \sigma^m_{\tilde{\gamma}_m^e-1}(y_{d_m}^{\tilde{\gamma}_m^e-1} - \xi^m_{\tilde{\gamma}_m^e-1}) + \cdots + \sigma^m_0(y_{d_m} - \xi^m_1) \end{bmatrix}
\end{aligned} \tag{20}
$$

where $(s^{\tilde{\gamma}_i^e} + \sigma^i_{\tilde{\gamma}_i^e-1}s^{\tilde{\gamma}_i^e-1} + \cdots + \sigma^i_0)$ is a Hurwitz polynomial for $i = 1, \cdots, m$.
Let us define the tracking errors to be

$$
e_1^i := \tilde{\xi}_1^i - y_{d_i}, \quad 1 \le i \le m \tag{21}
$$

Let us define the error coordinates for system $\tilde{\Sigma}_e$ to be

$$
\begin{bmatrix} e_1^i \\ \vdots \\ e^i_{\tilde{\gamma}_i^e-1} \end{bmatrix} = \begin{bmatrix} \tilde{\xi}_1^i \\ \vdots \\ \tilde{\xi}^i_{\tilde{\gamma}_i^e-1} \end{bmatrix} - \begin{bmatrix} y_{d_i} \\ \vdots \\ y_{d_i}^{\tilde{\gamma}_i^e-1} \end{bmatrix} \quad 1 \le i \le m \tag{22}
$$

Thus the system $\tilde{\Sigma}_e$ with the feedback (20) can be expressed in $(e, \tilde{\eta})$ coordinates by

$$
\begin{aligned}
\dot{e}^i &= \bar{A}^i e^i \quad i = 1, \cdots, m \\
\dot{\tilde{\eta}} &= q(\tilde{\xi},\tilde{\eta}) + P(\tilde{\xi},\tilde{\eta})\tilde{u}^e(\tilde{\xi},\tilde{\eta},v)
\end{aligned} \tag{23}
$$

where $\bar{A}^i \in \Re^{\tilde{\gamma}_i^e \times \tilde{\gamma}_i^e}$ given by

$$
\bar{A}^i = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ & & \vdots & & \\ 0 & 0 & 0 & 0 & 1 \\ -\sigma_0^i & -\sigma_1^i & \cdot & \cdots & -\sigma^i_{\tilde{\gamma}_i^e-1} \end{bmatrix}
$$

It can be shown (e.g. see [6]) that if

- The reference trajectory and its derivatives are bounded and small enough.
- Zero dynamics of (23) (i.e. the equilibrium point $\tilde{\eta}_0 = 0$ of the system)

$$
\dot{\tilde{\eta}} = q(0,\tilde{\eta}) + P(0,\tilde{\eta})\tilde{u}^e(0,\tilde{\eta},0) \tag{24}
$$

  is exponentially stable

- $q(\tilde{\xi},\tilde{\eta}) + P(\tilde{\xi},\tilde{\eta})\tilde{u}^e(\tilde{\xi},\tilde{\eta},v)$ is locally Lipschitz continuous in ξ, η

then $\lim_{t\to\infty} e_1^i(t) = 0 \ \forall i.$ and the states ξ, η remain bounded.

The controller of (20) is designed for the approximate extended system $\tilde{\Sigma}_e$. If we apply this controller to the exact system, we get the system equations in the $(e, \tilde{\eta})$ coordinates given by

$$
\begin{aligned}
\dot{e}^i &= \bar{A}^i e^i + \epsilon\beta^i(x^e)\tilde{u}^e(x^e) \quad 1 \le i \le m \\
\dot{\tilde{\eta}} &= q(\tilde{\xi},\tilde{\eta}) + P(\tilde{\xi},\tilde{\eta})\tilde{u}^e(\tilde{\xi},\tilde{\eta},v)
\end{aligned} \tag{25}
$$

where

$$\beta^i = \begin{bmatrix} 0_{1\times m} \\ \vdots \\ 0_{1\times m} \\ \beta^i_{\tilde{\gamma}_i^0} \\ \vdots \\ \beta^i_{\tilde{\gamma}_i^e - 1} \\ 0_{1\times m} \end{bmatrix}$$

represents the dynamics which was neglected during the approximate decoupling algorithm. Each β^i_j is $1 \times m$ row vector of functions of x, the first w_0 elements of which are identically zero.

The tracking control law (20) was designed for a system of the form (25) with $\epsilon = 0$. The following theorem shows that it works for the approximate system with nonzero ϵ as well. This theorem is motivated by and is similar to the one for *slightly nonminimum phase systems* as in [6].

Theorem 3 *If*

- *zero dynamics of the system (25) (i.e. the equilibrium point $\tilde{\eta}_0 = 0$ of the equation (24)) is exponentially stable*
- *The functions $\beta^i(x^e)\tilde{u}^e(x^e)$ are locally Lipschitz continuous with $\beta^i(x^e_0)\tilde{u}^e(x^e_0) = 0 \;\; \forall i = 1, \cdots, m$*
- *$q(\tilde{\xi}, \tilde{\eta}) + P(\tilde{\xi}, \tilde{\eta})\tilde{u}^e(\tilde{\xi}, \tilde{\eta}, v)$ is locally Lipschitz continuous in $\tilde{\xi}, \tilde{\eta}, v$*

Then for ϵ sufficiently small and for desired trajectories with derivatives small enough, the states of system (25) are bounded and the tracking errors satisfy

$$\|e^i_1\| = \|\tilde{\xi}^i_1 - y_{d_i}\| \le K\epsilon$$

for some $K < \infty$

Note: Since we know that f, g, h are smooth functions of x to start with, the functions $\beta^i(x^e)\tilde{u}^e(x^e)$ will be locally Lipschitz so long as the matrix $G_k(x^e)$ is a smooth function of x^e at each iteration of the approximate decoupling algorithm.

Proof:

From (22) and the fact that the desired trajectory and its derivatives are bounded (by b_d), we get

$$\|\tilde{\xi}\| \le \|e\| + b_d \tag{26}$$

The transformation which transforms $\tilde{\Sigma}_e$ into $(\tilde{\xi}, \tilde{\eta})$ coordinates is a diffeomorphism, thus there exits $l_x > 0$ such that,

$$\|x\| \le l_x(\|\tilde{\xi}\| + \|\tilde{\eta}\|) \tag{27}$$

As the functions $\beta^i(x^e)\tilde{u}^e(x^e)$ are locally Lipschitz continuous with $\beta^i(x_0^e)\tilde{u}^e(x_0^e) = 0$, there exists a positive constant l_β such that,

$$\|2\bar{P}\beta(x^e)\tilde{u}^e(x^e)\| \leq l_\beta\|x^e\| \tag{28}$$

where $\beta(x^e)$ is the block diagonal matrix with $\beta^i(x^e)$ being its diagonal blocks.
Since the zero dynamics

$$\dot{\tilde{\eta}} = q(0,\tilde{\eta}) + P(0,\tilde{\eta})\tilde{u}^e(0,\tilde{\eta},0) \tag{29}$$

is exponentially stable, by a converse Lyapunov theorem [18]
there exists, $\bar{v}(\tilde{\eta})$ and positive constants k_1, k_2, k_3, k_4 such that:

$$\begin{aligned} k_1\|\tilde{\eta}\|^2 \leq \bar{v}(\tilde{\eta}) &\leq k_2\|\tilde{\eta}\|^2 \\ \frac{\partial \bar{v}}{\partial \tilde{\eta}}[q(0,\tilde{\eta}) + P(0,\tilde{\eta})\tilde{u}^e(0,\tilde{\eta},0)] &\leq k_3\|\tilde{\eta}\|^2 \\ \|\frac{\partial \bar{v}}{\partial \tilde{\eta}}\| &\leq k_4\|\tilde{\eta}\| \end{aligned}$$

Thus from the exponential stability of zero dynamics and the Lipschitz continuity of $q + Pu^k$, we get,

$$\begin{aligned} \dot{v} &= \frac{\partial \bar{v}}{\partial \tilde{\eta}}[q(\tilde{\xi},\tilde{\eta}) + P(\tilde{\xi},\tilde{\eta})u^k(\tilde{\xi},\tilde{\eta},v)] \\ &= \frac{\partial \bar{v}}{\partial \tilde{\eta}}[q(0,\tilde{\eta}) + P(0,\tilde{\eta})u^k(0,\tilde{\eta},0)] \\ &+ \frac{\partial \bar{v}}{\partial \tilde{\eta}}[q(\tilde{\xi},\tilde{\eta}) + P(\tilde{\xi},\tilde{\eta})u^k(\tilde{\xi},\tilde{\eta},v) - \{q(0,\tilde{\eta}) + P(0,\tilde{\eta})u^k(0,\tilde{\eta},0)\}] \\ &\leq -k_3\|\tilde{\eta}\|^2 + k_4\|\tilde{\eta}\|l_\eta(\|\tilde{\xi}\| + \|v\|) \end{aligned}$$

To show the states of (25) remain bounded, let

$$V(e,\tilde{\eta}) = e^T\bar{P}e + \psi\bar{v}(\tilde{\eta})$$

be a Lyapunov function for the system (25) where $\bar{P} > 0$ satisfies the following lyapunov equation

$$\bar{A}^T\bar{P} + \bar{P}\bar{A} = -I$$

and ψ is a positive constant to be specified later. Then

$$\begin{aligned} \dot{V} &= -\|e\|^2 + 2\epsilon e^T\bar{P}\beta(x)u^k(x) + \psi\frac{\partial \bar{v}}{\partial \tilde{\eta}}\dot{\tilde{\eta}} \\ &\leq -\|e\|^2 + \epsilon\|e\|l_\beta\|x\| + \psi[-k_3\|\tilde{\eta}\|^2 + k_4\|\tilde{\eta}\|l_\eta(\|\tilde{\xi}\| + \|v\|)] \\ &\leq -\|e\|^2 + \epsilon\|e\|l_\beta l_x(\|e\| + b_d + \|\tilde{\eta}\|) + \psi[-k_3\|\tilde{\eta}\|^2 + k_4\|\tilde{\eta}\|l_\eta\{\|e\| + b_d + l_v(\|e\| + b_d)\}] \\ &\leq -(\frac{\|e\|}{2} - \epsilon l_\beta l_x b_d)^2 + (\epsilon l_\beta l_x b_d)^2 \end{aligned}$$

$$
\begin{aligned}
& -(\frac{\|e\|}{2} - (\epsilon l_\beta l_x + \psi k_4 l_\eta (1+l_v))\|\tilde{\eta}\|)^2 + (\epsilon l_\beta l_x + \psi k_4 l_\eta (1+l_v))^2 \|\tilde{\eta}\|^2 \\
& -\psi k_3 (\frac{\|\tilde{\eta}\|}{2} - \frac{k_4 l_\eta b_d (1+l_v)}{k_3})^2 + \psi \frac{(k_4 l_\eta b_d (1+l_v))^2}{k_3} \\
& -(\frac{1}{2} - \epsilon l_\beta l_x)\|e\|^2 - \frac{3}{4}\psi k_3 \|\tilde{\eta}\|^2 \\
\leq \; & -(\frac{1}{2} - \epsilon l_\beta l_x)\|e\|^2 - (\frac{3}{4}\psi k_3 - (\epsilon l_\beta l_x + \psi k_4 l_\eta (1+l_v))^2)\|\tilde{\eta}\|^2 \\
& +(\epsilon l_\beta l_x b_d)^2 + \psi \frac{(k_4 l_\eta b_d (1+l_v))^2}{k_3}
\end{aligned}
$$

Let

$$
\tilde{\psi} = \frac{k_3}{4(l_\beta l_x + k_4 l_\eta (1+l_v))^2}
$$

Then, for all $\psi \leq \tilde{\psi}$ and all $\epsilon \leq min(\tilde{\psi}, \frac{1}{4 l_\beta l_x})$, we have

$$
\dot{V} \leq -\frac{\|e\|}{4} - \frac{\psi k_3 \|\tilde{\eta}\|^2}{2} + (\epsilon l_\beta l_x b_d)^2 + \psi \frac{(k_4 l_\eta b_d (1+l_v))^2}{k_3} \tag{30}
$$

Thus $\dot{V} < 0$ whenever $\|\tilde{\eta}\|$ or $\|e\|$ is large. This implies that $\|\tilde{\eta}\|$ and $\|e\|$ and also, $\|\tilde{\xi}\|$ and $\|x\|$ are bounded. The above analysis is valid in U. Thus by choosing initial conditions inside U and b_d sufficiently small, we guarantee that the states will remain in U. Using the boundedness of x and the continuity of $\beta^i(x)u^k(x)$, we see that

$$
\dot{e}^i(x) = \bar{A}^i e^i + \epsilon \beta^i(x) u^k(x)
$$

are m SISO exponentially stable linear systems driven by order ϵ input. Thus we conclude that the tracking errors e^i converge to a ball of order ϵ $\Box$.

6 MIMO Linear systems

The analysis for linear system can be carried out in the same fashion as the nonlinear system analysis presented in previous sections. A recent report [19] describes the software which is developed for designing approximate decoupling controllers for linear systems. We propose a *precompensation and normalization technique* which gives a controller of smaller dimension. Start with the square linear system Σ_0 of the form:

$$
\begin{aligned}
\dot{x} &= Ax + Bu \quad x \in \Re^n, \; u \in \Re^m \\
y &= Cx \qquad y \in \Re^m
\end{aligned} \tag{31}
$$

Assuming that each component of the output has a finite relative degree, we get,

$$
\begin{aligned}
\begin{bmatrix} y_1^{\gamma_1} \\ \vdots \\ y_m^{\gamma_m} \end{bmatrix} &= \begin{bmatrix} C_1 A^{\gamma_1} \\ \vdots \\ C_m A^{\gamma_m} \end{bmatrix} x + \begin{bmatrix} C_1 A^{\gamma_1 - 1} B \\ \vdots \\ C_m A^{\gamma_m - 1} B \end{bmatrix} u \\
&= b_0 x + A_0 u
\end{aligned} \tag{32}
$$

If the decoupling matrix A_0 is nonsingular, then the static feedback law

$$u = A_0^{-1}\left[-b_0 x + v\right] \tag{33}$$

decouples the system into m SISO systems of chains of γ_j integrators each.

If A_0 is singular or close to being singular, then the application of approximate decoupling algorithm presented in section 3 results in a numerically robust dynamical controller. In case of linear systems, the singular value decomposition of A_k also provides the transformation matrix G_k, thereby simplifying the implementation (cf. [19] for details).

We propose a precompensation technique for linear systems which will reduce the computations further.

6.1 Precompensation and Normalization

If the given system Σ_0 is observable in addition to being controllable and invertible, then it is possible to precompensate for the far off transmission zeros before applying the decoupling algorithm. The idea is to transform the system into a canonical form in which one can identify the individual elements responsible for the far off transmission zeros. The precompensated system will need a controller of smaller dimension.

Recall (see [20], page. 333) that a completely observable square MIMO linear system can be transformed into the following "canonical form"

$$\begin{aligned} \dot{x}_c &= A_c x_c + B_c u \\ y &= C_c x_c \end{aligned}$$

where x_c are the states corresponding to the m outputs y_i and their derivatives upto the order of $\rho_i - 1$.

$$\begin{aligned} C_c &= \begin{bmatrix} c_1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & c_m \end{bmatrix} \\ c_i &= [1\,0\cdots0] \end{aligned}$$

with each c_i being a row vector of dimension ρ_i, where ρ_i are the observability indices of the system. A_c is in a MIMO "controllable canonical form" given by

$$A_c = \begin{bmatrix}
0 & 1 & & & \vdots & & & & & & & \\
 & & \ddots & & \vdots & & & & & & & \\
 & & & 1 & \vdots & & & & & & & \\
\ast & \ast & \ast & \ast & \vdots & \ast & \ast & \ast & \ast & \ast & \ast & \ast \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
 & & & & \vdots & \ddots & & \vdots & & & & \\
 & & & & \vdots & & \ddots & \vdots & & & & \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
 & & & & & & & \vdots & 0 & 1 & & \\
 & & & & & & & \vdots & & & \ddots & \\
 & & & & & & & \vdots & & & & 1 \\
\ast & \ast & \ast & \ast & \ast & \ast & \ast & \ast & \ast & \ast & \ast & \ast
\end{bmatrix}$$

The diagonal blocks are of the size $\rho_i \times \rho_i$. All the other elements which are not shown are zero except the m rows of A_c corresponding to the last row of each block which might have nonzero elements.

Define the set of integers $\delta_i, i = 1, \cdots, m$ to be $\delta_1 = \rho_1$ and $\delta_i = \sum_{j=1}^{i} \rho_i, i = 2, \cdots, m$. Complete controllability of the system implies that the m rows of B_c, corresponding to the row numbers δ_i are linearly independant. Without loss of generality we assume that δ_j^{th} element of the j^{th} column of B_c is non zero. This means that we can control the j^{th} output and its derivatives by using the j^{th} input.

In this formulation, the structure of B_c provides important information about the transmission zeros of the system. Let B^j denote the j^{th} column of B_c. Consider the following two cases:

1. If for each column B^j, all the elements B_i^j are zero except for $i = \delta_{j-1} + 1, \cdots, \delta_j$ and $i = \delta_j, \forall\, j = 1, \cdots, m$, then

   - The system Σ_0 is decouplable by static state feedback.
   - The m polynomials

   $$B^j_{\delta_{j-1}+1} s^{\rho_j - 1} + B^j_{\delta_{j-1}+2} s^{\rho_j - 2} + \cdots + B^j_{\delta_j} = 0 \quad j = 1, \cdots, m \tag{34}$$

   determine the finite transmission zeros of the system. The precompensation scheme given below can then be applied.

2. If the above condition in case 1 is not valid, then

   - A dynamic controller may be needed to decouple the system Σ_0.
   - The precompensation scheme similar to case 1 can still be applied and works for most of the cases. But these few exceptional cases might make a system noninvertible. The categorization of all these exceptions is not as yet complete.

Detailed calculations to establish one-to-one mapping between the neglected dynamics and the formal structure at infinity are still to be worked out. The ongoing work on the CAD package called AP_LIN ([21]) based on this approach will help automate the application of this theory so that it can be easily used in practice. Although this approximate algorithm is based on the Descusse-Moog dynamic decoupling algorithm, a similar algorithm based on the dynamic extension algorithm (see [1], Chapter 7) can be worked out in similar fashion.

A Existence of a smooth matrix $G_k(x)$.

The proof of existence of $\hat{G}_k(x)$ is given by Descusse and Moog in [3]. Thus we have a matrix $\hat{A}_k(x)$ whose ϵ numerical rank is w_k and the last $m - r_k$ columns are identically zero.

From the definition of ϵ numerical rank of $\hat{A}_k(x)$, it is possible to find a $w_k \times w_k$ minor, say, $\Delta_k(x)$, such that all the singular values of $\Delta_k(x)$ are bigger than ϵ in U. Without loss of generality, we assume that $\Delta_k(x)$ is the block formed by the first w_k rows and the first w_k columns of $A_k(x)$, and let $\delta(x)$ represent its determinant.

By definition, any minor of $\hat{A}_k(x)$ having size bigger than w_k, will have at least one singular value smaller than ϵ uniformly in U. Thus the determinant of this minor will be of the order of $\epsilon \times \delta(x)$. Thus we get,

$$\det \begin{bmatrix} & & & \vdots & A_{1j}(x) \\ & \Delta_k(x) & & \vdots & \vdots \\ & & & \vdots & A_{w_k j}(x) \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ A_{i1}(x) & \cdots & A_{iw_k}(x) & \vdots & A_{ij}(x) \end{bmatrix} = \epsilon\, \delta(x) \qquad (35)$$

$$\forall i \in \{w_k + 1, \cdots, m\},\ \forall j \in \{w_k + 1, \cdots, r_k\}$$

Consider the top left $r_k \times r_k$ block of $\hat{A}_k(x)$. We get

$$\delta(x) A_{ij}(x) + \sum_{i=1}^{r_k} \lambda_l(x) A_{il}(x) = \text{order } \epsilon \times \delta(x),\ \forall i \in \{w_k + 1, \cdots, m\},\ \forall j \in \{1, \cdots, r_k\} \quad (36)$$

where $\lambda_l(x)$ is the cofactor of $A_{il}(x)$, calculated with respect to the top left $r_k \times r_k$ block of $\hat{A}_k(x)$.

For case 1, we can get a numerically robust controller by the following precompensation technique.

Precompensation: Recall that the polynomials (34) determine the finite transmission zeros of the system. The precompensation procedure approximates those coefficients of each polynomial that give rise to far off zeros. The precompensation works for each polynomial individually.

1. If all the coefficients of j^{th} polynomial are smaller than ϵ, then normalize the j^{th} input by $\frac{u_j}{\epsilon}$.
2. Divide each coefficient of the j^{th} polynomial by $B^j_{\delta_j}$ and denote the normalized coefficients by $p^j_0 = 1$, $p^j_i = \frac{B^j_i}{B^j_{\delta_j}}$ $i = \delta_{j-1}+1, \cdots, \delta_j - 1$.

   Thus the normalized j^{th} polynomial will be given by

   $$p^j_{\rho_j-1}\ s^{\rho_j-1} + p^j_{\rho_j-2}\ s^{\rho_j-2} + \cdots + p^j_1\ s + 1 = 0$$

   Start with $p^j_{\rho_j-1}$ and then examine the magnitude of these coefficients all the way upto p^j_0. Ignore the zero coefficients.

   If the first nonzero coefficient is bigger than ϵ, stop! This polynomial does not need precompensation.

   If there exists two positive integers $k, l < \rho_j; k \leq l$ such that

   - $|p^j_k| < \epsilon$, $|p^j_{k+1}| < \epsilon^2, \cdots, |p^j_l| < \epsilon^{l-k+1}$
   - $|p^j_{k-1}| \geq 1$

   then we can approximate the first k elements of p^j by zero. The approximated j^{th} polynomial will be given by

   $$p^j_{k-1}\ s^{k-1} + \cdots + p^j_1\ s + 1 = 0$$

   Thus we have to approximate the corresponding elements of B_c by zero. This corresponds to neglecting the transmission zeros whose real parts are of the order of $\frac{1}{\epsilon}$.

7 Conclusions

A numerically robust algorithm for input output decoupling of nonlinear dynamical systems has been proposed. This algorithm provides low gain, practically implementable controllers which in addition does not cancel far off zeros. The use of this algorithm for a slightly nonminimum phase system (i.e. one which has far off right half plane zeros), results in an overall stable closed loop system. It is shown that the tracking controllers constructed by using this approximate decoupling algorithm result in bounded tracking with stability. Controllers based on this theory already exist for a few specific examples in the literature and this paper can be thought of as an attempt to formalize the techniques used in those particular examples.

Define the elementary column operation by

$$g_j(x) := \begin{bmatrix} 1 & & & & & \lambda_1(x) & & & \\ & \ddots & & 0 & & \vdots & & & \\ & & 1 & & & \lambda_{r_k}(x) & & & \\ & & & 1 & & 0 & & & \\ & & & & \ddots & \vdots & & 0 & \\ & & & & 1 & 0 & & & \\ & 0 & & & & \delta(x) & & & \\ & & & & & 0 & 1 & & \\ & & & & & \vdots & & \ddots & \\ & & & & & 0 & & & 1 \end{bmatrix}$$

Where $\delta(x)$ is in the j^{th} row and column. The first $r_k - 1$ elements in the j^{th} column of $\hat{A}_k(x)g_j(x)$ will be zero. The r_k^{th} element will be the determinant given by (36), which is of the order of ϵ. The rest of the elements in this column must be of the order of ϵ, else there will be a minor of $\hat{A}_k(x)g_k(x)$ having all its singular values more than ϵ and having more than w_k columns and rows. This will contradict the definition of the ϵ numerical rank of a matrix. Thus this particular procedure makes the elements of j^{th} column of the order of ϵ as compared with $\delta(x)$. We can have $r_k - w_k$ matrices of these form making one column of $\hat{A}_k(x)g_i(x)$ *small* at a time. It is clear that the matrix $\tilde{G}_k(x)$ will thus be nonsingular, square and a smooth function of x.

Thus $G_k(x) := \hat{G}_k(x)\tilde{G}_k(x)P_k(x)$ is a square invertible matrix of smooth functions of x.

References

[1] A. Isidori, *Nonlinear Control Systems.* Springer-Verlag, 2nd ed., 1989.

[2] H. Nijmeijer and A. J. van der Schaft, *Nonlinear Dynamical Control Systems.* Springer-Verlag, 1990.

[3] J. Descusse and C. Moog, "Decoupling with dynamic compensation for strong invertible affine non-linear systems," *International Journal of Control*, vol. 42, no. 6, pp. 1387–1398, 1985.

[4] H. Nijmeijer and J. M. Schumacher, "The regular local non-interacting control problem for nonlinear control systems," *SIAM Journal of Control and Optimization*, vol. 24, no. 1986, pp. 1232–1245, 1986.

[5] H. Nijmeijer and W. Respondek, "Decoupling via dynamic compensation for nonlinear control systems," in *IEEE Control and Decision Conference*, pp. 192–197, 1986.

[6] J. Hauser, S. Sastry, and G. Meyer, "Nonlinear control design for slightly non minimum phase systems- applications to V/STOL aircraft," *Automatica*, vol. 28, no. 4, pp. 665–679, 1992.

[7] S. N. Singh, "Control of nearly singular decoupling systems and nonlinear aircraft maneuver," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 24, no. 6, pp. 775–784, 1988.

[8] J. J. Romano and S. N. Singh, "I-O map inversion, zero dynamics and flight control," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 26, no. 6, pp. 1022–1029, 1990.

[9] J. P. Barbot, N. Pantalos, S. Monaco, and D. Normand-Cyrot, "On the control of singularly perturbed nonlinear systems," in *Proceedings of the IFAC Workshop on Nonlinear Control Systems Design, NOLCOS II*, 1992.

[10] J. W. Grizzle and M. D. D. Benedetto, "Approximation by regular input-output maps," *IEEE Transactions on Automatic Control*, vol. 37, no. 7, pp. 1052–1055, 1992.

[11] S. Sastry, J. Hauser, and P. Kokotović, "Zero dynamics of regularly purturbed systems may be singularly purturbed," *Systems and Control Letters*, vol. 13, pp. 299–314, 1989.

[12] A. Isidori, S. Sastry, P. Kokotović, and C. Byrnes, "Singularly purturbed zero dynamics of nonlinear systems," *IEEE Transactions on Automatic Control*, vol. AC-37, no. 10, pp. 1625–1631, 1992.

[13] J. Descusse and C. Moog, "Dynamic decoupling for right-invertible nonlinear systems," *Systems and Control Letters*, vol. 8, pp. 345–349, 1987.

[14] M. D. Benedetto and J. Grizzle, "Intrinsic notions of regularity for local inversion, output nulling and dynamic extension of nonsquare systems," *Control Theory and Advanced Technology*, vol. 6, pp. 357–381, 1990.

[15] A. Isidori and C. Moog, "On the nonlinear equivalent of the notion of transmission zeros," in *Lecture notes in Control and Information Sciences* (C. Byrnes and A. Kurzhanski(Eds), eds.), pp. 147–158, New York: Springer-Verlag, 1988.

[16] P. Kokotović, H. Khalil, and J. O'Reilly, *Singular perturbation methods in control: analysis and design.* Academic Press, 1978.

[17] J. Grizzle, M. D. D. Benedetto, and F. Lamnabhi-Lagarrigue, "Necessary conditions for asymptotic tracking in nonlinear systems," tech. rep., Control Group report no. GCR 91-4, University of Michigan, Ann Arbor, 1991.

[18] M. Vidyasagar, *Nonlinear Systems Analysis.* Prentice-Hall, 2nd ed., 1992.

[19] C. Damour and S. Vieules, "Software for exact and approximate dynamic decoupling and inversion," tech. rep., Department of EECS, University of Michigan, Ann Arbor, 1992.

[20] C. T. Chen, *Linear System Theory and Design.* Holt, Rinehart, and Winston, 1984.

[21] R. R. Kadiyala, *AP_LIN: A CAD Tool Box for Nonlinear Control Design*. PhD thesis, Department of Electrical Engineering, University of California, Berkeley, California, 1992.