

# Zero Finding via Feedback Stabilisation

T. Mylvaganam\* A. Astolfi\*\*

\* *Department of Aeronautics, Imperial College London, London SW7 2AZ, UK (Email: thulasi.mylvaganam06@imperial.ac.uk).*

\*\* *Department of Electrical and Electronic Engineering, Imperial College London, London SW7 2AZ, UK and Dipartimento di Ingegneria Civile e Ingegneria Informatica, Università di Roma “Tor Vergata”, Via del Politecnico 1, 00133 Roma, Italy (Email: a.astolfi@imperial.ac.uk).*

---

**Abstract:** Two iterative algorithms for solving systems of linear and nonlinear equations are proposed. For linear problems the algorithm is based on a control theoretic approach and it is guaranteed to yield a converging sequence for any initial condition provided a solution exists. Systems of nonlinear equations are then considered and a generalised algorithm, again taking inspiration from control theory, is proposed. Local convergence is guaranteed in the nonlinear setting. Both the linear and the nonlinear algorithms are demonstrated on a series of numerical examples.

Keywords: Numerical algorithms, numerical methods, convergence of numerical methods, control system design, numerical analysis

---

## 1. INTRODUCTION

A key objective which is central to control theory is to feedback stabilise an equilibrium of a system, *i.e.* given a system one seeks to design a *control input* which renders an equilibrium of the system globally (or locally) asymptotically stable. A consequence of this is that the state of the system converges to a certain equilibrium point. Moreover, it may be desirable that a certain *convergence rate* be achieved. For linear systems the rate of convergence is determined by the locations of the eigenvalues associated with the closed-loop system (Dorf and Bishop [2011]). In contrast, stability analysis for nonlinear systems often relies on Lyapunov methods (Isidori [1995], Khalil [1996]). On the other hand, numerical analysis concerns algorithms used to obtain solutions to mathematical problems when closed-form solutions are not readily available. In general, numerical methods can be divided into two approaches: direct methods and iterative methods. Direct methods, such as Gaussian elimination and the simplex method, allows to compute the solution of a problem in a finite number of steps. Moreover, given infinite precision, a solution found via such methods is exact. Differently from direct methods, iterative methods yield sequences which converge to a solution of the underlying mathematical problem. Examples of such methods include Newton’s method. In practice, if an iterative method is convergent, it is terminated when a sufficiently accurate solution is obtained, thus providing an approximate solution of the problem.

Several links can be drawn between numerical analysis and dynamical systems theory: for example, the solution of a system of equations can be interpreted as the equilibrium of a dynamical system. There is a further similarity

between the *dynamics* of a system and *iterations* of an iterative numerical method. Thus, there is a clear analogy between the control theoretic problem of designing a stabilising controller and that of determining a convergent iterative method. In this paper we explore the similarities between the two notions and propose a numerical method for obtaining the solution of a system of linear equations which is centred about a control theoretic approach. Due to its simplicity (in terms of computational requirements), the developed technique is well-suited for large-scale problems. Exploiting the system theoretic interpretation of the proposed numerical method, it is modified to solve systems of nonlinear equations, with local convergence results.

Different iterative methods for solving systems of equations exist. The Jacobi method and the Gauss-Seidel method, which fall within the category of stationary iterative algorithms, are two such examples (see, for instance, Ortega [1990]). The convergence of these algorithms can only be guaranteed for certain classes of problems (see Ortega [1990]). Iterative methods for nonlinear systems of equations include the well-known Newton’s method and variations thereof (see, for example, Dennis Jr. and Schnabel [1996], Ortega and Rheinboldt [1970]). However, the convergence of Newton’s method for nonlinear equations relies heavily on the selection of the initial condition and the method is not generally globally convergent (Dennis Jr. and Schnabel [1996]). Other iterative methods include a range of algorithms for solving unconstrained, convex optimisation problems (see, for example, Boyd and Vandenberghe [2004]) or Krylov subspace methods (see, for example, Van Der Vorst [2000]). A summary of developments made in the 20th century, relating to iterative methods for solving linear systems of equations can be found in Saad and Van Der Vorst [2000]. In Smale [1976] a so-called *global Newton method*, which relies on the introduction of

---

\* A preliminary version of this paper, which we have now withdrawn, was submitted to the American Control Conference.

an ordinary differential equations is provided. Properties of the method are further considered in Keenan [1981]. With roots in dynamical systems theory the global Newton method considered in Smale [1976] and Keenan [1981] has been developed in the context of mathematical economics.

The parallels between control theory and iterative algorithms have been studied from several different perspectives in the literature. In Bhaya and Kaszkurewicz [2007] iterative methods for solving linear and nonlinear equations are viewed as regulation problems for dynamical systems with feedback and several iterative methods are studied using this approach. The connections between the convergence of iterative methods and stability analysis based on Lyapunov functions are considered in Ortega [1973]. More recently, the convergence properties of several well-known iterative methods have been studied from a system theoretic point of view in Kashima and Yamamoto [2007]. The link between iterative algorithms and dynamical systems is also utilised in Harbor [1989] and Han and Han [2010]. Differently from the aforementioned literature, in Hasan et al. [2013] the effects of finite precision on numerical methods are investigated from a control theoretic point of view.

In this paper, iterative algorithms are proposed based on the discretisation of the ordinary differential equations (ODEs) describing dynamical systems with stabilising state feedback. The parallels between control theory and numerical analysis have not yet been fully exploited: the results in this paper are intended to highlight new ways in which these parallels can be taken advantage of. Moreover, this work is intended to pave the way for the construction and the study of convergence properties of iterative methods for solving systems of coupled algebraic Riccati equations (AREs) which arise in linear quadratic differential games (see, for example, Mylvaganam et al. [2015], Engwerda [2005]). Coupled AREs are, in general, not straight-forward to solve and, consequently, there is a need for novel numerical methods to deal with such problems.

The remainder of this paper is organised as follows. The problem of solving a system of linear equations is introduced along with a brief summary of available numerical methods before a novel approach for solving such systems of equations, based on a control theoretic approach, is provided in Section 2. The approach results in a globally convergent algorithm for solving systems of linear equations. Systems of nonlinear equations are then considered. The approach introduced in Section 2 is generalised and applied to nonlinear problems in Section 3, where a locally convergent algorithm for solving systems of nonlinear equations is provided. Both methods (for linear and nonlinear problems, respectively) and the resulting algorithms are illustrated on a series of numerical examples in Section 4. Finally, some concluding remarks and directions for further research are provided in Section 5.

The following standard notation is used in the remainder of this paper. The set of real numbers is denoted by  $\mathbb{R}$  and the open left half complex plane is denoted by  $\mathcal{C}^-$ . The identity matrix is denoted by  $I$ . The spectrum of the matrix  $A$  is denoted by  $\sigma(A)$ , *i.e.*  $\sigma(A) = \{\lambda_1, \dots, \lambda_N\}$ , where  $\lambda_i$ ,  $i = 1, \dots, N$ , are the eigenvalues of  $A$ . The

spectral radius of the matrix  $A$  is denoted by  $\rho(A)$ , *i.e.*  $\rho(A) = \max_i\{|\lambda_1|, \dots, |\lambda_N|\}$ . The notation  $A = \{a_{ij}\}$  is used as a compact representation of the matrix  $A$ . The weighted norm of a vector  $x \in \mathbb{R}^n$  is denoted by  $\|x\|_R$ , *i.e.*  $\|x\|_R = \sqrt{x^\top R x}$ , with  $R \in \mathbb{R}^{n \times n}$ . Finally, given a vector  $x \in \mathbb{R}^n$ ,  $\max_i\{|x_i|\}$  denotes the element of  $x$  with the largest magnitude.

## 2. AN ITERATIVE ALGORITHM FOR SYSTEMS OF LINEAR EQUATIONS

Consider the system of linear equations

$$Ax + d = 0, \quad (1)$$

where  $A \in \mathbb{R}^{n \times n}$ ,  $x \in \mathbb{R}^n$  and  $d \in \mathbb{R}^n$ , and suppose we want to find the solution  $x^*$  of equation (1). In what follows the following assumption is made.

*Assumption 1.* The matrix  $A$  in (1) is non-singular, *i.e.*  $\det(A) \neq 0$  (or, equivalently,  $\text{rank}(A) = n$ ).

Clearly, the solution of (1) is

$$x^* = -A^{-1}d. \quad (2)$$

However, it is of interest to obtain the solution without performing computationally demanding operations, such as matrix inversions. Alternative, numerical methods become particularly important for large-scale systems of equations, *i.e.* when  $n$  is large. In what follows, iterative numerical methods for solving the system of equations (1) are considered.

In Harbor [1989] iterative numerical methods for solving equations of the form (1) are considered from a control theoretic perspective. In particular, it is demonstrated that an iterative method, such as the well-known Jacobi iteration and the Gauss-Seidel iteration, can be interpreted as a feedback control system. An iterative method can thus be described as a discrete-time system and its convergence can be studied using standard tools for stability analysis. In Han and Han [2010] an algorithm for solving systems of nonlinear equations is provided. The algorithm proposed therein relies on the observation that a solution of such a system of equations can be interpreted as an equilibrium of a dynamical system (see also Deuffhard [2011]). In the remainder of this paper we pursue the idea of developing numerical methods based on *control theoretic concepts*. In particular, we seek to solve the following problem.

*Problem 1.* Consider the system of linear equations (1), where  $A$  satisfies Assumption 1. Derive an iterative numerical method which allows to determine the solution  $x^*$  (given by (2)) of the system without performing matrix inversions.

Drawing inspiration from Harbor [1989] and Han and Han [2010] an iterative method for solving Problem 1 which is centred around dynamical systems and feedback control is presented in this section. In particular, the problem of solving the system of linear equations (1) is recast as a problem of designing a dynamic feedback controller for a linear dynamical system. A similar, yet preliminary, approach has been adopted in Mylvaganam and Astolfi [2016a] and Mylvaganam and Astolfi [2016b] to obtain solutions for coupled algebraic Riccati equations (AREs), which are nonlinear.

To this end, consider the variables  $x(t) \in \mathbb{R}^n$  and  $u(t) \in \mathbb{R}^n$  and the extended system

$$\begin{aligned}\dot{x} &= Au + d, \\ \dot{u} &= w,\end{aligned}\quad (3)$$

with input  $w$ .

*Problem 2.* Consider the extended system (3). Design a state feedback  $w$  such that the closed-loop system has an exponentially stable equilibrium.

*Remark 1.* In the dynamic setting introduced above, the constant term  $d$  in (1) can be interpreted as a static disturbance affecting the dynamical system (3).  $\blacktriangle$

Let  $K_i \in \mathbb{R}^{n \times n}$ , for  $i = 1, 2$ , and let  $A_{cl} = \begin{bmatrix} 0 & A \\ K_1 & K_2 \end{bmatrix}$ . A solution of Problem 2 is provided by the following simple statement.

*Lemma 1.* Let  $K_i$ ,  $i = 1, 2$ , be such that  $\sigma(A_{cl}) \in \mathcal{C}^-$  and suppose  $K_1$  is invertible. Then the state feedback

$$w = K_1 x + K_2 u, \quad (4)$$

is such that the dynamical system (3) is exponentially stable and  $\lim_{t \rightarrow \infty} u(t) = x^*$ .  $\diamond$

Substituting (4) into (3), the closed-loop system is described by

$$\begin{bmatrix} \dot{x} \\ \dot{u} \end{bmatrix} = A_{cl} x + \begin{bmatrix} d \\ 0 \end{bmatrix}. \quad (5)$$

A selection of  $K_1$  and  $K_2$  satisfying the assumptions in Lemma 1, and which does not require computing matrix inverses, is provided in the following statement.

*Proposition 1.* The selection

$$K_1 = -A^\top, \quad K_2 = -M, \quad (6)$$

for some positive definite matrix  $M$ , is such that  $\sigma(A_{cl}) \in \mathcal{C}^-$ .  $\diamond$

*Remark 2.* Any positive definite matrix  $M$  is such that the equilibrium of the closed-loop system (5) is exponentially stable. A simple selection is then provided by

$$M = \lambda I, \quad (7)$$

where  $\lambda > 0$ .  $\blacktriangle$

Lemma 1 implies that the solution of (1) can be obtained by considering the evolution of the states of the closed-loop system (5). Thus, the discretisation of (5) starting from the initial conditions  $x(0) = x_0$  and  $u(0) = u_0$  results in an iterative algorithm for obtaining the solution of (1) as shown in Table 1, where  $T$  is the step size and  $\epsilon > 0$  is a *tolerance value*, indicating the accuracy required by the algorithm. In what follows the algorithm is referred to as Algorithm A-LI. Algorithm A-LI in Table 1 results from applying the Euler method for differential equations to the ordinary differential equation (5) and it is guaranteed to converge for a sufficiently small step size (provided  $K_1$  and  $K_2$  are such that  $A_{cl} \in \mathcal{C}^-$ ). Thus, Algorithm A-LI provides a solution for Problem 1, *i.e.* it provides a simple algorithm for obtaining an asymptotic solution of (1). The conditions for Algorithm A-LI to converge, for any initial condition  $x_0$  and  $u_0$ , are provided in the following statement.

Table 1. A-LI: Iterative algorithm for linear equations

---

<b>Input:</b> Parameters $A, d, K_1, K_2, T, \epsilon$ .
<b>Output:</b> Estimated solution $x^*$ .
1 : <b>Initialization:</b> Set initial conditions $x_0, u_0$ .
2 : <b>Repeat</b>
3 : $x_{k+1} := x_k + (Au_k + d)T$
4 : $u_{k+1} := u_k + (K_1 x_k + K_2 u_k)T$
5 : $k := k + 1$
6 : <b>Until</b> $\max_i \{ Ax_k + d _i\} < \epsilon$
7 : $x^* = u_k$
8 : <b>End</b>

---

*Lemma 2.* (Chapter 7, Ortega [1990]). Suppose that the parameters  $K_1, K_2$  and  $T$  in Algorithm A-LI are selected such that  $\rho(I + A_{cl}T) < 1$ . Then Algorithm A-LI is convergent, *i.e.*  $\lim_{k \rightarrow \infty} x_k = x^*$ .  $\diamond$

*Remark 3.* The matrices  $K_1$  and  $K_2$  and the step size  $T$  can be selected to maximise the speed of convergence of Algorithm A-LI. In fact, the rate of convergence is determined by  $\rho(H)$  and the aforementioned parameters can be selected to minimise  $\rho(H)$ , thus maximising the rate of convergence (see Chapter 7, Ortega [1990]). In some cases it may be useful to apply Gershgorin's Theorem, or generalisations thereof (see, for instance Feingold and Varga [1962]), to minimise the spectral radius of  $H$  without the direct computation of its eigenvalues.  $\blacktriangle$

The speed of the algorithm may be further improved according to the following two remarks, essentially relating to the method of discretising the continuous-time dynamical system (3) and the the selection of the step size  $T$ .

*Remark 4.* Different schemes for discretising the continuous-time system (3) resulting in iterations different from those in Algorithm-A-LI can be considered and may result in improved convergence rates. Higher order one-step methods such as the classic Runge-Kutta method or multistep methods may be considered (see, for example, Chapter 5, Ortega [1990]).  $\blacktriangle$

*Remark 5.* The step size  $T$  is constant in Algorithm A-LI. Utilising an adaptive step size may prove useful in improving the algorithm. Adaptive schemes are, for example, considered in Ilie et al. [2008]. Note that the system (5) with  $K_1$  and  $K_2$  in (6) is a Hamiltonian system. It may therefore be of interest to explore discrete-time models for port-controlled Hamiltonian systems as given in Laila and Astolfi [2006] and Stuart and Humphries [1998].  $\blacktriangle$

### 3. AN ITERATIVE ALGORITHM FOR SYSTEMS OF NONLINEAR EQUATIONS

Algorithm A-LI in Section 2 is applicable only to systems of linear equations. However, the algorithm can be generalised to deal with systems of nonlinear equations. To this end, consider the system of equations

$$f(x) = 0, \quad (8)$$

where  $x \in \mathbb{R}^n$  and  $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$  is a smooth mapping, and suppose we want to find a solution  $x^*$  satisfying (8), *i.e.*  $f(x^*) = 0$ . As in the case of linear systems discussed in Section 2, it is often of interest to obtain such a solution

numerically. That is, we seek to solve the following problem (which is a direct extension of Problem 1).

*Problem 3.* Consider the system of nonlinear equations (8). Derive an iterative numerical method which allows to determine the solution  $x^*$  of the system.

In the following a second algorithm, generalised to handle systems of nonlinear equations, is provided.

*Remark 6.* Unlike the case with systems of linear equations, systems of nonlinear equations of the form (8) may have several solutions. Therefore, in what follows, *local solutions* for Problem 2 are considered.  $\blacktriangle$

Let  $x^*$  denote a solution of (8). Consider, as in the linear case in Section 2, the variables  $x(t) \in \mathbb{R}^n$  and  $u(t) \in \mathbb{R}^n$  and the extended system

$$\begin{aligned}\dot{x} &= f(u), \\ \dot{u} &= w,\end{aligned}\tag{9}$$

with  $w$  as input. The “nonlinear equivalent” of Problem 2 follows.

*Problem 4.* Consider the extended system (9). Design a state feedback  $w$  such that the closed-loop system has a (locally) asymptotically stable equilibrium at  $u = x^*$ .

*Remark 7.* There exists a matrix-valued function  $F(u) : \mathbb{R}^n \rightarrow \mathbb{R}^{n \times n}$  such that the system (9) can be written in the form<sup>1</sup>

$$\begin{aligned}\dot{x} &= F(u)u + d, \\ \dot{u} &= w,\end{aligned}\tag{10}$$

where  $d = f(0)$ .  $\blacktriangle$

Suppose an equilibrium, denoted by  $(x_e^\top, u_e^\top)^\top$  of the system (9) exists and consider the function  $V : \mathbb{R}^n \rightarrow V$

$$V(x, u) = \frac{1}{2} \begin{bmatrix} (x - x_e)^\top & (u - u_e)^\top \end{bmatrix} \begin{bmatrix} \lambda I & \delta F(u_e) \\ \delta F(u_e)^\top & \lambda I \end{bmatrix} \begin{bmatrix} x - x_e \\ u - u_e \end{bmatrix},\tag{11}$$

with  $\lambda > 0$  and  $\delta > 0$ . Let  $M : \mathbb{R}^n, \mathbb{R}^n \rightarrow \mathbb{R}^{n \times n}$  denote a positive definite matrix-valued function, *i.e.*  $M(x, u) > 0$  for all  $x$  and  $u$ , and let  $\Delta_i : \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}^{n \times n}$  be matrix valued mappings such that  $\Delta_1(u, u_e)(u - u_e) = F(u)u - F(u_e)u_e$  and  $\Delta_2(u, u_e, x_e)^\top(x - x_e) = F(u)^\top x - F(u_e)^\top x_e$ . Finally, let  $\Gamma_{11} = \frac{1}{2}\delta(F(u_e)\Delta_2^\top + \Delta_2F(u_e)^\top)$ ,  $\Gamma_{12} = \frac{1}{2}(\delta F(u_e)M + \lambda(\Delta_2 - \Delta_1))$  and  $\Gamma_{22} = \lambda M - \frac{1}{2}\delta(F(u_e)^\top \Delta_1 + \Delta_1^\top F(u_e))$ .

*Proposition 2.* Let

$$w = -F(u)^\top x - M(x, u)u.\tag{12}$$

Suppose  $F(u_e)$  is invertible,  $\lambda$  and  $\delta$  are such that

$$\lambda I - \frac{\delta^2}{\lambda} F(u_e)^\top F(u_e) > 0,\tag{13}$$

and suppose the trajectories of the closed-loop system (9)-(12) satisfy

$$\Gamma_{11} > 0, \quad \Gamma_{22} - \Gamma_{12}^\top (\Gamma_{11})^{-1} \Gamma_{12} \leq 0\tag{14}$$

holds in a neighbourhood  $\mathcal{N}$  containing the equilibrium  $(x_e^\top, u_e^\top)^\top$ . Then the dynamic feedback  $(u, w)$  is such that

<sup>1</sup> The statement follows directly from the observations that  $\dot{x} = f(u) - f(0) + f(0)$  and that the first two terms can be written as  $f(u) - f(0) = F(u)u$ .

Table 2. A-NLI: Iterative Algorithm for Non-linear Equations

---

<b>Input:</b> Parameters $f(x)$ , $M(x_k, u_k)$ , $T, \epsilon$ .
<b>Output:</b> Estimated solution $x^*$ .
1 : <b>Initialization:</b> Set initial conditions $x_0, u_0$ .
2 : <b>Repeat</b>
3 : $x_{k+1} := x_k + f(u_k)T$
4 : $u_{k+1} := u_k - (F(u_k)^\top x_k + M(x_k, u_k)u_k)T$
5 : $k := k + 1$
5 : <b>Until</b> $\max_i \{ f(u_k) _i\} < \epsilon$
6 : $x^* = u_k$
7 : <b>End</b>

---

the equilibrium  $(x_e, u_e)$  is (locally) exponentially stable and  $\lim_{t \rightarrow \infty} u(t) = x^*$  where  $x^*$  is the solution of (8).  $\diamond$

*Remark 8.* In the linear case  $F$  is constant and  $\Delta_1 = \Delta_2 = A$ . Thus, in the linear case the inequalities (14) is trivially satisfied (for suitable values of  $\lambda$  and  $\delta$ ). From a control theoretic perspective the challenge, which becomes apparent in the nonlinear case, is that we seek to stabilise an *unknown* equilibrium of the system.  $\blacktriangle$

Similarly to what has been done in Section 2 for systems of linear equations, the above statement is utilised to develop an algorithm for solving Problem 2. In particular, the iterative numerical algorithm, given in Table 2, for obtaining an approximate solution of (8) follows from the discretisation of the closed-loop systems (9)-(12). Henceforth the algorithm will be referred to as Algorithm A-NLI. Similarly to what has been seen in Section 2 the Algorithm A-NLI in Table 2 simply represents the Euler approximation of the dynamical system (9).

*Remark 9.* As seen in Lemma 2 in the linear case, convergence of algorithm A-NLI depends on the step size  $T$ . Similarly to the observations made in Remark 3 the parameters may be selected to maximise the rate of convergence and, as seen in Remarks 4 and 5, alternative methods of discretising the dynamical system (9) may yield further improvements.  $\blacktriangle$

## 4. NUMERICAL EXAMPLES

Three numerical example are provided in this section. A system of linear equations, a nonlinear equation, which is a scalar ARE, and a system of nonlinear equations are considered separately in the following.

### 4.1 System of Linear Equations

Consider first a system of equations of the form (1) with 1000 unknowns, *i.e.*  $n = 1000$  and the error

$$e_k = \max_i \{|Au_k + d|_i\},\tag{15}$$

which quantifies the maximum (absolute) residue at each time. Note that this provides also the termination criterion of the algorithm. A matrix  $A = \{a_{ij}\}$ , such that  $A \in \mathbb{R}^{1000 \times 1000}$ , and such that its diagonal elements satisfy  $1 < a_{ii} < 10$  and its non-diagonal elements satisfy  $-0.1 < a_{ij} < 0.1$ , and a vector  $d \in \mathbb{R}^{1000}$  are randomly generated in MATLAB. Algorithm A-LI shown in Table 1

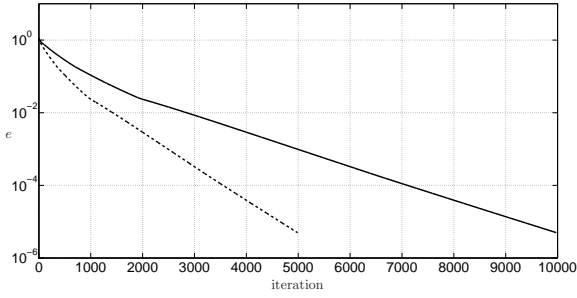


Fig. 1. Time history of the error  $e$  associated with Algorithm A-LI (solid line) and with an alternative algorithm based on the Runge-Kutta method (dashed line).

is applied with the following selection of parameters. The initial condition  $x_0 = u_0 = 0$  is selected, the sampling time is taken to be  $T = 0.05$  seconds and  $\epsilon = 5 \cdot 10^{-6}$ . In line with Proposition 1  $K_1 = -B^\top$  and  $M = 10I$ . Following Remark 4 an alternative to Algorithm A-LI in which the discretisation is done using the classical (fourth order) Runge-Kutta method in place of the Euler method (see, for instance, Chapter 5, Ortega [1990]), is also considered, for the same selection of parameters with the exception of the step size which is taken to be  $T = 0.1$ . Note that once the desired accuracy is reached the algorithms are terminated. The time history of the error  $e$  is shown in Figure 1 for Algorithm A-LI (solid line) and for the alternative algorithm based on the fourth order Runge-Kutta method (dashed line). Note that, despite the parameters not having been selected to optimise the convergence rate, the alternative algorithm requires half the number of iterations compared to Algorithm A-LI, indicating that, as noted in Remarks 4 and 5, the convergence rate may be further improved.

#### 4.2 Scalar ARE

Consider the case in which  $n = 1$  and

$$f(x) = 2ax + 2b^2x^2 + q, \quad (16)$$

where  $a \in \mathbb{R}$ ,  $b \in \mathbb{R}$  and  $q \in \mathbb{R}$ . Note that, similarly to one of the examples considered in Mylvaganam and Astolfi [2016b], the quadratic equation (16) is a scalar ARE. Note also that (16) can be written in the form (10) in Remark 7. Thus, Algorithm A-NLI can be implemented with  $F(u) = 2a + 2b^2u$  and  $d = q$ . Consider the case in which  $a = -0.1$ ,  $b = 0.2$  and  $q = 10$ . The parameters for Algorithm A-NLI are selected as  $M = 10$ ,  $\epsilon = 10^{-6}$  and  $T = 0.2$ . The resulting time histories of  $x$  (top) and  $u$  (bottom) are shown in Figure 2, whereas the time history of the nonlinear error

$$e_k = \max_i \{|f(u_k)|_i\}, \quad (17)$$

is shown in Figure 3. The circular markers and dashed vertical lines in Figures 2 and 3 indicate the iteration at which Algorithm A-NLI terminates and returns a solution of (16), namely  $x^* = 13.5078$ . In this case the algorithm converges to the positive root. Note that in the context of optimal control and differential games one seeks to obtain positive definite solutions of AREs.

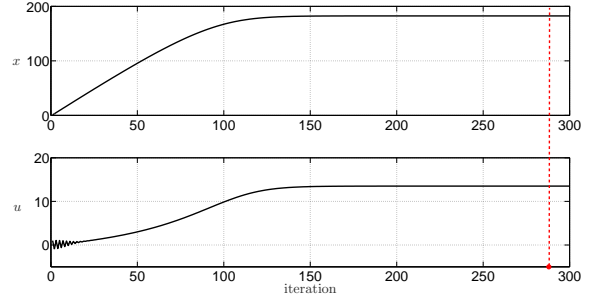


Fig. 2. Time histories of  $x$  (top) and  $u$  (bottom).

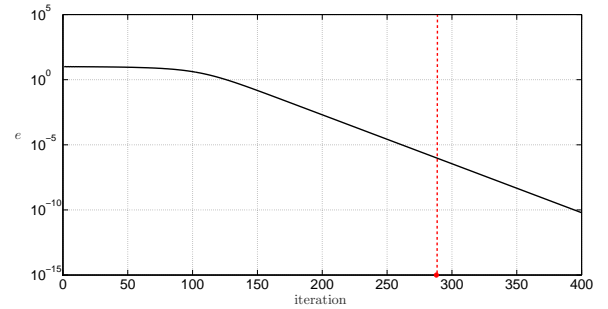


Fig. 3. Time history of the error  $e$ .

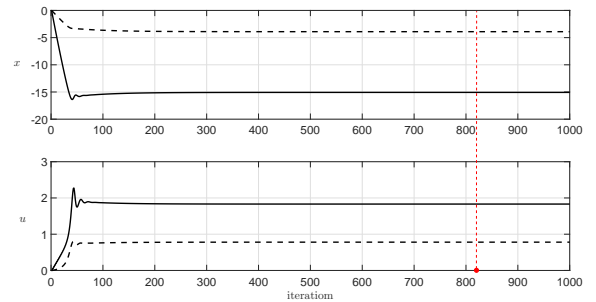


Fig. 4. Top: time histories of  $x_1$  (solid line) and  $x_2$  (dashed line). Bottom: time histories of  $u_1$  (solid line) and  $u_2$  (dashed line).

#### 4.3 System of Nonlinear Equations with Two Unknowns

Consider the system

$$f(x) = \underbrace{\begin{bmatrix} a_1x_1x_2 + a_2 \sin x_2 \\ 0 \\ a_3x_2 + a_4 \end{bmatrix}}_{F(x)} x + \underbrace{d}_{f(0)},$$

and consider the case in which  $a_1 = a_2 = a_3 = 1$ ,  $a_4 = 0.5$  and  $d = [-5, -1]^\top$ . Algorithm A-NLI is applied to the problem with  $M = 20I$ ,  $\epsilon = 10^{-6}$  and  $T = 0.1$ . Let  $u = [u_1, u_2]^\top$ . The time histories of  $x_1$  (solid line) and  $x_2$  (dashed line) are shown Figure 4 (top) whereas the time histories of  $u_1$  (solid line) and  $u_2$  (dashed line) are shown Figure 4 (bottom). The time history of the error (17) is shown in Figure 5. The circular markers and dashed vertical lines in Figures 4 and 5 indicate the iteration at which Algorithm A-NLI terminates, returning the solution  $x^* = [1.831, 0.7808]^\top$ . Note that the error response is not monotonic.

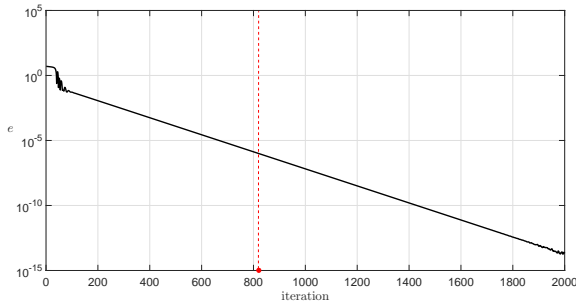


Fig. 5. Time history of the error  $e$ .

## 5. CONCLUSION

In this paper an iterative numerical algorithm for solving systems of linear equations are proposed. The algorithm, which takes its inspiration from control theory, is then generalised to obtain solutions for systems of nonlinear equations. The developed theory is demonstrated on a series of numerical examples.

Directions for future work include improving the performance of the proposed algorithms by using methods different from the Euler method for discretising the dynamical systems which form the foundations of the algorithms, following the observations made in Remarks 4 and 5 and in Section 4.1. It is also of interest to perform a more thorough analysis of complexity and convergence of the proposed algorithms, including a comprehensive comparison with other numerical methods. Underdetermined and overdetermined systems of linear equations will also be considered. Extensions of the proposed idea to matrix equations and, in particular, coupled AREs, such as those encountered in Mylvaganam and Astolfi [2016b,a], will be investigated. Such systems of AREs typically arise in linear quadratic nonzero-sum differential games. As mentioned in Section 4.2, in optimal control and differential games, one typically seeks positive definite (and symmetric) solutions of the AREs. Therefore, systematic ways of selecting particular roots of the coupled AREs (if they exist) will be sought.

## REFERENCES

- A. Bhaya and E. Kaszkurewicz. A control-theoretic approach to the design of zero finding numerical methods. *IEEE Transactions on Automatic Control*, 52(6):1014–1026, 2007.
- S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge university press, 2004.
- J. E. Dennis Jr. and R. B. Schnabel. *Numerical methods for unconstrained optimization and nonlinear equations*, volume 16. Society for Industrial and Applied Mathematics, 1996.
- P. Deuffhard. *Newton methods for nonlinear problems: affine invariance and adaptive algorithms*, volume 35. Springer Science & Business Media, 2011.
- R. C. Dorf and R. H. Bishop. *Modern Control Systems*. Pearson Prentice Hall, 12th edition, 2011. ISBN 9780136024583.
- J.C. Engwerda. *LQ dynamic optimization and differential games*. Chichester: John Wiley & Sons, 2005.
- D. G. Feingold and R. S. Varga. Block diagonally dominant matrices and generalizations of the Gerschgorin circle theorem. *Pacific Journal of Mathematics*, 12(4):1241–1250, 1962.
- T. Han and Y. Han. Solving large scale nonlinear equations by a new ODE numerical integration method. In *Applied Mathematics*, volume 1, pages 222–229, 2010.
- R. D. Harbor. A control theory perspective of iterative numerical methods. In *IEEE Proceedings of the Twenty-First Southeastern Symposium on System Theory*, pages 71–73, 1989.
- A. Hasan, E. C. Kerrigan, and G. A. Constantinides. Control-theoretic forward error analysis of iterative numerical algorithms. *IEEE Transactions on Automatic Control*, 58(6):1524–1529, 2013.
- S. Ilie, G. Söderlind, and R. M. Corless. Adaptivity and computational complexity in the numerical solution of ODEs. *Journal of Complexity*, 24(3):341 – 361, 2008.
- A. Isidori. *Nonlinear Control Systems*. Springer, 1995.
- K. Kashima and Y. Yamamoto. System theory for numerical analysis. *Automatica*, 43(7):1156 – 1164, 2007.
- D. Keenan. Further remarks on the global Newton method. *Journal of Mathematical Economics*, 8(2):159 – 165, 1981.
- H. K. Khalil. *Nonlinear Systems*. Prentice-Hall, New Jersey, 1996.
- D. S. Laila and A. Astolfi. Construction of discrete-time models for port-controlled Hamiltonian systems with applications. *Systems & Control Letters*, 55(8):673–680, 2006.
- T. Mylvaganam and A. Astolfi. Dynamic algorithms for solving coupled algebraic Riccati equations arising in mixed  $H_2/H_\infty$  control for scalar linear systems. In *Submitted to the 55th Annual Conference on Decision and Control*, 2016a.
- T. Mylvaganam and A. Astolfi. A Nash game approach to mixed  $H_2/H_\infty$  control for input-affine nonlinear systems. In *Proceedings of the 10th IFAC Symposium on Nonlinear Control Systems*, 2016b.
- T. Mylvaganam, M. Sassano, and A. Astolfi. Constructive  $\epsilon$ -Nash equilibria for nonzero-sum differential games. *IEEE Transactions on Automatic Control*, 60(4):950–965, 2015.
- J. M. Ortega. Stability of difference equations and convergence of iterative processes. *SIAM Journal on Numerical Analysis*, 10(2):268–282, 1973.
- J. M. Ortega. *Numerical analysis: a second course*, volume 3. Siam, 1990.
- J. M. Ortega and W. C. Rheinboldt. *Iterative solution of nonlinear equations in several variables*, volume 30. Society for Industrial and Applied Mathematics, 1970.
- Y. Saad and H. A. Van Der Vorst. Iterative solution of linear systems in the 20th century. *Journal of Computational and Applied Mathematics*, 123(1):1–33, 2000.
- S. Smale. A convergent process of price adjustment and global Newton methods. *Journal of Mathematical Economics*, 3(2):107–120, 1976.
- A. Stuart and A. R. Humphries. *Dynamical systems and numerical analysis*, volume 2. Cambridge University Press, 1998.
- H. A. Van Der Vorst. Krylov subspace iteration. *Computing in Science and Engineering*, 2(1):32–37, 2000.