In this case, we can view $m_k = 0$ which is the smallest nonnegative integer for $\beta^{m_k}\lambda < (K_2/K)$. Now, for any $\lambda > 0$, $0 < \beta < 1$, and let m_k be either the m_k determined by (A12) or $m_k = 0$ by (A13); from (A11), there must exist an $m' \le m_k$ (or $m' = 0$ if $m_k = 0$) such that

$$J(x^k + \beta^{m'}\lambda\hat{s}^k) \le J(x^k) - \frac{K_2}{2}\beta^{m'}\lambda\|\hat{s}^k\|_2. \tag{A14}$$

Note that $0 < \gamma^k < (K_2/K)$ is sufficient for (A11) to hold explains why $m' \le m_k$. This shows that Step 8 of our algorithm will terminate for certain m'. Since $\beta^{m'} \ge \beta^{m_k}$, from (A12) and (A13), $\beta^{m'}\lambda \ge \min\{\beta\lambda, \beta(K_2/K)\}$. Let $\tau \equiv \frac{1}{2}K_2 \cdot \min\{\beta\lambda, \beta(K_2/K)\}$, then τ is finite and positive, and

$$J(x^k + \beta^{m'}\lambda\hat{s}^k) \le J(x^k) - \tau\|\hat{s}^k\|_2^2. \tag{A15}$$

Then, each iteration of our algorithm ensures a decrement of the objective function by at least the amount $\tau\|\hat{s}^k\|_2^2$. Since $J(x)$ is bounded from below, we assume $c \in \Re$ is a lower bound of $J(x)$, then from (A15), we have

$$0 \le J(x^{k+1}) - c \le J(x^k) - c - \tau\|\hat{s}^k\|_2^2, \quad \forall k. \tag{A16}$$

Then, by (A16), $\sum_{k=0}^{\infty}\|\hat{s}^k\|_2^2 \le (J(x^0) - c/\tau) < \infty$, and (A7) shows that $\lim_{k\to\infty}\nabla J(x^k) = 0$. □

REFERENCES

[1] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*. London: Prentice-Hall, 1989.

[2] L. O. Chua and P. M. Lin, *Computer Aided Analysis of Electronic Circuits*. Englewood Cliffs, NJ: Prentice-Hall, 1975.

[3] W. F. Tinney and J. W. Walker, "Direct solutions of sparse network equations by optimally ordered triangular factorization," *Proc. IEEE*, vol. 55, no. 11, pp. 1801–1809, Nov. 1967.

[4] C.-H. Lin, "A new parallel processing algorithm for power system state estimation problems," master's thesis, Dept. of Elec. Engr., Nat'l Tsing Hua Univ., Hsinchu, Taiwan, 1991 (in Chinese).

[5] F. C. Schweppe and J. Wildes, "Power system static-state estimation, part I: exact model," *IEEE Trans. Power App. Syst.*, vol. PAS-89, no. 1, pp. 120–125, Jan. 1970.

[6] F. C. Schweppe, " Power system static-state estimation, part III: implementation," *IEEE Trans. Power App. Syst.*, vol. PAS-89, no. 1, pp. 130–135, Jan. 1970.

[7] A. Simoes-Costa and V. H. Quintana, "A robust numerical technique for power system state estimation," *IEEE Trans. Power App. Syst.*, vol. PAS-100, pp. 691–698, Feb. 1981.

[8] A. Garcia, A. Monticelli, and P. Abreu, "Fast decoupled state estimation and bad data processing," *IEEE Trans. Power App. Syst.*, vol. PAS-98, pp. 1645–1652, Sep. 1979.

[9] S.-Y. Lin, C.-H. Lin, and S.-L. Yu, "An efficient descent algorithm for a class of unconstrained optimization problems of nonlinear large mesh-interconnected systems," in *Proc. 32nd IEEE Conf. Dec. & Contr.*, Dec. 1993.

[10] S.-Y. Lin and C.-H. Lin, "An implementable distributed state estimator and bad data processing schemes for electric power systems," *IEEE Trans. Power Syst.*, vol. 9, no. 3, pp. 1277–1284, Aug. 1994..

[11] S.-Y. Lin, "A parallel processing multi-coordinate descent method with line search – algorithm and convergence," in *Proc. 30th IEEE Conf. Deci. Contr.*, Dec. 1991, pp. 2096–2097.

[12] D. Luenberger, *Linear and Nonlinear Programming*, 2nd ed. Reading, MA: Addison-Wesley, 1984.

On the Possible Divergence of the Projection Algorithm

Erjen Lefeber and Jan Willem Polderman

***Abstract*—It is shown by means of an example that the projection algorithm does not always converge.**

I. INTRODUCTION

It is well known that parameter identification of linear systems depends very much on the excitation of the signals. Generally speaking, all identification algorithms require the signals to be sufficiently exciting. In applications such as adaptive control, however, excitation is often not possible. The question then arises how useful the standard identification schemes are. In this note we consider the case where the data can be modeled exactly by a linear time invariant discrete-time model. It is a fact, that for such systems recursive least squares always produce a convergent sequence of parameter estimates, although it is of course not guaranteed that the limit will be the true parameter [1].

For the projection algorithm a similar result or its negation is to the best of our knowledge not available in the literature. Properties that can be derived without any assumptions on the signals can be found in [1]. Nothing is said about convergence there (see also [2, Problem 12.14]). In [3], the algorithm is used for adaptive pole assignment. Since the adaptive algorithm could be analyzed without proving convergence of the parameter estimates, the possible convergence is not studied there either.

In this note we show by means of an example that the projection algorithm does not necessarily converge. This is in contrast with recursive least squares.

The construction of the counter example is as follows. Firstly we construct a sequence of real vectors that satisfies at least some of the properties of the projection algorithm and which does not converge. Secondly we show that the sequence could as well have been obtained by applying the projection algorithm to an appropriate input/output system. Hence, rather than fitting the estimates to the data, we fit the data to the estimates.

II. THE PROJECTION ALGORITHM

For the sake of completeness, we briefly describe the projection algorithm. Let the system be described by

$$y(k+1) = \bar{\theta}^T\phi(k) \quad \bar{\theta} \in \mathbb{R}^n. \tag{1}$$

The projection algorithm is defined as follows: Suppose that the estimate of $\bar{\theta}$ at time k is θ_k, define $G_{k+1} := \{\theta \in \mathbb{R}^n \mid y(k+1) = \theta^T\phi(k)\}$. Define θ_{k+1} as the orthogonal projection of θ_k on G_{k+1}. The recursion is given by

$$\theta_{k+1} = \theta_k + \frac{\phi(k)}{\|\phi(k)\|^2}(y(k+1) - \theta_k^T\phi(k)). \tag{2}$$

Notice that G_{k+1} contains the true parameter $\bar{\theta}$. Regardless of the input sequence, the following two properties hold.

Property 2.1: 1) For all k: $\|\bar{\theta} - \theta_{k+1}\| \le \|\bar{\theta} - \theta_k\|$.
2) $\lim_{k\to\infty}(\theta_{k+1} - \theta_k) = 0$.

It is obvious that from Property 2.1 we cannot conclude that θ_k is a fundamental sequence, and in fact we will see that it need not be.

Manuscript received November 10, 1993.

The authors are with the Department of Applied Mathematics, University of Twente, P.O. Box 217, 7500 AE Enschede, The Netherlands.

IEEE Log Number 9407235.

0018–9286/95$04.00 © 1995 IEEE

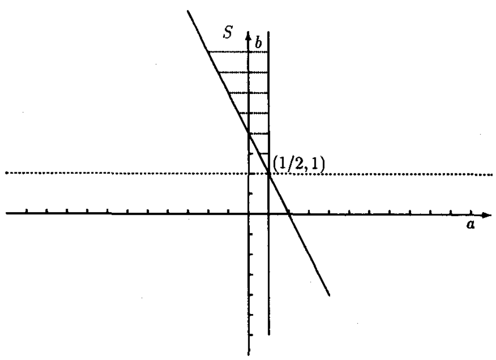

Fig. 1. The set S.

III. A Counterexample

The idea of the counterexample is that we will first construct a sequence $(a_k, b_k) \in \mathbb{R}^2$ with the properties that:

1) (a_{k+1}, b_{k+1}) is obtained from (a_k, b_k) by orthogonally projecting the latter onto a line passing through a fixed point.
2) The sequence does not converge.

Notice that this sequence is constructed in a similar way as the sequence of estimates in the projection algorithm. Subsequently we will show that the particular sequence is equal to the sequence of estimates produced by applying the projection algorithm to a particular first-order system. That will establish the claim that the algorithm does not necessarily produce a convergent sequence of estimates. The key idea is that we fit the data to the estimates rather than the estimates to the data.

A. Construction of the Sequence

The sequence $\{a_k, b_k\}$ will be defined inductively

$$(a_0, b_0) := (1/2, 4). \tag{3}$$

Suppose now that (a_k, b_k) has been constructed. Let L_k be the line passing through $(1/2, 1)$ and (a_k, b_k). Define (a_{k+1}, b_{k+1}) as the orthogonal projection on a line L_{k+1} yet to be defined. L_{k+1} will be a line passing through $(1/2, 1)$ with the property that the distance between (a_k, b_k) and its orthogonal projection on L_{k+1} is exactly $1/(k+1)$. There are two possibilities for L_{k+1}, one which requires a clockwise rotation of L_k to obtain L_{k+1} and one for which this rotation would be counter clockwise. This freedom of choice will now be used as follows. Define the region $S := \{(a, b) \mid -1 < -2a < b - 2 \wedge b > 1\}$. See Fig. 1. Determine the two possibilities for (a_{k+1}, b_{k+1}). When both points are in S, rotate L_k in the same direction as L_{k-1} was rotated to obtain L_k, to get L_{k+1}, otherwise rotate L_k in the opposite direction. L_1 of course requires a counter clockwise rotation of L_0. Notice that now every $(a_k, b_k) \in S$. Of course, the recursion could in principle be written in formulas; we feel, however, that this would not add much to our understanding.

Lemma 3.1: i) The sequence $\{(a_k, b_k)\}$ is well-defined. ii) The sequence $\{(a_k, b_k)\}$ does not converge.

Proof:

i) Define $r_k := \sqrt{(1/2 - a_k)^2 + (1 - b_k)^2}$ and $\delta_{k+1} := \sqrt{(a_{k+1} - a_k)^2 + (b_{k+1} - b_k)^2}$. From the construction it follows that

$$r_{k+1}^2 + \delta_{k+1}^2 = r_k^2. \tag{4}$$

Since $\delta_{k+1} = 1/(k+1)$ it follows that

$$r_{k+1}^2 = r_k^2 - 1/(k+1)^2. \tag{5}$$

If we disregard for a moment the restriction imposed by S, we conclude that (a_{k+1}, b_{k+1}) can be constructed from (a_k, b_k) provided $r_k^2 - 1/(k+1)^2 > 0$. Now, from (5) it follows that

$$r_{k+1}^2 = r_0^2 - \sum_{j=0}^{k} 1/(j+1)^2 \tag{6}$$

hence we should have $r_0^2 > \pi^2/6$, since in our case $r_0^2 = 9$, this condition is satisfied.
As a byproduct we obtain that $\lim_{k\to\infty} r_k^2 = 9 - \pi^2/6$. It should be clear that from this we can also conclude that the requirement that $(a_k, b_k) \in S$ does not impose a restriction on the existence of the sequence.

ii) From the fact that $r_k \to \sqrt{9 - \pi^2/6}$ and since $\delta_{k+1} = 1/(k+1)$, it follows that $\angle(L_k, L_{k+1})$ is $O(1/k+1)$. Therefore the sequence of lines $\{L_k\}$ does not converge and hence nor does $\{(a_k, b_k)\}$. □

Lemma 3.2: Consider the i/o system

$$y(k+1) = (1/2)y(k) + u(k), \quad y(0) = 1.$$

There exists an input sequence $\{u(k)\}$, such that the projection algorithm, initialized in (a_0, b_0) generates $\{(a_k, b_k)\}$ as the sequence of estimates.

Proof: This is now easy. All we have to do is make sure that at time $k+1$, $G_{k+1} = L_{k+1}$, or equivalently, $(a_{k+1}, b_{k+1}) \in G_{k+1}$ and $y(k) \neq 0$. Otherwise stated $u(k)$ has to be such that

$$1/2y(k) + u(k) = a_{k+1}y(k) + b_{k+1}u(k). \tag{7}$$

Hence we should take

$$u(k) = \frac{a_{k+1} - 1/2}{1 - b_{k+1}} y(k). \tag{8}$$

Since $(a_{k+1}, b_{k+1}) \in S$, this can indeed be done.

To complete the proof we have to check that for all k the output $y(k)$ will be nonzero. From (8) it follows that

$$y(k+1) = \left(1/2 + \frac{a_{k+1} - 1/2}{1 - b_{k+1}}\right) y(k). \tag{9}$$

Since $y(0) = 1$, and since $(a_{k+1}, b_{k+1}) \in S$, it follows from (9) that $y(k) \neq 0$.

Notice that since $(a_k, b_k) \in S$, we actually have that the sequences u and y are bounded. □

We have now proved the following theorem.

Theorem 3.3: There exists a system of the form (1), a bounded input sequence u and an initialization of the projection algorithm, such that the resulting sequence of estimates does not converge.

IV. Conclusion

By means of an example, we have shown that the sequence of estimates generated by the projection algorithm does not necessarily converge. Of course, the sequence of inputs needed for the example is fairly artificial. In applications such as adaptive control, however, it is most desirable to derive as many properties of the identification part as possible without having to rely on the specific nature of the input. For the input will depend in a highly nonlinear fashion on the estimates. Our construction shows that convergence is not automatically among the properties that can be derived without additional assumptions on the input sequence.

References

[1] G. C. Goodwin and K. S. Sin, *Adaptive Filtering Prediction and Control.* Englewood Cliffs, NJ: Prentice-Hall, 1984.

[2] G. C. Goodwin and R. H. Middleton, *Digital Control and Estimation.* Englewood Cliffs, NJ: Prentice-Hall, 1990.

[3] J. W. Polderman, "A state space approach to the problem of adaptive pole assignment," *Math. Contr., Signals, Syst.*, vol. 2, pp. 71–94, 1989.

A Comment on the Method of the Closest Unstable Equilibrium Point in Nonlinear Stability Analysis

E. Noldus and M. Loccufier

Abstract—**A counterexample is presented to a theorem which has been proposed as a theoretical basis for the method of the closest unstable equilibrium point to estimate asymptotic stability regions in nonlinear systems. An additional condition is formulated under which the theorem is valid. Its implications on the applicability of the method are discussed.**

I. Introduction

The method of the closest unstable equilibrium point (c.u.e.p.) is a well-known direct method of the Lyapunov type for estimating regions of asymptotic stability (RAS) in nonlinear systems analysis. The method has been described, among others, by Chiang *et al.* [1], [2] and various applications, for example to the power system transient stability problem have been reported [3]–[5]. Its basic principle is the following: Consider an autonomous nonlinear dynamical system

$$\dot{x} = f(x) \tag{1}$$

where $x \in R^n$ represents the state and $f(\cdot)$ satisfies the sufficient conditions for the existence and the uniqueness of the solutions for given initial conditions. Suppose that a scalar function $V(x) \in C^r$, $r \geq 1$, can be found such that along the solutions of (1)

$$\begin{aligned}\dot{V}(x) &\leq 0, \forall x \in R^n\\ &= 0 \Leftrightarrow \dot{x} = 0.\end{aligned} \tag{2}$$

By (2), $V(x)$ is a Lyapunov function of (1) in R^n. Let $\hat{x}_s$ be a locally asymptotically stable (l.a.s.) equilibrium state and let $\Omega(\hat{x}_s) \subset R^n$ be its exact RAS. Suppose that on the stability boundary $\partial\Omega(\hat{x}_s)$, $V(x)$ reaches an absolute minimum at $x = \hat{x}_e$ and let

$$\min_{x \in \partial\Omega(\hat{x}_s)} V(x) = V(\hat{x}_e) = V_{\min}. \tag{3}$$

Then it is well known that $\hat{x}_e$ is an unstable equilibrium point of the system (1) [1]. Furthermore for any k in the interval

$$V(\hat{x}_s) < k \leq V_{\min}$$

the set

$$S \triangleq \{x; V(x) < k\}$$

is the union of a number of connected, disjoint subsets

$$S = S_1 \cup S_2 \cup \cdots \cup S_r;$$

$$S_i \cap S_j = \emptyset \quad \text{for} \quad i \neq j$$

Manuscript received December 20, 1993; revised April 11, 1994.

The authors are with the University of Ghent, Department of Control Engineering and Automation, Technologiepark-Zwijnaarde 9, B-9052 Gent, Belgium.

IEEE Log Number 9407220.

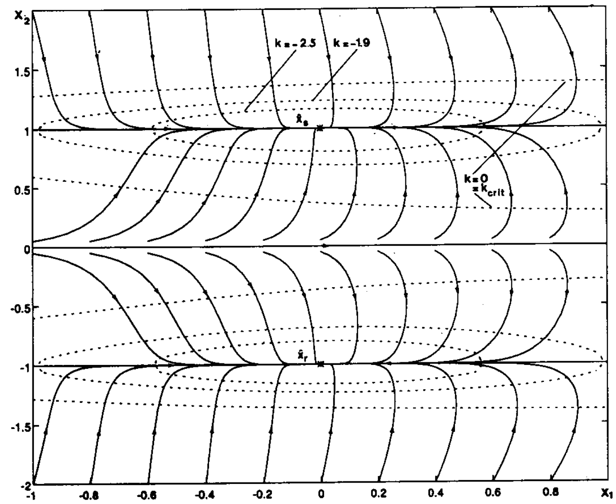

Fig. 1. Phase portrait of the system (5), (6) and level sets ∂S_1 for varying level values k, for the numerical values $a = 2$, $b = 1$, $\mu = 0.8$.

one of which, say S_1 contains $\hat{x}_s$. This subset S_1 is a RAS for $\hat{x}_s$

$$S_1 \subseteq \Omega(\hat{x}_s).$$

The largest stability region S_1 is obtained for $k = V_{\min}$. In [1] Chiang and Thorp have reported a theorem pertaining to the existence of the minimum $V_{\min}$, and a scheme for computing the corresponding stability region S_1 based on it.

Theorem [1]: If system (1) has a Lyapunov function $V(x)$ in R^n which satisfies (2) and if $\Omega(\hat{x}_s)$ is not dense in R^n, then $V_{\min}$ as defined by (3) exists and $\hat{x}_e$ is an unstable equilibrium state.

The proof relies on the property that if for $k = q$ the set $\overline{S}_1$ is a closed and bounded neighborhood of $\hat{x}_s$ which contains no other equilibria, and if for some $p > q$ there are no equilibrium states in the set $\overline{S}_1|_{k=p} - \overline{S}_1|_{k=q}$ then

$$\overline{S}_1|_{k=p} \quad \text{is also closed and bounded.} \tag{4}$$

In Section II a counterexample to this result and to the property (4) is presented. It is pointed out, however, that the theorem is valid under the additional assumption that all trajectories on the stability boundary $\partial\Omega(\hat{x}_s)$ are bounded for $t \geq 0$. Section III discusses the implications of this proposition for the c.u.e.p. method.

II. Example

Consider an example of the form

$$\dot{x} = f(x) \triangleq -\frac{\partial V(x)}{\partial x} \tag{5}$$

where $x \in R^2$ and

$$V(x) \triangleq e^{-x_1} - (2x_2^2 - x_2^4)v_1(x_1) \tag{6}$$

with

$$v_1(x_1) = [e^{-x_1} + ae^{-\mu x_1^2} + b]$$

and $a > 0$, $b > 0$ and $\mu > 0$. Then

$$\dot{V}(x) = \left[\frac{\partial V(x)}{\partial x}\right]' \dot{x} = -\dot{x}'\dot{x} \tag{7}$$