On asymptotic stability of a linear difference equation depending on parameters

Michał Przyłuski

Warsaw University of Technology, Faculty of Electronics and Information Technology,
Institute of Control and Computation Engineering, e-mail: M.Przyluski@elka.pw.edu.pl

The paper focuses on a linear difference equation depending on parameters. The equation is related to Goodwin's theory of extrapolative expectations. The stability region of the equation is investigated. Conditions for asymptotic stability are formulated and presented as an optimisation problem, which is further analysed. Despite employing state-of-the-art solvers, numerical results have turned out to be too ambiguous to provide the basis for definite conclusions about the investigated stability region.

Keywords and phrases: difference equation, asymptotic stability, optimisation.

2000 Mathematics Subject Classification: Primary 90C90; Secondary 65K99

Introduction and preliminary remarks

We will investigate equations related to Goodwin's theory of extrapolative expectations. The reader can find background information on market price expectations in [1] and details on Goodwin's extrapolative approach in his paper [2].

Let x be a variable characterizing the state of a system. We consider an agent that is trying to predict the value $x(t)$ of x at time t. One should observe that $x(t)$ does not denote a stochastic process for the moment t (which is actually a random variable), but the most probable realization of the stochastic process at the moment t, which is a real number.

Let $x^E(t)$ denote the expected (by the agent) value of x, at time t. We postulate that the actual value $x(t)$ of x at time t is influenced[1] by the expected value $x^E(t)$ of the variable x. The postulate is widely accepted[2] by economists. A simple version of the postulate can be stated as the assumption that

$$x(t) = ax^E(t) + b, \qquad (*)$$

for $t = 0, 1, \ldots$. Here $a, b \in \mathbb{R}$, and we assume in what follows that $a < 0$.

When a, b are known, one can "guess" $x(t)$. In that case the proper choice of $x^E(t)$ is the number $\overline{x} = \frac{b}{1-a}$, since

$$a\frac{b}{1-a} + b = \frac{b}{1-a},$$

i.e. $x(t) = x^E(t) = \overline{x}$. The so-defined $\overline{x}$ is called the equilibrium state of our system $(*)$, since it characterizes the case where expected and actual values of x are identical.

The main purpose of the present paper is to examine the case where the parameters a, b are unknown. Then various rules defining $x^E(t)$ can be applied. We will consider causal rules, i.e. we will assume that $x^E(t)$ can be generated using only the past values of $x(t)$.

The simplest way of estimating $x^E(t)$ is to assume that $x^E(t) = x(t - 1)$, which means that estimation of $x(t)$ is based on exactly one previous value of x. It follows that

$$x(t) = ax(t - 1) + b,$$

for $t = 0, 1,\ldots$ This difference equation is asymptotically stable [4] if and only if $a \in (-1, 0)$, since we only consider $a < 0$. Observe also that the equilibrium point of the above equation coincides with

[1] We can consider $x(t)$ as a given state and $x^E(t)$ can be perceived as a transition matrix in Markov process theory.

[2] See e.g. [3].

the equilibrium point of system (∗) i.e. with $\overline{x} = \frac{b}{1-a}$. Hence, for all $a \in (-1, 0)$, $x(t)$ tends to $\overline{x}$, independently of the initial condition $x(-1)$.

Consider now a more sophisticated estimate of $x(t)$. For this, let us assume that estimation is based on two historical values: $x(t-1)$ and $x(t-2)$. More precisely, we consider

$$x^E(t) = x(t-1) + \rho(x(t-1) - x(t-2));$$

here $\rho \in \mathbb{R}$ is a parameter, to be determined.

Using the relation $x(t) = ax^E(t) + b$, we obtain (for $t = 0, 1, \ldots$)

$$x(t) = a[x(t-1) + \rho(x(t-1) - x(t-2))] + b.$$

We rewrite the above difference equation in the following form:

$$x(t) = a(1+\rho)x(t-1) - a\rho x(t-2) + b.$$

It is easy to check that the equilibrium point of the equation coincides with $\overline{x} = \frac{b}{1-a}$. The equation is asymptotically stable if and only if the roots of its characteristic equation $\lambda^2 - a(1+\rho)\lambda + a\rho = 0$ lie in the open unit disc of $\mathbb{C}$. The dependency of the location of roots of the characteristic equation on the parameter ρ was extensively investigated by R. M. Goodwin in 1947, as presented in monograph [5]. It happens that for $\rho = -\frac{1}{3}$ the difference equation is asymptotically stable, for all $a \in (-3, 0)$.

Summing up, the above paragraph establishes that for any $a \in (-3, 0)$ the equation

$$x(t) = a[x(t-1) - \frac{1}{3}(x(t-1) - x(t-2))] + b,$$

is always asymptotically stable. It is essential to observe that for $a \leqslant -3$ the equation is never asymptotically stable, regardless of ρ.

It is also remarkable that when the estimate $x^E(t)$ of $x(t)$ is determined using only $x(t — 1)$ value, the corresponding difference equation can be stable for $a \in (-1, 0)$, and when using two historical values of x the equation is stable for $a \in (-3, 0)$.

A question that naturally arises is whether or not the use of more than two past values of the variable x to obtain $x^E(t)$ results in a greater range of parameter values a for which the equation is asymptotically stable.

Estimation based on three past values

Let us consider a more complex scenario, when estimation of $x(t)$ is based on three past values: $x(t — 1)$, $x(t — 2)$ and $x(t — 3)$. Namely, we assume that the equation linking $x^E(t)$ with $x(t — 1)$, $x(t — 2)$ and $x(t — 3)$ is as follows:

$$x^E(t) = x(t-1) + \rho(x(t-1) - x(t-2)) + \gamma(x(t-2) - x(t-3));$$

here $\rho, \gamma \in \mathbb{R}$ are arbitrary parameters.

Using the relation $x(t) = ax^E(t) + b$, we obtain

$$x(t) = a[x(t-1) + \rho(x(t-1) - x(t-2)) + \gamma(x(t-2) - x(t-3))] + b. \tag{$**$}$$

One can rewrite the above difference equation in the following form:

$$x(t) = a(1+\rho)x(t-1) + a(\gamma - \rho)x(t-2) - a\gamma x(t-3) + b,$$

where $t = 0, 1, \ldots$. The equilibrium point of the equation remains the same as before, namely $\overline{x} = \frac{b}{1-a}$. The characteristic polynomial of (∗∗) is as follows:

$$\lambda^3 - a(1+\rho)\lambda^2 - a(\gamma - \rho)\lambda + a\gamma.$$

Finding explicit relation between parameters ρ and λ and the roots of characteristic polynomial is possible, but difficult. On the other hand, it is straightforward to formulate simple necessary and sufficient conditions for asymptotic stability of the above difference equation.

For this, let us recall (see e. g. [6], [7, Ch. 3.5]) that for a third-degree polynomial $P(\lambda) = a_0 + a_1\lambda + a_2\lambda^2 + a_3\lambda^3$, the necessary and sufficient conditions for all its roots to be located in the open unit disc of $\mathbb{C}$ are as follows:

$$\begin{aligned} &a_0 + a_1 + a_2 + a_3 > 0, \\ &-a_0 + a_1 - a_2 + a_3 > 0, \\ &|a_0| < a_3, \\ &a_0^2 - a_3^2 < a_0 a_2 - a_1 a_3. \end{aligned}$$

Applying the above conditions to the difference equation (∗∗) we obtain the inequalities

$$\begin{aligned} &\alpha > 0, \\ &\alpha(2\rho - 2\gamma + 1) < 1, \\ &\alpha^2\gamma^2 < 1, \\ &\alpha^2\gamma^2 - 1 < -\alpha^2\gamma(1+\rho) - \alpha(\gamma - \rho), \end{aligned}$$

where $\alpha = |a|$ (as usual, we are considering only $a < 0$).

Our aim is to find the largest value of α satisfying the above inequalities for some ρ and λ. Solving these inequalities directly seems to be difficult, so instead we employ numerical methods to determine the largest α.

The rest of the paper will be devoted to numerical experiments that have been carried out in order to find the appropriate value of α. Therefore, the following optimisation problem will been analysed:

$$\begin{aligned} \text{maximize} \quad & \alpha \\ \text{subject to} \quad & \alpha \geqslant 0, \\ & \alpha(2\rho - 2\gamma + 1) \leqslant 1, \\ & \alpha^2\gamma^2 \leqslant 1, \\ & \alpha^2\gamma^2 - 1 + \alpha^2\gamma(1+\rho) + \alpha(\gamma - \rho) \leqslant 0. \end{aligned}$$

Since the above formulation of the optimisation problem contains weak inequalities, it may happen that the solution α of the optimisation problem lies at the boundary of the stability region.

Let us note that $(\alpha, \gamma, \rho) = (3, 0, -\frac{1}{3})$ belongs to the feasible region of the above optimisation problem.

Numerical experiments

The above formulated optimisation problem was, initially, formulated as a problem using MATLAB and YALMIP. YALMIP is a MATLAB package that allows for fairly simple representation of numerous kinds of optimisation problems and has provisions for solving them using various solvers available in MATLAB. It allows for easy solver-change, without the need to change original problem's formulation according to solver's expectations.

MATLAB with fmincon

The first attempt to solve the problem was to use default, built-in, MATLAB procedure fmincon, which is a general conditional extremum-finding procedure. It uses the trust region reflective algorithm by default.

With default settings, the trust region reflective algorithm states that the point $(\alpha, \gamma, \rho) = (21.3087, -0.0469, -0.5235)$ is the solution to the optimisation problem. The very same point is returned by the active set algorithm.

The other two algorithms (sqp and interior point) supported by fmincon produce worse results. The sqp finds the obviously worse solution $(\alpha, \gamma, \rho) = (7.3302, -0.1364, -0.5682)$, while the interior point algorithm cannot find any solution exceeding its limitations on the number of iterations or function evaluations (by default set to 1000 and 3000, respectively). Raising the limits gave no new results, as it has still reached them.

However, if we add an additional constraint ($\gamma = 0$ or $\gamma = -0.0469$) we observe a different behaviour.

In the case $\gamma = 0$, fmincon found the optimal solution at $(\alpha, \gamma, \rho) = (3, 0, -0.3333)$, which agrees with analytical calculations.

More surprising result was obtained for the problem with the additional constraint $\gamma = -0.0469$. One could expect the solver to return $\alpha = 21.3087$, or at least a very similar value. However, it turned out that the solution calculated by fmincon is quite different. More precisely the obtained solution was $(\alpha, \gamma, \rho) = (3.6928, -0.0469, -0.4115)$.

It has to be emphasised that fmincon does not report any numerical problems (assuming trust region reflective algorithm) during calculations and presents a solution of the optimisation problems without any information that could raise doubts about its correctness.

MATLAB with GloptiPoly3

The next attempt was taken using a specialised library called GloptiPoly3. It is supposed to solve optimisation problems with the objective function and constraints being polynomials. The library is described in [8], while the algorithm it uses (a relaxation that can be formulated as an SDP problem) can be found in [9]. Once the problem is formulated in terms of SDP, GloptiPoly3 solves it using any semidefinite solver, usually either SeDuMi or SDPT3 (see [10] and [11], respectively).

At first the base optimisation problem (without additional constraints on γ) was formulated and solved with GloptiPoly3 (both SeDuMi and SDPT3 were used as semidefinite solvers). However, both failed to find an optimal solution, returning after long calculations and numerous iterations with errors: run into numerical problems and stop: lack of progress in dual infeas, homrp = Inf (SeDuMi and SDPT3, respectively). Relaxations of various orders were used as well, giving no improvement in the quality of results, only lengthening the computation time.

One should note that GloptiPoly3 was able to solve the optimisation problem with additional constraints. In the case when $\gamma = 0$ both semidefinite engines found the optimal solution. Introducing the additional constraint $\gamma = -0.0469$ results in GloptiPoly3 finding an acceptable (i.e. feasible) solution $(\alpha, \gamma, \rho) = (21.3220, -0.0469, -0.5274)$.

Ipopt

Finally, the optimisation problem in question was implemented using the C language library called Ipopt. It is a powerful library (developed by IBM) described in [12] that solves many kinds of optimisation problems and can be directly interfaced with the C language. We have encountered severe numerical issues during an attempt to solve the problem.

The problem without extra constraints was particularly difficult to deal with. Both standard and alternative (Mehrotra predictor-corrector) Ipopt's algorithms failed to find the solution. Changing strategy for a barrier parameter (mu_strategy) or reformulating the problem as min $-\alpha$ made no improvement. In each case results were different and confusing.

Interesting results appeared after introducing the extra constraint $\alpha \leq 4 \cdot 10^6$. In that case, Ipopt erroneously reported $(\alpha, \gamma, \rho) = (4 \cdot 10^6, -2.5 \cdot 10^{-7}, -0.5)$ as the optimal solution. It appears that the constraint $\alpha(2\rho - 2\gamma + 1) \leq 1$ is not satisfied by the above solution! Unfortunately, Ipopt does not report any problems during computations, what leads to an illusion that it is indeed a correct solution. In depth analysis showed that the error is caused by numerical problems that rose in the constraint evaluation. It turned out that the sequence of operations (−1 + small_number + 1) gives incorrect result. Changing the order fixes the issue, but then Ipopt does not converge for the problem. It is surprising, though, that the order of addition operations has so much effect on the final result. It suggests a very unfortunate numerical characteristic of the considered optimisation problem.

Several other values of α were tested as an upper bound and various settings (algorithm, barrier parameter strategy, etc.) were considered. Those tests resulted in Ipopt presenting both feasible and infeasible solutions that required manual assessment of whether all constraints are satisfied.

Finally, after introducing the extra constraint α 10000 Ipopt found a solution (reported as optimal) that turned out to be feasible. All previous problems urged the need to ensure that it is correct, thus it has been double-checked that $(\alpha, \gamma, \rho) = (10000, -0.0001, -0.50005)$ indeed satisfies all conditions.

Summing up, the optimisation problem is ill-conditioned and even its slight seemingly equivalent reformulation can lead to unexpected results. It appears to be very difficult to take effective countermeasures against these problems and properly formulate the problem in C language to avoid various (hard to predict) errors.

Concluding remarks

In light of the numerical experiments described above one could hypothesize that the original optimisation problem is unbounded. However, all tested solvers failed to give a definitive answer to the problem, returning poor or even infeasible solutions. It is surprising that GloptiPoly3 — a solver supposed to deal with this kind of problems — could not cope with it.

References

[1] Muth, J.F. "Rational Expectations and the Theory of Price Movements". *Econometrica* 29 (3), 1961: 315–335.

[2] Goodwin, R.M. "Dynamical Coupling with Especial Reference to Markets Having Production Lags". *Econometrica* 15 (3), 1947: 181–204.

[3] Evans, G.W. and S. Honkapohja. *Learning and Expectations in Macroeconomisc*. Princeton: Princeton University Press, 2001.

[4] LaSalle, J.P. *The Stability and Control of Discrete Processes*. New York: Springer, 1986.

[5] Gandolfo, G. *Mathematical Methods and Models in Economic Dynamics*. Amsterdam: North-Holland, 1971.

[6] Ackermann, J. *Sampled-Data Control Systems*. Berlin: Springer-Verlag, 1985.

[7] Jury, E.I. *Theory and Application of the z-Transform Method*. Huntington New York: Krieger Publishing Co, 1964.

[8] Henrion, D., J.B. Lasserre, and J. Loefberg. "GloptiPoly 3: moments, optimization and semidefinite programming". *Optimization Methods and Software* 24 (4–5), 2009: 761–779.

[9] Lasserre, J.B. "A Semidefinite programming approach to the generalized problem of moments". *Mathematical Programming* 112 (1), 2008: 65–92.

[10] http://sedumi.ie.lehigh.edu/

[11] http://www.math.nus.edu.sg/˜mattohkc/sdpt3.html

[12] Wächter, A. and L.T. Biegler. "On the Implementation of a Primal-Dual Interior Point Filter Line Search Algorithm for Large-Scale Nonlinear Programming". *Mathematical Programming* 106 (1), 2006: 25–57.