

# Optimal Control of Nonlinear Dynamic Econometric Models: An Algorithm and an Application

V. Blueschke-Nikolaeva<sup>a</sup>, D. Blueschke<sup>a</sup>, R. Neck<sup>a\*</sup>

<sup>a</sup>Department of Economics, Klagenfurt University  
Universitätsstrasse 65-67, A-9020 Klagenfurt, Austria

July 10, 2010

**Abstract** OPTCON is an algorithm for the optimal control of nonlinear stochastic systems which is particularly applicable to econometric models. It delivers approximate numerical solutions to optimum control problems with a quadratic objective function for nonlinear econometric models with additive and multiplicative (parameter) uncertainties. The algorithm was programmed in *C#* and allows for deterministic and stochastic control, the latter with open-loop and passive learning (open-loop feedback) information patterns. The applicability of the algorithm is demonstrated by experiments

---

\*Corresponding author: Prof. Reinhard Neck, Department of Economics, Klagenfurt University, Universitätsstrasse 65-67, A-9020 Klagenfurt, Austria/Europe, Ph: +43 463 2700 4121, Fax: +43 463 2700 994121, Email: reinhard.neck@uni-klu.ac.at

with a small quarterly macroeconomic model for Slovenia. This illustrates the convergence and the practical usefulness of the algorithm and (in most cases) the superiority of open-loop feedback over open-loop controls.

**Keywords** Optimal control; Stochastic control; Algorithms; Econometric modeling; Policy applications

# 1 Introduction

Optimum control theory has a great number of applications in many areas of science from engineering to economics. In particular, there are many studies on determining optimal economic policies for econometric models. Most of these optimum control applications use algorithms for linear dynamic systems or those that do not take the full stochastic nature of the econometric model into account. Examples of the former are Kendrick (1981), Coomes (1987), and the references in Amman (1996), and Chow (1975, 1981) for the latter. An algorithm that is explicitly aimed at providing (approximate) solutions to optimum control problems for nonlinear econometric models and other dynamic systems with different kinds of stochastics is OPTCON, as introduced by Matulka and Neck (1992). However, so far OPTCON has been severely limited by being based on very restrictive assumptions about the information available to the decision-maker. In particular, learning about the econometric model while in the process of controlling the economy was ruled out by assumption. In reality, however, new information arrives in each period, and econometric models are regularly re-specified using this information. Therefore, extensions of the OPTCON algorithm to include various possibilities of obtaining and using new information about the system to be controlled are highly desirable.

The present extension of the OPTCON algorithm from open-loop control only (OPTCON1) to the inclusion of passive learning or open-loop feedback control where the estimates of the parameters are updated in each period results in the algorithm OPTCON2. It can deliver approximately optimal

solutions to dynamic optimization (optimum control) problems for a rather large class of nonlinear dynamic systems under a quadratic objective function with stochastic uncertainty in the parameters and in the system equations under both kinds of control schemes. In the open-loop feedback part, it is assumed that new realizations of both random processes occur in each period, which can be used to update the parameter estimates of the dynamic system, i.e. of the econometric model. Following Kendrick's (1981) approach, the parameter estimates are updated using the Kalman filter in order to arrive at more reliable approximations to the solution of stochastic optimum control problems. Whether this hope will materialize depends upon the comparative performance of open-loop feedback versus open-loop control schemes in actual applications. Some indication of this will be provided by comparing the two control schemes within a control problem for a small econometric model. This also serves to show that the OPTCON2 algorithm and its implementation in *C#* actually deliver plausible numerical solutions, at least for a small problem, with real economic data.

The paper has the following structure: In Section 2, the class of problems to be tackled by the algorithm is defined. Section 3 briefly reviews the OPTCON1 algorithm. Section 4 explains the OPTCON2 algorithm. In Section 5, the small econometric model for Slovenia SLOVNL is introduced, the applicability and convergence of OPTCON2 as implemented in *C#* is shown, and the quality of open-loop and open-loop feedback (passive-learning) controls in Monte Carlo simulations for this model are compared. Section 6 concludes. More details about the mathematics of the algorithm are given in Blueschke-Nikolaeva et al. (2010).

## 2 The problem

The OPTCON algorithm is designed to provide approximate solutions to optimum control problems with a quadratic objective function (a loss function to be minimized) and a nonlinear multivariate discrete-time dynamic system under additive and parameter uncertainties. The intertemporal objective function is formulated in quadratic tracking form, which is quite often used in applications of optimum control theory to econometric models. It can be written as

$$J = E \left[ \sum_{t=1}^T L_t(x_t, u_t) \right], \quad (1)$$

with

$$L_t(x_t, u_t) = \frac{1}{2} \begin{pmatrix} x_t - \tilde{x}_t \\ u_t - \tilde{u}_t \end{pmatrix}' W_t \begin{pmatrix} x_t - \tilde{x}_t \\ u_t - \tilde{u}_t \end{pmatrix}. \quad (2)$$

$x_t$  is an  $n$ -dimensional vector of state variables that describes the state of the economic system at any point in time  $t$ .  $u_t$  is an  $m$ -dimensional vector of control variables,  $\tilde{x}_t \in R^n$  and  $\tilde{u}_t \in R^m$  are given 'ideal' (desired, target) levels of the state and control variables respectively.  $T$  denotes the terminal time period of the finite planning horizon.  $W_t$  is an  $((n + m) \times (n + m))$  matrix, specifying the relative weights of the state and control variables in the objective function. In a frequent special case,  $W_t$  is a matrix including a discount factor  $\alpha$  with  $W_t = \alpha^{t-1}W$ .  $W_t$  (or  $W$ ) is symmetric.

The dynamic system of nonlinear difference equations has the form

$$x_t = f(x_{t-1}, x_t, u_t, \theta, z_t) + \varepsilon_t, \quad t = 1, \dots, T, \quad (3)$$

where  $\theta$  is a  $p$ -dimensional vector of parameters whose values are assumed to be constant but unknown to the decision-maker (parameter uncertainty),  $z_t$  denotes an  $l$ -dimensional vector of non-controlled exogenous variables, and  $\varepsilon_t$  is an  $n$ -dimensional vector of additive disturbances (system error).  $\theta$  and  $\varepsilon_t$  are assumed to be independent random vectors with expectations  $\hat{\theta}$  and  $O_n$  respectively and covariance matrices  $\Sigma^{\theta\theta}$  and  $\Sigma^{\varepsilon\varepsilon}$  respectively.  $f$  is a vector-valued function,  $f^i(\dots)$ , is the  $i$ -th component of  $f(\dots)$ ,  $i = 1, \dots, n$ .

### 3 OPTCON1

The basic OPTCON algorithm determines approximate solutions to optimum control problems with a quadratic objective function and a nonlinear multivariate dynamic system under additive and parameter uncertainties. It combines elements of previous algorithms developed by Chow (1975, 1981), which incorporate nonlinear systems but no multiplicative uncertainty, and Kendrick (1981), which deals with linear systems and all kinds of uncertainty. The version OPTCON1 is described in detail in Matulka and Neck (1992); here only its basic idea is presented.

It is well known in stochastic control theory that a general analytical solution to dynamic stochastic optimization problems cannot be achieved even for very simple control problems. The main reason is the so-called dual effect of control under uncertainty, meaning that controls do not only contribute directly to achieving the stated objective but also affect future uncertainty and hence the possibility of indirectly improving on the system performance by providing better information about the system (see, for instance, Aoki

(1989); Neck (1984)). Therefore only approximations to the true optimum for such problems are feasible, with various schemes existing to deal with the problem of information acquisition.

A useful distinction was adapted from the control engineering literature by Kendrick (1981): open-loop policies preclude the possibility of receiving information (measurements) while the system is in operation; open-loop feedback (or passive learning) policies use current information to determine the control but do not anticipate future measurements; and closed-loop (or active learning) policies make some use of information about future measurements as well. Alternatively, Kendrick and Amman (2006) propose the terms optimal feedback and expected optimal feedback for open-loop and open-loop feedback respectively. Given the intricacies of the interplay between control and information, even for very simple stochastic control problems (for example, a linear scalar system with a time horizon of only two periods), an exact analytical or even numerical solution is impossible; hence numerical approximations are required that make use of simplifying assumptions.

The OPTCON1 algorithm determines policies belonging to the class of open-loop controls. It either ignores the stochastics of the system altogether (deterministic solution, identical to the Chow algorithm) or assumes the stochastics (expectation and covariance matrices of additive and multiplicative disturbances) to be given once and for all at the beginning of the planning horizon (stochastic solution). The problem with the nonlinear system is tackled iteratively, starting with a tentative path of state and control variables. The tentative path of the control variables is given for the first iteration. In order to find the corresponding tentative path for the state variables, the

nonlinear system is solved numerically using the Newton-Raphson method. Alternatively, the Gauss-Seidel method or perturbation methods (e.g. Chen and Zadrozny (2009)) may be used for this purpose.

After the tentative path is found, the iterative approximation of the optimal solution starts. The solution is sought from one time path to another until the algorithm converges or the maximal number of iterations is reached. During this search the system is linearized around the previous iteration's result as a tentative path and the problem is solved for the resulting time-varying linearized system. The criterion for convergence demands that the difference between the values of current and previous iterations be smaller than a pre-specified number. The approximately optimal solution of the problem for the linearized system is found under the above-mentioned simplifying assumptions about the information pattern; then this solution is used as the tentative path for the next iteration, starting off the procedure all over again.

Every iteration, i.e. every solution of the problem for the linearized system, has the following structure: the objective function is minimized using Bellman's principle of optimality to obtain the parameters of the feedback control rule. This uses known results for the stochastic control of LQG problems (optimization of linear systems with Gaussian noise under a quadratic objective function). A backward recursion over time starts in order to calculate the controls for the first period. Next, the optimal values of the state and the control variables are calculated by applying forward recursion, i.e. beginning with  $u_1$  and  $x_1$  at period 1 and finishing with  $u_T$  and  $x_T$  at the terminal period  $T$ . If the convergence criterion is fulfilled, the solution of the last iteration is taken as the approximately optimal solution to the prob-



lem and the algorithm stops. Finally, the value of the objective function is calculated for this solution. For more details, see Matulka and Neck (1992). Figure 1 summarizes the OPTCON1 algorithm.

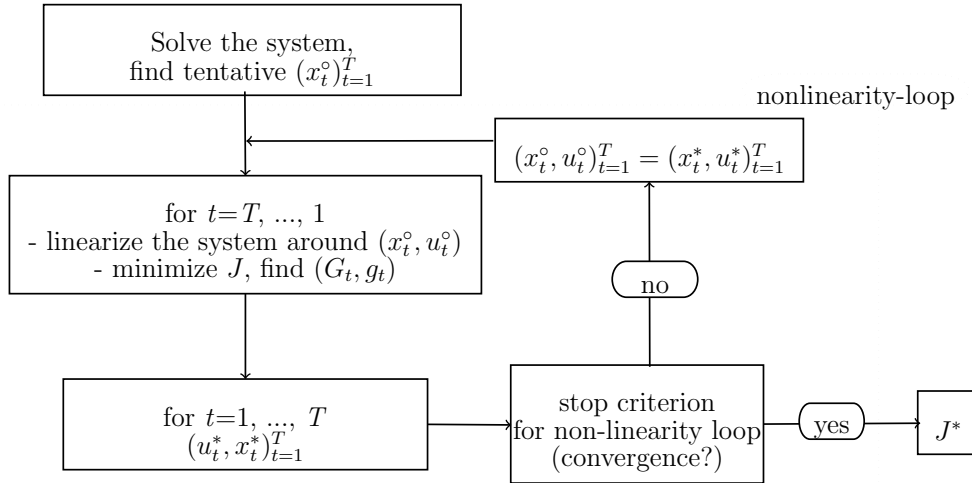


Figure 1: Flow chart of OPTCON1

## 4 OPTCON2

### 4.1 Characterization of the OPTCON2 algorithm

The new version of the algorithm, OPTCON2, incorporates both open-loop and open-loop feedback (passive-learning) controls. The idea of passive learning corresponds to actual practice in applied econometrics: at the end of each time period, the model builder (and hence the control agent) observes what has happened, that is, the current values of state variables, and uses this information to re-estimate the model and hence improve his/her knowledge of

the system. It should be mentioned that two kinds of errors, namely additive (random system errors) and multiplicative (‘structural’ errors in parameters), are taken into account but not possible specification errors, hence it is assumed that no re-specifications of the model are performed. Whether taking passive learning into account really improves the performance of the system is, however, an open question because the ‘true’ optimum is not known.

The main research aim is to obtain evidence as to whether applying passive learning can indeed improve the final solution. The prediction and optimization procedures for open-loop control assume that the model is not affected by random disturbances occurring during the optimization process. But in reality some random errors will disturb the optimization process. OPTCON2 can deal with two kinds of uncertainties, parameter and system errors. The passive-learning strategy implies observing current information and using it in order to adjust the optimization procedure. For the purpose of comparing open-loop and open-loop feedback results, it is not possible to observe current and true values, so one has to resort to Monte-Carlo simulations. Large numbers of random time paths for the additive and multiplicative errors are generated, representing what new information could look like in reality. In this way ‘quasi-real’ observations are created and both types of controls, open-loop and passive-learning (open-loop feedback), are compared. The procedure applied is as follows.

$M$  (a number, usually between 100 and 1000) different sets of realizations of random noises  $(\varepsilon_t^m)_{t=1}^T$  and  $\mu^m$ ,  $m = 1, \dots, M$ , are generated. It is assumed that there is an unknown ‘real’ model with the ‘true’ constant parameter vector  $\hat{\theta}$ . But the policy-maker does not know these ‘true’ parameters  $\hat{\theta}$  and

works with the ‘wrong’ parameters  $\theta^m$  resulting from the estimates using the realization of the random variable  $\mu^m$ :  $\theta^m = \hat{\theta} + \mu^m$ . After this, the following procedure is run for every random scenario: a forward loop is started from 1 to  $T$ . In each time period  $S$  an (approximately optimal) open-loop solution for the subproblem is determined, i.e the problem for the time periods from  $S$  to  $T$ . Then the predicted  $x_S^*$  and  $u_S^*$  for the time period  $S$  are fixed. The assumption is that the policy-maker knows the realized values of the state variables  $x_S^{a*}$  at the end of period  $S$ , which are, however, disturbed by the additive errors: the difference between  $x_S^* = f(x_{S-1}^{a*}, x_S^*, u_S^*, \theta^m, z_S)$  and  $x_S^{a*} = f(x_{S-1}^{a*}, x_S^{a*}, u_S^*, \hat{\theta}, z_S) + \varepsilon_S^m$  comes from the realization of the random numbers  $\varepsilon_S^m$  and  $\mu^m$ . Next, the new information is used by the policy-maker to update and adjust the parameter estimate  $\theta^m$ . After that, the same procedure is applied for the remaining subproblem from  $S + 1$  to  $T$ , and so on.

## 4.2 Detailed description of the algorithm

**STEP I:** Compute a tentative state path  $(\overset{\circ}{x}_t)_{t=1}^T$  by solving the system of equations  $f(\dots)$  with the Newton-Raphson algorithm (or Newton-Raphson with line-search expansion), given the tentative policy path  $(\overset{\circ}{u}_t)_{t=1}^T$ .

**STEP II:** Generate  $M$  paths of random normally distributed system noises  $(\varepsilon_t^m)_{t=1}^T$  and parameter noises  $\mu^m$  ( $\theta^m = \hat{\theta} + \mu^m$ ) using the given means and covariance matrices. The given covariance matrix is Cholesky decomposed in order to get the lower-triangular matrix. Applying this to uncorrelated random numbers produces a vector with the covariance properties of the system

being modeled.

**STEP III:** For each independent random scenario with  $(\varepsilon_t^m)_{t=1}^T$  and  $\mu^m$ , i.e. for each Monte Carlo run  $m$  ( $m = 1, \dots, M$ ), perform the following steps:

**Step III-1:** For each  $S$  from 1 to  $T$ , find the open-loop solution for the subproblem  $(S, \dots, T)$ , according to the procedure already implemented in OPTCON1; cf. Section 3 above.

**Step III-1a:** The *nonlinearity loop* is run until the stop criterion is fulfilled, i.e. until the difference between the values of the current and the previous iteration is smaller than a pre-specified number or the maximal number of iterations is achieved.

When the stop criterion has been achieved, the approximately optimal solution  $(x_t^*, u_t^*)_S^T$  has been found. Then go to the next step **III-1b**. It should be noted that after several runs of the nonlinearity loop only the solution  $(x_S^*, u_S^*)$  for the time period  $S$  will be taken as the optimal solution. The calculations of the pairs  $(x_{t'}^*, u_{t'}^*)$  for other periods ( $t' > S$ ) have to be done again, taking into account the re-estimated parameters for all periods.

**Step III-1b:** Calculate the following for only one time period  $S$ :

$$x_S^{a*} = f(x_{S-1}^{a*}, x_S^{a*}, u_S^*, \hat{\theta}, z_S) + \varepsilon_S^m.$$

**Step III-1c:** Update the parameter estimates  $\theta^m$  using the Kalman filter and the current (realized) values of the variables  $x_S^{a*}$ :

[1] Prediction:

$$\begin{aligned}
\hat{x}_{S/S-1} &= f(x_{S-1}^{a*}, \hat{x}_{S/S-1}, u_S^*, \theta_{S-1/S-1}^m, z_S) = x_S^*, \quad \theta_{S/S-1}^m = \theta_{S-1/S-1}^m, \\
\Sigma_{S/S-1}^{xx} &= F_{\theta_{S-1/S-1}}^x \Sigma_{S-1/S-1}^{\theta\theta} (F_{\theta_{S-1/S-1}}^x)' + \Sigma_S^{\varepsilon\varepsilon} \\
\text{and} \\
\Sigma_{S/S-1}^{x\theta} &= (\Sigma_{S/S-1}^{\theta x})' = F_{\theta_{S-1/S-1}}^x \Sigma_{S-1/S-1}^{\theta\theta}, \quad \Sigma_{S/S-1}^{\theta\theta} = \Sigma_{S-1/S-1}^{\theta\theta}.
\end{aligned} \tag{4}$$

[2] Correction:

$$\begin{aligned}
\Sigma_{S/S}^{\theta\theta} &= \Sigma_{S/S-1}^{\theta\theta} - \Sigma_{S/S-1}^{\theta x} (\Sigma_{S/S-1}^{xx})^{-1} \Sigma_{S/S-1}^{x\theta} \\
\text{and} \\
\theta_{S/S}^m &= \theta_{S/S-1}^m + \Sigma_{S/S-1}^{\theta x} (\Sigma_{S/S-1}^{xx})^{-1} [x_S^{a*} - x_S^*] \quad \text{and} \quad \hat{x}_{S/S} = x_S^{a*}.
\end{aligned} \tag{5}$$

Thus the update results in the new values  $\theta_{S/S}^m$  and  $\Sigma_{S/S}^{\theta\theta}$ .

**Step III-1d:** Set  $\theta^m = \theta_{t/t}^m$  and  $\Sigma^{\theta\theta} = \Sigma_{t/t}^{\theta\theta}$

and run the procedure for the period  $S+1$ . This loop is finished when  $S = T$  and the approximately optimal open-loop feedback control and state variables for all periods have been found.

**Step III-2:** Compute the expected (approximately) minimal welfare loss:

$$J^* = \sum_{t=1}^T L_t(x_t^{a*}, u_t^*) \tag{6}$$

with

$$L_t(x_t^{a*}, u_t^*) = \frac{1}{2} \begin{pmatrix} x_t^{a*} - \tilde{x}_t \\ u_t^* - \tilde{u}_t \end{pmatrix}' W_t \begin{pmatrix} x_t^{a*} - \tilde{x}_t \\ u_t^* - \tilde{u}_t \end{pmatrix}. \quad (7)$$

The main steps of OPTCON2 are summarized in Figure 2.

## 5 An application

The OPTCON2 algorithm was implemented in *C#*. In order to test its convergence, a very simple, small macroeconomic model for Slovenia was used. Section 5.1 gives the details of this model. The optimization results of this model for two different open-loop policies, a deterministic and a stochastic case, are presented in Section 5.2. In Section 5.3, the results of open-loop and passive-learning (open-loop feedback) control solutions are compared.

### 5.1 The model SLOVNL

The small nonlinear macroeconomic model of the Slovenian economy, called SLOVNL (**SLO**Venian model, **Non-Linear** version), consists of 8 equations, 4 of them behavioral and 4 identities. The model includes 8 state variables, 4 exogenous non-controlled variables, 3 control variables, and 16 unknown (estimated) parameters. The quarterly data for the time periods 1995:1 to 2006:4 yield 48 observations and admit a full-information maximum likelihood (FIML) estimation of the expected values and the covariance matrices for the parameters and system errors. The start period for the optimization is 2004:1 and the end period is 2006:4 (12 periods).

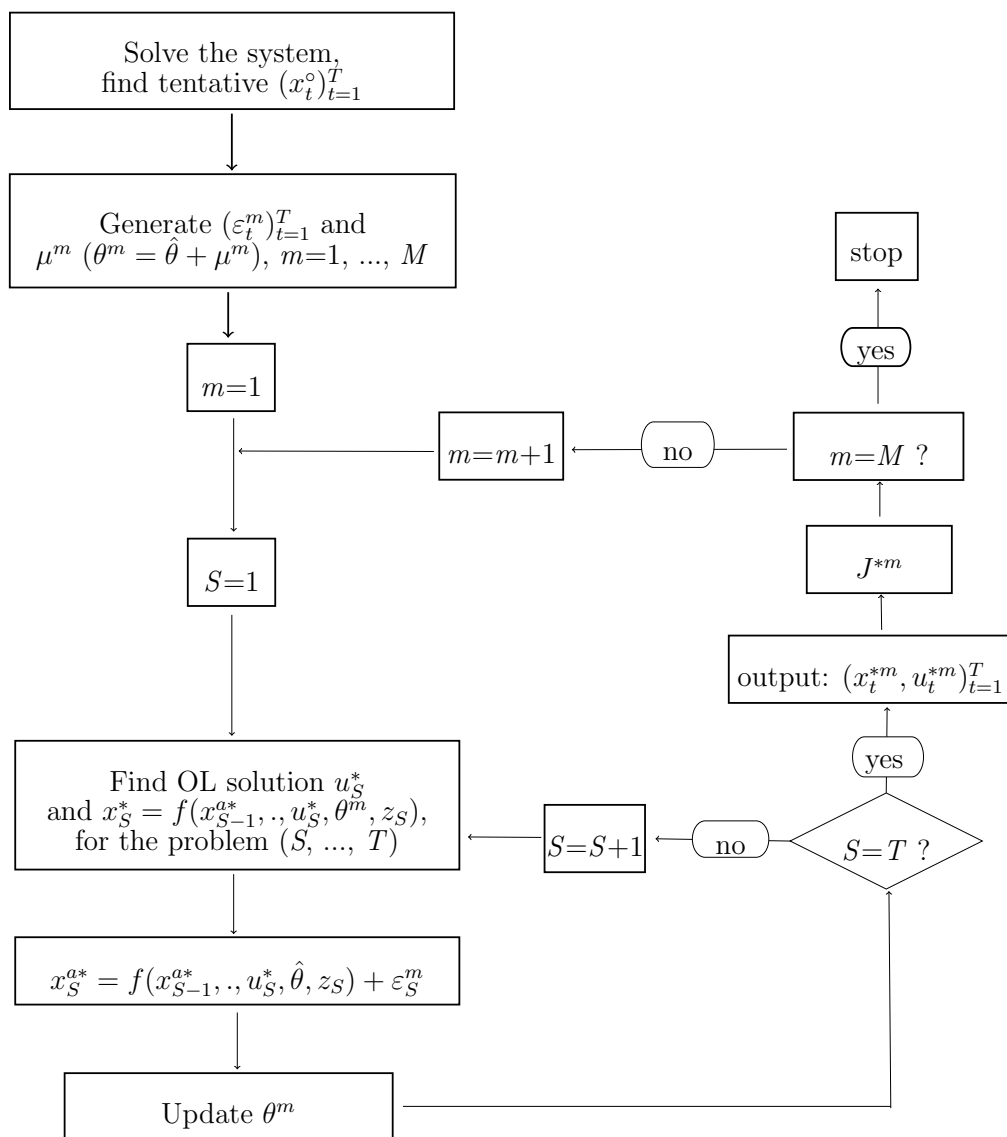


Figure 2: Flow chart of OPTCON2

Model variables used in SLOVNL

*Endogenous (state) variables:*

$x[1]$	<i>CR</i>	real private consumption
$x[2]$	<i>INVR</i>	real investment
$x[3]$	<i>IMPR</i>	real imports of goods and services
$x[4]$	<i>STIRLN</i>	short term interest rate
$x[5]$	<i>GDPR</i>	real gross domestic product
$x[6]$	<i>VR</i>	real total aggregate demand
$x[7]$	<i>PV</i>	general price level
$x[8]$	<i>Pi4</i>	rate of inflation

*Control variables:*

$u[1]$	<i>TaxRate</i>	net tax rate
$u[2]$	<i>GR</i>	real public consumption
$u[3]$	<i>M3N</i>	money stock, nominal

*Exogenous non-controlled variables:*

$z[1]$	<i>EXR</i>	real exports of goods and services
$z[2]$	<i>IMPDEF</i>	import price level
$z[3]$	<i>GDPDEF</i>	domestic price level
$z[4]$	<i>SITEUR</i>	nominal exchange rate SIT/EUR

*Model equations:*

The first four equations are estimated by FIML, the remaining equations are identities.

Standard deviations are given in brackets.



$$\begin{aligned}
CR_t = & 240.9398 + 0.740333 CR_{t-1} + 0.111727 GDPR_t \left(1 - \frac{TaxRate_t}{100}\right) \\
& (189.7449) \quad (0.1115) \quad (0.0330) \\
& - 1.007353 (STIRLN_t - Pi4_t) - 4.773533 Pi4_t \\
& (2.5848) \quad (2.4966)
\end{aligned}$$

$$\begin{aligned}
INVR_t = & 75.41731 + 0.932211 INVR_{t-1} + 0.264523 (VR_t - VR_{t-1}) \\
& (176.8549) \quad (0.1423) \quad (0.0924) \\
& - 0.455511 (STIRLN_t - Pi4_t) - 2.981241 Pi4_t \\
& (6.9044) \quad (3.1277)
\end{aligned}$$

$$\begin{aligned}
IMPR_t = & IMPR_{t-1} + 0.826449 (VR_t - VR_{t-1}) - 38.14954 SITEUR_t \\
& (0.0724) \quad (18.9336)
\end{aligned}$$

$$\begin{aligned}
STIRLN_t = & 0.811606 STIRLN_{t-1} - 0.000877 \frac{(M3N)_t}{PV_t} \cdot 100 \\
& (0.1375) \quad (0.0008) \\
& + 0.002746 GDPR_t \\
& (0.0026)
\end{aligned}$$

$$GDPR_t = CR_t + INVR_t + GR_t + EXR_t - IMPR_t$$

$$VR_t = GDPR_t + IMPR_t$$

$$PV_t = \frac{GDPR_t}{VR_t} \cdot GDPDEF_t + \frac{IMPR_t}{VR_t} \cdot IMPDEF_t$$

$$Pi4_t = \frac{PV_t - PV_{t-4}}{PV_{t-4}} \cdot 100$$

The objective function penalizes deviations of objective variables from their ‘ideal’ (desired, target) values. The ‘ideal’ values of the state and control

variables ( $\tilde{x}_t$  and  $\tilde{u}_t$  respectively) are chosen as follows:

Table 1: ‘Ideal’ values

<i>CR</i>	<i>INVR</i>	<i>IMPR</i>	<i>STIRLN</i>	<i>GDPGR</i>	<i>VR</i>	<i>PV</i>	<i>Pi4</i>	<i>TaxRate</i>	<i>GR</i>	<i>M3N</i>
1%	1%	2%	-0.25	1%	1.5%	0.75%	3	25.2	1%	1.75%

The ‘ideal’ values for most variables are defined in terms of growth rates (denoted by % in Table 1) starting from the last given observation (2003:4). For *Pi4* and *TaxRate*, constant ‘ideal’ values are used; for *STIRLN*, a linear decrease of 0.25 per quarter is assumed to be the goal.

The weights for the variables, i.e. the matrix  $W$  in the objective function, are first chosen as shown in Table 2a (‘raw’ weights) to reflect the relative importance of the respective variable in the (hypothetical) policy-maker’s objective function. These ‘raw’ weights have to be scaled or normalized according to the levels of the respective variables to make the weights comparable. To do so, the ‘raw’ weights are multiplied by normalization coefficients  $NC^i = (ML/MA^i)^2$ , where  $ML$  is the mean of a reference series and  $MA^i$  is the mean of the respective series  $i$ . The ‘correct’ weights obtained in this way are shown in Table 2b. The weight matrix is assumed to be constant over time (no discounting).

Next, the OPTCON2 algorithm is applied to this optimization problem in order to determine approximately optimal fiscal and monetary policies for Slovenia under the assumed objective function and the econometric model SLOVNL. Two different experiments are run: in experiment 1, two open-loop solutions are compared, a deterministic one where the variances and covariances of the parameters are ignored, and a stochastic one where the

Table 2: Weights

2a: ‘raw’ weights		2b: ‘correct’ weights	
variable	weight	variable	weight
<i>CR</i>	1	<i>CR</i>	3.457677
<i>INVR</i>	1	<i>INVR</i>	12.16323
<i>IMPR</i>	1	<i>IMPR</i>	1.869532
<i>STIRLN</i>	1	<i>STIRLN</i>	216403.9
<i>GDPR</i>	2	<i>GDPR</i>	2
<i>VR</i>	1	<i>VR</i>	0.333598
<i>PV</i>	1	<i>PV</i>	423.9907
<i>Pi4</i>	0	<i>Pi4</i>	0
<i>TaxRate</i>	2	<i>TaxRate</i>	37770.76
<i>GR</i>	2	<i>GR</i>	63.77052
<i>M3N</i>	2	<i>M3N</i>	0.090549

estimated parameter covariance matrix is taken into account. In experiment 2, the properties of the open-loop and the open-loop feedback solutions are compared. This exercise is not meant to determine actually optimal policies for Slovenia during the period considered (for this, the quality of the econometric model is not sufficient); instead, it should deliver some information about the convergence and the applicability of the OPTCON2 algorithm as implemented in *C#*.

## 5.2 Experiment 1: open-loop control

For experiment 1, two different open-loop solutions are calculated: deterministic and stochastic. The deterministic scenario assumes that all parameters of the model are known with certainty. In the stochastic case, the covariance matrix of the parameters as estimated by FIML is used but no updating of information occurs during the optimization process.

The results for two variables,  $GR$  and  $GDPR$ , are shown in Figures 3 and 4. Each figure shows the optimization results for the deterministic and the stochastic case and the ‘ideal’ and historical values of the respective variable.

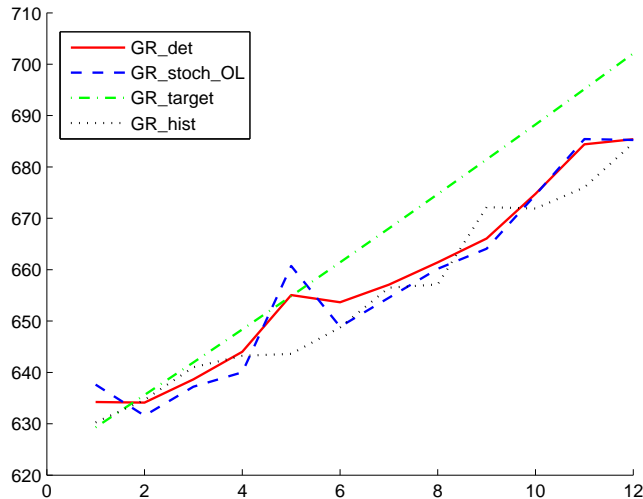


Figure 3: Deterministic vs. stochastic open-loop case ( $GR$ )

Figures 3 and 4 show that both the deterministic and the stochastic solution approximate the ‘ideal’ values rather well. This is also supported by the values of the objective function, which is 2,618,460.238 in the uncontrolled solution, 904,385.766 in the deterministic solution and 918,296.046 in the stochastic solution, showing a considerable improvement in system performance. An interesting detail is that the deterministic and the stochastic open-loop solutions are very similar, which goes in line with previous findings in a related study by Neck and Karbuz (1995).

In both cases (deterministic and stochastic) the algorithm needs 3 non-linearity runs to converge. The entire procedure took 2 seconds for the de-

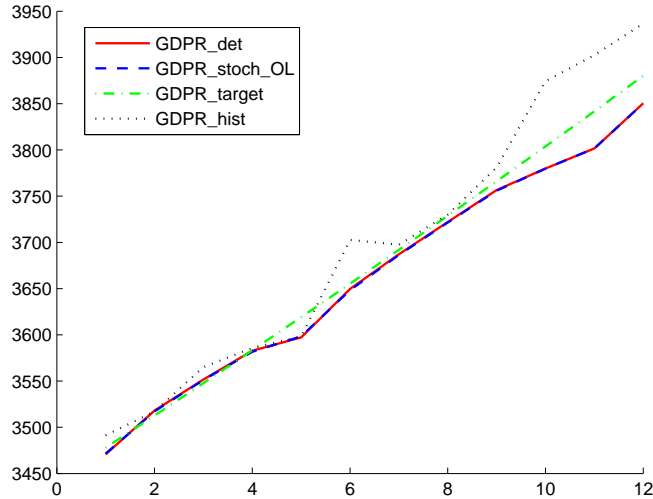


Figure 4: Deterministic vs. stochastic open-loop case (*GDPR*)

terministic case and 4 seconds for the stochastic case on a personal computer with 2 GHz Intel Core 2 Duo CPU and 4GB RAM. The results show that the OPTCON2 algorithm (OL-strategy) can be used to determine optimal open-loop controls, at least for small nonlinear econometric models.

### 5.3 Experiment 2: open-loop feedback control

The aim of experiment 2 is to compare open-loop and open-loop feedback controls. There is a problem concerning how to compare both strategies because open-loop controls do not take random disturbances into account during the optimization process. In order to make both strategies comparable, some adjustments to the open-loop controls are necessary: first the open-loop controls  $(u_t^*)_{t=1}^T$  are calculated for all periods, using the generated  $\theta^m$  ( $\theta^m = \hat{\theta} + \mu^m$ ). Then, assuming that the ‘true’ model is known with

the parameters  $\hat{\theta}$  and system error  $\varepsilon_t^m$ , the actual values of open-loop states ( $x_t^{a*} = f(x_{t-1}^{a*}, x_t^{a*}, u_t^*, \hat{\theta}, z_t) + \varepsilon_t^m, t = 1, \dots, T$ ) are determined, the only information which is available to the decision-maker. So the policies remain unchanged but the state variables are calculated taking the disturbances  $\mu^m$  and  $\varepsilon^m$  into account. The open-loop feedback solution is determined according to the algorithm sketched in Section 4. In this way, a comparison of both strategies under simulated ‘real’ uncertainty (disturbances) becomes possible.

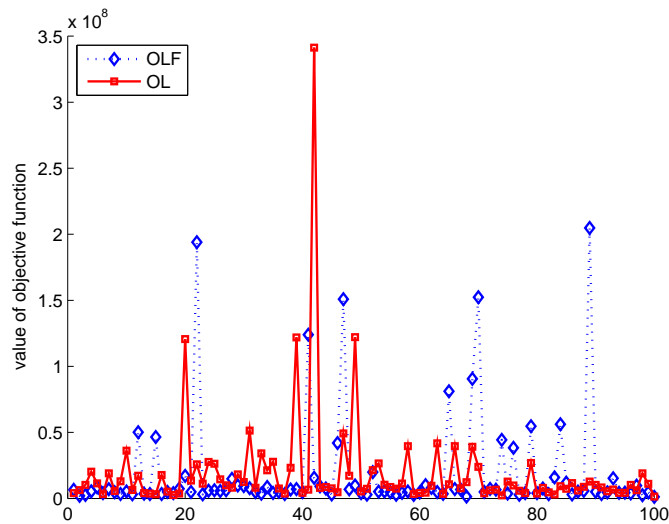


Figure 5: Open-loop vs. open-loop feedback control, value of objective function (100 Monte Carlo runs)

Figure 5 shows the results of a representative Monte-Carlo simulation, displaying the value of the objective function (loss) arising from applying OPTCON2 under 100 independent random Monte Carlo runs. Diamonds represent open-loop feedback results and squares open-loop results. One can see from this figure that in most runs the diamonds are below the squares (here: in 66 cases out of 100). This means that open-loop feedback con-

trols give better results (lower values of the cost function) in the majority of the cases investigated. But one can also see that there are many cases where either control scheme results in high losses. Many simulations having been run (with different numbers of Monte Carlo runs), the findings can be summarized as follows:

- In 60-75 percent of the cases, open-loop feedback controls give better results than open-loop controls.
- High losses occur in both the open-loop and the open-loop feedback case.
- For open-loop controls, high losses seem to be more frequent.

This result is somewhat unexpected because it means that (passive) learning does not necessarily improve the quality of the final results; it may even worsen them. One reason for this is the presence of the two types of stochastic disturbances: additive (random system error) and multiplicative error ('structural' error in the parameters). The decision-maker cannot distinguish between realizations of errors in the parameters and in the equations as he just observes the resulting state vector. Based on this information, he learns about the values of the parameter vector but may be driven away from the 'true' parameter due to the presence of random system error. The possibility of such a diversion can be expected to decrease during the planning horizon as new information (new realizations of the errors) are granted relatively less weight in the updating procedure as time passes by.

One possible way out of this dilemma is to introduce weights in the update

structure. In particular, in the correction procedure the correction term for the parameter estimate  $\theta$  is extended by a weighting parameter  $V_t$ :

$$\theta_{t/t} = \theta_{t/t-1} + V_t \Sigma_{t/t-1}^{\theta x} (\Sigma_{t/t-1}^{xx})^{-1} [x_t^{a*} - x_t^*], \quad 0 < V_t \leq 1. \quad (8)$$

Adjusting the updating procedure in this way means that better results can be expected under open-loop feedback control. The updating procedure aims at reducing the ‘structural’ error but can be disturbed by the random system error. Usually, this influence of the random system error can be expected to be especially strong at the beginning of the planning period. Introducing a time-dependent weighting parameter  $V_t$  serves to give less weight to revisions called for by learning during the earlier periods of the planning horizon than during the later periods. Different schemes were tried, and the weighted open-loop feedback scheme with the parameter  $V_t = \frac{t}{T-1}$  gave the best results, so in the simulations presented next this variant was used.

In a Monte-Carlo simulation to compare system performance under open-loop (OL) and weighted open-loop feedback (wOLF) control, the results are presented in Figure 6, again showing the values of the objective function against the number of the Monte Carlo run. Diamonds represent wOLF and squares OL results. In this simulation wOLF control gives better results: there are more diamonds below squares (77 out of 100) and only a few wOLF controls result in a very high loss.

Another way to show the improvement achieved by open-loop feedback, and even more so by weighted open-loop feedback control over open-loop control was introduced by Amman and Kendrick (1999), viz. scatter diagrams



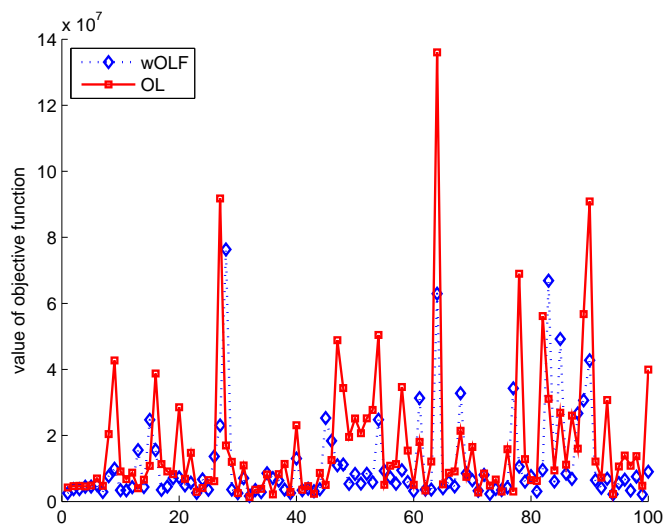


Figure 6: Open-loop vs. weighted open-loop feedback, value of objective function (100 Monte Carlo runs)

of values of the objective function in different runs. Figures 7 and 8 show such scatter diagrams for 1000 runs in each case. They show the main mass below the 45 degree line, meaning that OL results in higher losses than OLF (in 66.4% of the runs, Figure 7), and wOLF results in higher losses than OL (in 75.3% of the runs, Figure 8). These results corroborate those obtained by Amman and Kendrick (1999).

After running many simulations the results can be summarized as follows:

- In 70-80 percent of the cases considered, weighted open-loop feedback controls give better results than open-loop controls.
- Weighted open-loop feedback controls result in fewer cases of high loss than open-loop controls.

Using a weighting scheme for parameter updating thus increases the num-

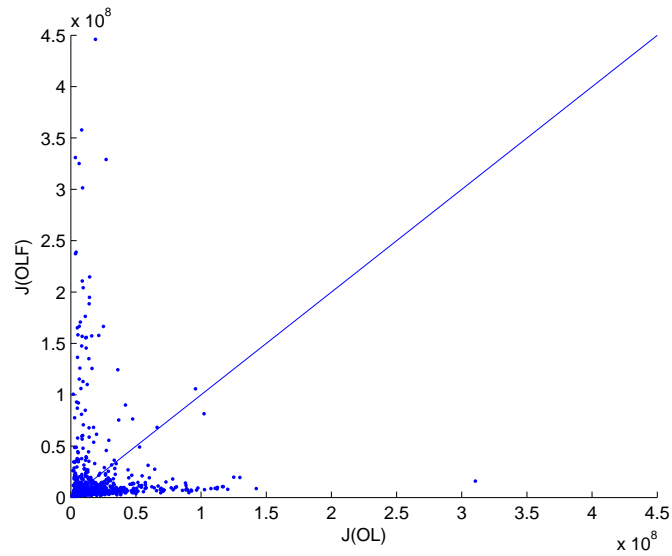


Figure 7: Scatter diagram of OL vs. OLF, values of objective function (1000 Monte Carlo runs)

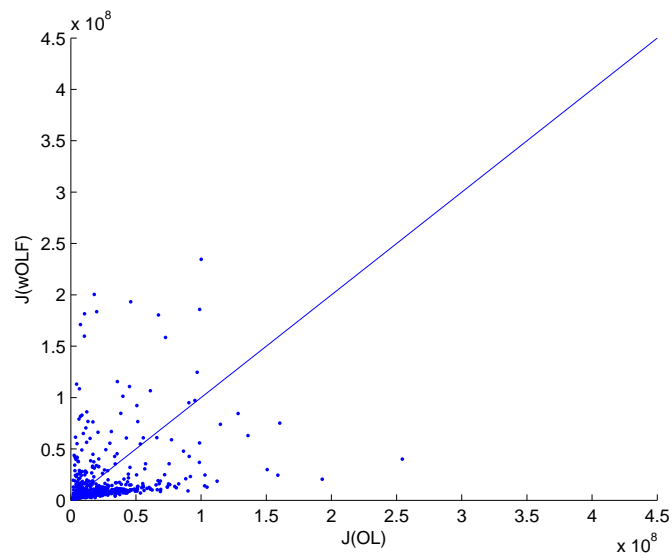


Figure 8: Scatter diagram of OL vs. wOLF, values of objective function (1000 Monte Carlo runs)

ber of runs where passive learning controls result in better values of the objective function than the control scheme without learning. Additionally, a decrease in the number of runs with very high losses can be achieved by introducing wOLF controls.

## 6 Concluding remarks

The present extension of the OPTCON algorithm, OPTCON2, calculates open-loop feedback in addition to open-loop control policies. It was programmed in the computer language *C#* and shown to converge for a small econometric model. The main improvement lies in learning about stochastically disturbed parameters during the control process. A comparison of open-loop control (without learning) and open-loop feedback control (with passive learning) shows that weighted open-loop feedback control outperforms open-loop control in a majority of the cases investigated for the small econometric model of Slovenia.

The next task is to apply OPTCON2 to larger and better macroeconomic models (in terms of their theoretical and statistical quality). Additional comparisons of the policy performance with respect to the postulated objective function are desirable; for example, it may be interesting to calculate controls by straightforward heuristic optimization procedures (see Gilli and Winker (2009), Winker and Gilli (2004), Lyra et al. (2010), among others) and assess their performance compared to the more sophisticated ones calculated by OPTCON2. Moreover, major extensions will have to include various schemes of active learning to deal with the dual nature of the control under

uncertainty along the lines of Kendrick (1981) for the linear case. Another challenge consists in incorporating rational (forward-looking) expectations and hence a non-causal structure in the dynamic system; see Amman and Kendrick (2000) for the linear case.

## 7 Acknowledgements

An earlier version of this paper was presented at the 3rd International Conference on Computational and Financial Econometrics (CFE 09), Limassol, Cyprus. Financial support from the EU-HRM Program, Marie Curie Research Training Network COMISEF is gratefully acknowledged. Thanks are due to very helpful comments and suggestions for improvement by three anonymous referees and an associate editor of this journal; the usual caveat applies.

## References

- Amman, H., 1996. Numerical methods for linear-quadratic models. In: Amman, H. M., Kendrick, D. A., Rust, J. (Eds.), *Handbook of Computational Economics*. Vol. I. Elsevier, Amsterdam, Ch. 13, pp. 587–618.
- Amman, H. M., Kendrick, D. A., 1999. Should macroeconomic policy makers consider parameter covariances? *Computational Economics* 14, 263–267.
- Amman, H. M., Kendrick, D. A., 2000. Stochastic policy design in a learning environment with rational expectations. *Journal of Optimization Theory and Applications* 105 (3), 509–520.

- Aoki, M., 1989. Optimization of Stochastic Systems. Topics in Discrete-Time Dynamics, 2nd Edition. Academic Press, New York.
- Blueschke-Nikolaeva, V., Blueschke, D., Neck, R., 2010. OPTCON2: An algorithm for the optimal control of nonlinear stochastic dynamic systems with passive learning. Tech. rep., Department of Economics, Klagenfurt University.
- Chen, B., Zadrozny, P. A., 2009. Multi-step perturbation solution of nonlinear differentiable equations applied to an econometric analysis of productivity. Computational Statistics and Data Analysis 53, 2061–2074.
- Chow, G. C., 1975. Analysis and Control of Dynamic Economic Systems. John Wiley & Sons, New York.
- Chow, G. C., 1981. Econometric Analysis by Control Methods. John Wiley & Sons, New York.
- Coomes, P. A., 1987. PLEM: A computer program for passive learning, stochastic control experiments. Journal of Economic Dynamics and Control 11, 223–227.
- Gilli, M., Winker, P., 2009. Heuristic optimization methods in econometrics. In: Belsley, D. A., Kontoghiorghes, E. J. (Eds.), Handbook of Computational Econometrics. John Wiley, Chichester, Ch. 3, pp. 81–119.
- Kendrick, D. A., 1981. Stochastic Control for Economic Models. McGraw-Hill, New York, second edition 2002 online at [www.eco.utexas.edu/faculty/Kendrick](http://www.eco.utexas.edu/faculty/Kendrick).

- Kendrick, D. A., Amman, H. M., 2006. A classification system for economic stochastic control models. *Computational Economics* 27, 453–481.
- Lyra, M., Paha, J., Paterlini, S., Winker, P., 2010. Optimization heuristics for determining internal rating grading scales. *Computational Statistics and Data Analysis* 54, 2693–2706.
- Matulka, J., Neck, R., 1992. OPTCON: An algorithm for the optimal control of nonlinear stochastic models. *Annals of Operations Research* 37, 375–401.
- Neck, R., 1984. Stochastic control theory and operational research. *European Journal of Operations Research* 17, 283–301.
- Neck, R., Karbuz, S., 1995. Optimal budgetary and monetary policies under uncertainty: a stochastic control approach. *Annals of Operations Research* 58, 379–402.
- Winker, P., Gilli, M., 2004. Applications of optimization heuristics to estimation and modelling problems. *Computational Statistics and Data Analysis* 47, 211–223.