Centre for Computational Finance and Economic Agents

WP010-06

Working Paper Series

Dietmar Maringer
Mark Meyer

Smooth Transition Autoregressive Models – New Approaches to the Model Selection Problem

October 2006

CCFEA

www.ccfea.net

Smooth Transition Autoregressive Models – New Approaches to the Model Selection Problem

Dietmar Maringer* Mark Meyer**

October 10, 2006

Abstract

It has been shown in the literature that the task of estimating the parameters of nonlinear models may be tackled with optimization heuristics. Thus, we attempt to carry these intuitions over to the estimation procedure of smooth transition autoregressive (STAR, Teräsvirta, 1994) models by introducing the following three stochastic optimization algorithms: Simulated Annealing, (Kirkpatrick, Gelatt, and Vecchi, 1983), Threshold Accepting (Dueck and Scheuer, 1990) and Differential Evolution (Storn and Price, 1995, 1997). Besides considering the performance of these heuristics in estimating STAR model parameters, our paper additionally picks up the problem of identifying redundant parameters which, according to our view, has not been addressed in a satisfactory way by now. The resulting findings of our simulation studies seem to argue for an implementation of heuristic approaches within the STAR modeling cycle. In particular for the case of STAR model specification, an application of these heuristics might offer valuable information to empirical researchers.

Keywords: Univariate time series modeling, Regime-switching models, Model specification, Heuristic optimization.

JEL classification: C22, C51, C63

1 Introduction

Nonlinear modeling approaches attracted a great deal of attention over the past decades. Hence, at least for prominent representatives like the threshold model of

*Centre for Computational Finance and Economic Agents (CCFEA), University of Essex, Wivenhoe Park, Colchester CO4 3SQ, UK, dmaring [at] essex.ac.uk

**Department of Economics, Justus-Liebig-University of Giessen, Licher Strasse 64, D-35394 Giessen, Germany, Mark.Meyer [at] wirtschaft.uni-giessen.de

Tong (1978, 1983) and Tong and Lim (1980), the Markov-switching model of Hamilton (1989), the artificial neural network model (see White, 1989 for an exposition) or the smooth transition model (see Teräsvirta, 1998 for an exhaustive introduction) we might assert that they have been well established not only in the theoretical time series literature but also on many areas of application.[1] The smooth transition model which takes center stage in the following originated from Bacon and Watts (1971) and was incorporated into an empirical modeling cycle by Teräsvirta (1994). Basically, it can be described as a regime-switching model with the current state of the system being defined by an observable variable and the transition between different regimes being determined by a continuous parametric function. Within this framework we are going to argue that the "problem of sensible starting values" (corresponding to the well know fact that results of the optimization algorithms commonly used for estimation are affected by the choice of initial starting values) as well as the "problem of redundant parameters" (i.e., the problem of imposing zero restrictions on individual parameters) have not been addressed satisfactorily yet: Both requirements depend on personal judgement by the model builder. Moreover, with rising lag lengths both working stages turn out to be quite elaborate and time consuming.

Smooth Transition Autoregressive (STAR) models may be taken as generalized TAR models. For the later, Wu and Chang (2002) and Baragona, Battaglia, and Cucina (2004) already suggested an application of heuristic approaches for parameter estimation. Thus, we consider it worthwhile an attempt carrying forward the general perceptions of those authors to the STAR framework. Yet, our study supplements theirs by introducing new heuristics within the STAR context. Moreover, our suggested algorithms explicitly target the "problem of redundant parameters" which has neither been considered by Wu and Chang (2002) nor by Baragona et al. (2004).

Based on a summarized representation of the methodological properties of STAR models in section 2, section 3 introduces two optimization heuristics which seem generally capable of estimating STAR-model parameters. Given these encouraging findings, section 4 broadens the conceptual formulation to the "problem of redundant parameters" and introduces a third heuristic estimation approach: Based on two STAR-specifications taken from the literature we demonstrate by means of Monte Carlo simulations that heuristics optimization techniques might be adopted to the challenge of finding parsimonious model specifications. Section 5 concludes.

2 STAR Models

2.1 Methodological Representation

This section summarizes the methodological framework under consideration. The chosen specifications are based on the article of van Dijk, Teräsvirta, and Franses

[1] See Franses and van Dijk (2000) or Teräsvirta (2006) for overviews of the most common nonlinear models in applied econometrics.

(2002) which represents a recent overview of the STAR models literature. See also Teräsvirta (1994, 1998) for further reading.

Let y_t, s_t denote two time series. If we assume y_t to depend on a vector of covariates $\mathbf{x}_t$ with the functional relationship between y_t and $\mathbf{x}_t$ being determined by the "transition variable" s_t, a general approximation to this interrelationship might be carried out as

$$y_t = \phi_1' \mathbf{x}_t + \phi_2' \mathbf{x}_t \cdot G(s_t; \gamma, c) + \epsilon_t, \tag{1}$$

with ϕ_i $(i \in 1,2)$ marking two different parameter vectors, $G(s_t; \gamma, c)$ denoting a continuous "transition function" and ϵ_t representing the error term.[2] Whereas equation (1) already depicts the original smooth transition approach of Teräsvirta (1994), a minor modification results in the following specification

$$y_t = \phi_1' \mathbf{x}_t \cdot (1 - G(s_t; \gamma, c)) + \phi_2' \mathbf{x}_t \cdot G(s_t; \gamma, c) + \epsilon_t. \tag{2}$$

For $G(s_t; \gamma, c) \in [0,1]$, equation (2) has the straightforward interpretation as a two-regime-switching model allowing for a smooth passage between those regimes appointed to the extreme values of the transition function.

A very popular choice for the transition function is given by the logistic function

$$G(s_t; \gamma, c) = (1 + \exp\{-\gamma(s_t - c)\})^{-1}, \quad \gamma > 0. \tag{3}$$

With increasing values for s_t, function (3) changes monotonically from 0 to 1 with the so-called "threshold parameter" c locating a well balanced situation between both regimes as $G(c; \gamma, c) = 0.5$. γ can be interpreted as a steepness parameter which determines the speed of transition between both regimes. For $\gamma \to \infty$, function (3) collapses into a discrete indicator function whereby equation (4) reproduces a two-regime case of the threshold autoregressive (TAR) model introduced by Tong (1978) and Tong and Lim (1980).

Within this general class of models we turn our attention to the smooth transition autoregressive (STAR) model. Along the lines of specification (2), STAR models can be represented as

$$\begin{aligned} y_t = &(\phi_{1,0} + \phi_{1,1} y_{t-1} + \ldots + \phi_{1,p} y_{t-p}) \cdot (1 - G(s_t; \gamma, c)) \\ &+ (\phi_{2,0} + \phi_{2,1} y_{t-1} + \ldots + \phi_{2,p} y_{t-p}) \cdot G(s_t; \gamma, c) + \epsilon_t. \end{aligned} \tag{4}$$

Equations (3) and (4) constitute a "logistic smooth transition autoregressive" (LSTAR) model. With regards to alternative STAR model variants (see, e.g., van Dijk et al. (2002) for an overview of alternative STAR specifications) like "exponential

[2] Basically, ϵ_t has to meet the requirements of a martingale difference sequence but most applications consider normally distributed error terms as this condition establishes equality between nonlinear least squares estimation approaches and maximum likelihood estimation. Regularity conditions for nonlinear least squares estimates being consistent and asymptotically normal are given by (among others) Pötscher and Prucha (1997) or Wooldridge (1994).

smooth transition autoregressive models" (ESTAR), it seems that LSTAR specifications reached an outstanding popularity in econometric applications: On the area of business cycle analysis Teräsvirta and Anderson (1992) prefer LSTAR specifications for modeling industrial production growth rates in nine out of the 14 cases under consideration. Lütkepohl, Teräsvirta, and Wolters (1999) prefer a logistic transition function for modeling non-linearity in a German M1 demand equation. Finally, within the field of financial econometrics Sarantis (2001), e.g., estimates STAR models of annual stock price growth rates. His results indicate that the dynamics of the indices under consideration point to LSTAR models in five out of seven cases. Hence, due to its outstanding prominence, the logistic transition function does provide a natural starting point for our Monte Carlo examples.[3] Moreover, there are also methodological inducements for a closer examination of the LSTAR model. As indicated by Chan and McAleer (2002), the general problems of gradient based optimization techniques in estimating STAR models might especially arise in the LSTAR framework.

The smooth transition regression framework might of course also be applied to multivariate vector processes. Additionally, an introduction of further transition functions to equation (1) would allow to incorporate further regime dependencies. However, a majority of empirical applications has been confined to univariate settings with the number of regimes being restricted to two (see, e.g., Skalin and Teräsvirta, 1999, Sarantis, 2001 or Skalin and Teräsvirta, 2002). Hence, our study also confines its focus to the class of smooth transition models given by equations (1) or (2).[4]

2.2 Conventional Modeling Procedure

A detailed description of the STAR modeling cycle is given by Teräsvirta (1994). The essential steps of this specification procedure might be summarized as

1. Specification of a linear AR model,
2. Testing the linear AR model against pre-selected STAR model candidates by a sequence of Lagrange multiplier tests,
3. STAR model estimation by nonlinear least squares and model evaluation.

It goes beyond the aims of this paper to give a comprehensive account of the particulars of this well-established modeling cycle. The interested reader is referred to the supplementary annotations of Escribano and Jordá (1999), Leybourne, Newbold, and Vougas (1998) and Chen (2003) as well as to the survey of van Dijk et al. (2002)

[3] See also Chan and McAleer (2003, p. 586): "*STAR models, especially LSTAR models, have been successfully applied in a number of areas.*"

[4] See the references in van Dijk et al. (2002) for further readings on more complex STAR specifications. A recent contribution to the issue of vector smooth transition models has also been published by Camacho (2004).

hereunto. Instead, we focus our attention to the latter of the mentioned topics: Model estimation, evaluation and re-specification.

The adjusted standard deviation of the residuals

$$\sigma_e = \sqrt{\mathbf{e}'\mathbf{e}/(T-k)}$$

with $\mathbf{e}$ denoting the vector of residuals, T marking the number of observations and k representing the number of estimated variables, usually serves as objective function for selecting the final STAR model specification . However, we also consider the information criteria[5] suggested by Akaike (AIC) and Schwartz (SBC)

$$\text{AIC} = \ln(\mathbf{e}'\mathbf{e}) + 2(T-k)/T$$
$$\text{SBC} = \ln(\mathbf{e}'\mathbf{e}) + (T-k)\ln(T)/T$$

in our specification experiments. These standard statistics have been chosen as they are usually being published in econometric analyses. Hence, as will be seen in the following sections, they provide natural benchmarks for our own simulation results.

For a given set of parameters (γ^0, c^0) the task of estimating equation (1) (or, alternatively equation (2)) reduces to a straightforward application of the ordinary least squares approach (Leybourne et al., 1998). The parameters of the transition function, however, are generally not known and cannot be derived from a closed form solution. Estimation conventionally starts with a two dimensional grid-search over (γ, c). Assuming plausible ranges for the values of c and γ, $[r_l^c; r_u^c]$ and $[r_l^\gamma; r_u^\gamma]$, n equidistant points can be selected within these intervals with $r_i = r_l + (i-1)\cdot\frac{r_u - r_l}{n-1}$ for $i = 1 \ldots n$. For any pair (r_i^c, r_j^γ), the parameter vectors (ϕ_1, ϕ_2) are then estimated via OLS. The outcome of this is a set of estimates $(\hat{\phi}_1^0, \hat{\phi}_2^0; \hat{\gamma}^0, \hat{c}^0)$. Eventually, a combination of (r_i^c, r_j^γ) is chosen that corresponds to the global minimum of the residual sum of squares. Yet, with n chosen too small the actual optimum might easily be overlooked whereas large values for n increase the computational time substantially (note that the grid consists of n^2 combinations). Therefore, the recommended procedure (see, e.g., van Dijk et al., 2002, p. 19-21) starts with a coarse grid over γ and c yielding initial starting values $(\hat{\phi}_1^0, \hat{\phi}_2^0; \hat{\gamma}^0, \hat{c}^0)$ which is then repeatedly refined by means of numerical optimization algorithms like downhill or gradient search (e.g., the popular Broyden-Fletcher-Goldfarb-Shanno (BFGS) method). However, the solution space tends to have many local minima and the literature is well aware of the associated problems of deterministic numerical optimization techniques that are either based on first order calculus (such as gradient search) or on complete enumeration

[5] See Anderson (2002) for a simulation study considering the performance of various information criteria in nonlinear settings.

(as mimicked by grid search).[6] Heuristic methods, on the other hand, are a new class of optimization techniques that are able to overcome these problems.

3 Heuristic Parameter Estimation

The following pages might be conceived as introductory notes to the topic of heuristic optimization in a STAR context. Subsection 3.1 transposes two single agent based optimization heuristics to the LSTAR estimation problem which has been constituted at the end of section 2. Applied details of the suggested algorithms as well as simulation results clarifying their persuasive performance in estimating the set of parameters $(\phi_1, \phi_2; \gamma, c)$ are presented within subsection 3.2.

3.1 Threshold Accepting and Simulated Annealing

Unlike traditional deterministic numerical methods, heuristic methods incorporate non-deterministic stochastic elements. This randomness might flow into the generation of new candidate solutions and/or into the acceptance criterion for new solutions.

3.1.1 The Threshold Accepting Algorithm

Table 1 opposes the gradient search approach to threshold accepting (TA), a method introduced in Dueck and Scheuer (1990).[7] $\mathbf{x}$ indicates the vector of objective variables and $f(\mathbf{x})$ is the corresponding value of the objective function. Both approaches are neighborhood search methods where the new solution is within a certain distance from the current solution.

The main differences between these two methods are that (i) TA starts off with a random initial value, and (ii) TA also allows impairments of $\mathbf{x}$ that are not too bad, i.e., do not exceed a certain threshold (hence the name). Consequently, local optima can be overcome, and, even if the initial solution is close to a local optimum, TA does not tenaciously drive the convergence process towards this inferior solution. Typically, the threshold is rather generous in the beginning, in the course of iterations, however, it is lowered towards zero such that in the last iterations the behavior of this approach assimilates the gradient search in that it does not allow impairments any longer. Yet, even if the threshold equals zero, due to its inherent stochastics, TA still does not equal gradient search

[6] Chan and McAleer investigated finite sample properties of this conventional estimation procedure by means of numerical simulations. For the STAR models under investigation they conclude: *As LSTAR is a more complicated model than ESTAR, it is more difficult for the algorithm to converge to the true values. BFGS is not a robust algorithm for estimating LSTAR.* (Chan and McAleer, 2002, p. 518f)

[7] For presentations and applications see, e.g., Winker (2001), Maringer (2005), and Winker and Maringer (2007).

Table 1: Numerical search methods for minimization problems $\min_{\mathbf{x}} f(\mathbf{x})$

(a) Gradient search

```
Initialize x with "good" guess

REPEAT
  make sophisticated guess for Δx
  x^n := x + Δx
  if f(x^n) < f(x)
     then x := x^n

UNTIL converged
```

(b) Threshold Accepting

```
Initialize x with random guess

REPEAT
  make random guess for Δx
  x^n := x + Δx
  if f(x^n) < f(x) + Threshold
     then x := x^n
  lower Threshold
UNTIL halting criterion met
```

In the STAR estimation problem with a given lag structure, the variables of interest are $\mathbf{x} = (c, \gamma)$ which are initialized randomly. In the search process itself, the generation of $\Delta\mathbf{x}$ consists of randomly deciding which $i \in (1,2)$ of the two variables to alter, and adding a normally distributed variable $\Delta x_i \sim N(0, r_i)$ to it. r_i indicates the range of the neighborhood from which the new solution will be drawn. Another main ingredient of the TA is the threshold sequence which determines how tolerant the acceptance criterion is towards impairments. The initial value for the threshold should be such that a certain percentage of impairments are accepted; this fraction ought then to be lowered towards zero implying that only improvements are accepted. The thresholds should therefore be chosen with respect to the distribution of the changes in the objective function, Δf, – which is obviously linked to the distribution of the Δx_i's via the r_i's.

3.1.2 The Simulated Annealing Algorithm

TA is a modified version of Simulated Annealing (SA, Kirkpatrick et al., 1983) which uses a stochastic acceptance rule: impairments are accepted with a certain probability $p = 1/(1 + \exp(\Delta f / T_i))$ (Boltzmann function) or $p = \min(1, \exp(-\Delta f / \theta_i))$ (Metropolis function) where θ_i is the "temperature" in iteration i. Initially, a high temperature makes the acceptance of even large impairments quite likely; in the course of iterations, the temperature is lowered – and so is the likelihood of accepting an impairing $\Delta\mathbf{x}$. For either function, the probability is the lower the more severe the impairment. Note however that with the Metropolis function, improvements are always accepted while the Boltzmann function accepts them with an increasing probability which is always above 0.5. TA is computationally slightly less demanding because of its deterministic acceptance rule. In SA, on the other hand, even very severe downhill steps have a non-zero probability of being accepted which might be advantageous if changes in some of the decision variables have more severe effects on the objective function than others (as we will find applies to our problem). Apart from the neigh-

borhood for generating new candidate solutions, the second main ingredient for this heuristic is the cooling sequence which, akin to TA's threshold sequence, ought to be linked to the magnitude and distribution of the Δf's.

3.2 Implementation and Computational Results

3.2.1 Configuration

For the heuristic approaches,the initial values for γ are randomly drawn from an exponential distribution with expected value 1 while c comes from a uniform distribution covering a 80% fractile of the observed transition variable series s_t. For the generation of new candidate solutions in the optimization process, one of the two variables was picked randomly and changed by a normally distributed random value $\epsilon \sim N(0, \sigma_v)$ where $\sigma_c = .5$ and $\sigma_\gamma = 1$. For the second main ingredient of the heuristics, the threshold sequence (TA) and the cooling sequence (SA), respectively, 100 000 random combinations of γ and c and a corresponding neighbor were generated, the remaining parameters of the STAR models where estimated and the differences in the corresponding objective functions where computed. This resulted in a rough estimate of the distribution of the Δf's. For TA, the initial threshold was set to the median of these values which is then linearly lowered to 0 in the course of 80% of the iterations, leaving the last 20% of the iterations for a strict downhill search. For SA, the stochastic Metropolis function replaced the deterministic Threshold rule; the temperature was chosen such that the corresponding TA Threshold T_i (or, if this is zero, a small positive value ε) has a 50% probability to be exceeded, i.e., $\theta_i = \max(\varepsilon, T_i)/\ln(0.5)$. After the last iteration, either algorithm reports the best of all solutions found in the search process which is often called the elitist. The number of iterations was limited to 10 000. Implemented with Matlab 7, the runtime on a Pentium 4 pc was approximately five seconds.

In this implementation, we impose neither upper nor lower limits on the values for γ in the search process (including the non-negativity constraint for γ). c, however, must be within the range between the 10% and the 90% percentiles of the threshold variable. Preliminary experiments showed that abandoning these limits in some cases resulted in extreme values for c and γ where the transition function G was no longer smooth or reserved one of the two regimes for a very limited number of observations when the lag structure is optimized simultaneously.

3.2.2 Valuation of Suitability

As heuristics are non-deterministic, different runs can yield different results because they, too, can converge to a local optimum. Repeated runs are therefore advisable, and frequent convergence towards the known optimum or a reliable benchmark can indicate the reliability of the method. For a first application, we chose the well-known

Canadian lynx data set described in and with the lag structure of Teräsvirta (1994)

$$y_t = \sum_{\ell \in \mathscr{L}_1} \alpha_\ell y_{t-\ell} + \sum_{\ell \in \mathscr{L}_2} \beta_\ell y_{t-\ell} \cdot G(\gamma, c, y_{t-3}) + e_t \tag{5}$$

$$G(\gamma, c, y_{t-3}) = \left(1 + \exp\left(\frac{-\gamma}{\sigma_y} \cdot (y_{t-3} - c)\right)\right)^{-1} \tag{6}$$

where $\mathscr{L}_1 = \{1\}$ and $\mathscr{L}_2 = \{2,3,4,9,11\}$ are the sets of included lags. The objective is to find values for γ and c that minimize the adjusted standard deviation of the residuals, σ_u. The reported optimal values are $\gamma^*/\hat{\sigma}_y = 1.73/1.8, c^* = 2.73$ and will be used as benchmark.

Table 2 summarizes some key statistics from a total of 8216 runs with different objective functions (standard deviation of the residuals, Akaike's information criterion; Schwarz Bayesian Criterion) and algorithms. The best results for the individual settings are given by the "minimum" line whereas "maximum" identifies the worst results found. As a measure of the algorithms' convergence rates we also computed the standard deviation ("SD") for each set of estimation results. Overall, these results indicate that both heuristics have no apparent problems in identifying the optimal solution $(\gamma^*, c^*) = (1.76, 2.73)$ for this instance. (Deviations of the optimized $\hat{\gamma}$ from the afore mentioned benchmark solution are due to the higher precision of $\hat{\sigma}_y$.)

Table 2: Results for the Lynx data set with given lag structure

criterion	σ_e		AIC		SBC	
heuristic	SA	TA	SA	TA	SA	TA
minimum	0.186924	0.186924	-3.279613	-3.279613	-3.074974	-3.074974
average	0.186961	0.186924	-3.279613	-3.279613	-3.074973	-3.074974
maximum	0.212929	0.186925	-3.279599	-3.279609	-3.074950	-3.074973
SD	0.000776	0.000000	0.000002	0.000000	0.000002	0.000000
runs	1378	1367	1368	1368	1368	1367

4 The Specification Problem

The previous section already enlightened the ability of TA and SA heuristics to find optimal parameter estimates $(\phi_1^*, \phi_2^*; \gamma^*, c^*)$ within a given lag structure. Yet, from the viewpoint of an application-oriented economist this seems to be only a necessary characteristic of our estimation proposal. Model evaluation and respecification surely represents the more demanding and time consuming challenge in empirical applications of the STAR modeling cycle. As complete enumeration of all possible lag structures usually proves unfeasible within reasonable time, the common way of finding a parsimonious model is to start with a full lag structure and then to keep

on estimating the parameters, removing lags whose estimates do not reach a minimum level of significance.[8] But this approach suffers from two severe shortcomings: relevant lags might be excluded prematurely, and the exclusion process might be misguided when one or more of the interim parameter estimations report local optima.

Indeed, experience taught us that this kind of "experimenting with more parsimonious models" bears its problems, as this task often turns out to be considerably time consuming. However, this instance seems to constitute an unstudied problem for a broader application of optimization heuristics. This section therefore analyzes the performance of heuristics whose underlying design simultaneously targets the estimation as well as the specification problem. Following a short description of the upgraded TA and SA configurations employed within this section, subsection 4.2 introduces the ideas of a complementary, population based heuristic (namely Differential Evolution). The performance of these three optimization heuristics is being analyzed by the two Monte Carlo studies of subsections 4.3 and 4.4, respectively. Subsection 4.5 completes the resulting findings with basic diagnostics of the heuristically optimized model specifications. Subsection 4.6, which summarizes, leads over to our general conclusions.

4.1 Joint Optimization of Parameters and Lags with TA and SA

In the context of VEC-models, Winker and Maringer (2005) have already shown that the lag selection problem can be solved heuristically. From the heuristic's point of view, the main difficulty with a joint optimization of the parameters and the lag structure is that modifications come with different changes of the fitness function: while a slight change in γ or c will not have a dramatic effect on f, including or excluding one lag can easily make the current parameter values useless and has therefore a much higher impact on f. Hence, given a certain threshold or temperature for the heuristic, a change in the parameters (with values Δf closer to zero) will have a higher chance of being accepted than a change in the lag structure. There would be several different remedies to overcome this problem. To name just two rather straightforward modifications: One might introduce one high and one low threshold (cooling) sequence and distinguish in the acceptance decision whether the parameters or the lag structure has been changed. Alternatively, one might allow for several search steps for the parameters after a change in the lag structure and only then decide over the acceptance of this sequence of modifications. In preliminary experiments, however, it turned out that sticking to the simple original version of TA and SA, respectively, and allowing

[8] To the best of our knowledge, the common procedure on the evaluation stage still initiates as follows: "*Because of the existence of local minima [...] the first task is to check whether the estimates look reasonable. [...] Excessively large standard deviation estimates for coefficient estimators [...] suggest that the model contains redundant parameters. All of the parameters estimated with large standard deviations need not be redundant; usually, a subset is. Experimenting with more parsimonious models reveals which variables can actually be omitted from the model.*" (Teräsvirta, 1994, p. 213).

Table 3: Pseudocode for the TA implementation

```
Initialize x = (c, γ) with random values;
Initialize binary string ℒ with random values;
FOR it = 1 to NumberCandidateSolutions
  with probability π_p
    add a random value to either c or γ;
  else
    randomly pick one element τ of ℒ and switch its value from 0 (1) to 1 (0);
  compute the resulting change in the objective function, Δs_u;
  if Δs_u < threshold
     keep modifications;
  else
     undo modifications and keep previous solution;
  lower threshold;
END
```

for more restarts was more effective than complicating the algorithm and increasing computational costs.

For the joint optimization problem at hand, this idea can be adopted as follows: In each iteration, the algorithm first chooses (randomly) whether to modify the parameters of the transition function (with probability π_p) or the lag structure $\mathscr{L}$ (with probability $1-\pi_p$). Modifications of c and γ are carried out as described in the previous section. For the modification of the lag structure, one lag ℓ is picked randomly and excluded if it is included in the current model and included otherwise. This can be implemented with a binary vector of length L where L is the maximum lag and the vector's ℓ-th element indicates whether or not lag ℓ is included; changing the status can then be done by simply negating this element. Table 3 summarizes the main steps of the algorithm.

For the application to the problem (5), two cases were distinguished were either only the lags in $\mathscr{L}_2$ are changed or one lag in a randomly picked set $\mathscr{L}$ is modified. The probability of whether to change a lag or one of the parameters γ and c is 0.5; if applicable, the conditional probability to pick either $\mathscr{L}_1$ or $\mathscr{L}_2$ is also 0.5. The number of iterations were increased to 50 000 and 100 000, respectively. The threshold and cooling sequences were determined akin to the original problem based on an estimation of the distribution of the Δf's which reflect the new set of decision variables.

4.2 Joint Optimization with Differential Evolution

Both TA and SA can be classified as trajectory methods where, figuratively speaking, a single agent traverses the solution space. While this type of optimization heuristic is considered quite powerful and easy to implement, a vast and rough solution space

with many local optima can make the search cumbersome: in order to overcome local optima, the acceptance criterion should be rather tolerant towards impairments – which, however, might also turn the search path into a (more or less directed) random walk. Also, when the initial solution is far away from the global optimum or the path between them is separated by regions of infeasible solutions, it is in the nature of local search algorithms that a high number of iterations is required.

Opposed to these single agent methods, population based methods consider several solutions at a time. Frequently in evolution or nature based methods, new candidate solutions are generated by (slightly) altering a solution ("mutation") or by combination of the properties of two or more existing solutions ("cross-over"). A rather novel approach among this group is Differential Evolution (DE), introduced in Storn and Price (1995) which has originally been developed for optimization in continuous spaces and which needs no assumption about the distribution of the Δf's. Unlike other optimized heuristics, DE is often claimed to require little or virtually no tuning. Vesterstrøm and Thomsen (2004) find in a computational study with 23 benchmark problems that DE outperforms other multi-agent methods. We therefore adapt DE as an alternative heuristic approach to the STAR model selection problem.

The basic idea of DE is that for each solution $\mathbf{x}_p$ of the current population, a new offspring solution $\mathbf{o}_p$ is generated. If $\mathbf{o}_p$ has a lower objective value f than $\mathbf{x}_p$, the offspring replaces $\mathbf{x}_p$, otherwise solution $\mathbf{x}_p$ survives. A new candidate solution $\mathbf{x}_c$ is generated as follows. First, three distinct members of the current population, m_1, m_2, and m_3, are randomly picked such that $p \neq m_1 \neq m_2 \neq m_3$. Then, an interim solution $\widetilde{\mathbf{o}}_p$ is computed according to $\widetilde{\mathbf{o}}_p = \mathbf{x}_{m_1} + F \cdot (\mathbf{x}_{m_2} - \mathbf{x}_{m_3})$. Finally, the new solution $\mathbf{o}_p$ consists of a crossed-over combination between $\mathbf{x}_p$ and $\widetilde{\mathbf{o}}_p$ where each element has a probability of π to come from $\widetilde{\mathbf{o}}_p$ and from $\mathbf{x}_p$ otherwise. The new candidate solution $\mathbf{o}_p$ therefore contains (combinations of) values of up to four current solutions: $\mathbf{x}_{m_1}$, $\mathbf{x}_{m_2}$, $\mathbf{x}_{m_3}$, and $\mathbf{x}_p$. If one element of $\mathbf{x}$ has the same value in all four of these parenting solutions, then so will the offspring's. If they disagree, the offspring will either inherit p's value (with a low probability of $(1-\pi)$) or will be given a new value which is in the neighborhood of the corresponding value in $\mathbf{x}_{m_1}$. The rule for generating new candidate solutions requires all decision variables to be continuous. For the lag selection problem (and, eventually for the dummy selection problem for our second data set), a vector of real numbers was used where each element represents one lag (and, where applicable, one dummy); positive (negative) values indicate inclusion (exclusion) of this lag and dummy, respectively. Table 4 lists the pseudo-code.

For the DE implementation, the crucial parameters are the population size, the number of iterations, the scaling factor F, and the cross-over probability π. As a general rule, the population size should be at least twice the number of decision variables. For the Lynx data set with at most 24 lags and STAR parameters to be optimized, the population was therefore set to $P = 50$. Typically, $F = 0.5$ and $\pi = 0.8$ or $\pi = 0.9$ are chosen; we used $F = 0.5$ and $\pi = 0.8$ for any of the following models. In order to make the results comparable to those from SA and TA, the number of iterations in DE was

Table 4: Pseudocode for the DE implementation

Initialize $\mathbf{x} = (c, \gamma)$ with random values;
Initialize real string $\mathscr{L}$ with random values;
FOR it = 1 to (NumberCandidateSolutions/PopulationSize)

Generate new offspring solutions:
FOR p = 1 to PopulationSize
randomly pick m_1, m_2, and m_3 with $p \neq m_1 \neq m_2 \neq m_3$
$\widetilde{\mathbf{o}}_c = \mathbf{x}_{m_1} + F \cdot (\mathbf{x}_{m_2} - \mathbf{x}_{m_3})$;
$\widetilde{\mathscr{O}}_c = \mathscr{L}_{m_1} + F \cdot (\mathscr{L}_{m_2} - \mathscr{L}_{m_3})$;
FOR i = 1 to 2
if RAND $< \pi$, $o_{p,i} = \widetilde{o}_{p,i}$ else $o_{p,i} = x_{p,i}$, end;
END;
FOR i = 1 to length($\mathscr{L}$)
if RAND $< \pi$, $\mathscr{O}_{p,i} = \widetilde{\mathscr{O}}_{p,i}$ else $\mathscr{O}_{p,i} = \mathscr{L}_{p,i}$, end;
END;
compute objective value $f(\mathbf{o}_p, \mathscr{O}_p)$
END

replace p-th current solution if offspring p is fitter:
FOR p = 1 to PopulationSize
if $f(\mathbf{o}_p, \mathscr{O}_p) < f(\mathbf{x}_p, \mathscr{L}_p)$
replace $\mathbf{x}_p$ with $\mathbf{o}_p$ and $\mathscr{L}_p$ with $\mathscr{O}_p$;
end;
END
END

set to $50\,000/P$ and $100\,000/P$, respectively, so that in all three of the methods, the same number of candidate solutions was generated.

4.3 Results for the Lynx Data Set

Based on a total of 2 645 runs, Table 5 reports individually optimized parameters and lag sets for the three objective functions under consideration.[9] Apparently, the allowance for alternative lag structures has a positive effect on model quality, as, compared to the benchmark result (indicated by "Ref"), a lower standard deviation of residuals and lower information criteria, respectively, have been achieved.

Table 5: Optimal sets and parameters for the Lynx data set; optimized values: (a) γ, c, (b) $\gamma, c, \mathscr{L}_2$, (b) $\gamma, c, \mathscr{L}_1, \mathscr{L}_2$

objective	criterion	γ^*	c^*	$\mathscr{L}_1^*$	$\mathscr{L}_2^*$
σ_ϵ					
Ref	0.187	1.73	2.73	1	2,3,4,9,11
σ_e					
(a)	0.18692448	1.75809577	2.72687199	1	2,3,4,9,11
(b)	0.17705925	2.33695618	2.75052253	1	1,2,3,4,5,6,10,11
(c)	0.17146943	2.05241257	2.82026793	1,2,5,6,7,8,10	1,2,3,4,5,6,7,8,10,11
AIC					
(a)	-3.27961345	1.75809577	2.72687199	1	2,3,4,9,11
(b)	-3.38222454	2.11907745	2.75224411	1	2,3,4,10,11
(c)	-3.40979719	2.15868458	2.6688362	1,2	1,2,3,4,10,11
SBC					
(a)	-3.0749743	1.75809577	2.72687199	1	2,3,4,9,11
(b)	-3.1775854	2.11907745	2.75224411	1	2,3,4,10,11
(c)	-3.1775854	2.11907737	2.75224412	1	2,3,4,10,11

However, both, the $\hat{\sigma}_e$ as well as the AIC criterion estimates eventually suggest less parsimonious models: In particular the residuals' standard deviation encourages the inclusion of a higher number of lags. The SBC, on the other hand, could be improved by including the 10th instead of the 9th lag; apart from this (and, of course, slightly different values for γ^* and c^*), the result is equal to the benchmark result.

4.4 Results for the Unemployment Data Set

van Dijk et al. (2002) illustrate the STAR modeling cycle by means of an exemplary analysis of monthly US unemployment rates (UER) for the period 1968 : 06 – 1989 : 12. Apart from employing a functional relationship according to equation (2), their general approach to this modeling task resembles the requirements already known

[9] As both algorithms appear to select equal solutions, we did not tabulate optimized sets subject to the respective heuristics. However, these results are available upon request.

from the Lynx data set. However, the complexity of the optimization set increases dramatically, as the large persistence of the series demands for an increased lag order whereas its noticeable seasonal pattern also demands for an additional inclusion of seasonal dummy variables. Starting with an unrestricted model for the first difference of the series (Δy) with 15 lagged first differences in both regimes, their multiple-stage approach of gradually removing all lagged first differences with an absolute value of the corresponding t-statistic less than or equal to 1 finally ends up in the specification given by equation (7). $\Delta_{12} y_{t-1}$ corresponds to the lagged twelve-month difference of the unemployment rate which, as an business cycle indicator, serves as transition variable. Note that our benchmarks for the residual standard deviation as well as for both information criteria differ from the reported values.[10]

$$\begin{aligned} \Delta y_t = \alpha_0 + \sum_{d \in \mathscr{D}} \alpha_d \mathscr{D}_d & \\ + \left(\beta_0 y_{t-1} + \sum_{\ell \in \mathscr{L}_1} \beta_\ell \Delta y_{t-\ell} \right) \cdot (1 - G(\gamma, c, \Delta_{12} y_{t-1})) & \\ + \left(\delta_0 y_{t-1} + \sum_{\ell \in \mathscr{L}_2} \delta_\ell \Delta y_{t-\ell} \right) \cdot G(\gamma, c, \Delta_{12} y_{t-1})) + e_t & \end{aligned} \tag{7}$$

$$G(\gamma, c, \Delta_{12} y_{t-1}) = \left(1 + \exp\left(\frac{-\gamma}{\hat{\sigma}_{\Delta_{12} y_{t-1}}} \cdot (\Delta_{12} y_{t-1} - c) \right) \right)^{-1}, \tag{8}$$

with $\mathscr{L}_1 = \{1, 6, 8, 10, 13, 15\}$ and $\mathscr{L}_2 = \{1, 2, 7, 13, 14, 15\}$ indicating the sets of included lags, $\mathscr{D} = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$ denoting the set of seasonal dummies and α_0 marking a constant.

Much akin to the Lynx data set, our simulation exercises addressed estimation problems with rising degrees of complexity as

(a) only the parameters γ and c;

(b) in addition the set $\mathscr{L}_2$;

(c) in addition the set $\mathscr{L}_1$; and

(d) in addition the set of dummy variables, $\mathscr{D}$,

were optimized. In accordance to the original approach of van Dijk et al. (2002), a maximum of 15 lags was considered in specifying $\mathscr{L}_1$ and $\mathscr{L}_2$. As each specification was restricted to contain a constant, a maximum of 11 seasonal dummy variables was taken into account for the specification of $\mathscr{D}$. Again, the threshold (TA) and cooling

[10] Running the original code (available at http://swopec.hhs.se/hastef/abs/hastef0380.htm) under OxGauss with m@ximize we recomputed benchmark values of $\hat{\sigma}_e = 0.1959486$, $AIC = -3.1541526$ and $SBC = -2.7625808$. The benchmarks for the set (γ^*, c^*), however, reproduced the published results and have been estimated as $(23.154240, 0.273687)$.

(SA) sequences were determined by preliminary simulations and the number of iterations was set to 10 000, 25 000, 50 000, and 100 000, respectively, with regards to the complexity of the optimization problem. The probability of modifying either the parameters or a set was again 0.5; if more than one set was to be optimized, the conditional probabilities were an even 0.5 and 1/3, respectively. For the DE heuristic, the scaling factor and the cross-over probability were not tuned to the considered optimization problem but set to the values suggested by the literature. The population size was chosen to be 30 plus two times the maximum number of lags to choose from; the number of iterations was again set such that the number of candidate solutions equals those in the TA and SA heuristics.

Optimized parameter values and lag structures for the different models are given by Table 6. Again, different objective functions lead to different specifications. Under the AIC, the benchmark result is confirmed. When the adjusted residuals' standard deviation is to be minimized, then the model becomes slightly less parsimonious than the benchmark with one additional lag in $\mathscr{L}_1^*$ (compare models (a) and (d)). The SBC, however, prefers more parsimonious models and reduces the number of lags in $\mathscr{L}_2^*$ and doing completely without lagged variables in the other block with $\mathscr{L}_1^* = \emptyset$.

4.5 Comparison of Models

As a matter of course, an appraisal of the heuristics' performance can not be accomplished without a closer inspection of the characteristics of the finally estimated models. Hence, this subsection aims at reporting some "common sense" diagnostics. As elementary diagnostic statistics, we present a subset of the battery of tests presented by van Dijk et al. (2002) in their results section. These measures are complemented by a basic visual inspection of the dynamical properties of the selected specifications.

The rows of Table 7 summarize residual based statistics for the specifications given by Tables 5 (case (c)) and 6 (case (d)). Apart from replicating the computed values for σ_e, AIC and SBC (rows two to four), we also added the models' $\bar{R}^2$-statistics as additional information about the goodness of fit. Considering misspecification tests, the following statistics are reported: The Lomnicki-Jarque-Bera test (LJB), residuals' skewness (SK) and excess kurtosis (EK), the results of LM tests for the absence of residual autocorrelation up to orders two (LM(2)) and twelve (LM(12)) as well as the results of ARCH tests up to order one (ARCH(1)) and four (ARCH(4)).[11]

Compared to the reference cases (columns two and six), we might state at least, that the specifications found by our heuristics seem to come off equally well.

In general, the graphical inspection of the optimized specifications discovers strong similarities in their respective dynamical patterns as well as with regards to

[11] Please note that, in order to achieve the comparability of measures between our results and those of van Dijk et al. (2002), all residual correlation tests have been based on auxiliary regressions which omitted the estimates of γ and c in the underlying gradient. See Eitrheim and Teräsvirta (1996) for the computational aspects of this test.

Table 6: Optimal sets and parameters for the UER data set

objective	criterion	γ^*	c^*	$\mathscr{D}^*$	$\mathscr{L}_1^*$	$\mathscr{L}_2^*$
Ref (AIC)	-3.154153	23.1542	0.2737	1, 2, 3, 4, 5, 6, 7, 8, 9, 10	1, 6, 8, 10, 13, 15	1, 2, 7, 13, 14, 15
σ_e						
(a)	0.195949	23.1549	0.273692	1, 2, 3, 4, 5, 6, 7, 8, 9, 10	1, 6, 8, 10, 13, 15	1, 2, 7, 13, 14, 15
(b)	0.195922	20.7199	0.275594	1, 2, 3, 4, 5, 6, 7, 8, 9, 10	1, 6, 8, 10, 13, 15	1, 2, 3, 4, 7, 13, 14, 15
(c)	0.195884	22.7009	0.276446	1, 2, 3, 4, 5, 6, 7, 8, 9, 10	1, 5, 6, 8, 10, 13, 15	1, 2, 7, 13, 14, 15
(d)	0.195884	22.7009	0.276447	1, 2, 3, 4, 5, 6, 7, 8, 9, 10	1, 5, 6, 8, 10, 13, 15	1, 2, 7, 13, 14, 15
AIC						
(a)	-3.154153	23.1549	0.273692	1, 2, 3, 4, 5, 6, 7, 8, 9, 10	1, 6, 8, 10, 13, 15	1, 2, 7, 13, 14, 15
(b)	-3.154153	23.1549	0.273692	1, 2, 3, 4, 5, 6, 7, 8, 9, 10	1, 6, 8, 10, 13, 15	1, 2, 7, 13, 14, 15
(c)	-3.154153	23.1549	0.273692	1, 2, 3, 4, 5, 6, 7, 8, 9, 10	1, 6, 8, 10, 13, 15	1, 2, 7, 13, 14, 15
(d)	-3.154153	23.1549	0.273692	1, 2, 3, 4, 5, 6, 7, 8, 9, 10	1, 6, 8, 10, 13, 15	1, 2, 7, 13, 14, 15
SBC						
(a)	-2.762581	23.1549	0.273692	1, 2, 3, 4, 5, 6, 7, 8, 9, 10	1, 6, 8, 10, 13, 15	1, 2, 7, 13, 14, 15
(b)	-2.786022	1152.23	0.267624	1, 2, 3, 4, 5, 6, 7, 8, 9, 10	1, 6, 8, 10, 13, 15	1, 2, 14, 15
(c)	-2.846709	608.38	0.621962	1, 2, 3, 4, 5, 6, 7, 8, 9, 10	8	2, 7, 14, 15
(d)	-2.859269	3452.11	0.623606	1, 2, 3, 4, 5, 6, 7, 8, 9, 10	–	2, 7, 14, 15

Table 7: Diagnostic Statistics of optimized models

	Lynx data series objective				UER data series objective			
	Ref (σ_e)	σ_e	AIC	SBC	Ref (AIC)	σ_e	AIC	SBC
σ_e	0.187	0.171	0.174	0.178	0.196	0.196	0.196	0.201
AIC	-3.280	-3.362	-3.410	-3.382	-3.154	-3.151	-3.154	-3.135
SBC	-3.075	-2.876	-3.154	-3.178	-2.763	-2.745	-2.763	-2.859
$\bar{R}^2$	0.890	0.907	0.905	0.900	0.816	0.816	0.816	0.806
	Misspecification Tests: Statistics with corresponding p-values							
LJB	0.138	0.059	0.397	0.161	19.460	17.881	19.460	35.546
p-value	0.933	0.971	0.820	0.923	0.000	0.000	0.000	0.000
SK	-0.020	-0.003	-0.046	-0.094	0.638	0.610	0.638	0.662
EK	-0.175	0.117	-0.290	-0.048	0.564	0.548	0.564	1.342
LM(2)	0.421	0.436	0.428	0.436	0.958	1.262	0.958	0.093
p-value	0.658	0.648	0.653	0.648	0.385	0.285	0.385	0.911
LM(12)	0.403	0.680	0.562	0.853	0.475	0.654	0.475	0.456
p-value	0.958	0.764	0.865	0.597	0.927	0.793	0.927	0.938
ARCH(1)	22.232	12.704	26.844	32.152	0.863	0.965	0.863	1.181
p-value	0.000	0.000	0.000	0.000	0.353	0.326	0.353	0.277
ARCH(4)	23.406	16.160	28.830	37.646	1.319	1.485	1.319	1.870
p-value	0.000	0.003	0.000	0.000	0.858	0.829	0.858	0.760

the reported benchmark models. Only the SBC-optimized model of the unemployment series makes a minor exception to this overall picture. Figure 1 visualizes fitted values (dashed line in panel (a)), a skeleton plot (i.e., a deterministic simulation with historical start values, solid line of panel (b)) and the estimated transition function of this model. Judged by the skeleton plot, this specification seems to establish a stable dynamical system with dominant seasonal pattern which does not differ much from the reference case (see Figure 8 in the appendix). Yet, the estimated transition function indicates that a threshold model might also be successfully fitted to this series: The time series plot of the transition function seems to jump between its extrem values (panel (d)). Analogously, the scatter plot of the transition function against the transition variable nearly depicts a two-regime indicator function. Nevertheless, this model basically seems to mirror the series' dynamics in a satisfactory manner.

As the corresponding figures of the remaining optimized models do not indicate any serious departures from their reference cases (see the appendix) we might therefore assess that our simulation exercises ended up with stable specifications which turn out to give close approximations to the dynamical properties of the analyzed series.

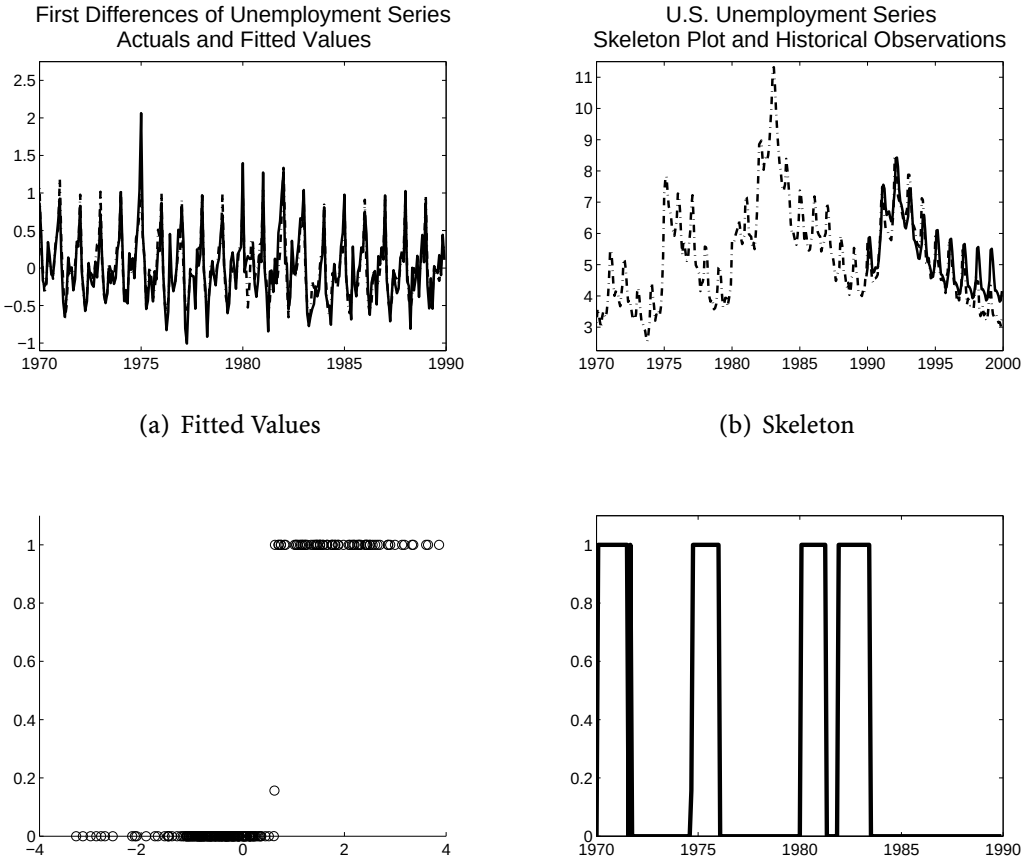
4.6 Comparison of Methods

The comparison of the heuristical optimized models with their benchmarks illustrated that TA, SA, and DE are well apt to tackle the STAR model selection and parameter estimation problem. For given lag structures all of the three heuristics under investigation identify the benchmark optimal values γ^* and c^* when being conceded the same number of candidate solution that would be needed for a traditional grid search. In particular DE for the Lynx data set and TA and DE for the UER data set have in addition a rather low standard deviation in the reported results; repeated runs therefore are likely to produce (more or less) equally good results.

When in addition the lag structures and, where applicable, the ideal number of dummies have to be identified, SA and TA tend to find slightly different results in repeated runs – which illustrates both the weakness and the strength of heuristic approaches. Like traditional methods, heuristics, too, can get trapped in local optima; however, unlike traditional methods, they are non-deterministic, and repeated runs might lead to different results since they include mechanisms that can overcome local optima and that (theoretically) will let them eventually converge to the global optimum. Hence, they are able to identify solutions that are at least as good as those found by traditional deterministic approaches (i.e., the benchmark results) or even better. However, it must be pointed out that these results, too, are not necessarily the global ones: in the UER data set, e.g., the optimal result found for model (d) could also have been identified under (c).

With respect to the individual convergence rates of our proposed algorithms, the population based DE heuristic excels at our experiments. The box plots of Figure 2

Figure 1: Suggested Model Specification, Objective: SBC



(a) Fitted Values

(b) Skeleton

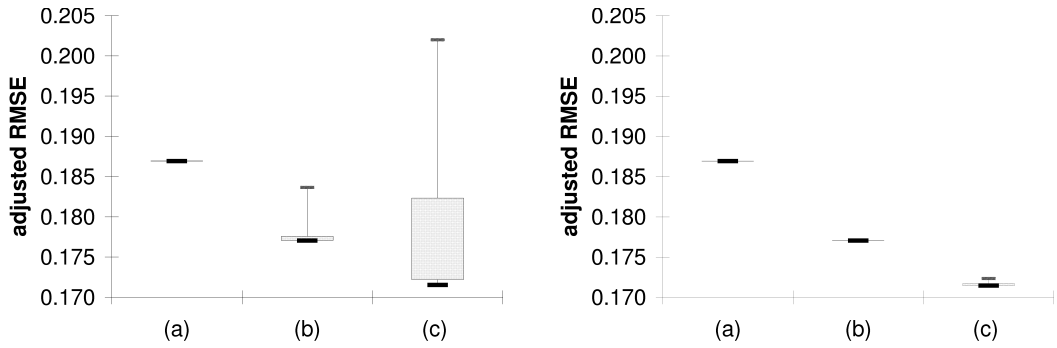
(c) Transition Function vs Trans. Variable

(d) Transition Function in Time

Source: Own calculations

visualize the distribution of results in our Lynx exercises for the TA as well as for the DE heuristic.

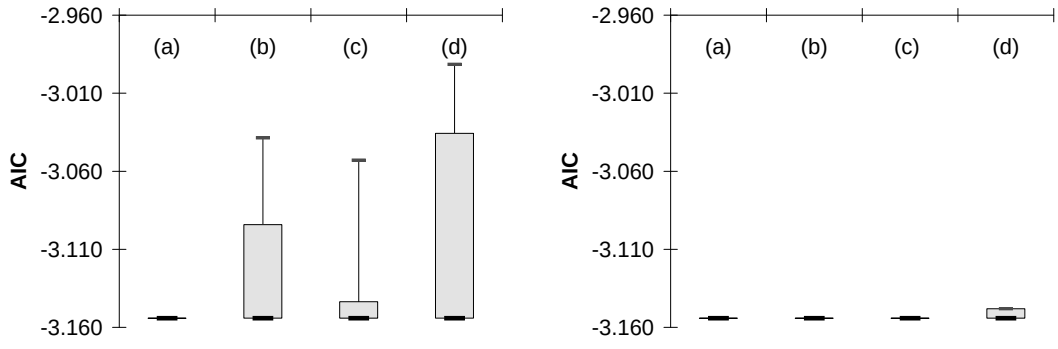
Figure 2: Distribution of Results: Lynx Series



(a) Threshold Accepting (b) Differential Evolution

The single agent approach obviously suffers from rising standard deviations of results for rising degrees of complexity whereas the population based DE algorithm seems to converge with almost constant rate irrespective of the degree of complexity. See the appendix (Table 8) for a detailed statistical summary of the observed results. For a total of 3 031 runs, the outcomes of the more complex unemployment exercises have been summarized by Tables 9.a and 9.b in the appendix. Obviously, the superior performance of the DE heuristic is confirmed by this study. See Figure 3 for box plots of the distribution of results.

Figure 3: Distribution of Results: Unemployment Series



(a) Threshold Accepting (b) Differential Evolution

Neither of the heuristic approaches suggested in this contribution requires substantially more implementation skills than an automated model selection based on

traditional deterministic approaches. TA and SA, however, require some parameter tuning which, as demonstrated, can be done by following some simple rules; for DE, high quality results could be achieved without any parameter tuning and simply using typical values.

5 Conclusion

This paper aimed at pointing out that optimization heuristics might prove beneficial on the specification stage of the STAR modeling cycle. Regarding our findings, we think that we did come up with a clear indication of the general capacities of our proposed heuristics: The results of simulated optimization tasks with different degrees of complexity indicated a reliable convergence behavior for each of the suggested heuristics. Heuristic methods therefore should be considered a reasonable approach for the STAR model selection problem, in particular when the number of models to choose from is rather big.

With regard to computational time, implementations were chosen where the number of candidate solutions did not exceed the one typically needed for a deterministic approach. With the computationally most expensive part of the algorithms being the evaluation of candidate solutions, the heuristic methods required roughly the same CPU time as their deterministic counterparts. Ideally, heuristic methods should be given repeated runs which linearly increases the run time. At the same time, CPU time could be more efficiently utilized by not having a fixed number of iterations but an additional criterion that interrupts and restarts the heuristic when the current run has seemingly converged (e.g., when no further improvement was achieved over a certain number of iterations). Deterministic methods, on the other hand, often do require substantially more computational time when the precision is to be increased: a finer grid for the grid search, a more investigative lag selection method and an increase in the number of lags (or dummies) to choose from frequently increase the necessary CPU time exponentially.

References

H. M. Anderson. Choosing lag lengths in nonlinear dynamic models. Monash Econometrics and Business Statistics Working Papers 21/02, Monash University, Department of Econometrics and Business Statistics, Dec. 2002.

D. Bacon and D. Watts. Estimating the transition between two intersecting straight lines. *Biometrika*, 58:525–534, 1971.

R. Baragona, F. Battaglia, and D. Cucina. Fitting piecewise linear threshold autoregressive models by means of genetic algorithms. *Computational Statistics & Data Analysis*, 47:277–295, 2004.

M. Camacho. Vector smooth transition regression models for US GDP and the composite index of leading indicators. *Journal of Forecasting*, 23(3):173–196, 2004.

F. Chan and M. McAleer. Maximum likelihood estimation of star and star-garch models: Theory and monte carlo evidence. *Journal of Applied Econometrics*, 17(5):509–534, 2002.

F. Chan and M. McAleer. Estimating smooth transition autoregressive models with garch errors in the presence of extreme observations and outliers. *Applied Financial Economics*, 13, 2003.

Y.-T. Chen. Discriminating between competing STAR models. *Economics Letters*, 79: 161–167, 2003.

G. Dueck and T. Scheuer. Threshold accepting: A general purpose algorithm appearing superior to simulated annealing. *Journal of Computational Physics*, 90:161–175, 1990.

Ø. Eitrheim and T. Teräsvirta. Testing the adequacy of smooth transition autoregressive models. *Journal of Econometrics*, 74:59–76, 1996.

A. Escribano and O. Jordá. Improved testing and specification of smooth transition regression models. In P. Rothman, editor, *Nonlinear Time Series Analysis of Economic and Financial Data*, pages 289–320. Kluwer Academic Publishers, Norwell, 1999.

P. Franses and D. van Dijk. *Nonlinear Time Series Models in Empirical Finance*. Cambridge University Press, Cambridge, 2000.

J. Hamilton. A new approach to the economic analysis of nonstationary time series and the business cycle. *Econometrica*, 57(2):357–384, 1989.

S. Kirkpatrick, C. Gelatt, and M. Vecchi. Optimization by simulated annealing. *Science*, 220(4598):671–680, 1983.

S. Leybourne, P. Newbold, and D. Vougas. Unit Roots and Smooth Transitions. *Journal of Time Series Analysis*, 19:55–62, 1998.

H. Lütkepohl, T. Teräsvirta, and J. Wolters. Investigating stability and linearity of a German M1 money demand function. *Journal of Applied Econometrics*, (14), 1999.

D. Maringer. *Portfolio Management with Heuristic Optimization*. Springer, Dordrecht, 2005.

B. Pötscher and I. Prucha. *Dynamic Nonlinear Econometric Models – Asymptotic Theory*. Springer-Verlag, Berlin, 1997.

N. Sarantis. Nonlinearities, cyclical behaviour and predictability in stock markets: International evidence. *International Journal of Forecasting*, 17(3):459–482, 2001.

J. Skalin and T. Teräsvirta. Modelling asymmetries and moving equilibria in unemployment rates. *Macroeconomic Dynamics*, 6:202–241, 2002.

J. Skalin and T. Teräsvirta. Another look at swedish business cycles, 1861–1988. *Journal of Applied Econometrics*, 14:359–378, 1999.

R. Storn and K. Price. Differential evolution – a simple and efficient adaptive scheme for global optimization over continuous spaces. Technical report, International Computer Science Institute, Berkeley, 1995.

R. Storn and K. Price. Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces. *J. of Global Optimization*, 11(4):341–359, 1997.

T. Teräsvirta. Univariate nonlinear time series models. In K. Patterson and T. Mills, editors, *Palgrave Handbook of Econometrics*, volume I. Palgrave Macmillan, New York, 2006.

T. Teräsvirta. Specification, estimation, and evaluation of smooth transition autoregressive models. *Journal of the American Statistical Association*, 89:208–218, 1994.

T. Teräsvirta. Modelling economic relationships with smooth transition regressions. In A. Ullah and D. Giles, editors, *Handbook of Applied Economic Statistics*, pages 507–552. Marcel Dekker, New York, 1998.

T. Teräsvirta and H. Anderson. Characterizing nonlinearities in business cycles using smooth transition autoregressive models. *Journal of Applied Econometrics*, 7:S119–S136, December 1992.

H. Tong. On a threshold model. In C. Chen, editor, *Pattern Recognition and Signal Processing*, pages 101–141. Sijhoff and Noordoff, Amsterdam, 1978.

H. Tong. *Threshold Models in Non-Linear Time Series Analysis*. Springer-Verlag, New York, 1983.

H. Tong and K. Lim. Threshold autoregressive, limit cycles and data. *Journal of the Royal Statistical Society*, B 42:245–292, 1980.

D. van Dijk, T. Teräsvirta, and P. Franses. Smooth transition autoregressive models – a survey of recent developments. *Econometric Reviews*, 21(1):1–47, 2002.

J. Vesterstrøm and R. Thomsen. A comparative study of differential evolution, particle swarm optimization, and evolutionary algorithms on numerical benchmark problems. In *Congress on Evolutionary Computation, 2004 – CEC2004*, 2004.

H. White. Some asymptotic results for learning in single hidden layer feedforward network models. *Journal of the American Statistical Association*, 84:1003–1013, 1989.

P. Winker. *Optimization Heuristics in Econometrics. Applications of Threshold Accepting*. John Wiley & Sons, ltd., Chichester et al., 2001.

P. Winker and D. Maringer. Optimal lag structure selection in VEC-models. In A. Welfe, editor, *New Directions in Macromodelling*. Elsevier, 2005.

P. Winker and D. Maringer. The threshold acceptance optimisation algorithm in economics and statistics. In E. J. Kontoghiorghes and C. Gatu, editors, *Optimisation, Econometric and Financial Analysis*. Springer, 2007.

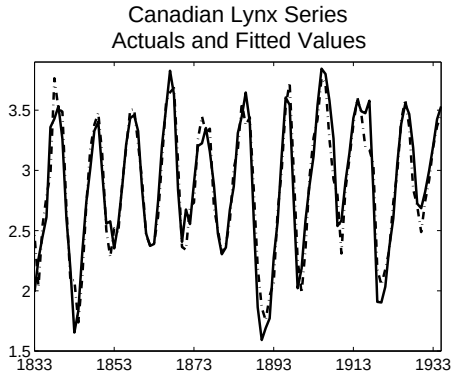
J. Wooldridge. Estimation and inference for dependent processes. In R. Engle and D. McFadden, editors, *Handbook of Econometrics*, volume IV. Elsevier Science, Amsterdam, 1994.

B. Wu and C.-L. Chang. Using Genetic Algorithms to Parameters (d,r) Estimation for Threshold Autoregressive Models. *Computational Statistics & Data Analysis*, 38: 315–330, 2002.

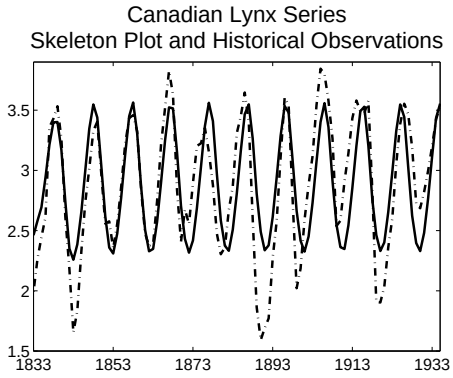
A Appendix

A.1 MC Results: Figures of Lynx Models

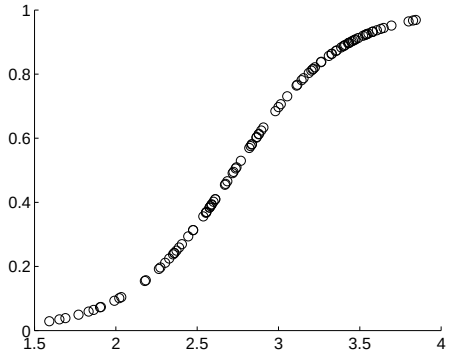
Figure 4: Suggested Model Specification: Reference Case



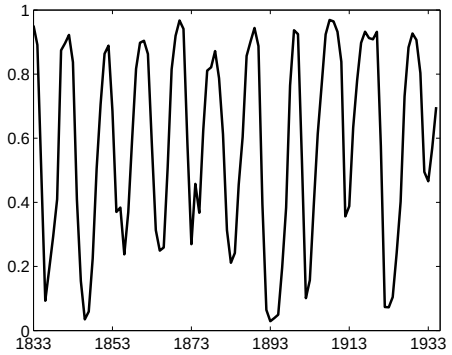
(a) Fitted Values



(b) Skeleton

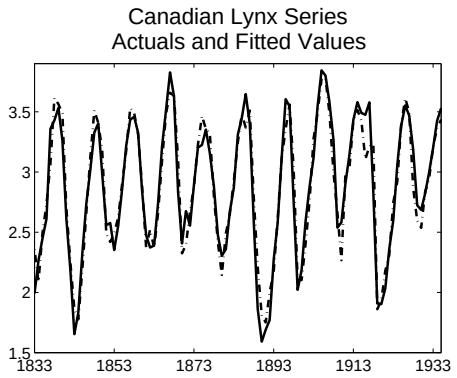


(c) Transition Function vs Trans. Variable

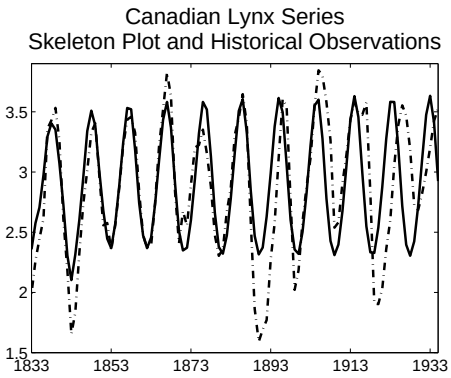


(d) Transition Function in Time

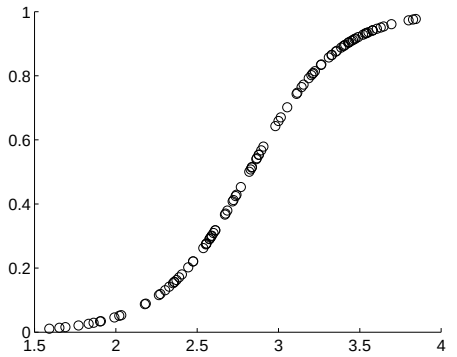
Figure 5: Suggested Model Specification, Objective: σ_e



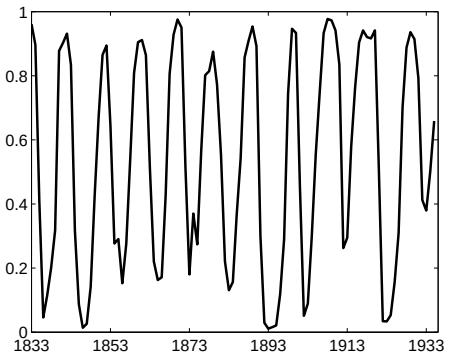
(a) Fitted Values



(b) Skeleton

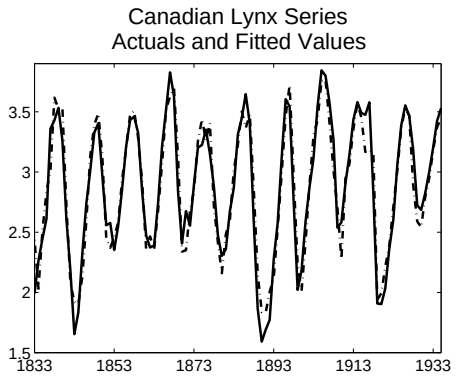


(c) Transition Function vs Trans. Variable

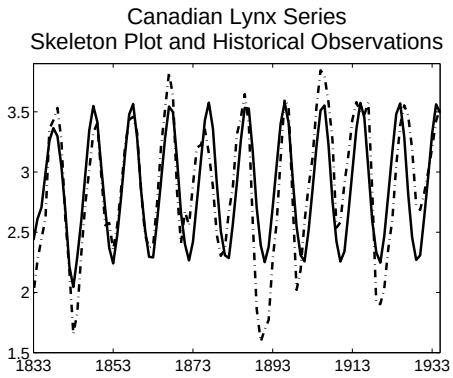


(d) Transition Function in Time

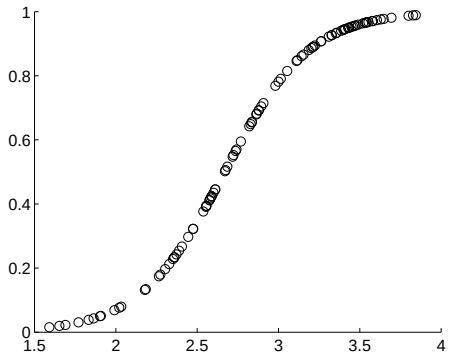
Figure 6: Suggested Model Specification, Objective: AIC



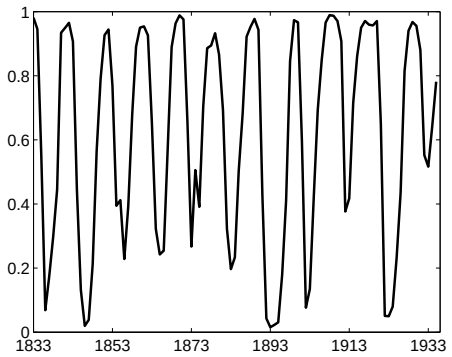
(a) Fitted Values



(b) Skeleton

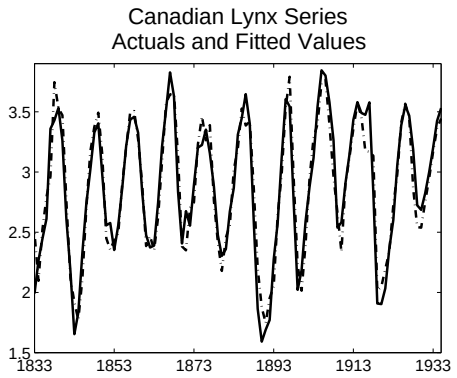


(c) Transition Function vs Trans. Variable

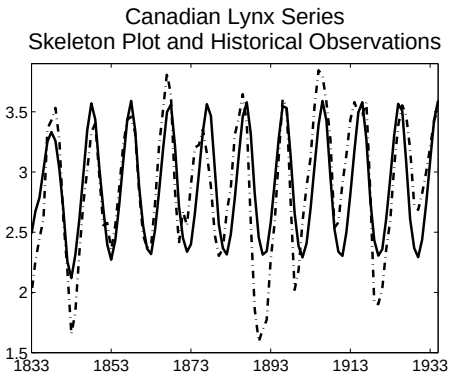


(d) Transition Function in Time

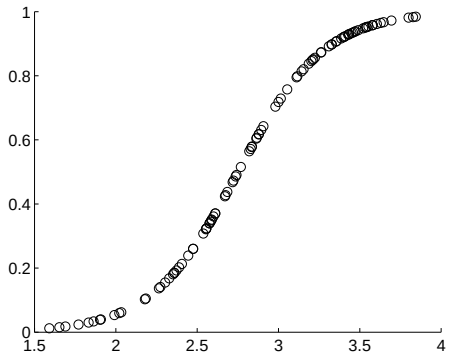
Figure 7: Suggested Model Specification, Objective: SBC



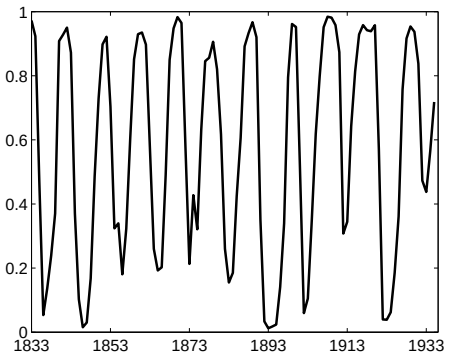
(a) Fitted Values



(b) Skeleton



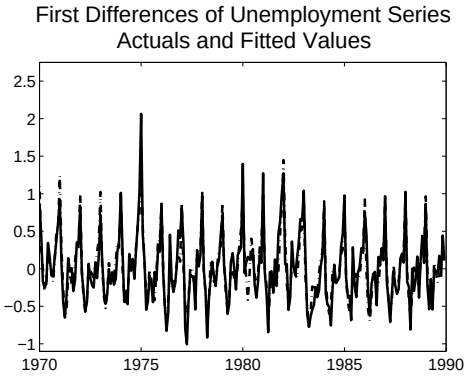
(c) Transition Function vs Trans. Variable



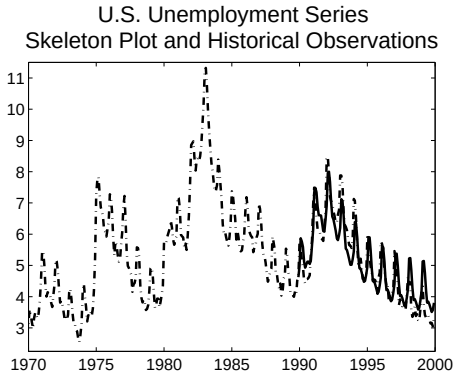
(d) Transition Function in Time

A.2 MC Results: Figures of Unemployment Models

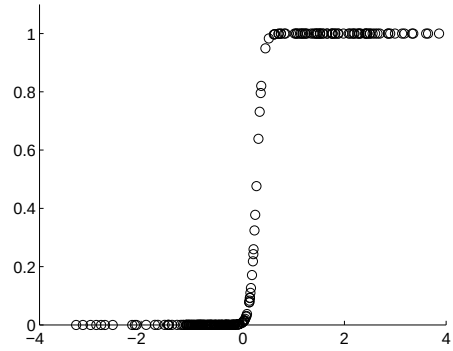
Figure 8: Suggested Model Specification: Reference Case



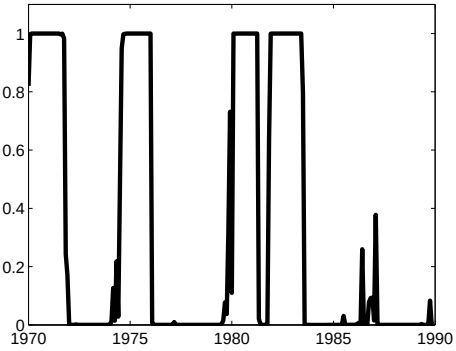
(a) Fitted Values



(b) Skeleton

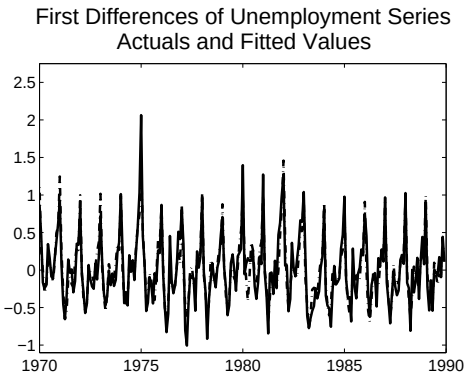


(c) Transition Function vs Trans. Variable

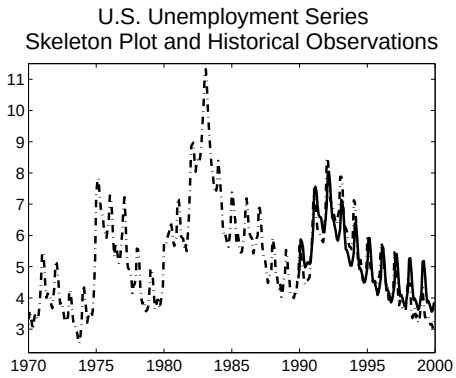


(d) Transition Function in Time

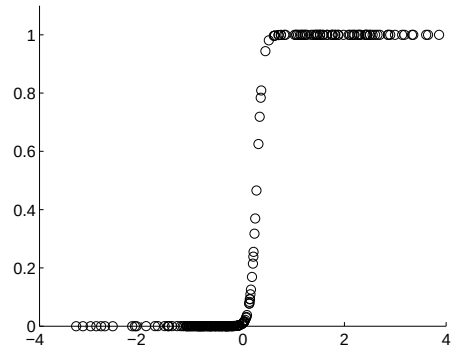
Figure 9: Suggested Model Specification, Objective: σ_e



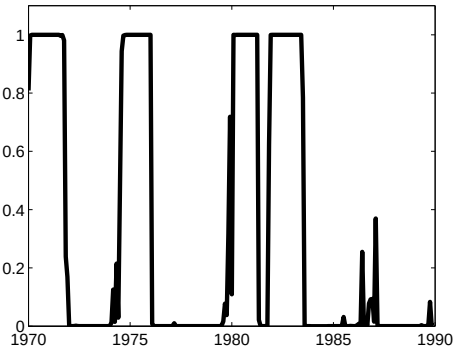
(a) Fitted Values



(b) Skeleton

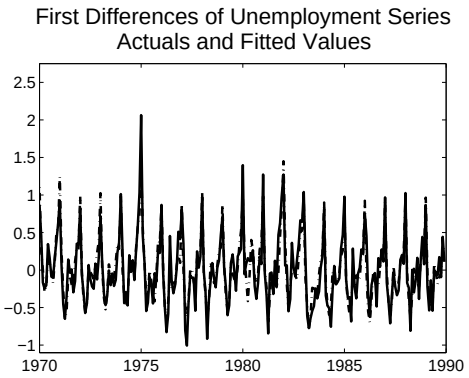


(c) Transition Function vs Trans. Variable

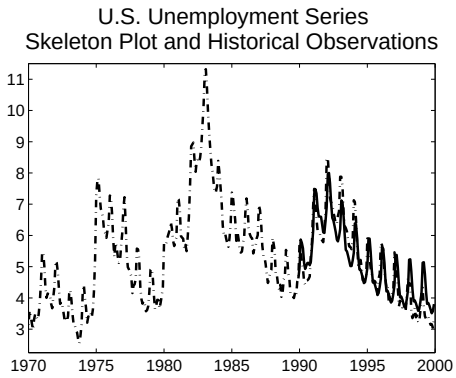


(d) Transition Function in Time

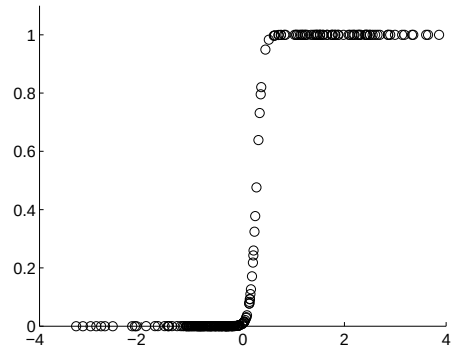
Figure 10: Suggested Model Specification, Objective: AIC



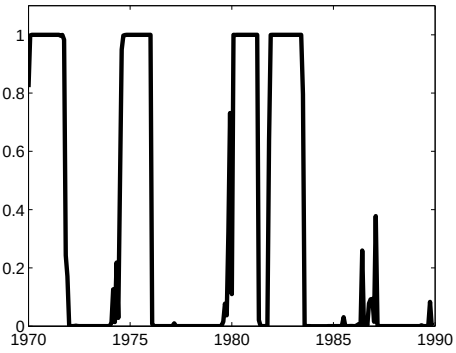
(a) Fitted Values



(b) Skeleton



(c) Transition Function vs Trans. Variable



(d) Transition Function in Time

A.3 MC Results: Tables

Table 8: Results for the Lynx data set with heuristically optimized lags from 2645 runs

(a) $\mathscr{L}_1 = \{1\}$, parameters (γ, c) and lags $\mathscr{L}_2$ optimized heuristically

criterion	σ_e			AIC			SBC		
heuristic	SA	TA	DE	SA	TA	DE	SA	TA	DE
minimum	0.177061	0.177059	0.177059	-3.382225	-3.382225	-3.382225	-3.177585	-3.177585	-3.177585
average	0.177493	0.177560	0.177059	-3.377767	-3.372027	-3.382225	-3.157380	-3.145678	-3.177585
maximum	0.177576	0.183648	0.177059	-3.361091	-3.037233	-3.382225	-3.077981	-2.858984	-3.177585
SD	0.000108	0.000707	0.000000	0.004698	0.028553	0.000000	0.023426	0.036163	0.000000
runs	156	156	129	156	156	130	156	156	129

(b) parameters (γ, c) and lags $\mathscr{L}_1$ and $\mathscr{L}_2$ optimized heuristically

criterion	σ_e			AIC			SBC		
heuristic	SA	TA	DE	SA	TA	DE	SA	TA	DE
minimum	0.172213	0.171532	0.171469	-3.409797	-3.409797	-3.409797	-3.177585	-3.177585	-3.177585
average	0.176577	0.176801	0.171561	-3.394871	-3.363007	-3.404188	-3.095555	-3.045786	-3.172527
maximum	0.199714	0.202005	0.172331	-3.374735	-3.151893	-3.393044	-2.960297	-2.910032	-3.136801
SD	0.004101	0.004703	0.000092	0.009564	0.035163	0.003361	0.043683	0.057060	0.013443
runs	156	156	129	156	155	129	156	155	129

Table 9.a: Results for the UER data set with heuristically optimized lags from 3013 runs

(a) parameters (γ, c) optimized heuristically

criterion	σ_e			AIC			SBC		
heuristic	SA	TA	DE	SA	TA	DE	SA	TA	DE
minimum	0.195949	0.195949	0.195949	-3.154153	-3.154153	-3.154153	-2.762581	-2.762581	-2.762581
average	0.196179	0.196159	0.195949	-3.151889	-3.154153	-3.154150	-2.760152	-2.762581	-2.762580
maximum	0.205093	0.205093	0.195950	-3.062929	-3.154152	-3.153977	-2.671352	-2.762579	-2.762558
SD	0.001294	0.001370	0.000000	0.013732	0.000000	0.000021	0.013963	0.000000	0.000003
runs	88	87	78	87	87	78	87	87	78

(b) parameters (γ, c) and lags $\mathscr{L}_2$ optimized heuristically

criterion	σ_e			AIC			SBC		
heuristic	SA	TA	DE	SA	TA	DE	SA	TA	DE
minimum	0.195922	0.195922	0.195922	-3.154153	-3.154153	-3.154153	-2.786015	-2.784330	-2.786022
average	0.196399	0.196271	0.195922	-3.147079	-3.148312	-3.154153	-2.779297	-2.779942	-2.784437
maximum	0.203999	0.203998	0.195922	-3.079230	-3.038599	-3.154153	-2.751932	-2.680074	-2.784330
SD	0.001605	0.001477	0.000000	0.019368	0.023858	0.000000	0.011137	0.014675	0.000412
runs	85	85	79	85	85	79	85	85	79

Table 9.b: Results for the UER data set with heuristically optimized lags from 3013 runs

(a) parameters (γ, c) and lags $\mathscr{L}_1$ and $\mathscr{L}_2$ optimized heuristically

criterion	σ_e			AIC			SBC		
heuristic	SA	TA	DE	SA	TA	DE	SA	TA	DE
minimum	0.195884	0.195884	0.195922	-3.154153	-3.154153	-3.154153	-2.846709	-2.846418	-2.786022
average	0.195898	0.196054	0.195922	-3.154054	-3.151495	-3.154153	-2.845128	-2.835390	-2.784460
maximum	0.195962	0.207666	0.195922	-3.151875	-3.052934	-3.154153	-2.841856	-2.775653	-2.784330
SD	0.000025	0.001260	0.000000	0.000371	0.011187	0.000000	0.001585	0.011007	0.000451
runs	86	86	79	86	86	78	86	86	78

(b) parameters (γ, c) and lags $\mathscr{L}_1$ and $\mathscr{L}_2$ and Dummies $\mathscr{D}$ optimized heuristically

criterion	σ_e			AIC			SBC		
heuristic	SA	TA	DE	SA	TA	DE	SA	TA	DE
minimum	0.195884	0.195884	0.195884	-3.154153	-3.154153	-3.154153	-2.846798	-2.834035	-2.859269
average	0.195906	0.197686	0.195884	-3.153999	-3.101239	-3.153308	-2.843648	-2.791522	-2.835200
maximum	0.195962	0.209444	0.195886	-3.152745	-2.991554	-3.148116	-2.827316	-2.725615	-2.812943
SD	0.000027	0.003542	0.000000	0.000379	0.043083	0.001978	0.004973	0.023425	0.012975
runs	88	88	79	88	86	79	86	85	79