INTERNATIONAL JOURNAL OF ADAPTIVE CONTROL AND SIGNAL PROCESSING
Int. J. Adapt. Control Signal Process. 2003; **00**:1–15

Estimation and prediction with ARMMAX model: a mixture of ARMAX models with common ARX part

Li He and Miroslav Kárný[1]

[1] *Adaptive System Department*
Institute Information Theory and Automation
Academy of Sciences of the Czech Republic
P. O. Box 18, 182 08 Prague, Czech Republic
Tel: (4202) 6605 2274, Fax: (4202) 6605 2268, E-mail: heli@utia.cas.cz, school@utia.cas.cz

SUMMARY

Bayesian parameter estimation and prediction of a linear-in-parameters model with colored noise is addressed. It is based on a novel mixture model called ARMMAX.

ARMMAX is a finite mixture with its ARMAX components having a common ARX part. It assumes that the common ARX part describes a fixed deterministic input-output relationship and allows for varying characteristics of the driving colored noise. ARMMAX model with fixed MA parts is estimated by a specific version of recursive Quasi-Bayes (ARMMAX-QB) algorithm. It rests on a classical Bayesian solution that requires no restrictions on MA part allowing it to be even at stability boundary.

For on line use, ARMMAX model offers flexibility with respect to varying characteristics of the model noise. The gained flexibility is paid by a slight increase of the computational burden comparing to single ARMAX with known MA part, which is, in this respect, close to recursive least squares.

For off line use, ARMMAX model offers the possibility to estimate unknown MA parts in a novel way. Exploiting the natural parallelism of ARMMAX model, robust, derivative free multi-directional search (MDS) is selected to deal with extensive data sets for which universal optimization tools are too cumbersome.

The paper motivates the model, describes algorithmic ingredients and illustrates the resulting algorithm on a simple example. Copyright © 2003 John Wiley & Sons, Ltd.

KEY WORDS: Bayesian estimation, ARMAX model, finite mixtures, multi-directional search.

1. INTRODUCTION

ARMAX model – auto-regression (AR) with moving average noise (MA) and external input (X) – is commonly adopted when describing linear-in-parameters stochastic systems driven by a colored noise. It is equivalent to the linear state-space model that forms the corner stone of modern estimation and control theory [1]. Estimation of its parameters is a hard, repeatedly addressed, problem. The difficulty stems from the lack of sufficient statistic with its dimension smaller than the number of data.

Contract/grant sponsor: Contract/grant sponsor: Project ProDaCTool IST-99-12058, AVČR S1075102

Approximate minimization of sample variance of prediction error (PE) method has become a golden standard among the multitude of estimation variants [1]. It is, however, oriented to point estimation of all model parameters. Consequently, only asymptotic information on precision of estimates is available. It implies that other tasks like structure estimation are weakly supported. Moreover, PE is restricted on ARMAX models with strictly stable MA part and faces a range of problems when this assumption is (almost) violated.

In 1981, Peterka [2] relaxed the stability restriction on MA part and provided real-time Bayesian estimation of the ARX part when the MA part is fixed. Essentially, he shown that LD factorization of the known correlation matrix acts as an optimal, time-varying, pre-whitening filter on the observed data. The filtered data are then used in standard Bayesian estimation of the ARX part. Consequently, its uncertainties are under the control, Bayesian structure estimation of the ARX part can be used [3], etc.

Unfortunately, the assumption that MA part is known is rarely met in practice. A Bayesian comparison of hypothesis was proposed for relaxing it in [4]. It evaluates posterior probabilities on hypotheses that a specific MA part is the best one in a finite set of candidates. No guide is, however, given how to select this set. Moreover, the posterior probabilities converge to a zero-one vector in generic case, see Proposition 3.2 below. Consequently, quality of the original choice of candidates is out of objective control.

The present paper prolongs this line and addresses Bayesian parameter estimation of the ARMAX model with an unknown MA term. It uses a novel finite mixture of ARMAX models with a common ARX part but different fixed MA parts. Quasi-Bayes estimation [5] is tailored to the introduced ARMMAX model. The resulting computational burden is well comparable to that is needed for estimation of a single ARMAX model. The estimation provides a quantitative measure of the descriptive quality of individual ARMAX models forming the mixture components. Thus, several ARMAX models are estimated in parallel. This parallelism offers a chance to employ derivative-free optimization procedures for generating new, hopefully better, MA candidates. We have selected *multi-directional search* (MDS) method [6] as such a generator. Its simplicity, robustness and guaranteed convergence properties have driven our choice.

The introduced ARMMAX model is richer than ARMAX model as it admits temporal variations of the stochastic MA part. Consequently, a valuable model is obtained even if the search in the MA space is stopped before converging to a single ARMAX model. Stopping may be enforced by a slow terminal convergence of the MDS method or by computational demands implied by the extensive data set processed. The latter case restricts substantially suitability of universal optimization algorithms to the addressed problem.

Bayesian estimation of ARMAX with a known MA part and the adopted multi-directional search are recalled in Section 2. They are used throughout the paper and help us to formulate the addressed problem in detail. In Section 3, ARMMAX model is introduced and studied. It supports our claim that ARMMAX provides a parallel environment that makes the practical use of the MDS procedure realistic on single processor machine. A special version of recursive Quasi-Bayes estimation (ARMMAX-QB) tailored to ARMMAX model is presented in Section 4. Coupling of Quasi-Bayes estimation with multi-directional search is described and discussed in Section 5. The last two Sections provide illustrative example and conclusions, respectively.

2. BACKGROUND AND PROBLEM DESCRIPTION

We assume that the relationships of a multi-variate external input u_t to a scalar output y_t can be described by the ARMAX model [2]

$$y_t = \theta' \psi_t + \underbrace{\sum_{i=1}^{n} c_i e_{t-i} + e_t}_{v_t}, \tag{1}$$

where $'$ denotes transposition. The data $d_t = [y_t, u_t]$ are observed at time instances $t < \infty$. The n_ψ-dimensional regression vector ψ_t is a known function of u_t and the past data $d(t-1) = (d_1, \cdots, d_{t-1})$. The n_ψ-dimensional vector of unknown parameters θ describes the deterministic part of the model. The n-dimensional vector of parameters $C = [c_1, \cdots, c_n]$ characterizes the *colored* stationary noise v_t. It is a moving average (MA) defined on a sequence of mutually un-correlated, zero-mean, Gaussian noise $\{e_t\}$ with an unknown but constant variance r.

Throughout the paper, structure of this and other models are assumed to be known. Bayesian structure estimation [3] can directly be used to relax this restriction.

Our estimation of the model (1) with an unknown MA term relies on the estimation results obtained when the MA part is known. This makes us to recall them briefly.

2.1. *Estimation and prediction of ARMAX model with a fixed MA part*

For a given C, the covariance matrix G of the noise v_t is Toeplitz matrix with rs_i, $s_i = \sum_{k=i}^{n} c_i c_{k-i}$, $c_0 = 1$ on i-th sub- and super-diagonals, $i \leq n$.

Let us consider LD decomposition of the covariance $G = rLDL'$, where L is a band, lower triangular matrix with unit diagonal and $D = \mathrm{diag}[D_1, D_2, \ldots, D_t]$ is a diagonal matrix with positive diagonal entries D_τ, $\tau = 1, \ldots, t$. The factors L, D are functions of the MA parameters only. Their entries can be evaluated recursively as follows

$$D_1 = s_0, \quad L_{1,2} = s_1 D_1^{-1}, \quad D_2 = s_0 - L_{1,2}^2 D_1$$

and for $\tau = 3, 4, .., i = n, n-1, ..., 1$, with $n_\tau = \min(n, \tau)$,

$$L_{i,\tau} \;=\; \left(s_i - \sum_{k=i+1}^{n_\tau} L_{k,\tau} D_{\tau-k} L_{k-i,\tau-i} \right) D_{\tau-i}^{-1}, \;\; D_\tau = s_0 - \sum_{k=i+1}^{n_\tau} L_{k,\tau} D_{\tau-k} L_{k,\tau}. \tag{2}$$

The recursive evaluation requires to store $n+1$ numbers.

Using this decomposition, Peterka [2] proved the following Proposition.

Proposition 2.1 (Relationship of ARMAX model with a known MA part to ARX model) *Using the filter (2), the* probability density function *(pdf) of the normal ARMAX model (1)* *equals to the pdf describing* ARX *model defined on filtered data, marked by* ˜,

$$
\begin{aligned}
f(y_t|u_t, d(t-1), \Theta_a, C) &= \mathcal{N}_{y_t}(\mu_t, rD_t) = (2\pi rD_t)^{-0.5} \exp\left[-\frac{(y_t-\mu_t)^2}{2rD_t}\right] \qquad (3)\\
\mu_t &= \theta'\tilde{\psi}_t + \underbrace{\sum_{i=1}^{n_t} L_{t,i}\tilde{y}_{t-i}}_{\Delta_t}, \quad n_t = \min(n,t)\\
\tilde{y}_1 &= y_1, \quad \tilde{y}_t + \sum_{i=1}^{n_t} L_{t,i}\tilde{y}_{t-i} = y_t, \quad \tilde{\psi}_t + \sum_{i=1}^{n_t} L_{t,i}\tilde{\psi}_{t-i} = \psi_t, \quad \tilde{\psi}_1 = \psi_1.
\end{aligned}
$$

Where the filtered data vector $\tilde{\Psi} = [\tilde{y}, \tilde{\psi}']'$ *is obtained by passing the observed data vector* $\Psi = [y, \psi']'$ *through the pre-whitening filter (3) determined by the* L, D *entries (2). The* ARX *model is parameterized by the pair* $\Theta_a = [\theta, r]$ *and acts on* $\tilde{\Psi}$.

The number of flops needed for filtering per single data sample is $\mathcal{O}\left(n_\psi + n^2\right)$, *where* n_ψ *is dimension of the regression vector* ψ_t *and* n *is the order of the* MA *part.*

The following properties of the Peterka's filter are vital in the case of unknown C-parameters.

The pre-whitening by this filter does not require stability of the polynomial (in z^{-1}) $1+C(z^{-1}) = 1+c_1z^{-1}+\cdots+c_nz^{-n}$. The time-evolution of the filter provides a sort of spectral factorization of the MA part that is able to cope with both strictly unstable roots and roots at stability boundary as well. Parameters of the filter are, however, time varying even when the covariance matrix G of the noise v_t is time-invariant. The variations are data independent and driven only by the time-invariant C-parameters. The evaluation of the filter is computationally cheap but the discussed variations hinder the attempts to estimate the unknown C-parameters recursively. Moreover, they make evaluation of derivatives of the related likelihood function at least impractical.

The transformation of the ARMAX model to ARX model allows us to use effectively Bayesian estimation. It can be performed in real-time as the general functional recursion reduces to updating of fixed dimensional sufficient statistics, [2].

In estimation, we assume that the generator of the external input u_t meets *natural conditions of control* [2]. They reflect the assumption that the sole knowledge of the value u_t without the knowledge of the value y_t brings no information on the unknown parameters, say Θ, i.e.

$$
f(\Theta|u_t, d(t-1)) = f(\Theta|d(t-1)) \;\Leftrightarrow\; f(u_t|d(t-1), \Theta) = f(u_t|d(t-1)). \qquad (4)
$$

Proposition 2.2 (Bayesian estimation and prediction with ARX model) *Consider the* ARX *model parameterized by unknown parameters* $\Theta_a = [\theta, r]$. *Let (4) hold for* $\Theta = \Theta_a$ *and let us select a prior pdf on unknown parameters* Θ_a *of the* ARX *model (2) in the conjugate Gauss-inverse-Wishart form*

$$
f(\Theta_a) = GiW_{\Theta_a}(V_0, \nu_0) \equiv \frac{r^{0.5(-\nu_0+n_\psi+2)}}{\mathcal{I}(V_0,\nu_0)} \exp\left\{-\frac{1}{2r}\mathrm{tr}\left(V_0[-1,\theta']'[-1,\theta']\right)\right\} \qquad (5)
$$

V_0 *is positive definite symmetric matrix,* $\nu_0 > 0$, n_ψ *is dimension of* θ *and*

$$
\mathcal{I}(V,\nu) = \Gamma(0.5\nu)\Lambda^{-0.5\nu}|V_\psi|^{-0.5}2^{0.5\nu}(2\pi)^{0.5n_\psi}, \quad V = \begin{bmatrix} V_y & V'_{\psi y} \\ V_{\psi y} & V_\psi \end{bmatrix}, \quad \Lambda = V_y - V'_{\psi y}V_\psi^{-1}V_{\psi y},
$$

where V_y *is scalar.*

Then, the posterior pdf of Θ_a*, conditioned on the observations* $d(t)$ *and known parameters* C*, is* $GiW_{\Theta_a}(V_t, \nu_t)$ *pdf with the* extended information matrix V_t *and the* number of degrees of freedom ν_t *updated according to the recursions*

$$V_t = V_{t-1} + \tilde{\Psi}_t \tilde{\Psi}'_t, \quad \nu_t = \nu_{t-1} + 1, \quad V_0, \ \nu_0 \text{ are a priori chosen.} \tag{6}$$

The predictive pdf of y_t*, given by* $u_t, d(t-1)$*, is the Student pdf fulfilling*

$$f(y_t|u_t, d(t-1)) \propto \mathcal{I}\left(V_{t-1} + \tilde{\Psi}_t \tilde{\Psi}'_t, \nu_{t-1} + 1\right), \tag{7}$$

where $\propto$ *means that the right hand side has to be normalized to have unit integral.*

The number of flops needed per single estimation and prediction step is $\mathcal{O}\left(n_a^2\right)$, $n_a = n_\psi + 1$.

Note that the extended information matrix V_t together with the degrees of freedom ν_t form sufficient statistics for estimation of Θ_a. Their evolution is equivalent to well-known recursive least squares. The updating is often poorly conditioned and its *$L'DL$ decomposition* has to be propagated in order to avoid the induced numerical troubles [2].

Let us assign to candidates $\{C\}$ of a "true" MA part a prior pdf $f(C)$. Observations $d(t)$ correct it to the posterior pdf $f(C|d(t))$ through the Bayes rule. Under the natural conditions of control (4), with $\Theta = C$, it reads

$$f(C|d(t)) \propto \prod_{\tau=1}^{t} f(y_\tau|u_\tau, d(\tau-1), C) f(C) \equiv \mathcal{L}(d(t), C) f(C). \tag{8}$$

For any fixed C, the value of the introduced *likelihood function* $\mathcal{L}(d(t), C)$ is simply a product of values of the predictive pdfs (7). Although the posterior pdf formally can be used for estimating the unknown C, the complex nature of $\mathcal{L}(d(t), C)$ makes its general analytical treatment difficult. Restricting the competing C's to a finite set, Peterka[4] made the formula (8) applicable where the posterior pdf $f(C|d(t))$ serves for selecting the most promising candidates among them. No rule is, however, given how to select a suitable finite set of competitors. Numerical maximization procedures offer themselves for generating the most interesting competitors around the maximum of the posterior pdf.

2.2. Possibilities of generating MA *candidates*

For presentation simplicity, we restrict ourselves to uniform prior pdf on C and search for the candidates in a neighborhood of

$$\operatorname{Arg}\max_{\{C\}} f(C|d(t)) = \operatorname{Arg}\max_{\{C\}} \mathcal{L}(d(t), C). \tag{9}$$

The maximization problem (9) amounts to nonlinear programming and numerical analysis. The adopted Peterka's filter allows us to consider it as unconstrained problem.

Since an efficient evaluation of the gradient is inhibited by the complex nature of the correctly evaluated likelihood $\mathcal{L}(d(t), C)$, we have to restrict ourselves to *derivative-free search* methods. Meanwhile, the optimized likelihood function is multi-modal in generic case. It exhibits both flat and sharp modes. Thus, we can rely at most on continuous differentiability of $\mathcal{L}(d(t), C)$ with respect to C. Moreover, the available supply of ready methods are further narrowed down to respect the fact that each evaluation of the objective function $\mathcal{L}(d(t), C)$ is relatively costly, especially for extensive data sets. The evaluation corresponds with a run of least-squares estimation.

These considerations have reduced the set of options more or less to direct search methods. Despite of their wide use in practice, the direct search methods have been often perceived as completely "heuristic", since they are often plagued with a weak convergence analysis and some other troubles [7]. Nelder-Mead (NM simplex) algorithm has been the most popular among them. Recently, a progress has been reported in [8] that proves convergence and robustness of NM in one-dimensional case, some general properties can also be proven in higher dimensions but convergence is not guaranteed there.

A variant of the simplex method, *multi-directional search* (MDS) [6], has brought a new interest in direct search methods since 1989. The following favorable properties [9] are offered by MDS:

- the method is derivative-free;
- strong convergence properties are guaranteed;
- "noisy" evaluations of function values do not spoil the search: the method is robust;
- method can be executed with a high degree of parallelism.

They turned our attention to the MDS algorithm as generator of candidates of the MA part. It is expected to be efficient in high dimensions of the search space even for extensive data sets.

2.3. MDS Algorithm

Here, we outline the basic MDS algorithm searching for $\min_x g(x)$, where $g(\cdot)$ is continuously differentiable function of the real n-vector x. It helps us to formulate the tackled problem and to understand the specific features of its variant called MDS-ARMMAX-QB, see Section 5.

As any simplex-based method, MDS evolves $n+1$ points in n-dimensional real space forming the vertices of a *simplex*. A *non-degenerate* simplex requires the set of edges adjacent to any of its vertex forms a basis of the space so that the simplex spans the space. Although the size of the simplex is modified all the time, the shape (angle) always remains the same as that of the original one.

In each iteration, MDS searches a point strictly improving over the best vertex, but the simple decrease is used as the acceptance criteria. There are three possible operations–*reflection, expansion, contraction*–in the procedure, and they involve the n edges of the simplex emanating from the best vertex so that the entire simplex is reflected, expanded, contracted. The function values of the generated trial points are then compared with the function's values at the vertices of the current simplex. After each operation, if at least one trial vertices has a better function value than the current best vertex, the operation is called *successful.* To *accept* one operation, we replace the vertices of current simplex by the trial points after the operation.

Low demands of on prior knowledge of the maximized function and guaranteed convergence are paid by a rather slow convergence, especially, in the terminal phase of the optimization when MDS *behaves like a gradient method.*

Algorithm 2.1 (MDS algorithm)
Initial phase

- *Select an initial guess x_1^0 and generate a non-degenerate initial simplex formed by $n+1$ vertices $\langle x_1^0, \cdots, x_{n+1}^0 \rangle$.*

- *Select expansion* $\chi \in (1, \infty)$ *and contraction* $\xi \in (0, 1)$ *factors as well as the stopping rule with its parameters.*
- *Evaluate* $g(x_i^0)$*, for* $i = 1, \cdots, n+1$ *and swap labels so that* $g(x_1^0) = \arg\min_i g(x_i^0)$.

Iterative phase
Do while the stopping rule is not met
Set $j := j + 1$

1. ***Reflection***
   - *Define* n *reflected vertices* $x_i^r = 2x_1^{j-1} - x_i^{j-1}$*, for* $i = 2, \cdots, n+1$.
   - *Evaluate* $g(x_i^r)$*, for* $i = 2, \cdots, n+1$.
   - *Go to the step 2, if* $\min g(x_i^r) < g\left(x_1^{j-1}\right)$. *Otherwise, go to the step 3.*
2. ***Expansion***
   - *Define* n *expanded vertices* $x_i^e = x_1^{j-1} + \chi\left(x_1^{j-1} - x_i^{j-1}\right)$*, for* $i = 2, \cdots, n+1$.
   - *Evaluate* $g(x_i^e)$*, for* $i = 2, \cdots, n+1$.
   - *Accept the expanded simplex, if* $\min g(x_i^e) < \min g(x_i^r)$*, i.e. replace* x_i^j *by* x_i^e*, for* $i = 2, \cdots, n+1$. *Otherwise, accept the reflected simplex, i.e. replace* x_i^j *by* x_i^r*, for* $i = 2, \cdots, n+1$.
   - *Go to step 4.*
3. ***Contraction***
   - *Define* n *contracted vertices* $x_i^c = x_1^{j-1} + \xi\left(x_1^{j-1} - x_i^{j-1}\right)$*, for* $i = 2, \cdots, n+1$.
   - *Evaluate* $g(x_i^c)$*, for* $i = 2, \cdots, n+1$.
   - *Accept the expanded simplex, replace* x_i^j *by* x_i^e*, for* $i = 2, \cdots, n+1$.
4. ***Swap***
   Swap the labels so that $g\left(x_1^j\right) = \arg\min_i g\left(x_i^j\right), i = 1, \cdots, n+1$.

2.4. Addressed problem

Generally, we want to overcome practical restriction of the Bayesian comparison of hypotheses [4] while preserving its ability to use full Bayesian solution with respect to ARX part.

Practically, we search for an efficient use of the multi-directional search method to generate suitable candidates of the MA part around maximizing arguments of the likelihood $\mathcal{L}(d(t), C)$. The solution is required to be implementable even for high dimensional cases.

2.5. Idea of the solution

We introduce a special mixture called ARMMAX model. Its components are ARMAX models having a common ARX part while the different fixed MA parts, given by different C_p's, $p = 1, \ldots, n+1$, that create vertices of the simplex propagated by the MDS method.

Besides to be an interesting model itself, ARMMAX provides an "algorithmic" parallel environment for MDS. We show that likelihood values assigned to individual components can be interpreted as the approximated values of the likelihood $\mathcal{L}(d(t), C_p)$ corresponding to individual

ARMAX's with its MA part defined by respective C_p's. Thus, MDS can perform a parallel search while being implemented on a single-processor machine.

Consequently, MDS in the "algorithmic" ARMMAX parallel environment generates a convergent sequence, which ideally converges to the C-parameters of the "true" MA part. The explored points define the best vertices of the simplex and the components having the highest component log-likelihood in the corresponding ARMMAX as well.

The MDS-driven evolution of simplex corresponds with the specification of C-parameters of ARMMAX's. With them, the evaluation of the likelihood values coincides with estimation of the resulting ARMMAX model. The estimation is based on tailored Quasi-Bayes estimation [5] called ARMMAX-QB algorithm. In particular, the estimation of the common ARX part consists of a run of least squares fed by the weighted outputs of several Peterka's filters running in parallel.

It has to be stressed that the presented idea can immediately be extended to ARMMAX model with more components than the number of vertices in simplex. For time being, we prefer to rely on the guaranteed properties of the MDS method.

The details of the solution are introduced step-wise in subsequent sections.

3. ARMMAX MODEL

A finite mixture model [10] describes the observed data by a convex combination of a finite number of pdfs, called components. Mixture can be interpreted as a universal approximation of the pdfs describing observations, often, of a radial-basis-functions type. ARMMAX model is a finite mixture with the common ARX part and different MA parts in all ARMAX components. At each time instance t, the *probability density function* (pdf) of ARMMAX model is thus given as

$$f\left(y_t|u_t, d(t-1), \Theta_a, \alpha, \Theta_c\right) \quad = \quad \sum_{p=1}^{k} \alpha_p f(y_t|u_t, d(t-1), \Theta_a, C_p), \quad k < \infty \tag{10}$$

with the component weights $\alpha = [\alpha_1, \cdots, \alpha_k]$ satisfying $\alpha_p \geq 0, \;\; p = 1, \cdots, k, \;\; \sum_{p=1}^{k} \alpha_p = 1$. All ARMAX components have the same parameters Θ_a describing the AR part and different C-parameters $\Theta_c = \{C_p\}_{p=1}^{k}$ characterizing the MA parts. Thus, ARMMAX model is parameterized by the compound parameter

$$\Theta \equiv \left(\Theta_a \equiv (\theta, r), \alpha \equiv [\alpha_1, \cdots, \alpha_k], \Theta_c \equiv \{C_p\}_{p=1}^{k}\right)$$

With the assumed normality and applying (3) for the known $C = C_p$, the p-th ARMAX component is described as $f(y_t|d(t-1), \Theta_a, C_p) = \mathcal{N}_{\tilde{y}_{p;t}}(\theta' \tilde{\psi}_{p;t}, rD_t)$.

The ARMMAX describes well the cases when the common ARX part has a physical meaning of interest. It provides more freedom in describing stochastic part of the input-output relationship. It is more flexible and richer for modeling of non-measured disturbances compared to a single ARMAX model. It is intuitively obvious but it can be shown also formally.

Proposition 3.1 (Moments of ARMMAX model) *For an ARMMAX model with a given selection of C-parameters $\Theta_c = \{C_p\}_{p \in p^*}$, an equivalent single ARMAX model exists in terms of the first moment. The equivalence does not hold with respect to variance.*

Proof: Results are implied by the mixture definition, the linearity of the expectation and the identity $E\left[y^2\right] = \mathrm{var}[y] + E^2[y]$.

For simplicity, the proof is presented with $k = 2$, when a pair $\Theta_c = (C_1, C_2)$ generates the filtered data $\tilde{\psi}_{p;t}, \Delta_{p;t}, \ p \in \{1,2\}$, c.f. Proposition 2.1. The mixing weights are $\alpha = [\alpha, 1-\alpha]$. The corresponding conditional expectation $E[\cdot|\cdot]$ and variance $\mathrm{var}[\cdot|\cdot]$ of the output y_t are

$$\begin{aligned} E\left[y_t|u_t, d(t-1), \Theta_a, \alpha, \Theta_c\right] &= \alpha\left(\theta'\tilde{\psi}_{1;t} + \Delta_{1;t}\right) + (1-\alpha)\left(\theta'\tilde{\psi}_{2;t} + \Delta_{2;t}\right) \\ &= \theta'\left(\alpha\tilde{\psi}_{1;t} + (1-\alpha)\tilde{\psi}_{2;t}\right) + \alpha\Delta_{1;t} + (1-\alpha)\Delta_{2;t} \\ \mathrm{var}\left[y_t|u_t, d(t-1), \Theta_a, \alpha, \Theta_c\right] &= \alpha(1-\alpha)\left(\theta'\left(\tilde{\psi}_{1;t} - \tilde{\psi}_{2;t}\right) + \Delta_{1;t} - \Delta_{2;t}\right)^2 + \\ &+ \ r\left(\alpha D_{1;t} + (1-\alpha)D_{2;t}\right). \end{aligned} \tag{11}$$

Smooth dependence of the filtered data vectors on Θ_c implies that a single equivalent C can be found generating a single filtered data vector equivalent to the convex combination in the first moment. Obviously, the dependence of the conditional variance on data cannot be neglected. □

The possibility to estimate in parallel k ARMAX models is the key property of ARMMAX we exploit. ARMMAX model can be interpreted as a parallel realization of several ARMAX models. It suits to the MDS procedure we intend to use for selecting adequate values of $\Theta_c = \{C_p\}_{p \in p^*}$, for an adequate filtering of the raw data vectors. The following propositions help us to support this claim.

Proposition 3.2 (Asymptotic of estimation) *Let natural conditions of control (4) hold and $0 < \underline{C}_\Theta \leq \overline{C}_\Theta \leq c < \infty$, $\bar{t}_\Theta \in \{1, 2, \ldots\}$ exist, for almost all $\Theta \in \Theta^*$, such that*

$$\underline{C}_\Theta f(y_t|u_t, d(t-1), \Theta) \leq {}^{[o]}f(y_t|u_t, d(t-1)) \leq \overline{C}_\Theta f(y_t|u_t, d(t-1), \Theta), \ \forall t > \bar{t}_\Theta, \ \forall d(t), \tag{12}$$

where ${}^{[o]}f(y_t|u_t, d(t-1))$ denotes the "true" generator of data.

Then, the posterior pdf $f(\Theta|d(t))$ converges almost surely (a.s.) to a pdf $f(\Theta|d(\infty))$. The asymptotic posterior pdf $f(\Theta|d(\infty))$ has the support $\mathrm{supp}\left[f(\Theta|u_\infty)\right] = \{\Theta : \ f(\Theta|u_\infty) > 0\}$ *coinciding with the following set of minimizing arguments*

$$\mathrm{supp}\left[f(\Theta|u_\infty)\right] = \mathrm{Arg} \inf_{\Theta \in \mathrm{supp}[f(\Theta)] \cap \Theta^*} \mathcal{H}_\infty\left({}^{[o]}f||\Theta\right), \ \textit{where } \mathcal{H}_\infty\left({}^{[o]}f||\Theta\right) \tag{13}$$

$$= \lim_{t \to \infty} \mathcal{H}_t\left({}^{[o]}f||\Theta\right) = \lim_{t \to \infty} \frac{1}{t} \sum_{\tau \leq t} \int {}^{[o]}f(y_\tau|u_\tau, d(\tau-1)) \ln\left[\frac{{}^{[o]}f(y_\tau|u_\tau, d(\tau-1))}{f(y_\tau|u_\tau, d(\tau-1, \Theta)}\right] dy_\tau.$$

Thus, if there is a unique consistent estimate of Θ then the Bayesian estimation provides it.

Proof: Under the natural conditions of control, the posterior pdf can be written in the form

$$\begin{aligned} f(\Theta|d(t)) &\propto f(\Theta)\exp\left[-t\mathcal{H}(d(t)||\Theta)\right] \quad \text{with} && (14)\\ \mathcal{H}\left(d(t)||\Theta\right) &= \frac{1}{t}\sum_{\tau\le t}\ln\left[\frac{{}^{[o]}f(y_\tau|u_\tau,d(\tau-1))}{f(y_\tau|u_\tau,d(\tau-1,\Theta)}\right]. && (15)\end{aligned}$$

This form exploits the fact that the non-normalized posterior pdf can be multiplied by any factor independent of Θ.

Let us fix the argument $\Theta\in\Theta^*$ and define the deviations $e_{\Theta;\tau}$ of values

$$\ln\left[\frac{{}^{[o]}f(y_\tau|u_\tau,d(\tau-1))}{f(y_\tau|u_\tau,d(\tau-1),\Theta)}\right]$$

from their conditional expectations ${}^{[o]}\mathcal{E}[\cdot|u_\tau,d(\tau-1)]$ with respected to ${}^{[o]}f(y_\tau|u_\tau,d(\tau-1))$,

$$\begin{aligned} e_{\Theta;\tau} &\equiv \ln\left[\frac{{}^{[o]}f(y_\tau|u_\tau,d(\tau-1))}{f(y_\tau|u_\tau,d(\tau-1),\Theta)}\right] - {}^{[o]}\mathcal{E}\left[\ln\left[\frac{{}^{[o]}f(y_\tau|u_\tau,d(\tau-1))}{f(y_\tau|u_\tau,d(\tau-1),\Theta)}\right]|u_\tau,d(\tau-1)\right]\\ &\equiv \ln\left[\frac{{}^{[o]}f(y_\tau|u_\tau,d(\tau-1))}{f(y_\tau|u_\tau,d(\tau-1),\Theta)}\right] - \int {}^{[o]}f(y_\tau|u_\tau,d(\tau-1))\ln\left[\frac{{}^{[o]}f(y_\tau|u_\tau,d(\tau-1))}{f(y_\tau|u_\tau,d(\tau-1),\Theta)}\right]dy_\tau.\end{aligned}$$

A direct check reveals that the introduced deviations $e_{\Theta;\tau}$ are zero mean and mutually non-correlated. With them,

$$\mathcal{H}(d(t),\Theta)=\mathcal{H}_t\left({}^{[o]}f||\Theta\right)+\frac{1}{t}\sum_{\tau\le t}e_{\Theta;\tau}.$$

The assumption (12) implies that the variance of $e_{\Theta;\tau}$ is bounded. Consequently, the last term in the above expression converges to zero almost surely (*a*.s.), see [11], page 417. The first term on the right hand side of the last equality is non-negative as it can be viewed as a sum of Kullback-Leibler distances, [12]. Due to (12), it is also finite. Thus, (15) converges almost surely to the non-negative value $\mathcal{H}\left({}^{[o]}f||\Theta\right)$. The posterior pdf remains unchanged if we subtract $\inf_{\Theta\in\text{supp}[f(\Theta)]\cap\Theta^*}\mathcal{H}_\infty\left({}^{[o]}f||\Theta\right)$ from the exponent of its non-normalized version (14). Then, the exponent contains $(-t\times$ an_asymptotically_non-negative_factor). Thus, the posterior pdf $f(\Theta|d(\infty))$ may be asymptotically non-zero on minimizing arguments (13) only.

If the unique ${}^{[o]}\Theta_0$ exists such that ${}^{[o]}f(y_\tau|u_\tau,d(\tau-1)=f\left(y_\tau|u_\tau,d(\tau-1),{}^{[o]}\Theta\right)$, then it is the unique minimizing argument due to the elementary properties of the Kullback-Leibler distance.

□

The following Proposition exploits the results of Proposition 3.2 to justify our claim on the parallel nature of the ARMMAX model.

Proposition 3.3 (Parallelism of ARMMAX model) *Under the natural conditions of control (4), the predictor, generated by an ARMMAX model parameterized by the collection $\Theta_c\equiv\{C_p\}_{p\in p^*}$, has the form*

$$f(y_t|u_t,d(t-1),\Theta_c)\propto\sum_{p=1}^{k}\hat{\alpha}_{p;t-1}(\Theta_c)\mathcal{I}\left(V_{t-1}(\Theta_c)+\tilde{\Psi}(C_p)_t\tilde{\Psi}'(C_p)_t,\nu_{t-1}+1\right),$$

where $\hat{\alpha}_{t-1}(\Theta_c) = \mathcal{E}[\alpha|d(t-1), \Theta_c]$. *The following inequality holds*

$$\begin{aligned} 0 &\leq \mathcal{H}\left({}^{[o]}f||\Theta_c\right) \leq \sum_{p\in p^*} \hat{\alpha}_{p;\infty}(\Theta_c)\mathcal{H}_p\left({}^{[o]}f||\Theta_c\right), \text{ with a.s. existing} \qquad (16) \\ \hat{\alpha}_{p;\infty}(\Theta_c) &= \lim_{t\to\infty} \hat{\alpha}_{p;t}(\Theta_c),\ 0 \leq \mathcal{H}_p\left({}^{[o]}f||\Theta_c\right) = \lim_{t\to\infty} \frac{1}{t}\sum_{\tau=1}^{t} \ln\left(\frac{f(y_\tau|u_\tau, d(\tau-1))}{f(y_\tau|u_\tau, d(\tau-1), C_p)}\right). \end{aligned}$$

Let the "true" system be described by a single ARMAX *model with* MA *part defined by some* ${}^{[o]}C$*, let the vectors* $\{C_p\}$ *forming* Θ_c *have the same order* n *as* ${}^{[o]}C$ *and create a non-degenerate simplex in* n*-dimensional real space.*

Then Θ_c *may belong to the support of the posterior pdf* $f(\Theta_c|d(t))$ *only if the weighted component entropy rates* $\hat{\alpha}_{p;\infty}(\Theta_c)\mathcal{H}_p(\Theta_c)$ *are simultaneously minimized. In other words,* Θ_c *may belong to the support of the posterior pdf if the vectors* $\{C_p\}$ *maximize asymptotic values of the weighted* component log-likelihood

$$l_p(d(t), \Theta_c) \equiv \hat{\alpha}_{p;t}(\Theta_c)\sum_{\tau=1}^{t} \ln[f(y_\tau|u_\tau, d(\tau-1), C_p)], \qquad (17)$$

with predictors $f(y_\tau|u_\tau, d(\tau-1), C_p) \propto \mathcal{I}\left(V_{t-1}(\Theta_c) + \tilde{\Psi}(C_p)_t\tilde{\Psi}'(C_p)_t, \nu_{t-1}+1\right),\ p \in p^*$.

Proof: The form of the mixture predictor is simply obtained by taking conditional expectation with respect to unknown α and ARX part.

Jensen inequality implies the inequality between finite sums defining asymptotically the involved entropy rates. Almost sure convergence of $\hat{\alpha}_{p;t}$ follows from the fact that, as conditional expectation of bounded variable, is a bounded martingale, [11]. Properties of a specific component entropy rate can be shown exactly as in the proof of Proposition 3.2.

The minimum can be reached if either the (asymptotic) estimate of components weights $\hat{\alpha}_{t-1}(\Theta_c)$ is zero-one probabilistic vector or all components with non-zero weights coincide. The former possibility is excluded by the use of C_p's defining a non-degenerate simplex.

The form of component predictors is implied by Proposition 2.2. □

In summary, by estimating a single mixture whose C-parameters form simplex recommended by the MDS method, we evaluate
quality of the competitive ARMAX model in *parallel*.
This gives us a chance to exploit the relatively slow MDS
procedure for practical estimation of the underlying ARMAX model.
The above propositions justify also the parallelism of general
mixture estimation with ARMAX components, i.e. without the
assumption on the common ARX part. We do not exploit this
possibility in order to keep the computational burden low. Assuming the common ARX part, ARMMAX has the computational complexity that is slightly larger than that needed for estimation of a single ARMAX model, see Section 4.
It is worth of stressing that finite data samples are always at
disposal and *approximate* mixture estimation is applied,
see section 4. Thus, only approximate component likelihood
values instead of their asymptotic values are available.
This makes robustness of the MDS procedure very important.

The parallelism can be used in an extended setting. Let us assume that the true description of the system is an ARMMAX model with m components distinguished by n-dimensional C-parameters ${}^{[m]}C = \{C_1, \ldots, C_m\}$. The joint dimension of the involved MA parts then equals to mn. Thus, we should use the MDS procedure with $mn + 1$ vertices in order to search for all involved MA parts. The same logic as above implies that the sub-mixture with m components having the highest weighted likelihood is the best approximation of the unknown "true" pdf. Especially in this case, the mild increase of the computational load, connected with the increase of the number of considered components, is invaluable.

4. ARMMAX-QB ESTIMATION

The possibility to use the ARMMAX model is supported by a recent progress in estimation of finite mixtures. The *Quasi-Bayes* (QB) estimation of mixtures with ARX components, [5], hints how to estimate the ARMMAX model. QB algorithm is a slight extension of classical mixture-estimation algorithms [10]. It has good properties, Bayesian motivation as well as predictable and feasible computational complexity. The following proposition modifies it for estimation of the Gaussian ARMMAX model with known C-parameters.

Proposition 4.1 (Quasi-Bayes estimation of ARMMAX model) *Let the modelled system be described by the ARMMAX model (10) with fixed MA parts $\Theta_c = \{C_p\}_{p=1}^{k}$ of individual components.*

Let us introduce the unobserved random pointer $p_t \in \{1, \ldots, k\} = p^*$ to the active component that takes the value $p \in p^*$ with the probability α_p. Let us define

$$
\begin{aligned}
& f(y_t, p_t|u_t, d(t-1), \Theta_a, \alpha, \Theta_c) \\
= \quad & f(y_t|p_t, u_t, d(t-1), \Theta_a, \alpha, \Theta_c) f(p_t|u_t, d(t-1), \Theta_a, \alpha, \Theta_c) = \mathcal{N}_{\tilde{y}_{p;t}}\left(\theta' \tilde{\psi}_{p;t}, rD_{p;t}\right) \alpha_{p_t}.
\end{aligned}
$$

Then, the marginal pdf $f(y_t|u_t, d(t-1), \Theta_a, \alpha, \Theta_c)$ is the ARMMAX model.

Let natural condition of control (4) hold and, at the time $t-1$, the posterior pdf on $\Theta_a = [\theta, r]$ and α have the form

$$
f\left(\Theta_a, \alpha|d(t-1), \Theta_c\right) = GiW_{\Theta_a}(V_{t-1}, \nu_{t-1}) Di_\alpha(\kappa_{t-1}),
$$

where the used Dirichlet pdf on the probabilistic vector α is defined as

$$
Di_\alpha(\kappa_{t-1}) \equiv \Gamma\left(\sum_{p=1}^{k} \kappa_{p;t-1}\right) \prod_{p=1}^{k} \frac{\alpha_p^{\kappa_{p;t-1}-1}}{\Gamma(\kappa_{p;t-1})}, \ \kappa_{p;t-1} > 0, \quad \Gamma(x) = \int_0^\infty z^{x-1} \exp(-z)\, dz, \ x > 0.
$$

Then, $f(\Theta_a, \alpha, p_t|d(t), \Theta_c)$ (18)

$$= GiW_{\Theta_a}\left(V_{t-1} + \sum_{p\in p^*} \delta_{p,p_t}\tilde{\Psi}_{p;t}\tilde{\Psi}'_{p;t}, \nu_{t-1}+1\right) Di_{\alpha}\left(\kappa_{t-1} + \sum_{p\in p^*}\delta_{p,p_t}[\underbrace{0,\ldots,0}_{p-1},1,0,\ldots]'\right),$$

where The Kronecker symbol $\delta_{p,p_t} = \left\{\begin{array}{ll} 1 & \textit{if}\, p = p_t \\ 0 & \textit{otherwise}\end{array}\right.$ *has the expectation*

$$w_{p;t} \quad \equiv \quad E\left[\delta_{p,p_t}|d(t),\Theta_c\right] \propto \mathcal{I}\left(V_{t-1} + \tilde{\Psi}_{p;t}\tilde{\Psi}'_{p:t}, \nu_{t-1}+1\right)(\kappa_{p;t-1}+1).$$

The approximation $\delta_{p,p_t} \approx w_{p;t}$ *preserves the assumed form of the posterior pdf even at time* t

$$f(\Theta_a, \alpha|d(t), \Theta_c) = GiW_{\Theta_a}(V_t, \nu_t)Di_{\alpha}(\kappa_t). \tag{19}$$

The associated statistics are updated according to the recursions

$$V_t \quad = \quad V_{t-1} + \sum_{p\in p^*} w_{p;t}\tilde{\Psi}_{p;t}\tilde{\Psi}'_{p;t}, \quad \nu_t = \nu_{t-1}+1, \quad \kappa_{p;t} = \kappa_{p;t-1} + w_{p;t}.$$

The last line describes the Quasi-Bayes estimation of the ARMMAX *model. The estimation uses the filtered data vectors* $\tilde{\Psi}_{p;t}$ *generated by the filters (3) determined by* $C = C_p,\ p \in p^*$.

Proof: The first statement is directly implied by marginalization over p_t.

The exact updating with the unknown value δ_{p,p_t} uses the normal ARX version of the ARMAX model (2), the form of the GiW pdf (5) and the Bayes rule, c.f. (6) applied under the natural conditions of control (4).

Marginalization over Θ_a and α giving the weight $w_{p;t}$ exploits the predictive Student pdf (7) and the elementary property of Dirichlet pdf $Di_{\alpha}(\kappa)$ stating that $E[\alpha_p|\kappa] \propto \kappa_p$.

The final updating is implied directly by the adopted approximation of δ_{p,p_t}. □

It is important that the above estimation requires $\mathcal{O}\left(n_a^2 + kn^2\right)$ flops. The computational burden increases linearly with the number of components k. Due to the presence of external input (often multi-variate), the dimension $n_a = n_{\psi}+1$ of the ARX part is usually much larger than the order n of MA. Since a single common ARX part is estimated, the computational complexity connected with estimation of ARMMAX model is slightly larger than that needed for estimation of a single ARMAX model.

5. MDS-ARMMAX-QB Estimation

Here, we combine the discussed essential ingredients – MDS method, the parallelism of ARMMAX and its efficient estimation algorithm ARMMAX-QB. It provides MDS-ARMMAX-QB procedure to estimate the ARMAX model with unknown C-parameters.

Specifically, the Quasi-Bayes algorithm, Proposition 4.1, is used for estimation of the ARMMAX model (10) defined by $\Theta_c = \{C_p\}_{p \in p^*}$ that is sitting on a simplex modified by the MDS algorithm, section 2.3. The MDS-ARMMAX-QB parallel estimation is applied in the "algorithmic" ARMMAX parallel environment.

- For a n-dimensional optimization problem, the number of vertices in the used simplex is generally determined as $n+1$. In our estimation of ARMAX setting, the dimension of the problem means the order of MA part. Notice that the structure estimation of the ARX part and selection of the order n of MA parts are assumed to be known as prior and not addressed here.
- The number of components in the corresponding ARMMAX is equal to the number of vertices in the current simplex, so that there are $k = n+1$ components in the ARMMAX with the n-dimensional MA parts.
- The vertices of the evolving simplex and the trial points from each of three possible operations in the iterations define a sequence of ARMMAX models correspondingly.

Recall that ARMMAX provides an "algorithmic" parallel environment in the sense that likelihood values assigned to individual components can be interpreted as the approximated values of the log-likelihood $\mathcal{L}(d(t), C_p)$ corresponding to respective ARMAX's whose MA parts defined by C_p, $p = 1, \ldots, k$. Thus, the component log-likelihood (17), i.e.

$$l_p(d(t), \Theta_c) = \hat{\alpha}_{p;t}(\Theta_c) \sum_{\tau=1}^{t} \ln\left[f(y_\tau | u_\tau, d(\tau-1), C_p)\right]$$

are used to judge of the quality of individual C_p's.

It is worth of stressing that they approximately reflect the global log-likelihood of interest as it has the form

$$\ln\left[\mathcal{L}(d(t), \Theta_c)\right] = \sum_{\tau=1}^{t} \ln\left[\sum_{p \in p^*} \hat{\alpha}_{p;\tau}(\Theta_c) f(y_\tau | u_\tau, d(\tau-1), C_p)\right]. \tag{20}$$

The value (20) is obtained as a byproduct of the Quasi-Bayes estimation and may serve for monitoring of success of the search made via the component log-likelihood values $l_p(d(t), \Theta_c)$.

5.1. MDS-ARMMAX-QB algorithm

This subsection describes formally the basic MDS-ARMMAX-QB algorithm. In Bayesian estimation context, we need to respect the prior and flattening issues, they are discussed together with the other implementation details in next subsection.

Algorithm 5.1 (MDS-ARMMAX-QB search)

Initial phase

- *Select a suitable initial guess C_1^0 of MA-part.*

- *Generate the other vertices $C_p^0, p = 2, \cdots, n+1$ of the initial simplex based on C_1^0. The non-degenerate simplex is formed by $n+1$ vertices to span the n-dimensional space. The number n is the common order of the C-polynomials defining* MA *parts of $n+1$-components of the* ARMMAX *model.*
- *Select expansion $\chi \in (1, \infty)$ and contraction $\xi \in (0, 1)$ factors.*
- *Select the stopping rule and define its parameters.*
- *Specify the $GiW_\Theta(V_0, \nu_0) Di_\alpha(\kappa_0)$ prior pdf on parameters of the* ARMMAX *model with the $k = n+1$ components determined by the vertices of the initial simplex. It is given by the extended information matrix $V_0 = L_0' D_0 L_0$, the scalar ν_0 and the vector κ_0.*
- *Perform the* ARMMAX-QB *estimation, Proposition 4.1, i.e. update the statistics $L_0, D_0, \nu_0, \kappa_0$ to $L_t, D_t, \nu_t, \kappa_t$, where t is the length of the available data sample $d(t)$. Accumulate the vector $l^0 = \left[l_1^0(d(t), \Theta_c), \ldots, l_{n+1}^0(d(t), \Theta_c)\right]$ of component log-likelihood values (17), corresponding to the vertices C_p^0 defining respective components $p = 1, \ldots, n+1$.*
- *Swap, the vertices according to the values of the entries of l^0 so that the vertex with the highest component likelihood l_p^0 is labeled as C_1^0,*
- *Set j, the counter of the total number of iterations, to zero.*

Iterative phase

Do while the stopping rule is not met

Set $j := j + 1$

1. ***Reflection***

   - *Define n reflected vertices $C_p^r = 2C_1^{j-1} - C_p^{j-1}$, for $p = 2, \cdots, n+1$.*
   - *Specify the* MA *parts of $k = n+1$ components in* ARMMAX *by the reflected vertices and the current best vertex C_1^{j-1}. Specify the corresponding prior pdf on the* ARX *part.*
   - *Perform the* ARMMAX-QB *estimation. Accumulate the vector l^r of component log-likelihood values (17) corresponding to the vertices C_p^r.*
   - *Set $maxr = \arg\max_{p \in p^*} l_p^r$.*
   - *Go to the step 2, if $maxr > 1$. Otherwise, go to the step 3.*

2. ***Expansion***

   - *Define n expanded vertices $C_p^e = C_1^{j-1} + \chi(C_1^{j-1} - C_p^{j-1})$, for $p = 2, \cdots, n+1$.*

- *Specify the* MA *parts of $k = n + 1$ components in* ARMMAX *by the expanded vertices and the current best vertex C_1^{j-1}.*
  Specify the corresponding prior pdf on the ARX *part.*
- *Perform the* ARMMAX-QB *estimation. Accumulate the vector l^e of component log-likelihood values (17) corresponding to the vertices C_p^e.*
- *Set maxe* $= \arg\max_{p \in p^*} l_p^e$.
- *Accept the expansion to replace C_p^j by C_p^e, for $p = 2, \cdots, n+1$, if $l^e_{maxe} > l^r_{maxr}$. Otherwise accept the reflection to replace C_p^j by C_p^r, for $p = 2, \cdots, n+1$.*
- *Go to step 4.*

3. ***Contraction***

- *Define n contracted vertices $C_p^c = C_1^{j-1} + \xi(C_1^{j-1} - C_p^{j-1})$, replace C_p^j by C_p^e, for $p = 2, \cdots, n+1$.*
- *Specify the* MA *parts of $k = n + 1$ components in* ARMMAX *by the expanded vertices and the current best vertex C_1^{j-1}.*
  Specify the corresponding prior pdf on the ARX *part.*
- *Perform the* ARMMAX-QB *estimation. Accumulate the vector l^c of component log-likelihood values (17) corresponding to the vertices C_p^c.*
- *Set maxc* $= \arg\max_{p \in p^*} l_p^c$.
- *Go to the step 4 if maxc > 1, otherwise go to the step 1.*

4. ***Swap***

Swap the new best point as C_1^j according to the components likelihood values of the accepted mixture.

In all cases, $l \equiv [l_1(d(t), \Theta_c), \ldots, l_{n+1}(d(t), \Theta_c)] = [l_1, \ldots, l_{n+1}]$ is the component log-likelihood vector. The upper index distinguishes the operation, for instance, l_p^r means the log-likelihood of the p-th component in *reflection* operation.

5.2. Implementation

Implementation follows predominantly the standard recommendations of general MDS [6]. The following implementation options are adopted.

- **Prior pdf and flattening of the posterior ones**: The critical choice of the prior pdf for the ARMMAX model is solved in the following way. The very initial prior pdf is chosen by incorporating prior knowledge on the ARX part, see [13]. In a generic iterative step, the latest posterior pdf on ARX part is flattened so that the iterations do not cause false over-confidence, see [14].

- **Initial simplex**: The multi-directional search starts at an externally supplied initial simplex $\langle C_1^0, \cdots, C_{n+1}^0 \rangle$. It is generated from an initial point C_1^0 and the simplex is chosen by deciding on:

  Shape: Good spanning of the C-space is guaranteed by selecting the initial simplex with *right angle*. The remaining vertices are determined as follows

  $$C_p^0 = C_1^0 + \beta_p \mathbf{1}_p, \quad p = 2, \cdots, n+1, \tag{21}$$

  where $\mathbf{1}_p$ denotes the unit coordinate vector and β_p's is a length scaling coefficient. It determines magnitude of the entry c_p in C_p^0 relative to the corresponding entry of C_1^0. It must be non-zero in order to get the needed non-degenerate initial simplex.

  Size and orientation: Formally, arbitrary magnitude of C_1^0 and scaling factors β_p can be chosen due to the use of the filter (2) that enables us to work with an unconstrained problem. We should, however, respect the fact that influence of the C-parameters is indeed scaling dependent. The fact that C can be chosen as asymptotic spectral factor helps us to determine a rough scaling of the C-parameters. The magnitude of the p-th entry element c_p in $C = (c_1, \cdots, c_n)$ need not in exceed the combination number $|c_p| \leq \begin{pmatrix} n \\ p \end{pmatrix}, p = 1, \cdots, n$. This makes us to define

  $$\beta_p = \pm h \begin{pmatrix} n \\ p \end{pmatrix}, \quad p = 1, \cdots, n. \tag{22}$$

  Thus, its magnitude $|\beta_p| = h \begin{pmatrix} n \\ p \end{pmatrix}$ determines the *size* of simplex. The scalar $h \in (0, 1)$ becomes the only scaling parameter as the order n is fixed. The signs of β_p determine the important *orientation* of the initial simplex. There is no universal rule how to select them. Thus, whenever possible, it makes sense to try several initial options differing just in orientation.

- **Scaling factors in algorithm (2.1)**: We stick to the usual choices: unit reflection factor, $\chi = 2$, $\xi = 1/2$ for expansion and contraction factors, respectively.
- **Stopping criteria**: Standard stopping criteria are adopted and enriched by a specific one:

The relative size of simplex is inspected. It is

measured by the length of longest edge adjacent to the best vertex C_1^j

$$\frac{1}{\Delta}\max_{1\leq i\leq n}\left|\left|C_i^j - C_1^j\right|\right| \leq \epsilon,\quad \epsilon\in(0,1), \tag{23}$$

where $\Delta = \max\left(1, \left|\left|C_1^j\right|\right|\right)$ and ϵ is a pre-selected tolerance.

This rule is used in the illustrative example, see section 6.

The number of search iterations is limited.

Such a number usually can be determined by the affordable computational time. Here, we benefit from the fact that the Quasi-Bayes estimation has fixed a priori known computational demands.

Increments of the global log-likelihood of the mixture

$\mathcal{L} = \mathcal{L}(d(t), \Theta_c)$ (20) are checked.

The evaluations are stopped if the increment among iteration steps

$$\left|\frac{\mathcal{L}^j - \mathcal{L}^{j-1}}{\mathcal{L}^j}\right| \leq \eta,\quad \eta\in(0,1) \tag{24}$$

is smaller than a pre-specified threshold η.

6. ILLUSTRATIVE EXAMPLE

Here, the simple simulated example illustrates the performance of the proposed estimation procedure that is implemented using the algorithmic basis of the toolbox Mixtools [15].
The data $d(t)$ of the length $t = 2000$ were generated by the following single-output ARMA, i.e. by ARMAX with no external input present,

$$y_t = 1.5y_{t-1} - 0.7y_{t-2} + e_t - 0.8e_{t-1} + 0.6e_{t-2}. \tag{25}$$

The variance $r = 0.1$ of the driving white Gaussian noise e_t was chosen.

The same structure as the simulated system is used for the ARMA models forming the components of the estimated ARMMA model.

$$y_t = a_1y_{t-1} - a_2y_{t-2} + e_t + c_1e_{t-1} + c_2e_{t-2}$$

Since the order of the MA term

is $n = 2$, there are $k = n + 1 = 3$ vertices in the used simplex.
Correspondingly, 3-component ARMMA are estimated by ARMMAX-QB algorithm.
Just for an informal comparison, the AR model

$$y_t = a_1 y_{t-1} - a_2 y_{t-2} + e_t$$

was estimated using Proposition 2.2 with $\tilde{\Psi}_t = \Psi_t = [y_t, y_{t-1}, y_{t-2}]'$.
The following implementation options, section 5.2, of the MDS-ARMMAX-QB algorithm 5.1, were made.

- Three different representative starting points were chosen

$$C_I^0 = (-1/2 \ \ 1/2),\ C_{II}^0 = (1/3 \ \ -1/3),\ C_{III}^0 = (-1/3 \ \ -1/3)$$

- The corresponding three right-angle initial simplex generated, according to (21). The relation (22) gives $\beta = [\pm 0.2 \ \ \pm 0.1]$ for the chosen $h = 1/10$.
- Stopping according to the relative size of simplex (23) was chosen with tolerance $\epsilon = 10^{-4}$.
  No limit on the number of iterations and likelihood of mixture were imposed in order to illustrate convergence properties of the algorithm.

The estimation results for considered starts are reflected in Table 1 where the estimates with the AR model are added.

	Initial C^0	$\hat{\theta} = E[\theta \vert d(t)]$	$\hat{r} = E[r \vert d(t)]$	Final best vertex C	No. j
True values	–	1.5 -0.7	0.1	-0.8 0.6	–
AR	–	0.8428 -0.0154	0.1382	0 0	-
ARMMA	-1/2 1/2	1.4642 -0.6543	0.1056	-0.7000 0.5000	13
ARMMA	1/3 -1/3	1.4737 -0.6795	0.1044	-0.7708 0.6021	24
ARMMA	-1/3 -1/3	1.4889 -0.7059	0.1053	-0.8333 0.6667	18

Table 1: *Point estimates for AR and ARMMA models obtained for different starting points* C^0.

Bad estimation results obtained by least-square corresponding to the estimation of AR model are not surprising. It indicates that the influence of the chosen MA part is not negligible.
The proposed MDS-ARMMAX-QB method behaves well. And even this simple example demonstrates that the quality of results depends on the choice of the initial simplex. For instance, $C_I^0 = (-1/2 \; 1/2)$ is the closest to the true C and required the fewest iterations, 13. It is also fair to report that other experiments indicate that the size and orientation of the initial simplex may have significant, sometimes rather adverse, influence on the quality of results.

7. CONCLUSIONS

A finite mixture of ARMAX components with the common ARX part is introdueced in the paper. The resulting ARMMAX model is an interesting model capturing deterministic input-output relationships while allowing temporal variations of colored noise part. Here, ARMMAX model has served us as a generator of numerically-quantified candidates of the MA parts of competitive ARMAX models. In combination with a Quasi-Bayes estimation and multi-directional search method, it leads to a novel MDS-ARMMAX-QB estimation algorithm. The algorithm is able cope with an unknown MA part of the ARMAX model while preserving Bayesian processing of its ARX part. The respective design steps are mostly supported theoretically. Preliminary experimental results are promising. Still a lot of work remains to be done to make MDS-ARMMAX-QB algorithm fully ready for routine real applications. Use of mixtures with more components than the number of vertices in simplex and standardization of tuning knobs are the main directions to be addressed. Richer modeling power of ARMMAX and the gained algorithmic parallelism are the main features to be exploited further on. For instance, completely different filters than the Peterka's one can be used within the current framework [16]. It opens a way to a wide set of novel adaptive filters dealing with outliers, with temporarily varying measurement noise etc. Practical impact of such possibilities can hardly be over-stressed.

REFERENCES

1. L. Ljung, *System Identification-Theory for the User.* Prentice-hall. Englewood Cliffs, N.J: D. van Nostrand Company Inc., 1987.
2. V. Peterka, "Bayesian approach to system identification," in *Trends and Progress in System identification* (P. Eykhoff, ed.), pp. 239–304, Oxford: Pergamon Press, 1981.
3. M. Kárný and R. Kulhavý, "Structure determination of regression-type models for adaptive prediction and control," in *Bayesian Analysis of Time Series and Dynamic Models* (J. Spall, ed.), New York: Marcel Dekker, 1988. chapter 12.
4. V. Peterka, "Self–tuning control with alternative sets of uncertain process models," in *Proceedings of IFAC Symposium on Adaptive Systems in Control and Signal Processing*, pp. 409–414, 1989.
5. M. Kárný, J. Kadlec, and E. L. Sutanto, "Quasi-Bayes estimation applied to normal mixture," in *Preprints of the 3rd European IEEE workshop on Computer-Intensive Methods in Control and Data Processing, J. Rojíček, M. Valečková, M. Kárný and K. Warwick, Eds.*, pp. 77–82, Prague: ÚTIA AV ČR, September 1998.
6. V. Torczon, *Multi-directional Search: A Direct Search Algorithm for Parallel Machines.* Rice University, Houston, Texas, USA: Ph.D thesis, 1989.
7. M. H. Wright, "Direct search methods: Once scorned, now respectable," *Numerical Analysis 1995: Proceedings of the 1995 Dundee Biennial Conference in Numerical Analysis*, pp. 191–208, 1996.
8. J. C. Lagarias, J. A. Reeds, M. H. Wright, and P. E. Wright, "Convergence properties of the Nelder-Mead simplex method in low dimensions," *SIAM Journal of Optimization*, vol. 9, pp. 112–147, 1998.

9. V. Torczon, "On the convergence of the multidirectional search algorithm," *SIAM journal on optimization*, vol. 1, pp. 123–145, 1991.
10. D. M. Titterington, A. F. M. Smith, and U. E. Makov, *Statistical Analysis of Finite Mixtures.* Chichester, New York, Brisbane, Toronto, Singapore: John Wiley & Sons, 1985. ISBN 0 471 90763 4.
11. M. Loeve, *Probability Theory.* Princeton, New Jersey: D. van Nostrand Company Inc., 1962. Russian translation, Moscow 1962.
12. S. Kullback and R. Leibler, "On information and sufficiency," *Annals of Mathematical Statistics*, vol. 22, pp. 79–87, 1951.
13. M. Kárný, N. Khailova, J. Böhm, and P. Nedoma, "Quantification of prior information revised," *International Journal of Adaptive Control and Signal Processing*, vol. 15, no. 1, pp. 65–84, 2001.
14. M. Kárný, P. Nedoma, I. Nagy, and M. Valečková, "Initial description of multi-modal dynamic models," in *Artificial Neural Nets and Genetic Algorithms. Proceedings* (V. Kůrková, R. Neruda, M. Kárný, and N. C. Steele, eds.), (Wien), pp. 398–401, Springer, April 2001.
15. P. Nedoma, M. Kárný, I. Nagy, and M. Valečková, "Mixtools. MATLAB Toolbox for Mixtures," Tech. Rep. 1995, ÚTIA AV ČR, Praha, 2000.
16. V. Šmídl, M. Kárný, T. Guy, and A. Quinn, "Mixture-based filter bank for estimation of ARX model," vol. –, pp. –, 2003. under preparation.