HORUS - High-dimensional model Order Reduction via low moment-matching Upgraded Sampling

Jorge Fernández Villena*, Luis Miguel Silveira*§
*INESC ID / IST - TU Lisbon. Rua Alves Redol 9, 1000-029 Lisbon, Portugal. Email:jorge.fernandez@inesc-id.pt
§Cadence Research Labs. Rua Alves Redol 9, 1000-029 Lisbon, Portugal. Email:lms@inesc-id.pt

Abstract— This paper describes a Model Order Reduction algorithm for multi-dimensional parameterized systems, based on a sampling procedure which incorporates a lower-order moment matching paradigm into a multi-point based methodology. The procedure seeks to maximize the subspace generated by a given number of samples, selected among an initial candidate set. The selection is based on a global criteria that chooses the sample whose associated vector adds more information to the existing subspace. However, the initial candidate set can be extremely large for high-dimensional systems, and thus the procedure can be costly. To improve efficiency we propose a scheme to incorporate information from low-order moments to the basis with small extra cost, in order to extend the approximation to a wider region around the selected point. This will allow reduction of the initial candidate set without decreasing the level of confidence. We further improve the procedure by generating the global subspace based on the composition of local approximations. To achieve this, the initial candidates will be split into subsets that will be considered as independent regions, and in a first phase the procedure applied locally thus enabling improved efficiency and providing a framework for almost perfect parallelization.

I. INTRODUCTION

Model Order Reduction (MOR) encompasses a set of techniques whose goal is to generate reduced models with lower complexity while ensuring that the I/O response and other characteristics of the original model (such as passivity) are maintained [1]. Within the EDA industry, the detailed descriptions obtained after the modeling and extraction steps are often too large for efficient simulation and verification, and thus the application of MOR is required. Techniques based either on moment matching [2] or on truncated balanced realizations [3] have achieved considerable success in this area. Multi-point based approaches [4], [5] have recently gathered renewed attention due to their robustness and reliability, in particular in cases of large frequency range.

However, for smaller technology processes the increasing effect of electrical, geometrical and environmental parameters has to be taken into account leading to parameterized models [6], [7]. In order to successfully use such models inside simulation environments, Parameterized Model Order Reduction (pMOR) methodologies must be used, which address these representations and the effect of the parameters.

In this context multi-dimensional sample-based techniques [8] have been developed from the multi-point linear approaches [5] in order to tackle the issue of pMOR. These algorithms seem less sensitive to the number of parameters, and become a good alternative to the multi-dimensional moment matching approaches, [7], [9], [10], [11], which seem unable to deal with what has been denoted as the curse of dimensionality (leading to oversized models when the number of parameters and the accuracy require matching a large set of moments). However, the multi-dimensional sampling can be expensive if there is no good indication of where to place the sample points. Random based sampling can defeat the reliability and efficiency of these algorithms, whereas trying to cover the complete subspace with a linear scheme is overwhelming beyond the single dimension (an area to which some effort was devoted [12], [13]).

Recently, in [14], an algorithm was proposed to efficiently deal with this multidimensional sampling, based on the maximization of the generated subspace. Starting from an initial candidate set, and using the system residue at each point as a proxy to the error, the procedure automatically selects, without any system knowledge, the point among a set of candidates whose solution will minimize the maximum residue, and (if the residue is a good proxy for the error) therefore the maximum error. A drawback faced by this procedure is the fact that, although reliable and robust, if system information is missed in the initial candidate set, it will be missed in the generated basis. This means that a relatively fine discretization of the multidimensional space was required to achieve a good level of confidence.

In this paper we will try to increase the confidence level by incorporating a low order moment matching procedure at each point, which will improve the approximation in a larger local region around each sample, allowing for a coarser but perhaps more comprehensive discretization of the multidimensional space. This will allow for a reduction both of the number of samples in the candidate set and of the number of solves required by the procedure, since the extra vectors provided by the moments will improve the global approximation. Also, the procedure will be combined with a partition of the initial domain into smaller candidate sets. This will allow for local approximations that can be combined into a globally accurate model, improving the efficiency and allowing for an almost perfect parallelization.

This paper is structured as follows: in Section II an overview of the parameterized MOR paradigm is presented, along with a discussion of existing techniques. In Section III the new scheme is introduced, along with a study of its complexity and computational issues. Automation, reliability and efficiency will be the focus of the algorithm. In Section IV several examples are shown that illustrate the efficiency of the proposed technique, and in Section V conclusions are drawn.
II. Background

A. Parameterized Systems and Representation

The main techniques in MOR are geared toward the reduction of a state space linear time-invariant system, obtained by some modeling methodology, and representing a physical system. In such representation, the output $y$ is related to the input $u$ via some inner states $x$. When parameterized variations are taken into account, the system is represented as a parametric state-space descriptor,

$$ C(\lambda) \dot{x}(\lambda) + G(\lambda)x(\lambda) = Bu , \quad y(\lambda) = Ex(\lambda), $$

(1)

where $C, G \in \mathbb{R}^{n \times m}$ are respectively the dynamic and static matrices, $B \in \mathbb{R}^{n \times m}$ is the matrix that relates the input vector $u \in \mathbb{R}^{m}$ to the inner states $x \in \mathbb{R}^{n}$ and $E \in \mathbb{R}^{p \times n}$ is the matrix that links those inner states to the outputs $y \in \mathbb{R}^{p}$. We assume here, as is common, that the elements of $C$ and $G$, as well as the states $x$, depend on a set of $P$ parameters $\lambda = [\lambda_1, \lambda_2, \ldots, \lambda_P] \in \mathbb{R}^P$ which model the effects of the uncertainty. The representation of parameterized dependence is usually obtained via first order sensitivity computation of the discretized elements with respect to the parameters [6], [7]. Therefore, the parameterized system can be described as a state space descriptor in which the $C$ and $G$ matrices of (1) are represented as a Taylor Series approximation with respect to the parameters:

$$ G(\lambda) = G_0 + \sum_{i=0}^{\infty} \lambda_i^1 G_{i0} \ldots 0 + \lambda_i^2 G_{i0} \ldots 0 + \ldots $$

$$ C(\lambda) = C_0 + \sum_{i=0}^{\infty} \lambda_i^1 C_{i0} \ldots 0 + \lambda_i^2 C_{i0} \ldots 0 + \ldots $$

(2)

where $G_0, C_0$ are the nominal values, and $G_{ij...k}$, and $C_{ij...k}$ are the sensitivities w.r.t. the parameters.

The parameterized time-domain description yields a frequency response modeled via the transfer function

$$ H(s, \lambda) = E(sC(\lambda) + G(\lambda))^{-1}B, $$

(3)

The objective of pMOR techniques is to generate a reduced order approximation of (3), able to accurately capture the input-output behavior of the system for any point in the joint frequency-parameter space,

$$ \hat{H}(s, \lambda) = \hat{E}(s\hat{C}(\lambda) + \hat{G}(\lambda))^{-1}\hat{B}, $$

(4)

where $\hat{C}, \hat{G} \in \mathbb{R}^{q \times q}, \hat{B} \in \mathbb{R}^{q \times m}$, and $\hat{E} \in \mathbb{R}^{p \times q}$, are the reduced set of matrices, with $q \ll n$ the reduced order.

B. Projection based Reduction

The most common procedure to obtain an accurate and structurally similar ROM is to use some form of projection scheme on a sensitivity-based Taylor Series Representation, as presented in [7]. Standard pMOR methodologies rely on the generation of a suitable low order subspace (spanned by the basis $V \in \mathbb{R}^{n \times q}$), in which the original system matrices $C(\lambda), G(\lambda), B$ and $E$ are projected,

$$ \hat{C}(\lambda) = V^T (C_0 + \sum_{i=0}^{\infty} \lambda_i^1 C_{i1} + \lambda_i^2 C_{i2} + \ldots) V $$

$$ \hat{G}(\lambda) = V^T (G_0 + \sum_{i=0}^{\infty} \lambda_i^1 G_{i1} + \lambda_i^2 G_{i2} + \ldots) V $$

(5)

$$ \hat{B} = V^T B \quad \hat{E} = EV \quad x(s, \lambda) = V \hat{x}(s, \lambda) $$

where $V \in \mathbb{R}^{n \times q}$ spans the projection subspace of reduced dimension $q$, and $\hat{C}, \hat{G} \in \mathbb{R}^{q \times q}, \hat{B} \in \mathbb{R}^{q \times m}, \hat{E} \in \mathbb{R}^{p \times q}$, and $\hat{x} \in \mathbb{R}^{q}$ define the Reduced Order Model associated to a transfer function (4) which accurately approximates the behavior of (3) under parameter variations. To ensure the accuracy of the ROM, the basis $V \in \mathbb{R}^{n \times q}$ must be able to capture the behavior of $x(s, \lambda)$ for the relevant $\{s, \lambda\}$ space, or in mathematical form,

$$ x(s, \lambda) \approx \sum_{i=0}^{\infty} \alpha_i(s, \lambda)V_i \quad \forall \{s, \lambda\}, $$

(6)

where $V_i$ is the $i$-th column of the projector $V$, and $\alpha_i(s, \lambda)$ is any complex value. Different approaches have been presented in order to generate the matrix $V$. Most of the techniques in the literature extend the moment matching paradigm [2] to the multi-dimensional case [7], [9], [10], [11]. They usually rely in the implicit or explicit matching of the moments of the parametric transfer function (3),

$$ x(s, \lambda_1, \ldots, \lambda_P) = \sum_{k=0}^{\infty} \sum_{k_1=0}^{k-k_0} \sum_{k_2=0}^{k-k_1-1} \cdots \sum_{k_{P-1}=0}^{k-k_{P-1}} M_{k,k_1,k_2,\ldots,k_{P-1}} s^{k-k_0-k_1-\cdots-k_{P-1}} \lambda_1^{k_1} \cdots \lambda_P^{k_{P-1}} $$

(7)

where $M_{k,k_1,k_2,\ldots,k_{P-1}}$ is a $k$-th ($k = k_0 + k_1 + \ldots + k_{P}$) order multi-parameter moment corresponding to the coefficient term $s^{k-k_0-k_1-\cdots-k_{P-1}} \lambda_1^{k_1} \cdots \lambda_P^{k_{P-1}}$. Following the same idea used in the nominal moment matching techniques, a basis for the subspace formed from these moments can be built

$$ \text{colspan}[V] = \text{span}\{M_{0,0,0,\ldots,0}, \ldots, M_{k,k_1,k_2,\ldots,k_{P-1}}\} $$

(8)

and the resulting matrix $V$ can be used as a projection matrix. The generated parameterized ROM matches up to the $k$-th order multi-parameter moment of the original system. Different approaches differ in which moments are matched and how these moments are generated. In general, these methods, which rely in local matching, suffer from oversize of the models when the number of moments to match is high, either because high order is required, or because the number of parameters is large.

A different paradigm, more appealing to our goals, is based on Multi-Point methods [8]. Multi-Point approaches seek to generate the basis either by generating the transfer function moments from multiple expansion points $(s_k, \lambda_k)$, or from solving the system at different sample points on the relevant frequency plus parameter space,

$$ z_k = z(s_k, \lambda_k) = (s_kC(\lambda_k) + G(\lambda_k))^{-1}B, $$

(9)

where $z(s_k, \lambda_k)$ is the sample vector generated at the sample point $(s_k, \lambda_k)$, i.e. the zero order moment. In general, the space of interest is sampled and the most relevant vectors among those generated are selected via Singular Value Decomposition (SVD) in order to build the projector $V$. This approach, less sensitive to the number of parameters, is more reliable but, on the other hand, depends on a good sampling selection scheme. Regrettably, bad or poor sampling (undersampling) may lead to loss of information and inaccurate results, whereas oversampling can lead to inefficiency.
C. Iterative Sample Selection Methodology

Recently, in [14], an automatic sampling scheme was presented. This methodology was built on the premise that computation of the system matrix

$$A(s, \lambda) = G(\lambda) + sC(\lambda)$$

(10)

for a sampling point is inexpensive (as we only need to evaluate the matrices in (2) and sum them), whereas obtaining the sample vector is expensive (it requires solving the system, and thus factorizing the system matrix and performing backsolves). As a consequence, the methodology aims at obtaining a minimum number of vectors (and thus minimizing the number of solves) so that we can obtain a good approximation of the states vector, i.e. we want to find the minimum set \( q \) so that (6) holds. Therefore, the scheme seeks to retrieve the best samples to solve for, but to do so before solving them.

The method starts from an initial candidate set \( \Psi = \{\psi_1, \ldots, \psi_k\} \), where each \( \psi_j \) is a point in the space of interest. \( \Psi \) is a set that covers such space of interest (joint frequency and parameter space), and at each step, we will select from it the most appropriate point for our goals. This is done by trying to find which point has an associated vector that is less similar to the vectors we have already computed,

$$x(\psi_j) \neq \sum_{i=0}^{k} \alpha_i V_i \quad \psi_j \in \Psi,$$

(11)

for any complex \( \alpha_i \), and with \( k \) the number of vectors we have already generated. Although this is not a perfect indicator of the optimum point to select, it will maximize our chances of adding rank to the subspace, and such information can be obtained in a cheap manner.

We take the \( j \)-th candidate sample point, \( \psi_j \), with an associate matrix \( A_j = A(\psi_j) \). If we have a vector \( z_i \in \mathbb{R}^n \), we can determine if it is a good solution for our system at this \( j \)-th point by computing the norm of the residue,

$$||r_{i,j}|| = ||B - A_j z_i||,$$

(12)

where \( r_{i,j} \) is the residue of the vector \( z_i \) for the system \( \{A_j, B\} \). The cost of this operation is relatively cheap since \( A_j \) is sparse. If the norm is small, it means that the system \( \{A_j, B\} \) is well approximated by vector \( z_i \), so in this case \( \psi_j \) is not a good new point.

In our case, we have a basis \( V \in \mathbb{R}^{n \times k} \). The error we commit in the approximation of the state vector at a given point \( \psi_j \) is

$$e_j = x(\psi_j) - \sum_{i=0}^{k} \alpha_i (\psi_j) V_i,$$

(13)

Multiplying by the system matrix evaluated in this point, \( A_j = A(\psi_j) \), we obtain

$$A_j e_j = A_j x(\psi_j) - \sum_{i=0}^{k} \alpha_i (\psi_j) A_j V_i,$$

(14)

Since \( A_j x(\psi_j) = B \), and if we denote \( r_j = A_j e_j \),

$$r_j = B - \sum_{i=0}^{k} \alpha_i (\psi_j) A_j V_i,$$

(15)

Therefore, to see if the system \( \{A_j, B\} \) is well approximated by a set of vectors \( V \), we simply orthogonalize vector \( B \) against the set of vectors \( A_j V \), and generate the norm

$$||r_j|| = ||B \perp A_j V||,$$

(16)

where \( r_j \) is the residue after the orthogonalization (\( \perp \)) of \( B \) against the set of vectors \( A_j V \). If the norm of \( r_j \) is small, the system \( \{A_j, B\} \) is well approximated by the already chosen set of vectors \( V \), and thus, the vector associated with the candidate point \( \psi_j \), will probably add "small" rank to the subspace spanned by the set of vectors \( V \).

The operation in (16) is repeated for all the candidate sample points, to know, for each of them, whose solution is the worst approximated by the set of vectors. This information can be obtained by finding the maximum among the norm of the residues given by (16) for each candidate point \( j \). This methodology ensures that the new vector generated by the selected sample point adds rank to the subspace, and furthermore, it is likely to be a relevant point to include in the sample set. Once we have selected the best suited candidate \( \psi_j \), we solve the system to generate the vector \( z_j \), and withdraw the sample point for the candidate set, \( \Psi = \{\psi_1, \ldots, \psi_{j-1}, \psi_{j+1}, \ldots, \psi_k\} \). Then we repeat the procedure to obtain the next point.

III. PROPOSED METHODOLOGY

This section introduces the proposed methodology, which enhances the scheme of [14] by the generation of low order moments at each sample point, and by a sub-domain partition.

A. Moment-Matching Upgraded Sampling

The method in [14], as depicted in the previous sections, only computes the zero order moment at each selected point. However, we notice that once we have selected a new suitable sample point to solve for, the main cost comes from solving the system at such point. This solve is usually done by performing an \( LU \) factorization of the matrix \( A_j = A(s_j, \lambda_j) \), plus the backsolves with the factors and the matrix \( B \). Let us suppose we have a state space description, with an explicit dependence on the variables, e.g. a Taylor Series approximation in the form of (2) in combination with a state space representation in (1). Once we have computed the \( LU \) factors for the system matrix, and computed the zero order moment by solving the system, we can obtain the multi-dimensional moments [7], [9] at such expansion point with cheap extra computations, via an Arnoldi procedure [2]. If we only take into account first order approximation and no cross-terms, for a given point \( (s_j, \lambda_j) \),

$$A_j = G_0 + \sum_{i=1}^{P} \lambda_i G_i + s_j (C_0 + \sum_{i=1}^{P} \lambda_i C_i)$$

$$L_j = L \psi (A_j),$$

$$M_{0,i} = z_j = U^{-1} L^{-1} B \quad \forall i = 0, \ldots, P$$

$$M_{f,i} = U^{-1} L^{-1} (G_i + s_j C_i) M_{f-1,i},$$

(17)

where \( G_i \) and \( C_i \) are the first order sensitivities of the Taylor Series in (1), \( M_{f,i} \) is the moment of order \( f \) with respect to the parameter \( i \) (frequency is taken as parameter 0). More refined approaches can take higher order approximations and cross terms into account, following some of the Multi-Dimensional moment matching approaches already referred (for more details see [7], [10]). Of course the implementation should be done so that numerically robustness is asserted (e.g. moment computation is performed implicitly).
Once all the desired moments are computed, the matrix $V$ is updated with
\[ V = SVD([V \ M_0 \ M_{1,0} \ M_{1,1} \ldots]) \] (18)
And now, this subspace is used in (16) for generating the residues and finding the next sample point.

B. Practical Considerations

Some issues may arise from this approach. For example, how many moments should we compute with respect to each parameter? With respect to this issue, we advocate to generate the first order moments with respect to each parameter, including the frequency, for several reasons: first of all, matching more moments or including cross derivatives will quickly increase the number of vectors in the subspace, with the consequent extra cost in the orthonormalization steps. Also, the parameters may be quite different, and thus we may need high order with respect to some of them, and low order for others. To determine these settings a-priori is a difficult task. Including cross derivatives would lead to more complicated implementations that may lead to efficiency loss. Furthermore, the effect captured by those higher order and cross moments can be addressed by the multi-dimensional sampling, which selects expansion points in areas poorly modeled, providing new vectors that will improve the global accuracy.

To generate first order moments with respect to each parameter is a straightforward and simple approach, that may help in some scenarios to capture linear information around the sampling point, with little extra cost, and avoiding to determine complicated settings for the methodology. This could also allow us to reduce the discretization of the space of interest using a sparser, therefore smaller candidate set, as we are capturing not only information at the point, but also on the nearby region. Systems in which the $LU$ factorization of the matrix is very expensive may benefit from this approach, since by generating more vectors at each point it is likely that we reduce the final number of points to sample.

Another issue is how this step modifies the stopping criteria. We propose to maintain the same criteria as in [14], although the norm to check should be the norm of the zero order moment, which is the one that contains the energetic information of the system at the sampling point. Furthermore, vectors related to higher order moments may suffer from deflation, leading to false information in terms of error convergence.

C. Domain Partition

An important feature of the methodology is the inherent parallelism and incrementality, which provides us with a large degree of flexibility. An interesting use of this characteristic is the possibility of dividing the domain of the multi-dimensional space of interest into several subdomains (i.e. the global initial candidate set into several local disjoint smaller candidate sets). The procedure can be applied on these disjoint regions in parallel, so that each independent procedure retrieves a projector which spans the subspace of interest for its region. After all the independent procedures are performed, an overall orthonormalization can be applied to get rid of redundant information and generate the global projector. The advantage of this approach is the reduction of the number of initial candidate samples at each subregion and the possibility of almost perfect parallelization.

As an illustration, if at we have a basis $V$ of 20 vectors and a candidate set of 100 points, in the worst case we have to perform the operation in (16) 100 times (one per candidate), with a cost of $100 \times O(20^2 n) = 100 \times O(400n)$, for a size $n$ of the system. But if we divide the domain in 4 regions, each with 25 candidates, and we have (in the perfectly balanced case) 5 vectors in each region, the cost of finding the next candidate at each region is $25 \times O(5^2 n) = 25 \times O(25n)$. On a single machine, running sequentially, the cost of finding the next 4 points (one at each domain) is $100 \times O(25n)$. But if we have 4 machines, we can do the operation in parallel to generate the 4 points, at cost of $25 \times O(25n)$ on each machine. In this scenario, solving can also be done in parallel, largely improving efficiency. This is a major efficiency boosting in the methodology, theoretically going from a cost of $100 \times O(400n)$ per point to a cost of $25 \times O(25n)$ per 4 points on 4 machines. On the other hand, the more regions we divide the domain in, the more chances we have of generating redundant information, and the less optimal the procedure is. Nonetheless, there is plenty of potential in this approach, and a good trade off can lead to fast reductions.

IV. Simulation Results

In this section we present the capabilities of the proposed algorithm on two different benchmarks. The figure of merit will be the maximum error of the reduced model with respect to the original parameterized model transfer function, in a fairly exhaustive Monte Carlo (MC) simulation covering the region of interest. The algorithms to test will be [8] based on Random and Uniform Sampling, [14] (denoted as ARMS), and the proposed methodology, HORUS.

A. RLC example

The first example is a 2-port 6002-states distributed RLC model of two coupled lossy lines, which depends on 6 geometric parameters (thus the parameter dimension is 7 if we take the frequency into account).

We set an initial candidate set of 3645 points, and generate several reduced models with ARMS and HORUS, with different number of samples. The results of the MC simulation using 112500 points is given in Table I, along with the ROM characteristics. It can be seen that the HORUS model generated with 3 and 5 samples provides good accuracy. On the other hand, ARMS models with 3 and 5 samples provide a worst approximation. In order to achieve a similar accuracy, ARMS needs to solve 12 and 18 samples, respectively. However, for the same accuracy, the size of the models generated by ARMS is smaller. This can be understood since HORUS generates moments to expand the subspace, which are mathematical approximations of the behavior of the system, whereas ARMS only generates the zero order moments, which contain true physical information of the system.
TABLE I
6002-STATES COUPLED LOSSY LINES: pMOR FEATURES

<table>
<thead>
<tr>
<th>ROM</th>
<th># Initial Samples</th>
<th># Solved Samples</th>
<th>ROM Size</th>
<th>Maximum Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARMS</td>
<td>3645</td>
<td>5</td>
<td>12</td>
<td>6.717e-2</td>
</tr>
<tr>
<td></td>
<td>3645</td>
<td>5</td>
<td>20</td>
<td>2.900e-2</td>
</tr>
<tr>
<td></td>
<td>3645</td>
<td>12</td>
<td>48</td>
<td>9.017e-3</td>
</tr>
<tr>
<td></td>
<td>3645</td>
<td>18</td>
<td>72</td>
<td>1.032e-3</td>
</tr>
<tr>
<td>HORUS</td>
<td>3645</td>
<td>3</td>
<td>53</td>
<td>9.327e-3</td>
</tr>
<tr>
<td></td>
<td>3645</td>
<td>5</td>
<td>78</td>
<td>1.004e-3</td>
</tr>
</tbody>
</table>

Fig. 1. SPIRAL and CMIM EM example: MC simulation on 1500 points in a local region. 3D plots represent the relative error of ROMs w.r.t. parameterized original model. From top to bottom, ARMS with 6 samples, ARMS with 18 samples, and HORUS with 6 samples (initial set of 50 samples).

B. EM example

Our second benchmark is an EM model of an industrial integrated planar spiral connected to a Metal-Insulator-Metal capacitor. It has 11005 states and depends on 3 parameter dimensions: the frequency and two very different parameters, one modeling the width of the capacitor insulator, and the other a design parameter modeling the length of the side of the spiral square (with quite a large range of variation).

First we show the improvement in overall approximation provided by HORUS. We take a smaller region depending on the frequency and the spiral parameter, and we cover it with an initial set of 50 samples. With this initial set, we apply ARMS and HORUS, and show a 3-D plot of the relative error with respect to the original parameterized model (Figure 1) in a MC simulation of 1500 points in this local region. The ROMs shown are ARMS, generated with 6 (size 12) and 18 samples (size 36), and HORUS generated with 6 samples (size 36 also). It can be seen that in the case of HORUS, for smaller number of samples, the extended subspace generated by the low order moments allows a much better local approximation. On the other hand, ARMS needs a larger number of samples to accurately approximate the region, although for equivalent size, the AMRS model is more accurate.

Second we compare the Random, Uniform Sampling, and ARMS with HORUS (for a single domain and a domain partition into 4 sub-domains). HORUS, with first order moments, is applied with an initial candidate set with 600 points. It takes 39 samples to yield a 309-states ROM. Its maximum relative error with respect to the original parameterized system for the MC simulation using 7700 samples is given in Table II. As a different configuration, we split the initial set in 4 domains, and apply sequentially the HORUS procedure on each of them, with 10 samples on each domain, and then do an overall orthonormalization of the generated subspace to obtain the projector. The speed-up of this approach with respect to single domain HORUS is 14X. Notice that the algorithms are prototypes implemented in MATLAB, which shares compiled and non-compiled routines. Also, the speed-up is very dependent on the example and the implementation, so this speed-up should not be taken literally. However, it gives an indication of the potential and possibilities of these schemes.

To compare the accuracy, ARMS is also applied with the same initial set. It requires 145 samples to yield a 290-states ROM, whose associate subspace captures the behavior of the parameterized system. Therefore, in order to obtain a subspace with accuracy comparable to HORUS it requires to solve 145 samples, which is far more expensive than the HORUS approach (see Table II). For comparison purposes, results of ROMs obtained with Random and Uniform sampling with 155 samples, which generate 310-state ROMs, are also presented. It is clear from Table II that although the size of the ROMs for the Random and Uniform approaches is the same as HORUS,
the accuracy is worst. This is caused by the absence of a good sample selection scheme, leading to a low level of confidence in the model. Figure 2 shows the distribution of the relative errors for the complete MC simulation. It is clear that Uniform and Random sampling schemes perform oversampling (which leads to very small errors) in some regions, and undersampling in others (with large error). ARMS and HORUS concentrate samples in bad approximated regions, and thus do a better job in maintaining a lower maximum error.

V. CONCLUSIONS

We have introduced a multi-point multi-dimensional linear moment-matching approach combined with an automatic residue minimization sampling scheme for model order reduction of parameterized models. The sampling scheme leads the samples to poorly approximated regions, whereas the low order moment generation efficiently generates an extended subspace that improves the local approximation around the sample. The advantages are twofold. On one hand we need to perform a lower number of samples for a desired accuracy, and thus we reduce the operations related to the most costly operation in multi-point approaches, i.e. the system solve. On the other hand, the improved local approximation allows for a reduction of initial candidates in the residue minimization sampling scheme, boasting the efficiency of the algorithm, in particular in the case of high parameter-dimensions.

We have also shown the possibility of improving the efficiency by splitting the domain of samples in disjoint regions (largely reducing the candidates in each region), in which the procedure is applied locally to generate a local projector. This subspaces can be combined into global subspace via an overall orthonormalization. The efficiency of the algorithm is largely improved, allowing for a quasi-perfect parallelization.

ACKNOWLEDGMENT

This work was supported by NXP Semiconductors Research under the EMOnIC project.

REFERENCES