Systematic Upscaling for Feynman Path Integrals

A Progress Report

Mark Zlochin and Achi Brandt

The Weizmann Institute of Science, Rehovot 76100, Israel

November 24, 2005

1 Path Integrals

1.1 The basic setup

The path integral approach was introduced by Feynman in his seminal paper (Feynman, 1948). It provides an alternative formulation of time-dependent quantum mechanics, equivalent to that of Schrödinger. Since its inception, the path integral has found innumerable applications in many areas of physics and chemistry. The reasons for its popularity are numerous. First, the path integral formulation offers a straightforward way of obtaining the classical limit of quantum mechanics. In addition, it provides a unified description of quantum dynamics and equilibrium quantum statistical mechanics. Finally, it avoids the use of wave functions and thus is often the only viable approach to many-body problems.

The path integral formulation builds on the principle of superposition, which leads to the celebrated quantum interference observed in the microscopic world. Thus, the amplitude for making a transition between two states is given by the sum of amplitudes along all possible paths connecting these states in the specified time. Specifically, the amplitude to get from a [vector valued] position x_a at the time t_a to the position x_b at the time t_b is expressed, in the path integral formulation, as a sum of contributions from all conceivable paths that connect these points. The contribution of each path $x(t)$ is proportional to a phase that is given by the action functional $S[x(t)]$ along that path in units of Planck's constant h:

$$K(x_a,t_a;x_b,t_b) \;\propto \sum_{\substack{x(t)\in C([t_a,t_b]) \\ x(t_a)=x_a, x(t_b)=x_b}} \exp\{iS[x(t)]/h\} \tag{1}$$

In practice, the paths are discretized using the Trotter theorem (Trotter, 1959), leading to the following

approximate path integral expression for the propagator:

$$K(x_a,t_a;x_b,t_b) = \left(\frac{mn}{2\pi i h(t_b-t_a)}\right)\int dx_1 \ldots \int dx_{n-1} \\ \times \exp\left\{\frac{i}{h}\frac{mn}{2(t_b-t_a)}\sum_{k=1}^{n}(x_k-x_{k-1})^2 - \frac{i}{h}\frac{t_b-t_a}{2n}\sum_{k=1}^{n}\left[V(x_k)+V(x_{k-1})\right]\right\}, \tag{2}$$

which becomes an equality in the limit $n \to \infty$. The exponent in the last expression is easily recognized as the trapezoid rule discretization of the action in Eq. (1).

Because the quantum mechanical time evolution operator $\exp(-iH(t_b - t_a)/h)$ has the same mathematical form as the Boltzmann operator $\rho = \exp(-\beta H)$, where $\beta = (k_B T)^{-1}$ is the inverse temperature in units of the Boltzmann constant k_B, the above path integral formalism can be straightforwardly generalized to yield equilibrium properties in the canonical ensemble (Feynman, 1972). Making the formal identification $t_b - t_a = -ih\beta$, the canonical density matrix is given by the following "imaginary time" path integral expression:

$$\rho(x_b,x_a) = \langle x_b \mid \exp(-\beta H) \mid x_a\rangle = \lim_{n\to\infty}\left(\frac{mn}{2\pi h^2\beta}\right)^{n/2}\int dx_1 \ldots \int dx_{n-1} \\ \times \exp\left\{-\frac{mn}{2h^2\beta}\sum_{k=1}^{n}(x_k-x_{k-1})^2 - \frac{\beta}{2n}\sum_{k=1}^{n}\left[V(x_k)+V(x_{k-1})\right]\right\} \tag{3}$$

Although identical in structure to Eq. 2 for the real time propagator, the path integral representation of the canonical density matrix involves a real-valued integrand, in which different paths enter with different weights. These features are extremely useful in numerical calculations as discussed in the next section.

In addition to calculation of purely dynamical quantities using the real-time propagators on one hand, and the equilibrium calculations using the imaginary-time expressions just described, path integrals can also be used to compute thermally averaged quantum time correlation functions (Thirumalai and Bern, 1983) and thus information about dynamical effects at finite temperature. It is often convenient from a computational point of view to express a time correlation function $G_{AB}(t)$ in a symmetrized form (Thirumalai and Bern, 1984) by introducing the complex time $t_c = t - ih\beta/2$:

$$G_{AB}(t) = Tr[A\exp(iHt_c^*/h)B\exp(-iHt_c/h)], \tag{4}$$

where A and B are quantum mechanical operators. The complex-time propagators appearing in Eq. 4 can be also represented using path-integrals similar to Eq. (2) and (3).

To summarize, many quantum mechanical problems can be reduced to a calculation of the path integrals of the form:

$$\int dx_1 \ldots \int dx_{n-1} A(\bar{x}) \exp\left\{\frac{i}{h}\frac{m}{2\delta}\sum_{k=1}^{n}(x_k-x_{k-1})^2 - \frac{i}{h}\frac{\delta}{2}\sum_{k=1}^{n}\left[V(x_k)+V(x_{k-1})\right]\right\}, \tag{5}$$

where:

- for real-time propagator, $\delta = \frac{t_b - t_a}{n}$
- for equilibrium calculations, $\delta = \frac{-ih\beta}{n} = \frac{ih}{k_B T n}$
- for symmetrized thermally averaged quantum time correlation functions, $\delta = \frac{t - ih\beta/2}{n}$

Within the path integral formulation, identical particles are dealt with by adding (or subtracting) the appropriate amplitudes corresponding to particle permutations in a way similar to that employed in the symmetrization (or antisymmetrization) of wave functions. This procedure introduces negative amplitudes in finite temperature calculations of many-fermion systems, leading to significant numerical difficulties, which are addressed below.

1.2 Numerical methods

1.2.1 Equilibrium properties

The discretized path integral expression of the canonical density matrix leads to extremely useful numerical algorithms for evaluating finite temperature properties of many particle spinless or boson systems not treatable by other techniques. The required multidimensional integrals are evaluated by Monte Carlo methods, which have been reviewed extensively elsewhere (see e.g., (Ceperley, 1995)). Here we outline the general features of such calculations, using as an example the Boltzmann average of a quantum mechanical operator A:

$$\langle A \rangle = \frac{\mathrm{Tr}(\exp(-\beta H)A)}{\mathrm{Tr}(\exp(-\beta H))} = \frac{\int dx \langle x \mid \exp(-\beta H)A \mid x \rangle}{\int dx \langle x \mid \exp(-\beta H) \mid x \rangle} \tag{6}$$

Assuming that the operator A is diagonal in the coordinate representation and using the path integral representation (3) of the canonical density matrix, the expectation equals to the limit:

$$\langle A \rangle = \lim_{n \to \infty} \frac{\int dx_1 \ldots \int dx_n W(x_1, \ldots, x_n) A(x_n)}{\int dx_1 \ldots \int dx_n W(x_1, \ldots, x_n)}, \tag{7}$$

where

$$W(x_1, \ldots, x_n) = \exp\Big\{ -\frac{mn}{2h^2\beta} \sum_{k=1}^{n} (x_k - x_{k-1})^2 - \frac{\beta}{2n} \sum_{k=1}^{n} [V(x_k) + V(x_{k-1})] \Big\} \tag{8}$$

Here $x_0 = x_n$, that is, the paths involved in the integrals are closed.

The weight function W is positive and, consequently, the ratio of the two integrals above can be calculated with Monte Carlo methods using sampling distribution proportional to W. Note that successive variables (x_{k-1}, x_k) are correlated under W, hence some sort of collective coordinate moves are needed in order to achieve rapid convergence. A variety of methods for improving the sampling efficiency have been proposed in the literature, the most effective being the multilevel sampling (Ceperley, 1995).

The approach outlined above can be easily generalized to bosonic systems, by augmenting the variables $x_1, \ldots, x_n$ with a permutation $\sigma(x_1, \ldots, x_n)$ and integrating over the augmented space. However, when dealing with fermions severe problems arise. As configurations corresponding to identical particle exchange enter with alternating signs, the integrand is no longer positive and the Monte Carlo procedure must be modified. Although positive sampling functions can still be identified, the frequent sign change causes dramatic cancellation and eventually renders Monte Carlo methods unstable. This behavior, known as the "sign problem", continues to plague quantum Monte Carlo simulations of many-electron systems. The Systematic Upscaling approach, described in the following, deals with the sign problem by absorbing the sign effect into the propagator at the time scales where this effect becomes visible.

1.2.2 Dynamical properties

Unlike finite temperature equilibrium properties of spinless or boson systems, which can be evaluated efficiently by Monte Carlo path integral methods, dynamical quantities present a considerable challenge. As all paths carry a complex weight with unity modulus, one needs to sample the entire volume of integration uniformly and importance sampling does not offer an advantage. Most importantly, the rapid phase oscillation of the integrand results in enormous cancellation, which cannot be dealt with by Monte Carlo procedures. This behavior is yet another manifestation of the sign problem and hinders dynamical path integral calculations.

The situation can be improved, for example, by constructing improved propagators which employ appropriate projection operators (Makri, 1991) or bias the sampling near classical paths, where the phase is stationary and therefore phase cancellation is minimal (Filinov, 1986; Makri and Miller, 1987; Doll and Frreeman, 1988). These schemes lead to path integral expressions where the integrand is relatively localized and only mildly oscillatory with respect to each path integral variable. However, the effect of the residual oscillations (which are essential for reproducing quantum interference effects) is amplified in multidimensional space, leading to dramatic cancellation, which renders Monte Carlo schemes inadequate for calculating the dynamics beyond a few time steps. The methods described above can be also applied in the context of thermally averaged quantum time correlation functions, but as the number of the time steps increases, the cancellation effects become destructive, similarly to the real-time propagation.

Let us finally note that including the fermionic effects further increases the sign problem, making the calculation unstable even for a moderate number of time steps.

2 Research motivation

2.1 The scale gap

The difficulties described above are a particularly pronounced example of a much more general problem plaguing many areas of scientific computation. This problem is *the scale gap* between the microscopic level, at which the mathematical description is given and the much larger scale of phenomena we wish to understand. For example, in Eq. (2), the long-time quantum propagator is expressed as a product of the short-time propagators. This introduces a huge number of new variables (e.g., coordinates of the intermediate points), whereas, in fact, we are only interested in the relation of the end-points.

The multiple particle systems exemplify another aspect of the scale gap - the "microscopic description" entails some combination of numerous single-particle descriptions, whereas in practice we are commonly interested not in the effect of each particular particle, but rather in a much higher-level description, such as, for example, electronic density (Kohn and Sham, 1965), in which averaging over the individual particles and/or their permutations is taking place.

In addition to introduction of a huge number of microscopic variables and possibly even a much larger number of interactions (e.g., one force between every pair of particles), one has to deal with some purely computational problems. Specifically, computers simulate physical systems by moving few variables at a time; as a result, each such move must be extremely small, since a larger move would have to take into account all the motions that should in parallel be performed by all other variables. Such a computer simulation is particularly incapable of moving the system across large-scale energy barriers, which can be crossed only by a large coherent motion of many variables. These obstacles make it impossible to carry out realistic calculations of the properties of elementary particles, atomic nuclei, etc., or, in other words, to *computerize chemistry and materials science*, so as to enable the design of materials, drugs and processes, with enormous potential benefits for medicine, biotechnology, nanotechnology, agriculture, materials science, industrial processing, etc. With the computational methods currently in use, the needed amount of computer processing often increases so steeply with the problem size, that in many cases even much faster computers will not do.

2.2 Multigrid methods

Past studies have demonstrated that scale-born slowness can be overcome by multiscale algorithms. Such algorithms have first been developed in the form of fast $multigrid\ solvers$ for discretized PDEs (Brandt,

1977; Brandt, 1982; Hackbusch, 1985; Briggs et al., 2000; Trottenberg et al., 2000). These solvers are based on two processes: (1) classical $relaxation$ schemes, which are generally slow to converge but fast to smooth the error function; (2) approximating the smooth error on a $coarser\ grid$ (typically having twice the meshsize), by solving there equations which are derived from the PDE and from the fine-grid residuals; the solution of these coarse-grid equations is obtained by using recursively the same two processes. As a result, large scale changes are effectively calculated on correspondingly coarse grids, based on information gathered from finer grids. Such multigrid solvers yield $linear\ complexity$ (solution work proportional to the number of unknowns). Moreover, since the local processing (relaxation, etc.) in each scale can be done in parallel at all points of the domain, the multiscale algorithms, based on such processing, proved ideal for implementation on massively parallel computers.

In many years of research, the multigrid methodology has been extended to cover most major types of linear and nonlinear large systems of equations appearing in sciences and engineering. The new developments include, for instance, grid-free solvers, called $algebraic\ multigrid$ (AMG; see (Brandt et al., 1982; Brandt, 1986; Ruge and Stüben, 1987)), non-deterministic statistical mechanics problems ((Brandt et al., 1986; Goodman and Sokal, 1986; Brandt, 1992; Brandt et al., 1994)) and multiple coarse-level representations (Brandt and Livshits, 1997).

To obtain even further generality, there emerged however two basic reasons to go much beyond these multigrid methods. First, they do not perform well for $highly\ nonlinear\ cases$, where configurations cannot be decomposed into weakly-interacting local and non-local parts. Second, for many systems, even linear complexity is not good enough, since the number of variables is huge. Such systems on the other hand are typically $highly\ repetitive$, in the sense that the same small set of governing equations (or Hamiltonian terms) keep repeating itself throughout the physical domain. This opens the way to the possibility of having, at the coarse level too, a small set of governing equations that are valid everywhere, and that can be derived from fine-level processing conducted only in some small representative "windows" (see below).

These two basic reasons point in fact in the same direction. Instead of relaxing the given system of equations so as to obtain a smooth error that can be approximated on a coarse level, one should use coarse level variables that represent the $full$ solution rather than the correction to any given current approximation. Such coarse variables can be chosen (as described below) so that the coarse-level equations can be derived just by local processing. We use the term "$upscaling$" for this type of direct (full-solution) transition from a fine level to a coarser one. Such a transition is valid even in those highly nonlinear cases, where all scales interact with each other so strongly that correction-based multileveling is inapplicable. These include, for

example, an Ising spin model (Brandt and Ron, 2001), simple fluids (Brandt and Ilyin, 2001) and polymer models (Bai and Brandt, 2000; Bai, 2004).

In the following we generalize these ideas under a single methodological approach named *Systematic Upscaling*.

2.3 Systematic Upscaling (SU): An outline

$Local\ equations\ and\ interactions$. Computationally, we deal only with discrete systems; their n variables $x_1, x_2, \ldots, x_n$ will typically be either values of discretized functions or locations of particles. An equation or interaction is called $local$, if it involves only $O(1)$ neighboring variables. In the following, we limit the discussion to the case of local interactions, as those appearing in Equation (5), although generalization to the non-local case is possible. Also the discussion is restricted to the stochastic problems, with the deterministic ones being regarded as a zero-stochasticity limit.

$Coarsening$. Similarly to multigrid, SU is based on two processes: The usual local processing (using appropriate Monte Carlo method) and repeated coarsening, creating increasingly coarser descriptions of the same physical system. At each coarsening stage, to each fine-level configuration $x = (x_1, \ldots, x_n)$ one defines a unique coarse-level configuration, denoted by $x^c = (x^c_1, \ldots, x^c_m)$, which is a vector with a reduced number of variables.

$Interpolation$. To any given coarse configuration $X = (X_1, \ldots, X_m)$, there are of course many fine-level configurations x, which are *compatible* with X. The interpolation (transition from X to a specific configuration x) is created by $compatible\ Monte\ Carlo$ (CMC), that is by a local processing, restricted to configurations compatible with X. The interpolation is completed once the CMC has reached its equilibrium.

$The\ general\ coarsening\ criterion$. The fine-to-coarse transformation $x \to x^c$ (and, in particular, the choice of coarse variables) is said to be $adequate$ if (and only to the extent that) the compatible Monte Carlo equilibrates fast.

A major problem in coarsening any system is to find a suitable set of coarse variables. The above criterion gives a general and very effective tool for developing such a set. The adequacy of that set implies essentially local dependence of the configuration only on coarse variables, and hence the feasibility to construct, just by local processing, a set of coarse interactions, that is, a Hamiltonian-like functional of the coarse variables, $H^c(x^c)$, that will govern simulations at the coarse level, namely, such that $Prob(x^c) \propto \exp(-H^c(x^c))$.

The actual derivations of such a coarse functional H^c is done by expressing H^c in the form $\sum a_j H_j^c$, where H_j^c are simple known local interactions and a_j are to-be-determined coefficients. Similarly to the group renormalization techniques (Wilson, 1983; Fisher, 1998), the coefficients are found by iterative comparisons of a sequence of averages (such as $< H_l^c >$) calculated through simulations on the coarse level with corresponding averages calculated at the fine level. The comparisons also suggest what new interactions H_j^c should be added to H^c to obtain better agreement (as in our work on upscaling polymer models (Bai and Brandt, 2000) described also in (Brandt, 2001)). Our previous experience suggests that the errors introduced in the Systematic Upscaling of a system are fully controllable; as the number of terms H_j^c in the coarse functional increases, the errors typically decay exponentially, whereas the amount of invested computation increases only algebraically.

In highly repetitive systems (defined above), the local processing need not be done everywhere: the coarse-level equations can iteratively be derived by comparing coarse-level with fine level simulations, where the latter are performed only in some relatively small $windows$ (subdomains, on the boundaries of which the fine level is kept compatible with the coarse level). New separate windows need only be opened at regions of different regional conditions (i.e., such that simulations in one region are not likely to sample local situations encountered in the other region).

In particular, separate windows should be opened at different *attraction basins* of the investigated system. By building a common Hamiltonian (through adding to H^c terms sensitive to changing the basin), the possibility for easy Monte Carlo transition between the basins will emerge at coarser levels, enabling accurate statistics of their relative probabilities.

Thus, the fine level simulations supply the operational rules (the Hamiltonian-like functional) for the next coarser level, while the latter supplies the windows for those simulations. Iterating back and forth between all the levels quickly settles into a self-consistent multilevel compatibility. If the coarsening ration n/n_c is not large, no slowdown should occur, and, at each level, the computations need extend only over a collection of representative windows, whose number depends on the diversity of regional conditions, not on the size of the problem.

$Long\ range\ interactions$ (e.g., between electrostatic charges) can each be decomposed into a smooth interaction and a local one ("smooth" and "local" being meant on the scale of the next coarse level; all familiar physical interactions can be decomposed this way (Brandt, 1991)). The smooth part can directly be transferred to the coarse level (e.g., smoothness allows accurately replacing the fine-level electrostatic charges by aggregated charges and dipoles), while the local part is coarsened, together with all other local

interactions, using the fine/coarse iterations described above.[1]

The SU approach produces *a direct macro-level description* of the system in question. That is, it leads to a formulation of new "physical laws" (such as H^c) describing *the physical phenomena of interest on the scale of interest*. Upscaling does not solve an isolated problem, but derives coarse equations for a wide family of situations. When needed, the family is enlarged by opening new windows. The upscaling accuracy can be fully measured and controlled by comparing the fine and the coarse level averages of a sequence of test functionals for each pair of successive levels.

2.4 Related work

The very idea of using multiscale methods in the context of path integral calculation is not entirely new. Below we review some important contributions in this directions and discuss their limitations and differences, as compared to the "Systematic Upscaling" paradigm.

First, within the MG community, several multigrid Monte Carlo methods have been proposed in the past (Mack and Pordt, 1985; Goodman and Sokal, 1986; Brandt et al., 1986; Brandt et al., 1994). Independently, Ceperley and co-workers have developed the "multilevel sampling" (Ceperley and Pollock, 1986; Ceperley, 1995; Ceperley, 2003). This approach uses a multilevel trial distribution, which is incorporated in the Metropolis-like Monte Carlo update. The trial paths are generated using a "bisection method", which first generates new midpoints for paths and then new midpoints for remaining halves and so on, with the possibility of rejecting the new paths at any stage in the construction. The bisection method insures that the improbable paths may be rejected at an early stage of their construction, hence allowing many more trial moves for a given amount of computer time. The rejection step ensures that the accepted paths reflect the correct density matrix. The conditional distribution of the midpoint is approximated by a multivariate Gaussian, whose parameters are chosen to approximate the moments of the exact conditional distribution. While producing some impressive results, the multiscale sampling suffers from an inherent drawback, stemming from the reliance on the Gaussian approximations - clearly if the conditional distributions are not well approximated by Gaussians, the algorithm will loose its effectiveness.

A different approach was employed in (Janke and Sauer, 1993), where a unigrid version of MG algorithm was implemented for the ϕ^4 model. While not relying on approximations, the unigrid implementation (i.e.,

[1]The advantage of such a smooth decomposition is that fine scale MC movements can be designed (distributive movements, as in (Brandt and Lubrecht, 1990)), such that they are hardly affected by the non-local part. As a result, only rare rejections will occur when the result of a series of fine-scale MC movements is subjected to a coarse level Metropolis test to obtain the exact detailed balance.

actually performing the fine-level moves corresponding to the coarse-level moves) can be quite demanding computationally for problems of considerable size. It is important to note that, in the cited article, the authors were forced to use the unigrid approach, since they had no way to directly estimate the coarse level Hamiltonian. Automatic construction of such estimates is at the core of the SU methodology described in Section 3.

While the two methods cited above did make some use of the multigrid methodology, the final calculation was still performed on the finest level. Though beneficial for bosonic imaginary-time path integrals, where those methods allow to reduce the sampling slowdown, they are not particularly useful for fermionic imaginary-time and general complex-time path integrals, where the main problem is the sign-cancelation.

A different approach, which is closer in spirit to the SU methodology, is the "multilevel blocking" algorithm (Mak and Egger, 1999). In this method, the final calculation is transferred to the coarsest level by "integrating out" the intermediate points in a multiscale manner. The crucial part is the exact mechanism for carrying out this integration - in the multilevel blocking, the integrals are approximated by weighted Monte Carlo averages based on a large fine-level sample. This point is, in our view, the main limitation of the multilevel blocking, since it could be expected that the number of the fine-level points used for "integrating out" should scale exponentially with the number of particles involved, hence making the method unsuitable for any, but really small problems.

The SU methodology described next bridges the gap between the methods based on parameterized approximations (as in "multilevel sampling") and the "non-parametric" methods like the "multilevel blocking". The solution is the use of flexible extensible semi-parametric approximations, which allow high-accuracy estimates, while maintaining reasonable computational costs.

3 Systematic Upscaling for Path Integrals

3.1 The Multiscale Structure

The general SU principles surveyed above can beautifully be adapted to path integrals, including even the more difficult case of *real-time* path integrals of *fermions*, which we outline next.

The starting point of our methodology is the existence of natural multiscale structures in the problem. First, obviously, the time dimension has a natural scale hierarchy, with the time step size as a scale parameter. The fine level steps are short enough to facilitate the use of the trapezoid rule in Eq. 5, and the coarsening amounts to increasing (e.g., doubling) the time step.

The coarsening of the time step could be accompanied by spatial coarsening, because at long time intervals individual particle location (or pointwise phase values) are hardly correlated. A natural spatial coarsening is by Gaussian averaging of particle locations, forming a weighted aggregation of the paths. As shown in (Filinov, 1986), such an averaging (without which the path summations are not even well defined) integrates over many mutually canceling contributions of neighboring paths and produces a real and positive weight for each aggregated path (similar to $W(x)$ in (5)). These weights, which correctly locally prefer paths closer to the classical one, make the system suitable for Monte-Carlo simulations that can be used for the SU local processing. It can be shown that the width of the Gaussian should increase proportionally to the time step in order to maintain the temporal locality of the interactions.

What is missing in (Filinov, 1986) and other related works are the multiscale aspects of the spatial averaging. In fact, Filinov's procedure can be regarded as a one-step coarsening, whereas in SU we develop the underlying ideas to their full realization in a truly multiscale manner. With each additional time-step coarsening, further corresponding averaging of paths takes place. Due to the possibility of restricting the derivation of coarsening to representative windows at each level (see below), it can be done *purely numerically*. It is worth noting that the restriction to one-step coarsening was not the only reason for the very modest performance of the previous approaches. A more serious limitation of these methods was the use of the *analytical* approximation to the coarse-level functional, which requires sufficiently small averaging width, in order to maintain reasonable approximation accuracy. In SU this limitation is alleviated by the numerical approximation methodology described above.

We will apply the general coarsening criterion cited above to design at each level an adequate set of variables. As noted in (Makri and Miller, 1987), one should not limit the averaging only to the path intermediate points $x_1, x_2, \ldots, x_{n-1}$, but also to the end points taking into account the initial (final) wave function $\Psi(x_0)$ ($\Psi(x_n)$). However, for problems with pronounced momentum (e.g., nearly classical particles), using this position-space averaging introduces spurious correlations, hence destroying the locality of the interactions. A more intelligent coarsening is obtained, if one uses coherent-states representation of the path-integral (Klauder and Skagerstam, 1985) and performs coarsening by Gaussian averaging in the combined position-momentum space. For simplicity, this type of situation is ignored in the description below.

At coarser levels, other types of variables will enter. Before discussing that, however, it is important to state the effectively local nature of upscaling here. At each level l the description will be given as an integral

of the form

$$\langle A\rangle = \int dx_0^{(l)} \int dx_1^{(l)} \ldots \int dx_{n_l}^{(l)} A(x_1^{(l)}, x_{n_l}^{(l)}) \exp\Big(\sum_{k=1}^{n_l-1} S^{(l)}[t_k^{(l)}]\Big), \tag{9}$$

where $x_k^{(l)}$ is the average path at time $t_k^{(l)} = t_0^{(l)} + 2^l\delta k$, and the functional $S^{(l)}$ is to be structured and computed similarly to H^c above. For any time $t_k^{(l)}$, the functional $S^{(l)}[t_k^{(l)}]$ depends on $x_k^{(l)}$ and several neighboring $x_j^{(l)}$. At the finest level, $S^{(0)}[t_k^{(0)}] = S(x_k, x_{k-1})$, as in (2) above.

Due to the locality and universality of the interactions, the construction of $S^{(l)}$ given $S^{(l-1)}$ can be done in small representative windows, i.e., employing relatively short paths. *Even in the case of many particles*, the functional $S^{(l)}$ is essentially a sum of interactions that are local not only in time, but also in space (after transferring a smooth non-local part directly to the coarse level, as discussed in Sec. 2.3 above). Hence the representative windows can be local in space too.

At coarser levels, where the spatial scale becomes comparable to the inter-particle distances, new variables and processes should take place. Whereas at finer levels averages are taken over individual particles, they should now involve the mutual density of several neighboring particles, the more of them the coarser the level. Other types of coarse variables will enter, such as averages of moments of interparticle distances. Eventually, at sufficiently coarse level, variables should emerge that depend only on particle densities (e.g., the electronic density, in the Born-Oppenheimer approximation) and their derivatives.

Also, exactly at such time and space scales where particle average locations start to mix, and only around those scales, *particle exchange* should be taken into account. At finer scales the exchange is unlikely, while in the long run exchanges are just the sum result of multiple neighbor exchanges at the mixing scales. As in other cases of different attraction basins (see Sec. 2.3 above), building at such scales common $S^{(l)}$ functionals for different permutations of particles is expected to give a very efficient way to incorporate the required exchanges.

The separation between local and smooth interactions is expected to persist to the level of particle densities, resulting there in *multiscale density functionals* (effectively incorporating all energy terms, including the kinetic energy and the exchange correlation). By this we mean functionals that sum interactions between *neighboring average densities* (neighboring at the scale of the averaging) over a sequence of scales. In terms of such functionals, concrete problems can be solved fast by multiscale solvers.

In the research results reported below we limit the discussion to the to the time-step-size multilevel structure, though in the future we plan to study its combination with the space/permutation averaging described above. Also we assume that the number of time steps at level $l = 0, \ldots, L$ is 2^{L-l}. Hence $l = 0$ corresponds to the finest level and $l = L$ to the coarsest one.

As a next step, one needs to specify the procedure for the fine-to-coarse transformation of the corresponding equations. Recall (see Eq. (5)) that the finest level description is given as an integral in the form

$$\langle A\rangle = \int dx_1 \int dx_2 \ldots \int dx_n A(x_1, x_n) \exp\Big(\sum_{k=1}^{n-1} S(x_{k-1}, x_k)\Big), \tag{10}$$

where $n = 2^L + 1$, $A(x_1, x_n)$ is some function of the end-points, and

$$S(x_{k-1}, x_k) = \frac{i}{h}\Big(\frac{m}{2\delta}(x_k - x_{k-1})^2 - \frac{\delta}{2}[V(x_k) + V(x_{k-1})]\Big). \tag{11}$$

It can be easily verified that the same form also holds for any intermediate levels, that is, for any l:

$$\langle A\rangle = \int dx_1 \int dx_{1+2^l} \ldots \int dx_n A(x_1, x_n) \exp\Big(\sum_{k=1,1+2^l,\ldots,n} S^{(l)}(x_{k-2^l}, x_k)\Big), \tag{12}$$

that is, the interaction on the intermediate levels remain local. Hence, if we could express $S^{(l)}$ in a closed form (or, at least, find a good approximation thereof), the calculation of $\langle A\rangle$ could be carried out on the level l, without the need to use the finer (hence, computationally more expensive) levels.

To complete the description of the Path Integral Systematic Upscaling, we need to specify a mechanism for approximating $S^{(l)}$. Specifically, we approximate $S^{(l)}$ by a finite sum $\sum c_j^{(l)} S_j^{(l)}$, where $\{S_j^{(l)}\}$ belong to some complete function family, as described in the next section.

3.2 Experimental design & methods

3.2.1 Estimation of coefficients

In section 2.3, we have briefly mentioned the general ideas behind our approximation methodology. In the following, a more detailed description is given.

Given a set of basis functions $\{S_j^{(l)}\}$, the coefficients $c_j^{(l)}$ can be determined using a generalization of the maximal entropy (ME) method (Jaynes, 1957). For probability distributions parameterized by a vector $\bar{c}$, the ME parameter estimate is the one, which maximizes the relative entropy:

$$\langle \log p_{\bar{c}}\rangle^{target} = \langle \sum_j c_j S_j^{(l)} - n(\bar{c})\rangle^{target} = \sum_j c_j \langle S_j^{(l)}\rangle^{target} - n(\bar{c}), \tag{13}$$

where $\langle\rangle^{target}$ denotes expectation with respect to the target distributions and $n(\bar{c}) = \log \int \exp(\sum_{j'} c_{j'} S_{j'}^{(l)})$. A common approach for finding the optimal ME parameters is by differentiating the relative entropy with respect to the parameters and equating the derivatives to zero. This leads to the following system of non-linear equations:

$$\langle S_j^{(l)}\rangle^{fine} = \langle S_j^{(l)}\rangle^{coarse} \tag{14}$$

For example, one may tackle the later problem in an iterative manner using local linearization - a method known in statistics as Fisher scoring (Rao, 1948). A simplified version of the Fisher scoring was successfully used in the systematic upscaling of the polymer models (Bai and Brandt, 2000).

While the ME method was originally designed for probability distributions, the idea of using Equation (14) can be naturally generalized to complex measures that appear in path integral calculations. Specifically, given an integrable complex measure ψ, we define an average with respect to ψ as:

$$< A >_{\psi} = \frac{\int A d\psi}{\int d|\psi|}. \tag{15}$$

In the following, we also present the generalized Fisher Scoring method for the complex measures.

As already stated earlier, there is no need to simulate the full system in order to estimate the coefficients. In fact, for the path integrals problems, the strictly local character of the interactions allows working with a window consisting of just two fine-level time steps and one coarse level step. In this case,

$$\begin{aligned} \langle S_j^{(l)} \rangle^{fine} &= \int S_j^{(l)}(x,z) \exp\Big(\textstyle\sum_{j'} c_{j'}^{(l-1)} [S_{j'}^{(l-1)}(x,y) + S_{j'}^{(l-1)}(y,z)] \Big) dy \, d\mu(x,z) \\ \langle S_j^{(l)} \rangle^{coarse} &= \int S_j^{(l)}(x,z) \exp\Big(\textstyle\sum_{j'} c_{j'}^{(l)} S_{j'}^{(l)}(x,z) \Big) d\mu(x,z) \end{aligned} \tag{16}$$

where $x = x_1, y = x_{1+2^{l-1}}, z = x_{1+2^l}$.

The measure $\mu(x,z)$ specifies the window boundary conditions - as a rule it should be chosen so as to give more weight to the areas from which the main contribution to the final integral (12) comes, and can be changed adaptively based on the results of the coarse-level simulation. It is important to note that, once we consider the more elaborate multiscale structures involving spatial averaging, the measure μ may become dependent on the location of the window and should incorporate information about the variables outside the window.

3.2.2 Choice of the basis functions

As we have seen in the introduction, for the finest level, the function $S^{(0)}(x,z)$ is a linear combination of $(x-z)^2, V(x)$ and $V(z)$. Hence it is natural to include these functions in the basis set. Naturally, as we ascend in the multiscale hierarchy, this restricted basis is not sufficient (apart from the trivial harmonic case) and should be augmented with additional basis functions, as described next.

Starting from the basis, spanning the space of the quadratic polynomials of x, z together with $V(x)$ and $V(z)$, we estimate the coefficients as described in the previous section. Next, in addition to the existing basis functions, we also consider their "refinements", that is the function multiplied by a tent function $T(x,z)$, whose support is limited to a square. If the approximation error for the average of the refinement is on

the same scale as the errors for the functions already in the basis set, the refinement is also added to the basis set. As the algorithm proceeds, it considers refinements with increasingly smaller supports, until some prescribed level of precision is achieved.

A somewhat different approach was employed in (Bai and Brandt, 2000; Bai, 2004), where the candidates for addition to the basis set were chosen from the correlations among the functions already in the basis as well as single-variable refinements. The advantage of this approach is that the list of candidates is smaller at each step, especially when x and z are higher dimensional, and the correlations are calculated anyway as a part of the "information matrix". However, in this work we did not make use of this alternative approach, since our examples allow the somewhat more intuitive method described above.

3.2.3 Solving the non-linear equations

For the purposes of present project, we have generalized the well-known Fisher scoring (Rao, 1948) algorithm. For simplicity of presentation the level index l is omitted in the following.

In the Fisher Scoring method, given the current parameters vector $\{c^{old}\}$, we seek the change of parameters δc that solves the local linearization of Eq. (14):

$$\langle S_j \rangle^{old} + I \cdot \delta c = \langle S_j \rangle^{coarse}, \tag{17}$$

where $\langle S_j \rangle^{old}$ is the current vector of statistics averages and I is the covariance matrix with entries

$$I_{ij} = \langle S_j S_j \rangle^{old} - \langle S_j \rangle^{old} \langle S_j \rangle^{old}. \tag{18}$$

Note that the equation (17) is equivalent to the following least square optimization problem:

$$\delta c = \text{argmin}\Big\{ \| \langle S_j \rangle^{old} + I \cdot \delta c - \langle S_j \rangle^{coarse} \|^2 \Big\} = \text{argmin}\Big\{ \| I \cdot \delta c - \delta \langle S_j \rangle \|^2 \Big\}, \tag{19}$$

where $\delta \langle S_j \rangle = \langle S_j \rangle^{coarse} - \langle S_j \rangle^{old}$.

Now, in the complex case, the general form remains the same, apart from the form of the matrix I. Let us recall that we consider complex measures of the exponential form:

$$d\psi = \exp(\sum c_j S_j) d\mu = \exp(\sum c_j^{mag} S_j + i \sum c_j^{ph} S_j) d\mu, \tag{20}$$

where c_j^{mag} and c_j^{ph} denote the real component, affecting the magnitude of the measure, and the imaginary component, affecting the phase. Hence, the parameter vector becomes

$$c = \begin{pmatrix} c^{mag} \\ c^{ph} \end{pmatrix} \tag{21}$$

Using our definition (15) of complex expectation, we obtain

$$< S_k >_\psi = \frac{\int S_k \exp(\sum c_j^{mag} S_j + i \sum c_j^{ph} S_j) d\mu}{\int \exp(\sum c_j^{mag} S_j) d\mu}. \tag{22}$$

When differentiating separately with respect to the magnitude and the phase one obtains the following form for the "covariance" matrix I:

$$I = \begin{pmatrix} I^{mag} & I^{ph} \end{pmatrix}, \tag{23}$$

where

$$\begin{aligned} I_{ik}^{mag} &= \frac{\partial < S_i >_\psi}{\partial c_k^{mag}} = < S_i S_k >_\psi - < S_i >_\psi < S_k >_{|\psi|} \\ I_{ik}^{ph} &= \frac{\partial < S_i >_\psi}{\partial c_k^{ph}} = i < S_i S_k >_\psi \end{aligned} \tag{24}$$

It is important to note that although the "generalized covariance" matrix I is complex in this case, the resulting optimization problem is still real-valued, hence well-defined. This can be verified by substituting equations (24) into the optimization problem (19), and obtaining, after some routine calculations, an equivalent simplified form:

$$\text{argmin} \Big\{ \delta c^T I^* I \delta c - 2Re(\delta S^* I) \delta c \Big\}. \tag{25}$$

Since the optimization problem (25 is derived by linearization of the original Eq. (14), the "optimal" solution δc_{opt} of (25) is only approximate solution of the original problem. Consequently, it is a common practice to employ an interactive update, where at iteration t, the new parameter vector is calculated as $c_{t+1} = c_t + C_t \cdot \delta c$, where $0 < C_t < 1$ and typically $C_t \to 0$. We have experimented with different choices of C_t and found the schedule $C(t) = \frac{1}{\sqrt{t}}$ to give the best results. One could potentially construct a more sophisticated and theoretically sound algorithm with quadratic convergence, which would increase the MC sample size quadratically with the number of iteration, but this line of research was not pursued yet.

3.3 Results

Before we proceed to present the results of our experiments, it is important to identify four potential sources of errors in the SU estimates of the coarse-level Hamiltonian:

Representation error - this is the residual error of the best estimate possible, given the functional form of the coarse-level Hamiltonian. Unlike the parametric estimation, where this error is typically non-zero, the expandable functional families employed in SU guarantee that this error may be made arbitrary small, provided enough terms are introduced into the coarse Hamiltonian. The multiscale construction of the approximation provides an intuitive control of the accuracy/complexity trade-off.

Optimization error - since in practice the parameters' value is determined by numerically solving the optimization problem (25), the optimization process, even using the exact estimates of the fine and coarse-level averages, may introduce a surplus error over the optimal estimate for the given basis of coarse-level Hamiltonian terms.

Coarse-level and fine level Monte Carlo errors - the additional errors due to the stochasticity of the MC estimates used by the optimization algorithm. Since coarse-level and fine-level estimates are calculated at different stages of the algorithm and with different frequency, it is beneficial to consider them separately.

3.3.1 Equilibrium calculations

Harmonic oscillator

The first problem we have considered, as a sanity-check for our approach, was the equilibrium distribution calculation for harmonic oscillator, $V(x) = \frac{1}{2}m\omega^2 x^2$. For simplicity of notation, in all the experiments to follow, we have used scaled measurement units, such that m, ω and β reduce to unity.

As a first step, in order to verify the correctness of our code, we have run the algorithm with all the averages being computed using the well-known Numerical Matrix Multiplication (NMM) algorithm (Thirumalai et al., 1983), commonly used as a reference algorithm in PI calculations. In this experiment, the exact coarse-level Hamiltonian is known to be quadratic, hence the representational error is zero, even when no refinements of the initial basis set are made. Similarly, because of the accuracy of the NMM algorithm, the MC approximation error is also taken out of the equation, hence we are testing the optimization error alone.

Every 30 iterations (of a type $c_{t+1} = c_t + C_t \cdot \delta c_t$) the algorithm inspected the errors of the refinements of the basis functions and expanded the basis set if necessary. The algorithm stopped, when the relative statistics error dropped below 10^{-6}. The experiment was performed for a variety of values of β and various coarsening levels - qualitatively the results were all the same.

Figures 1-4 present the results of this experiment for two representative cases: $\beta = 0.1$, with 8 slices and first coarsening level, and $\beta = 10$, with 128 slices and sixth coarsening level.

We have used several criteria to monitor the behavior of our algorithm. First, the most obvious criteria is the estimation errors of the average "statistics" - the basis functions used to approximate the coarse Hamiltonian. However, unless we know *a priori* that the given basis set is sufficient for approximating the coarse Hamiltonian, convergence of these errors to zero does not guarantee convergence of the approximation. Consequently, we have used two additional indicators. The first is the maximal variation error of the

coarse-level kernel

$$\sup_{x,z} |\rho_{coarse}(x,z) - \hat{\rho}_{coarse}(x,z)|,$$

where $\rho_{coarse}(x,z) = \exp(-H_{coarse})$ is the exact coarse-level kernel and $\hat{\rho}_{coarse}(x,z)$ is our numerical approximation thereof.

Since the object we attempt to approximate in the first place is the equilibrium distribution

$$d(x) = \rho(x,x) =< x|exp(-\beta H)|x >,$$

we have also chosen to monitor the maximal variation error in the estimated equilibrium distribution:

$$\sup_{x} |d(x) - \hat{d}(x)|.$$

It can be seen from these results that, for the harmonic oscillator, our algorithm was able to find approximation to the coarse level Hamiltonian, which is virtually indistinguishable from the exact one, with respect to all three measures. This shows that the optimization error of our algorithm is zero.

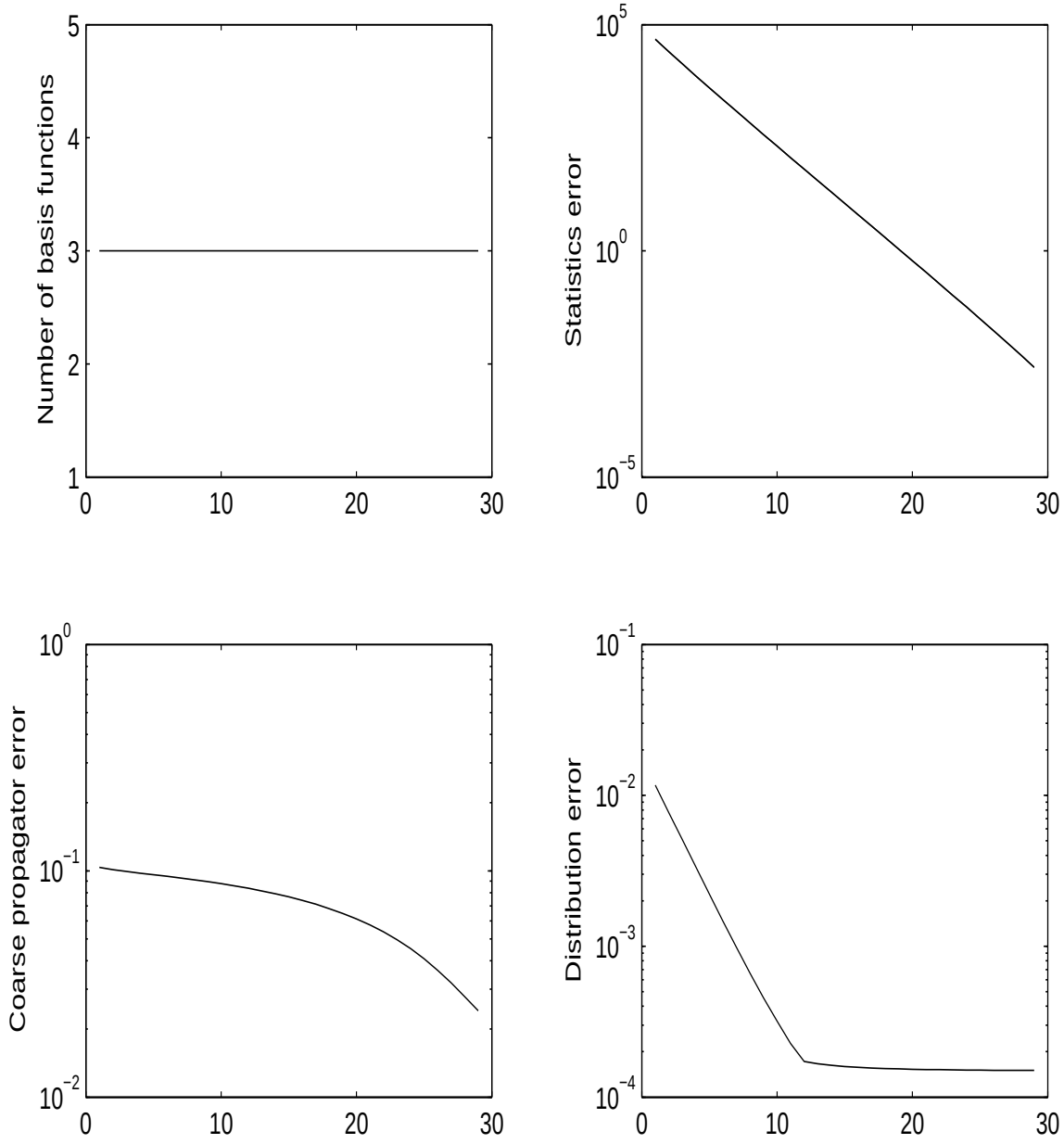

Figure 1: Harmonic oscillator - the progress of the algorithm with NMM estimates for the first coarsening level with 8 slices and $\beta = 0.1$: the number of the basis functions (top-left), the mean-squared error of the statistics in the basis set (top-right), maximal variation error in the coarse-level kernel (bottom-left) and maximal variation error in the estimated equilibrium distribution (bottom-right).

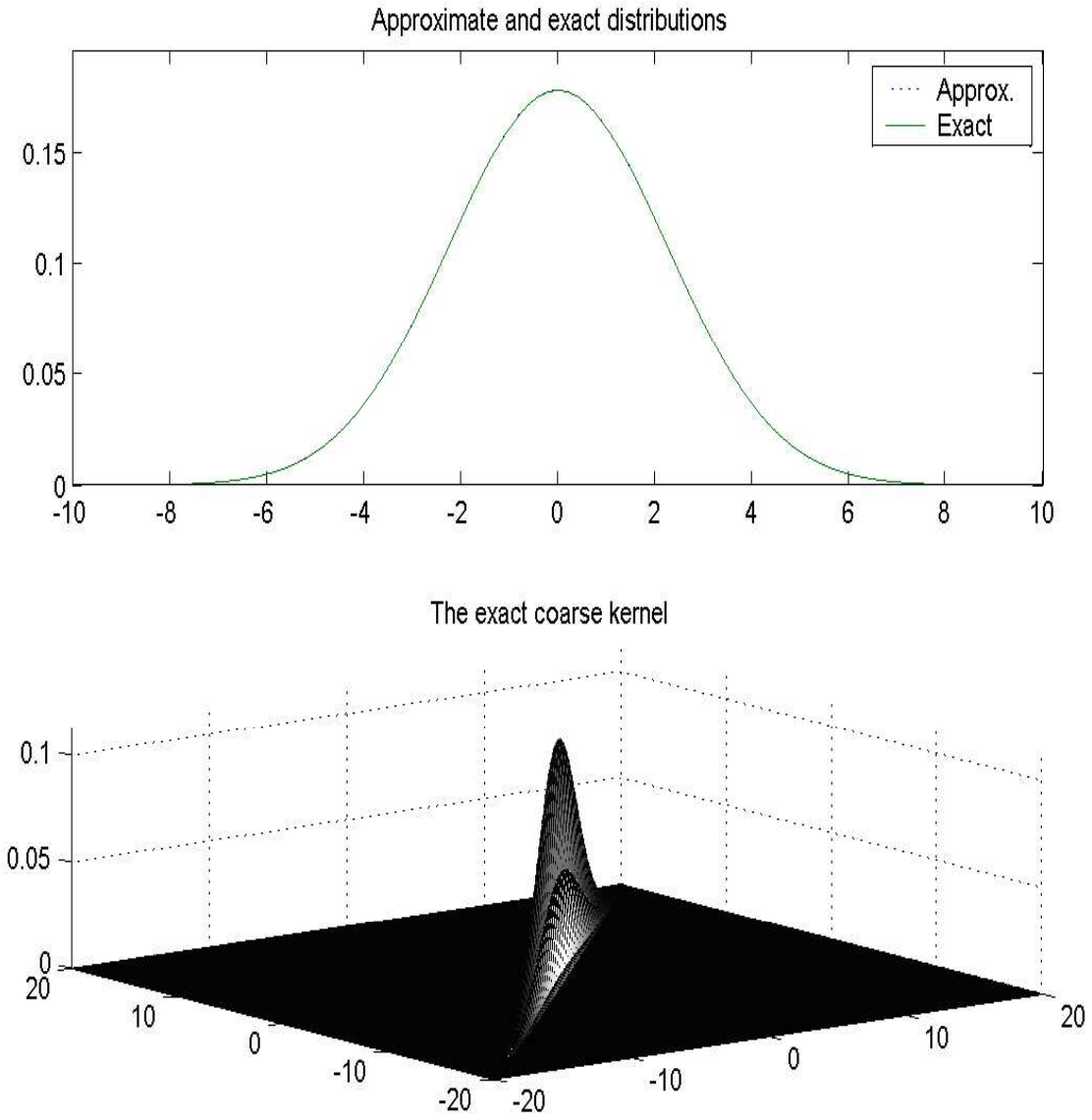

Figure 2: Harmonic oscillator - exact equilibrium distribution $d(x)$ versus its approximation $\hat{d}(x)$ (top) and exact coarse level kernel $\rho_{coarse}(x, z)$ (bottom) for the first coarsening level with $\beta = 0.1$ with NMM estimates.

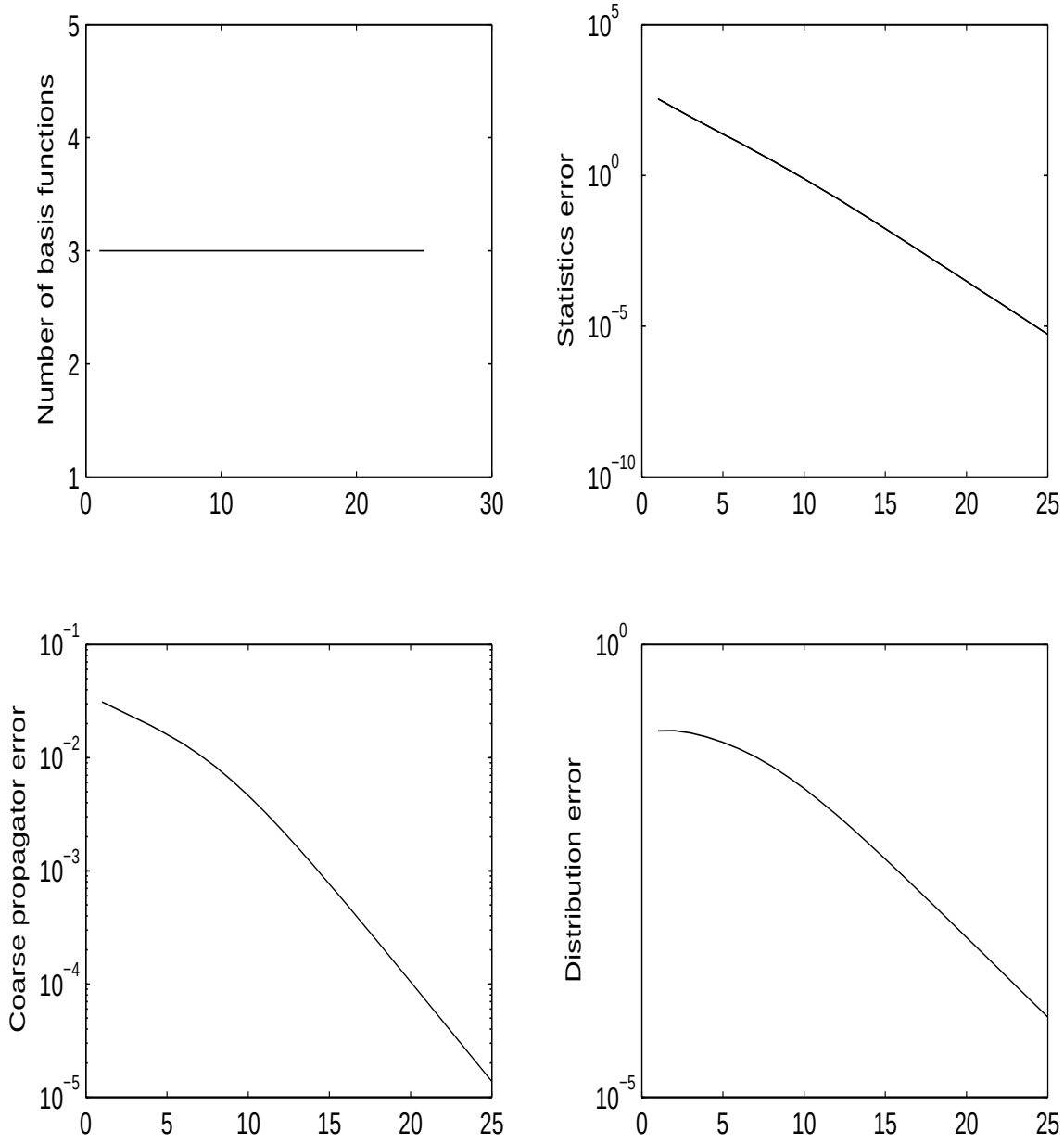

Figure 3: Harmonic oscillator - the progress of the algorithm with NMM estimates for the sixth coarsening level with 128 slices and $\beta = 10$: the number of the basis functions (top-left), the mean-squared error of the statistics in the basis set (top-right), maximal variation error in the coarse-level kernel (bottom-left) and maximal variation error in the estimated equilibrium distribution (bottom-right).

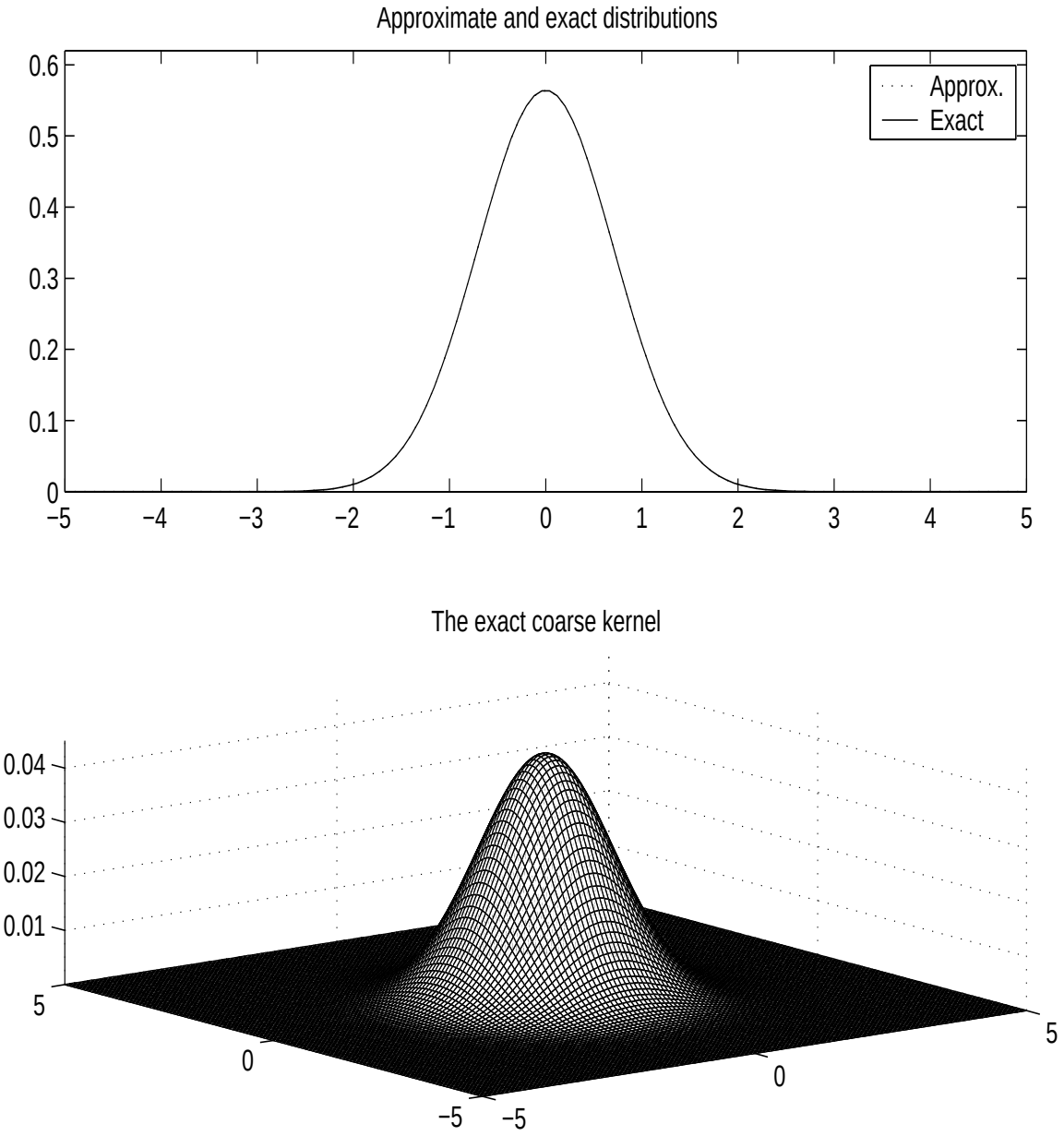

Figure 4: Harmonic oscillator - exact equilibrium distribution $d(x)$ versus its approximation $\hat{d}(x)$ (top) and exact coarse level kernel $\rho_{coarse}(x, z)$ (bottom) for the sixth coarsening level with $\beta = 10$ with NMM estimates.

Next, we have repeated this experiment with the coarse-level averages being computed using 10^4 MC samples per iteration and using NMM for calculating the fine-level averages. In other words, in this experiment we consider the effect of the coarse-level MC approximation error.

The results are shown in Fig. 5-8. It can be seen that, although with MC estimates of the statistics aver-

ages one could not achieve arbitrary accuracy (as in the "exact averages" case), still the coarse Hamiltonian approximations obtained in the process are very accurate and adequate for any practical purpose. If one wished to improve the accuracy even further, one could either simply increase the number of MC samples per iteration or employ the extended Kalman filtering update (Bertsekas, 1995), which ensures consistency of the stochastic optimization algorithm.

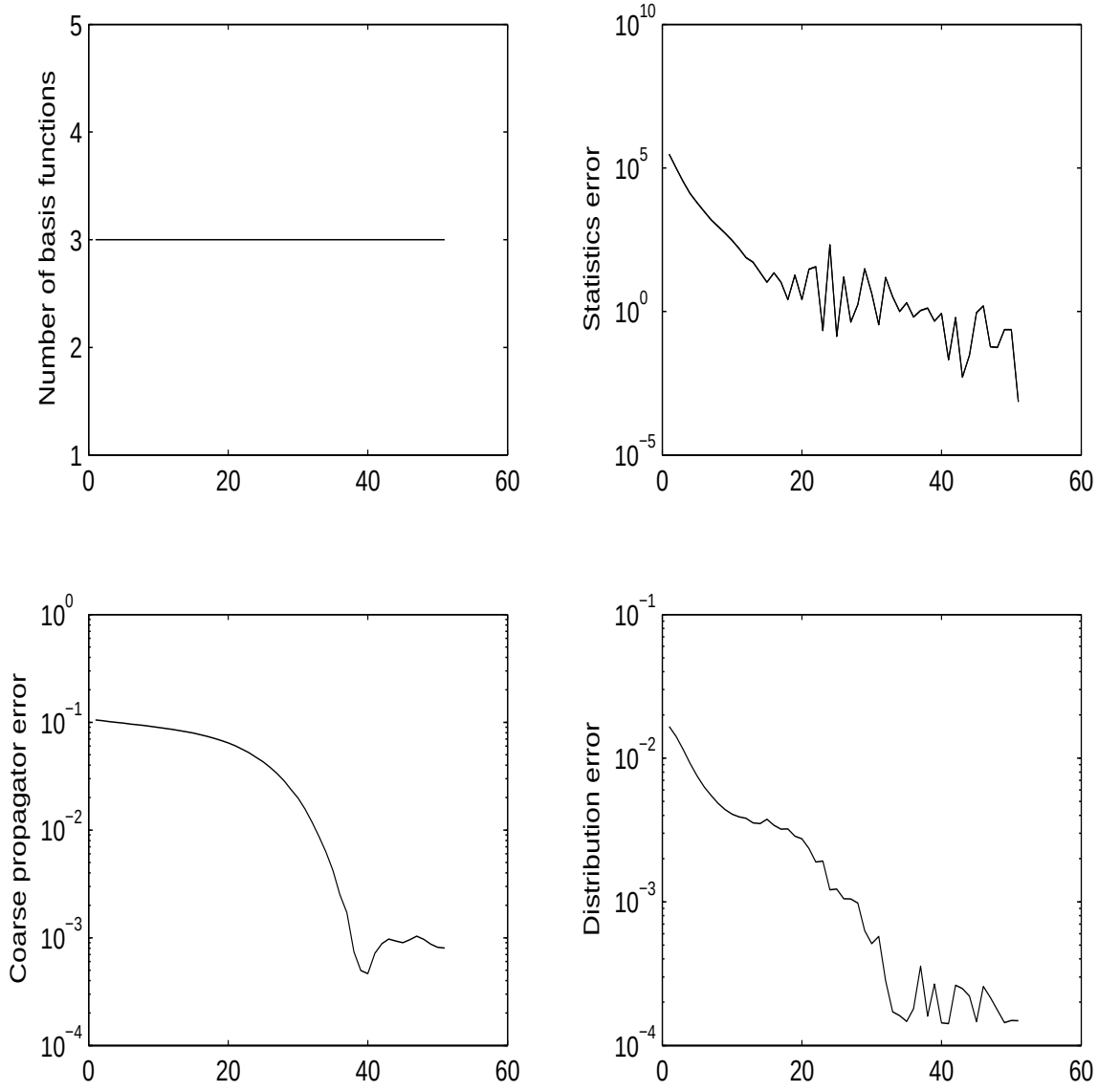

Figure 5: Harmonic oscillator - the progress of the algorithm with Monte Carlo coarse-level estimates for the first coarsening level with 8 slices and $\beta = 0.1$: the number of the basis functions (top-left), the mean-squared error of the statistics in the basis set (top-right), maximal variation error in the coarse-level kernel (bottom-left) and maximal variation error in the estimated equilibrium distribution (bottom-right).

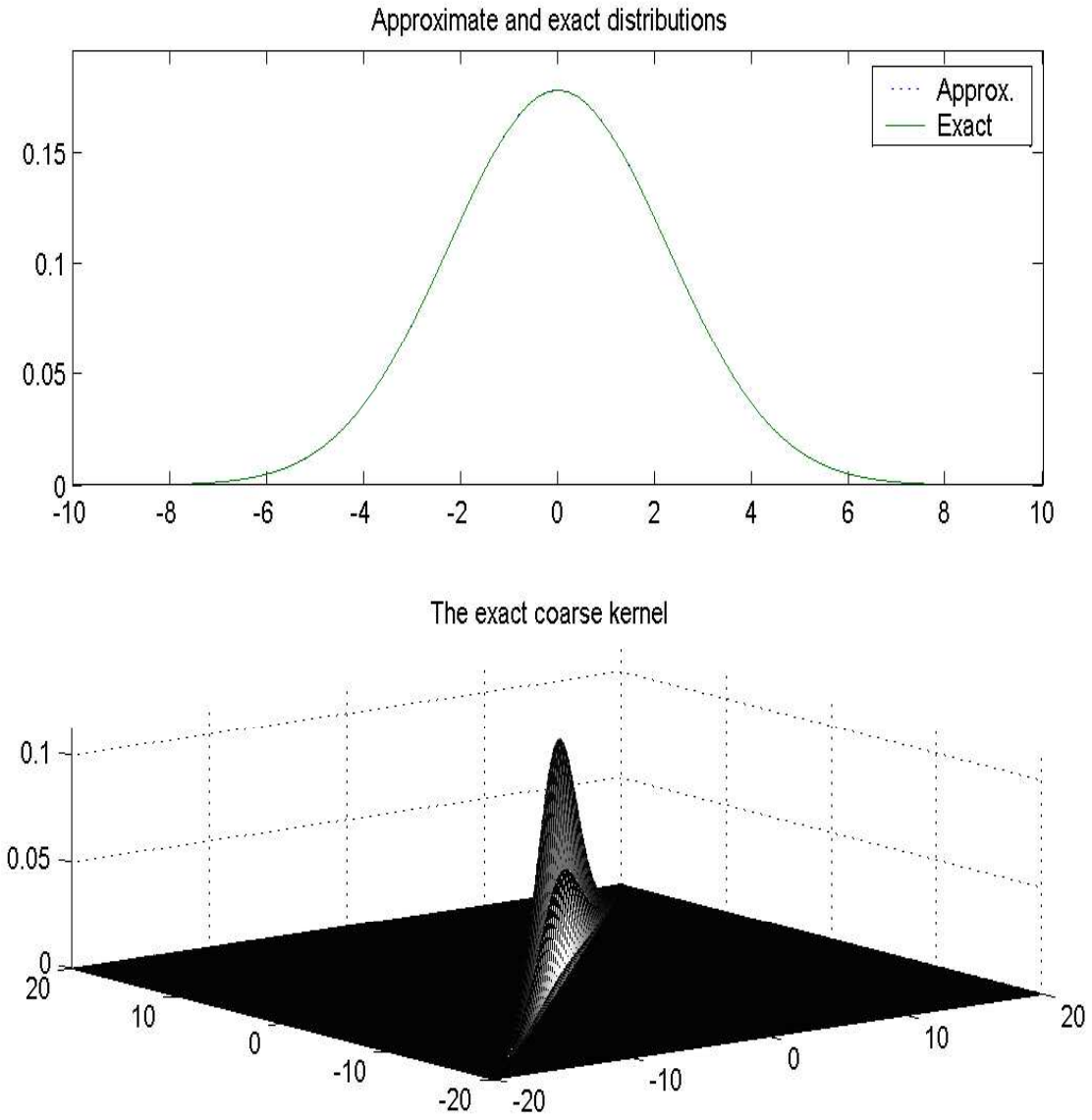

Figure 6: Harmonic oscillator - exact equilibrium distribution $d(x)$ versus its approximation $\hat{d}(x)$ (top) and exact coarse level kernel $\rho_{coarse}(x, z)$ (bottom) for the first coarsening level with $\beta = 0.1$ with Monte Carlo coarse-level estimates.

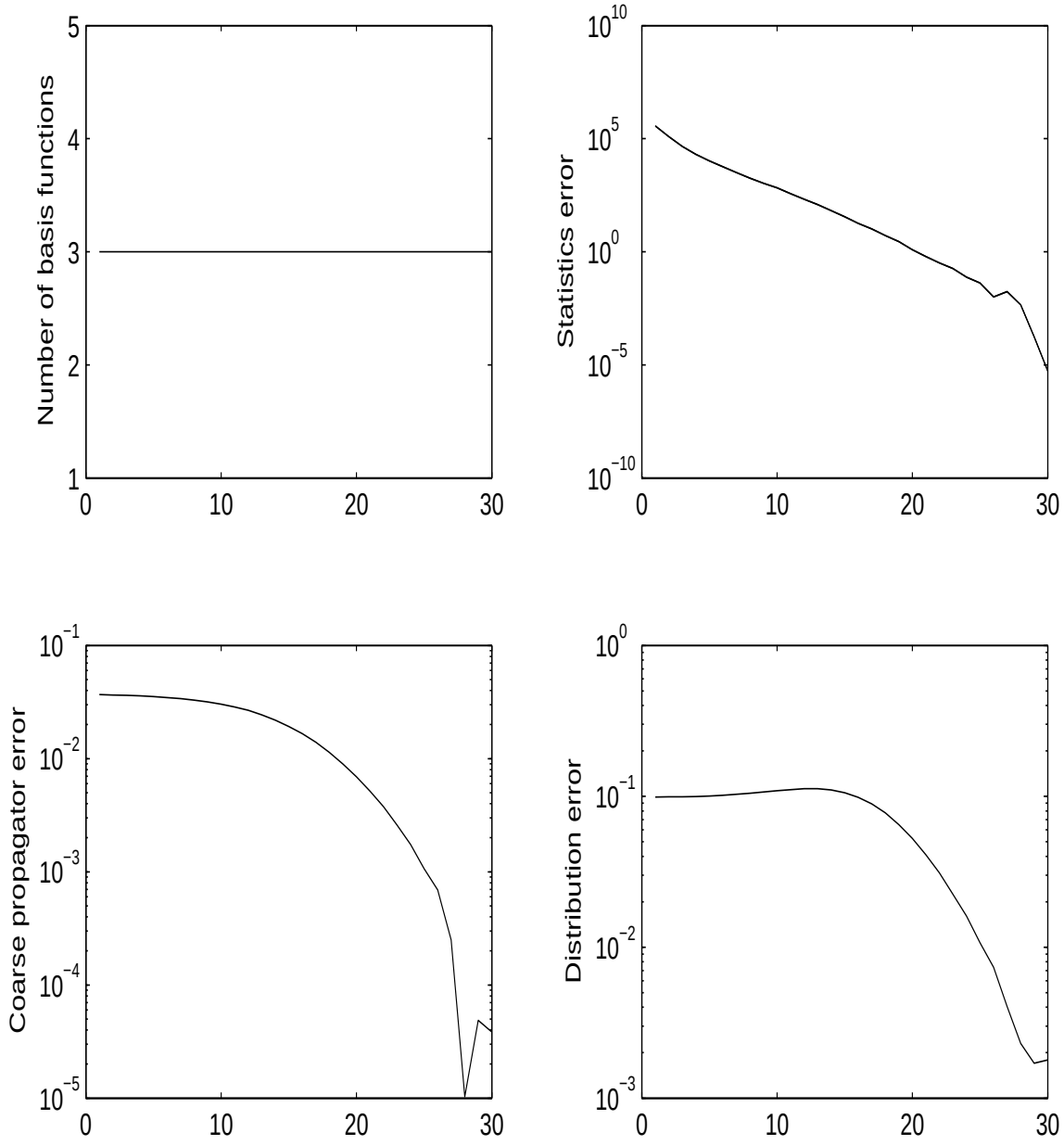

Figure 7: Harmonic oscillator - the progress of the algorithm with Monte Carlo coarse-level estimates for the sixth coarsening level with 128 slices and $\beta = 10$: the number of the basis functions (top-left), the mean-squared error of the statistics in the basis set (top-right), maximal variation error in the coarse-level kernel (bottom-left) and maximal variation error in the estimated equilibrium distribution (bottom-right).

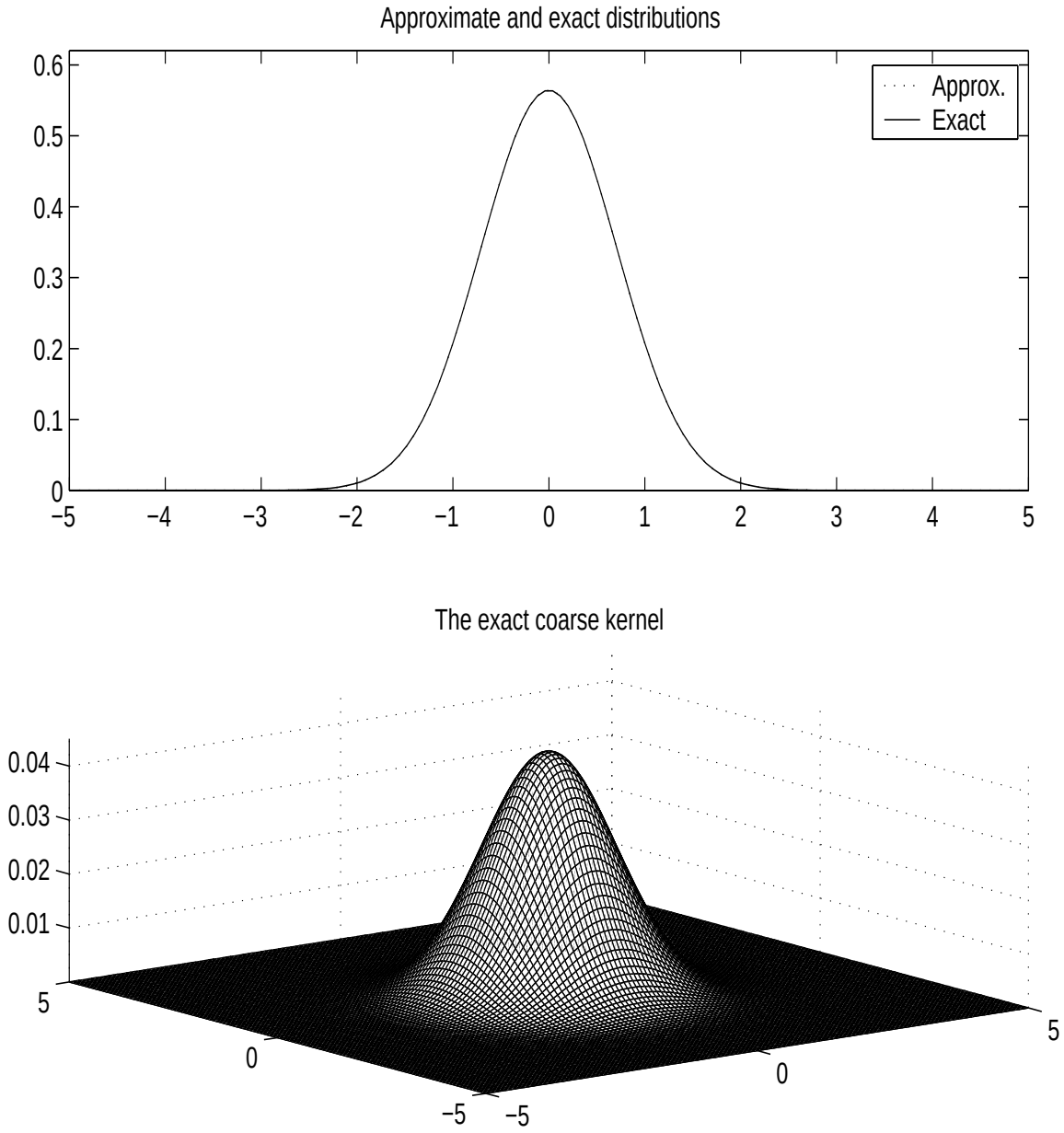

Figure 8: Harmonic oscillator - exact equilibrium distribution $d(x)$ versus its approximation $\hat{d}(x)$ (top) and exact coarse level kernel $\rho_{coarse}(x, z)$ (bottom) for the sixth coarsening level with $\beta = 10$ with Monte Carlo coarse-level estimates.

Let us conclude by observing that, in the full algorithm, the fine-level sample need only be generated once, hence the effect of the fine-level MC approximation error is predictable - it introduces a bias in the estimates, which could be made arbitrary small by increasing the size of the fine-level sample. Consequently, we have not carried out any experiment aimed at isolating the fine-level MC approximation error effect, and always calculated the fine-level averages using the NMM method.

Double well

As a next step, we have considered the double well potential:

$$V(x) = \frac{x^4}{4} - \frac{x^2}{2} \tag{26}$$

As a first step we considered the case $\beta = 0.1$ with 8 slices. The results for the second coarsening level are presented in Figures 9-13 (the lower coarse levels produced similar results). With a relatively small sample size (10^4), the stochastic sampling errors forced introduction of some superfluous basis functions. The situation was improved with a larger sample size, 10^5, which produced an early stop similarly to the NMM-based run. At this point we would like to emphasize the major accuracy improvement that our algorithm produced merely by optimizing the coefficients of the initial basis, $V(x), V(z), (x-z)^2, x^2, z^2$, without any refinements.

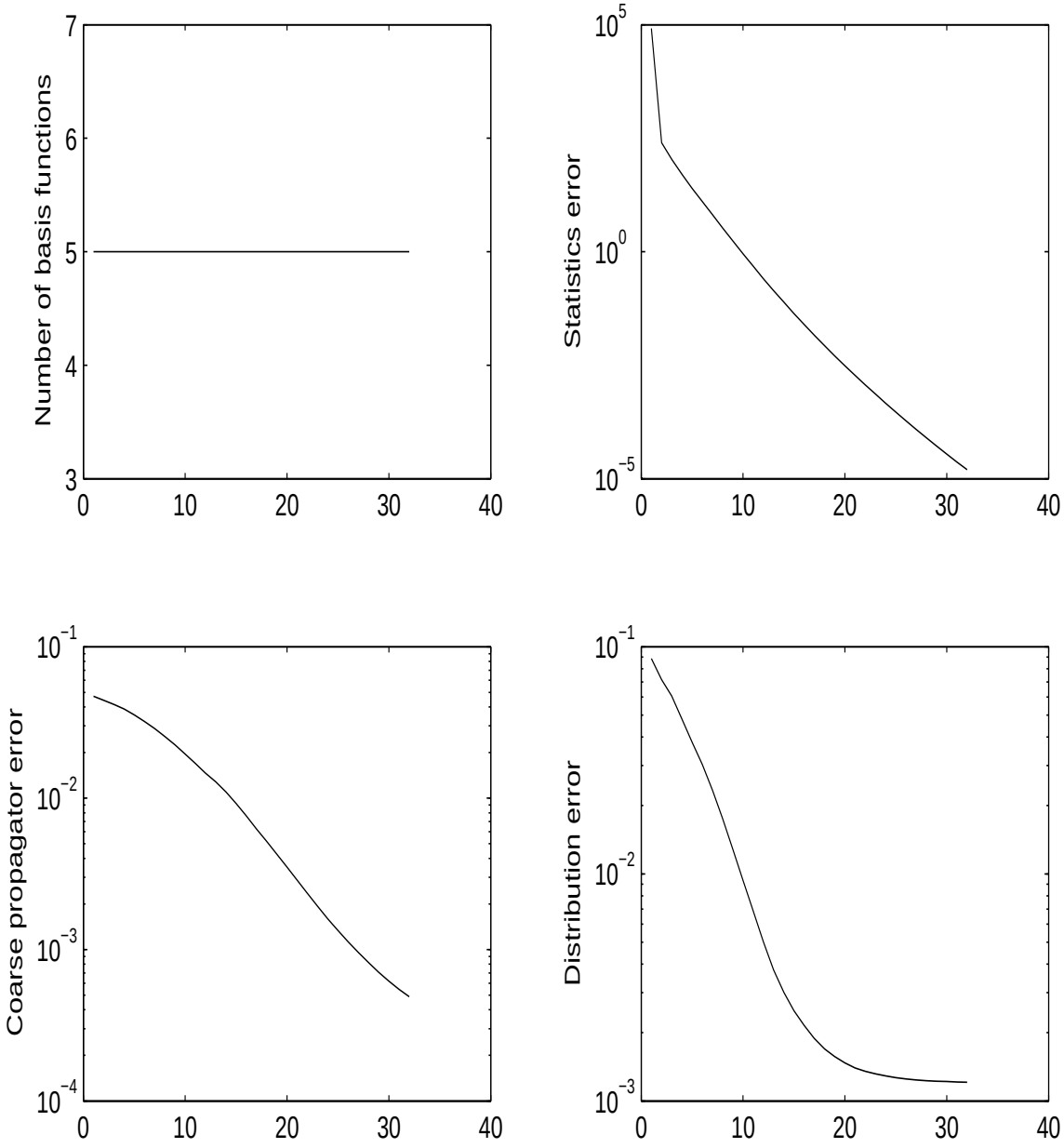

Figure 9: Double well - the progress of the algorithm with NMM estimates for the second coarsening level with 8 slices and $\beta = 0.1$: the number of the basis functions (top-left), the mean-squared error of the statistics in the basis set (top-right), maximal variation error in the coarse-level kernel (bottom-left) and maximal variation error in the estimated equilibrium distribution (bottom-right).

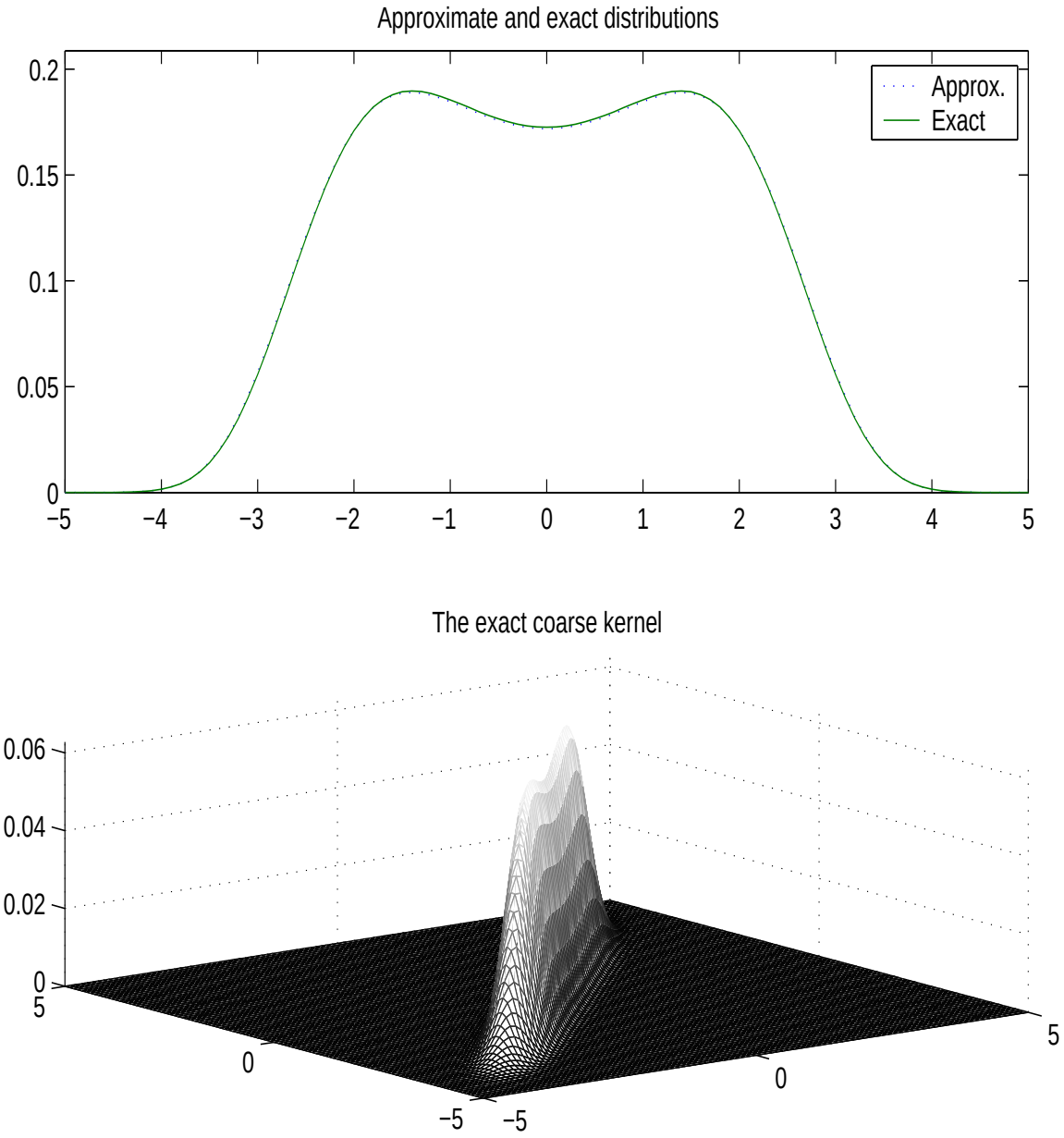

Figure 10: Double well - exact equilibrium distribution $d(x)$ versus its approximation $\hat{d}(x)$ (top) and exact coarse level kernel $\rho_{coarse}(x, z)$ (bottom) for the second coarsening level with $\beta = 0.1$ with NMM estimates.

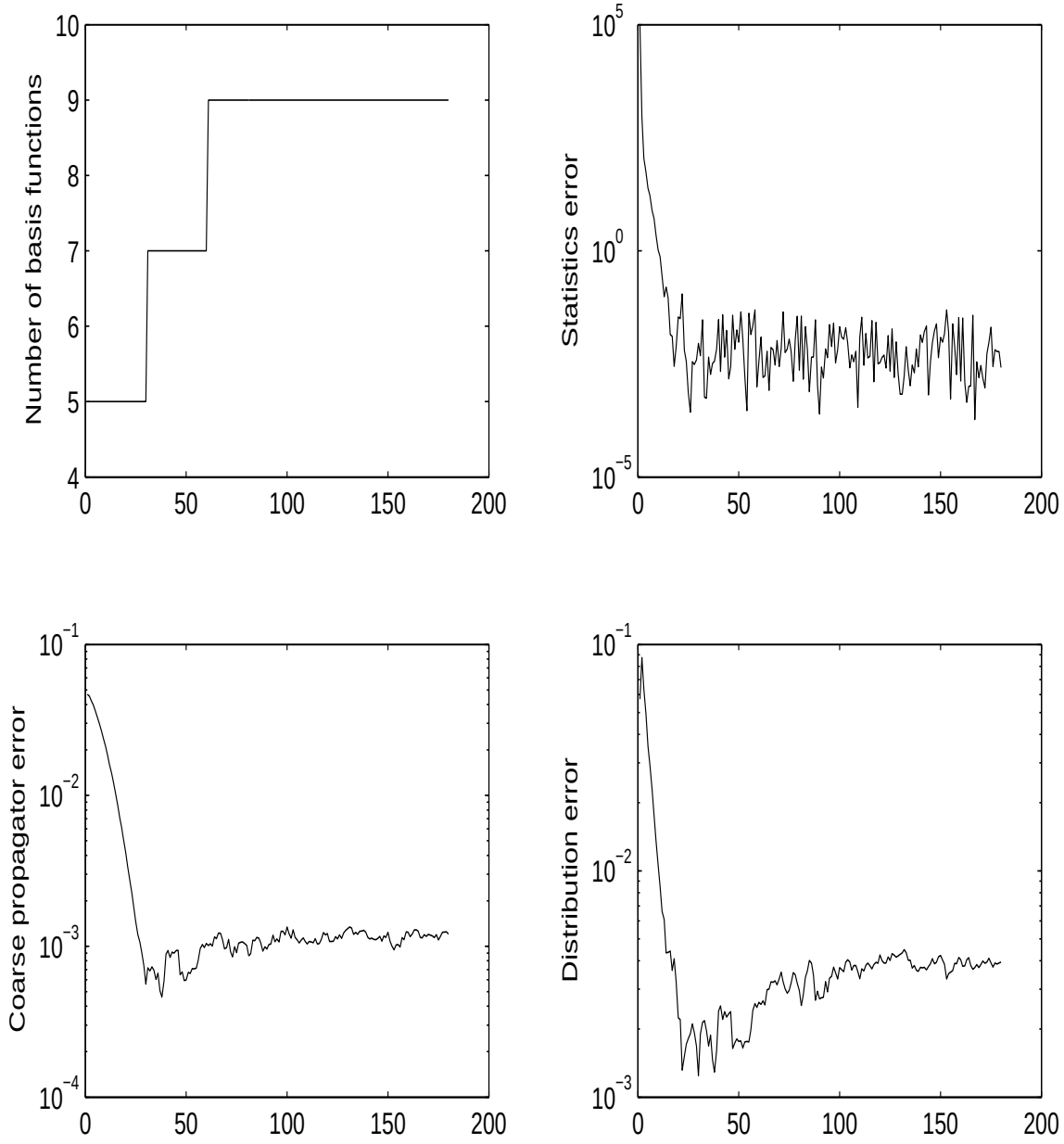

Figure 11: Double well - the progress of the algorithm with 10^4 MC samples per iteration with 8 slices and $\beta = 0.1$ for the second coarsening level: the number of the basis functions (top-left), the mean-squared error of the statistics in the basis set (top-right), maximal variation error in the coarse-level kernel (bottom-left) and maximal variation error in the estimated equilibrium distribution (bottom-right).

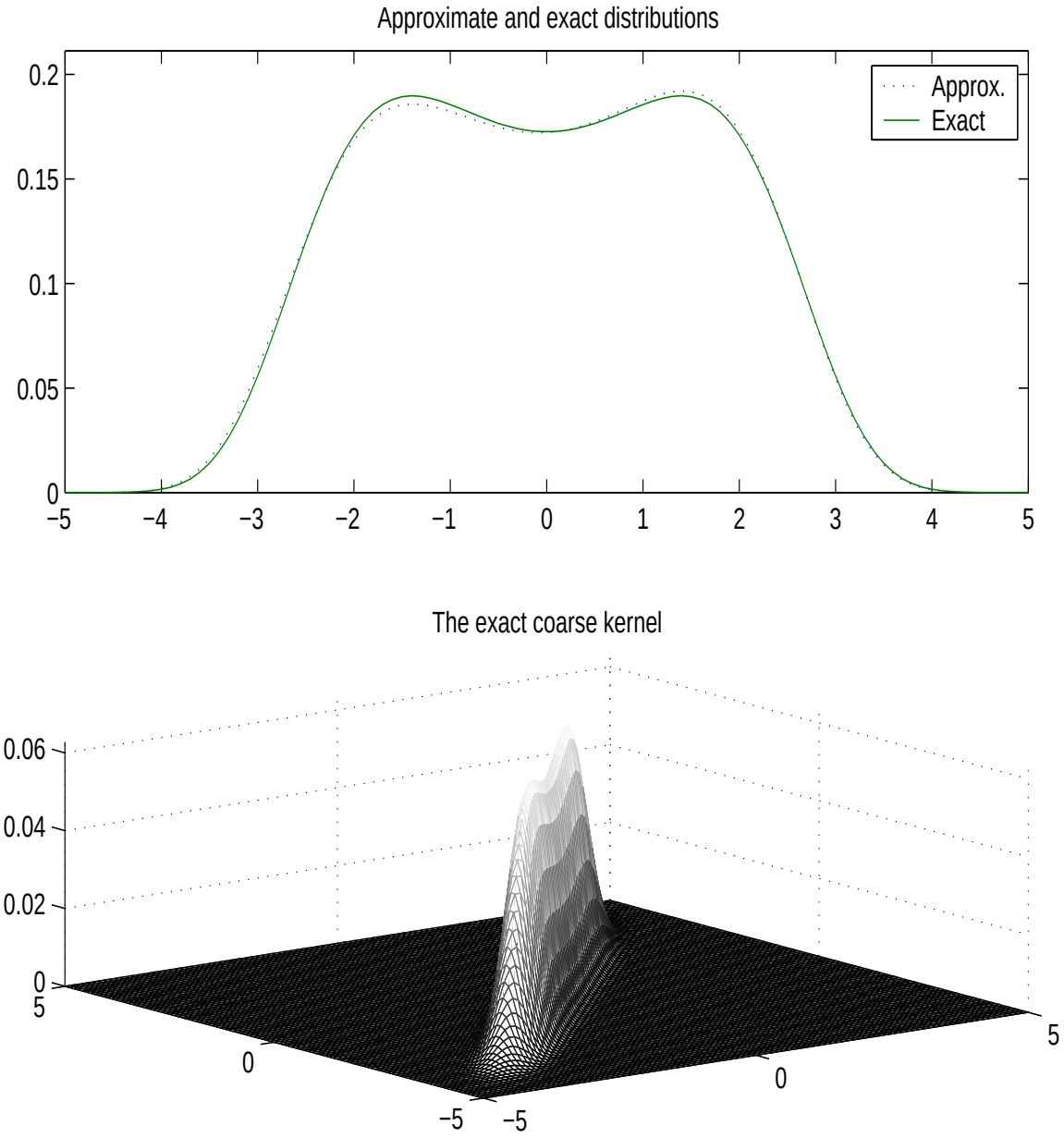

Figure 12: Double well - exact equilibrium distribution $d(x)$ versus its approximation $\hat{d}(x)$ (top) and exact coarse level kernel $\rho_{coarse}(x, z)$ (bottom) for the second coarsening level with $\beta = 0.1$ with Monte Carlo (10^4 samples per iteration) coarse-level estimates.

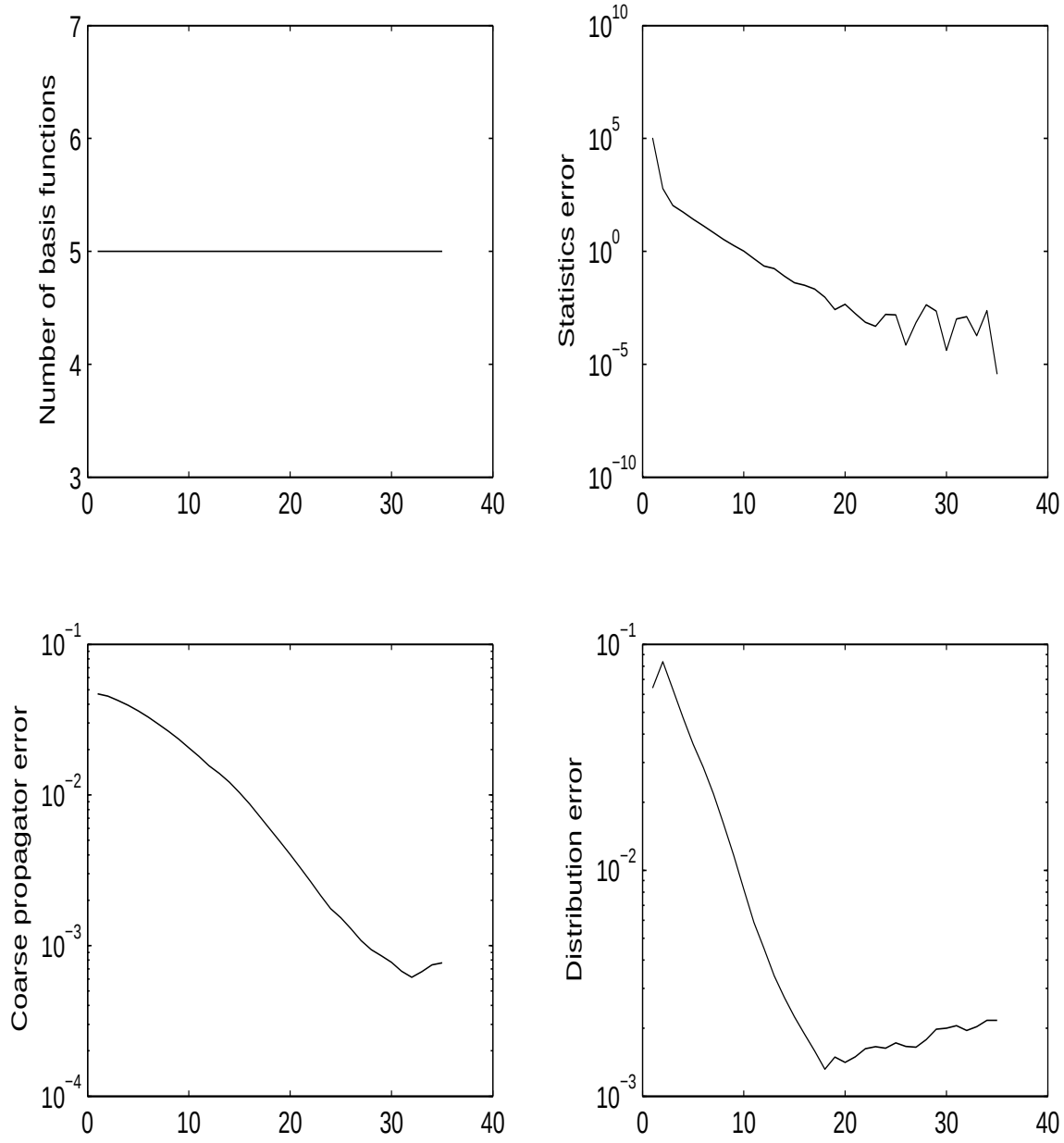

Figure 13: Double well - the progress of the algorithm with 10^5 MC samples per iteration with 8 slices and $\beta = 0.1$ for the second coarsening level: the number of the basis functions (top-left), the mean-squared error of the statistics in the basis set (top-right), maximal variation error in the coarse-level kernel (bottom-left) and maximal variation error in the estimated equilibrium distribution (bottom-right).

Next, we have explored the more difficult case of $\beta = 10$ with 128 slices. Figures 14,15 show the results of the NMM-based run for the third coarsening level, which revealed that the statistics error is no longer a reliable indicator in this case. While the relative statistics error dropped below 10^{-6}, the kernel

and, especially, the equilibrium distribution still had some non-negligible errors. Consequently, in the few experiments to follow, we have removed the early stopping condition and allowed the algorithm to run for a fixed number of iterations regardless of the statistics error.

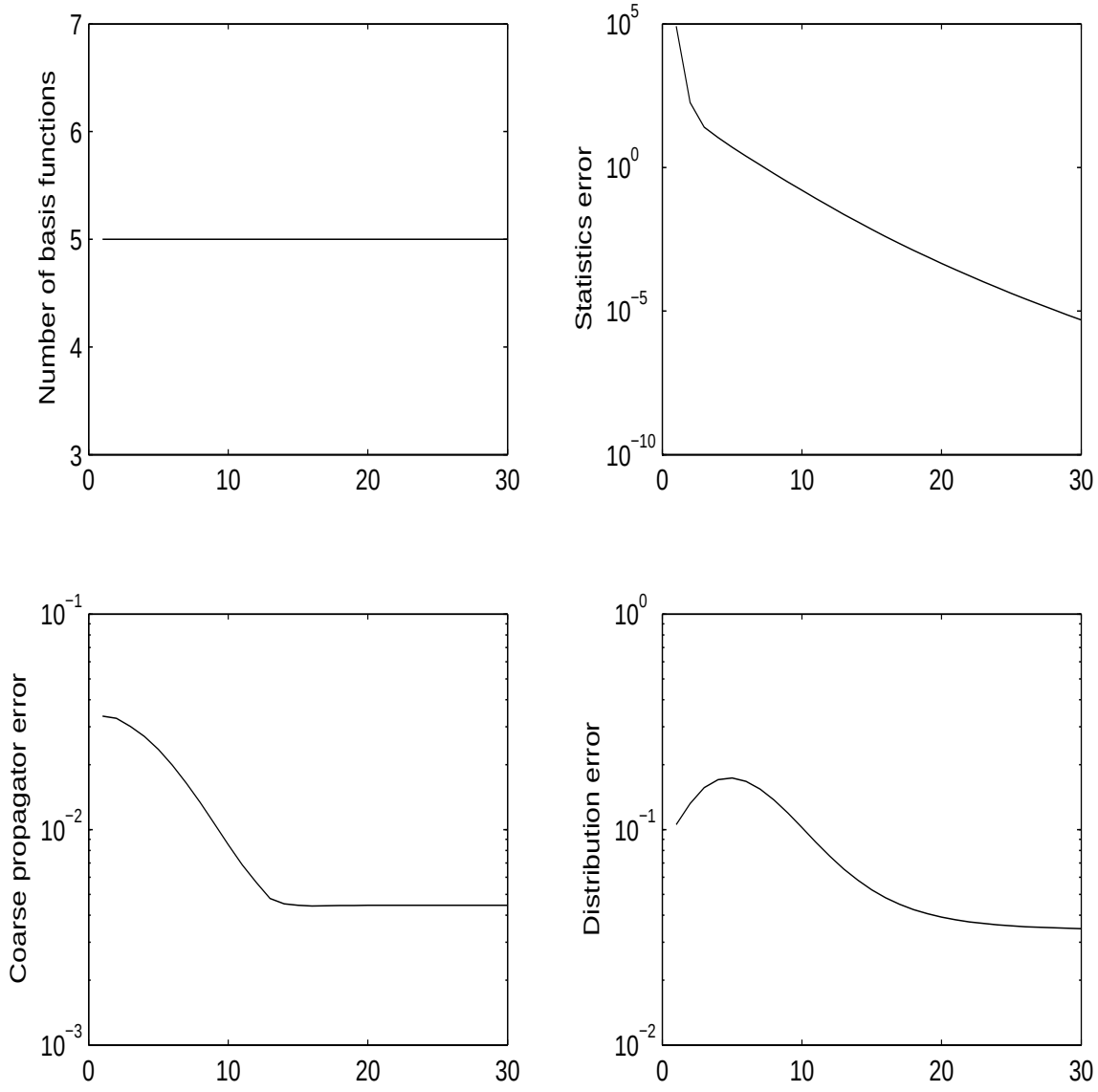

Figure 14: Double well - the progress of the algorithm with early stop and NMM estimates for the third coarsening level with 128 slices and $\beta = 10$: the number of the basis functions (top-left), the mean-squared error of the statistics in the basis set (top-right), maximal variation error in the coarse-level kernel (bottom-left) and maximal variation error in the estimated equilibrium distribution (bottom-right).

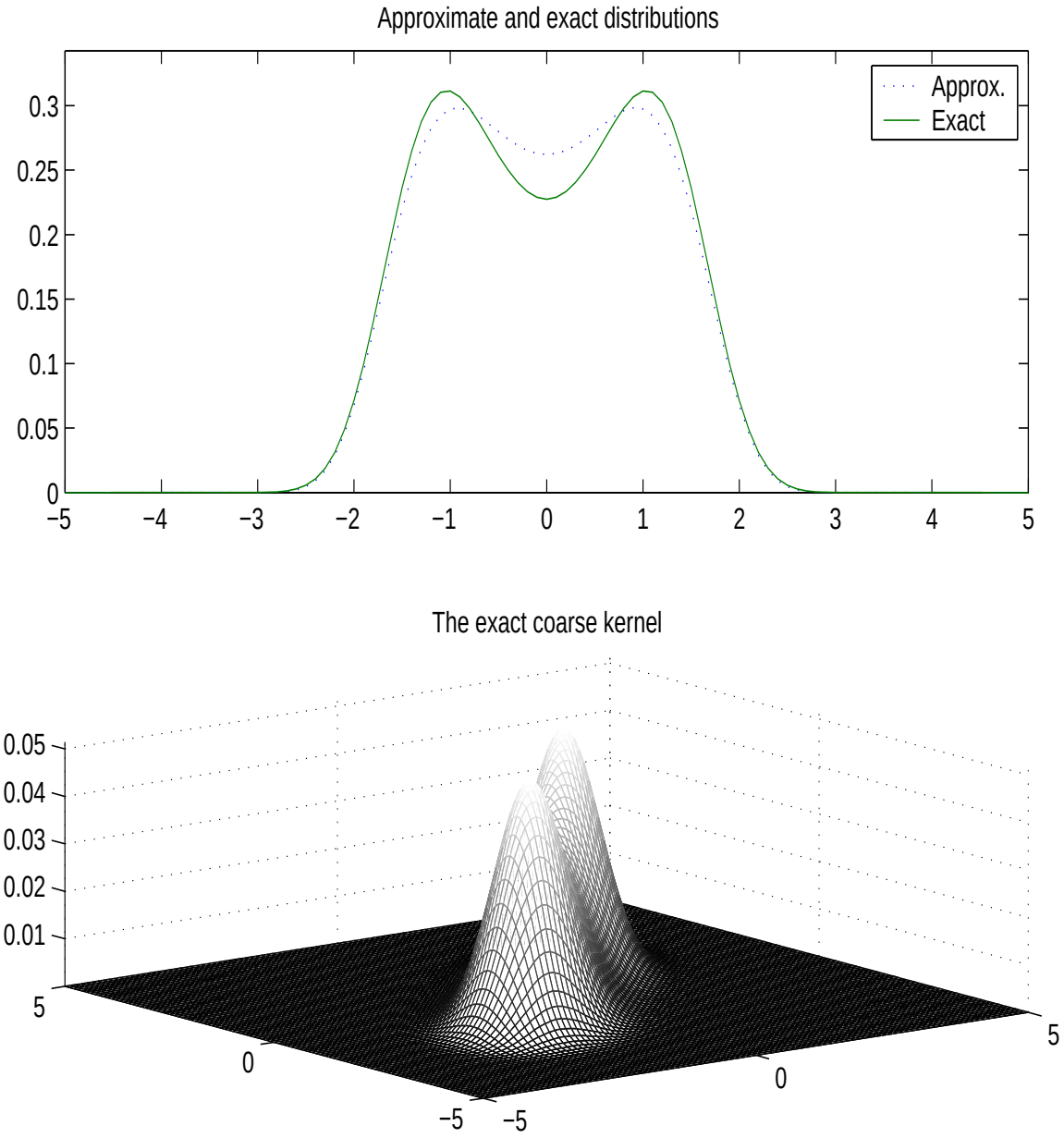

Figure 15: Double well - exact equilibrium distribution $d(x)$ versus its approximation $\hat{d}(x)$ (top) and exact coarse level kernel $\rho_{coarse}(x, z)$ (bottom) for the first coarsening level with $\beta = 0.1$ with NMM estimates and early stop.

Figures 16-19 illustrate the improved behavior of the algorithm without the stopping condition with NMM-based and MC-based estimates respectively. It can be seen that the algorithm succeeds to produce very accurate estimates with relatively small number of basis functions.

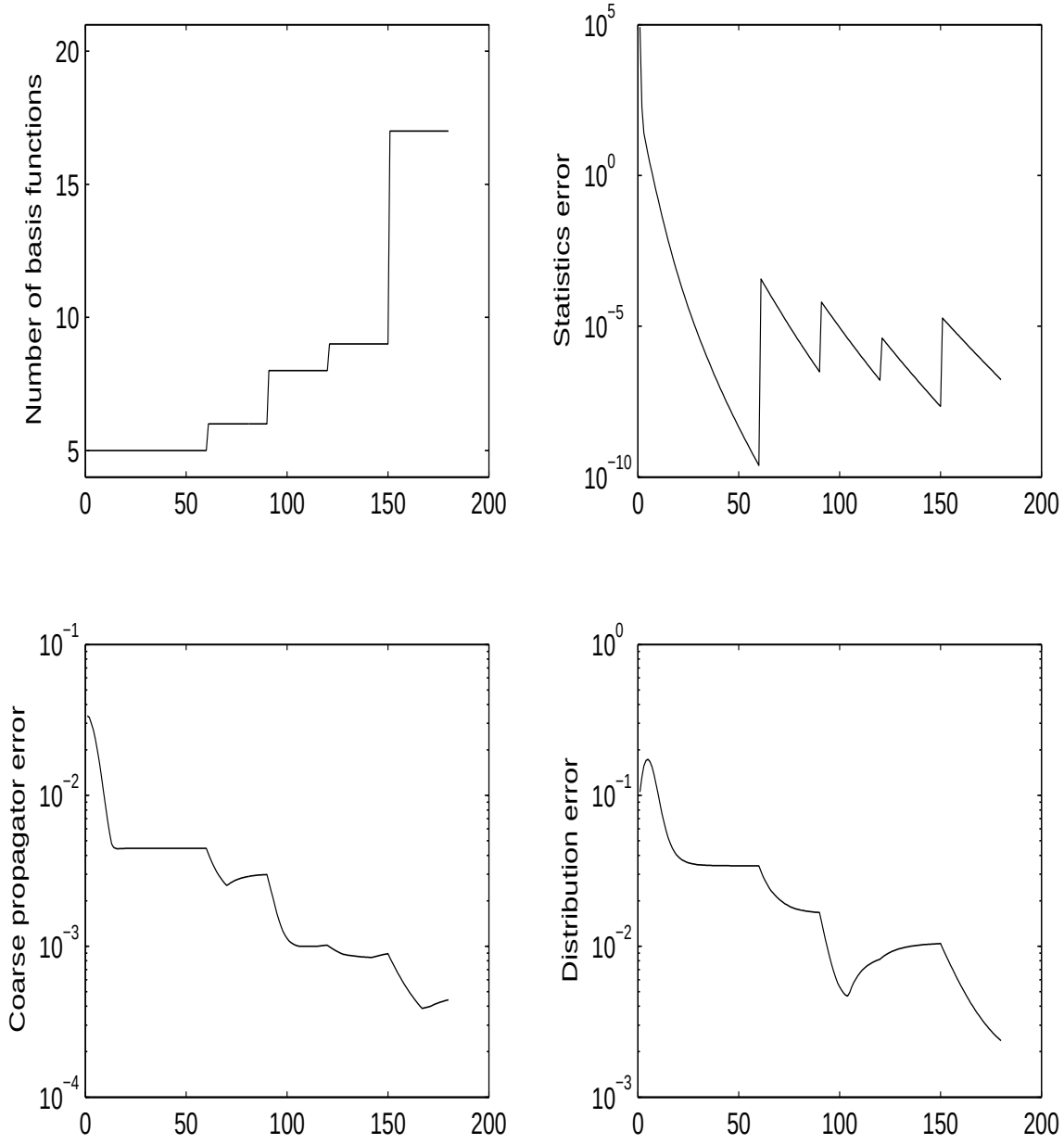

Figure 16: Double well - the progress of the algorithm without early stop and NMM estimates for the third coarsening level with 128 slices and $\beta = 10$: the number of the basis functions (top-left), the mean-squared error of the statistics in the basis set (top-right), maximal variation error in the coarse-level kernel (bottom-left) and maximal variation error in the estimated equilibrium distribution (bottom-right).

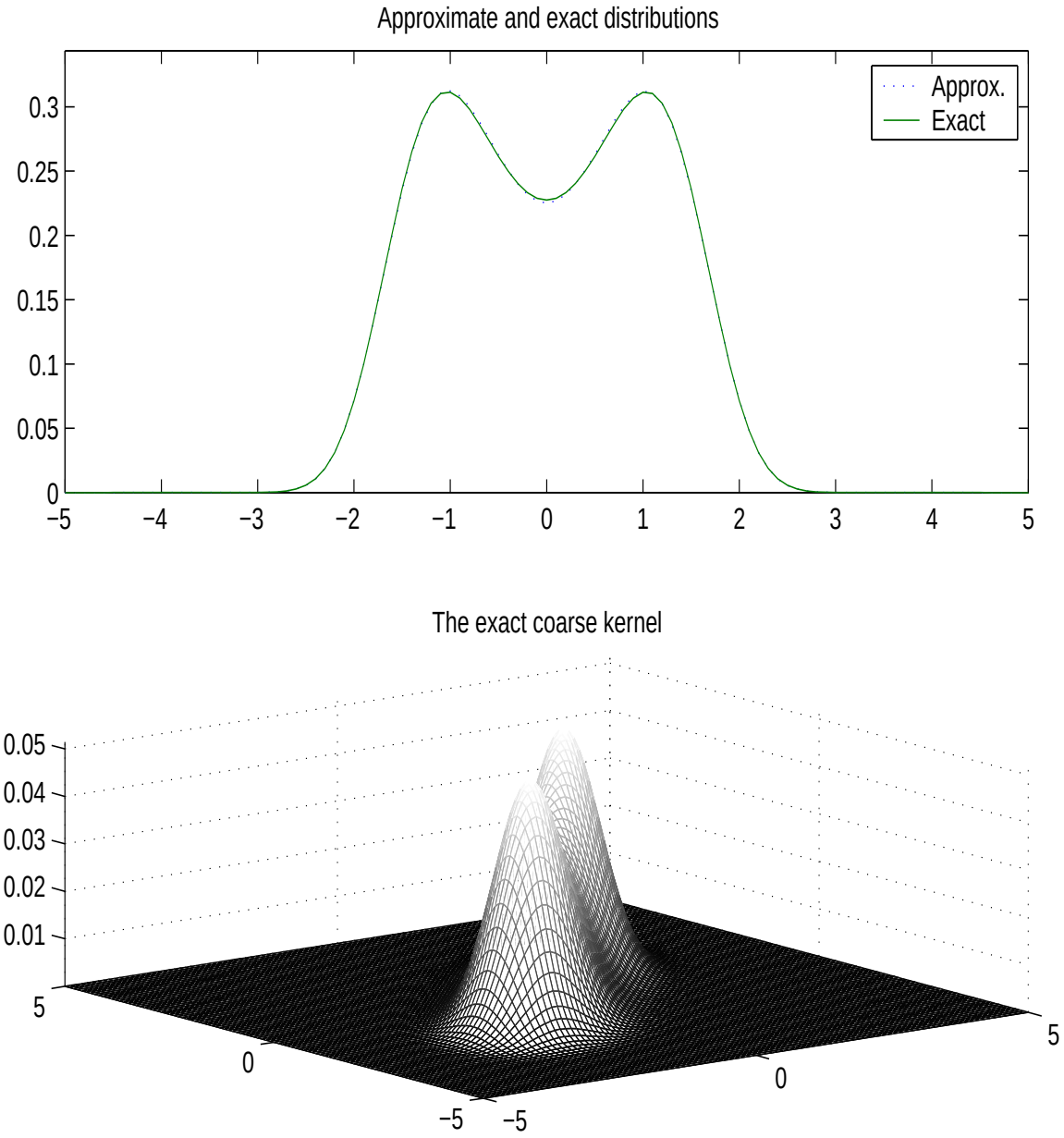

Figure 17: Double well - exact equilibrium distribution $d(x)$ versus its approximation $\hat{d}(x)$ (top) and exact coarse level kernel $\rho_{coarse}(x, z)$ (bottom) for the first coarsening level with $\beta = 0.1$ with NMM estimates and without the early stop.

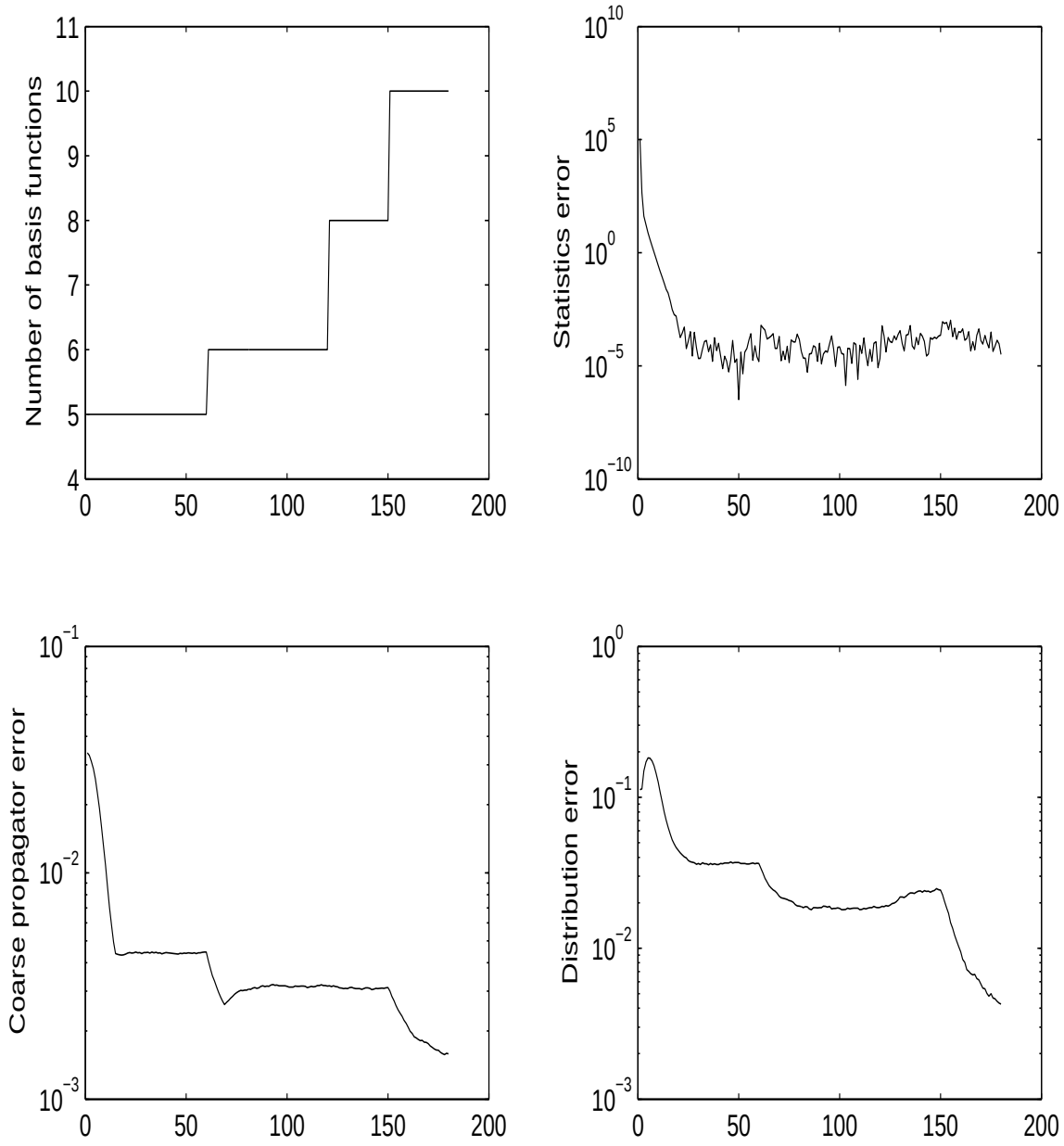

Figure 18: Double well - the progress of the algorithm without early stop and MC estimates (10^5 samples per iteration) for the third coarsening level with 128 slices and $\beta = 10$: the number of the basis functions (top-left), the mean-squared error of the statistics in the basis set (top-right), maximal variation error in the coarse-level kernel (bottom-left) and maximal variation error in the estimated equilibrium distribution (bottom-right).

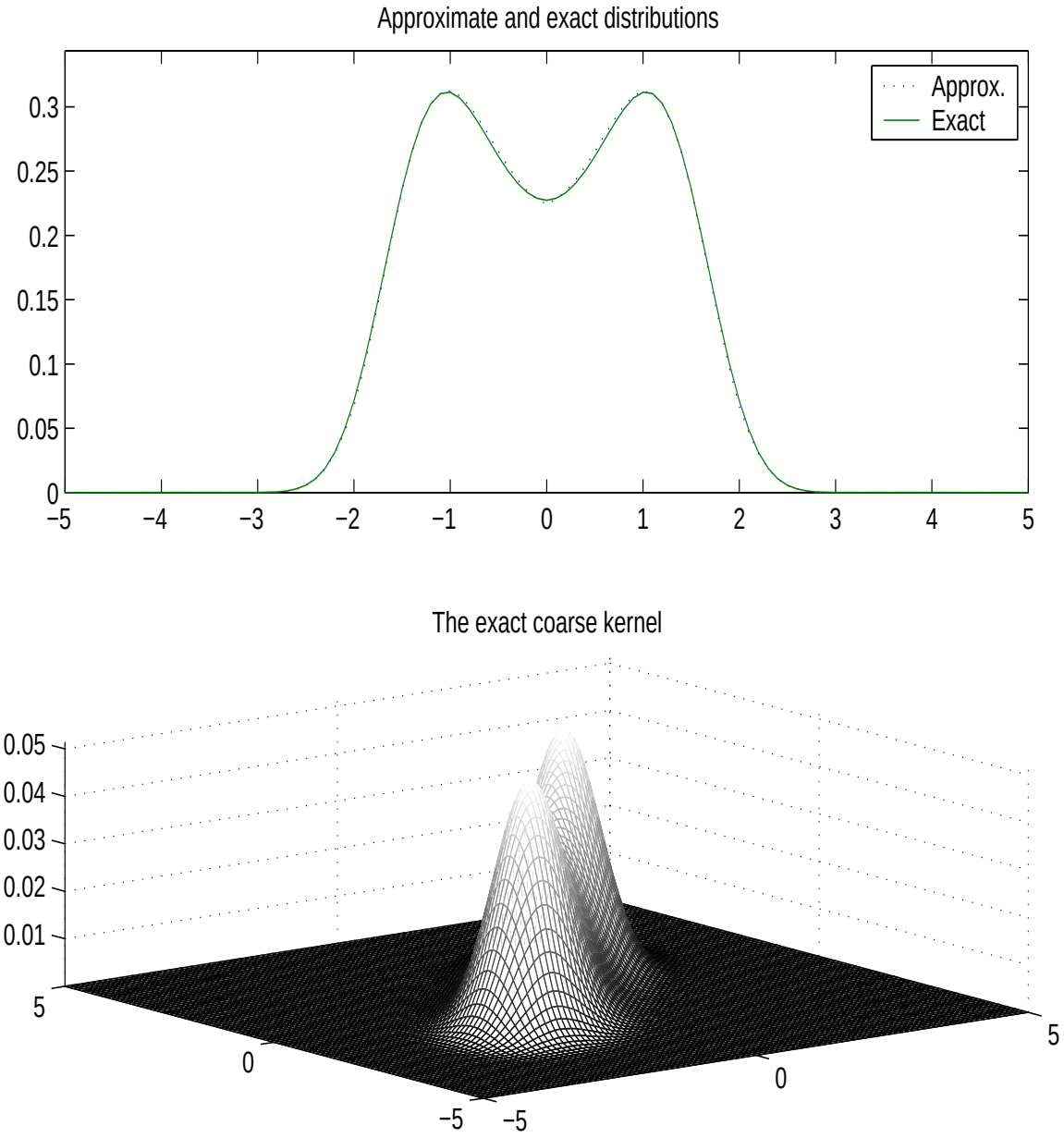

Figure 19: Double well - exact equilibrium distribution $d(x)$ versus its approximation $\hat{d}(x)$ (top) and exact coarse level kernel $\rho_{coarse}(x, z)$ (bottom) for the first coarsening level with $\beta = 10$ with MC estimates (10^5 samples per iteration) and without the early stop.

Finally, we have examined the limits of our algorithm by considering the sixth coarsening level. As can be seen from Figures 20-21, when using NMM estimates, the algorithm still managed to produce very accurate approximations. However, this success could not be reproduced when using the MC-based estimates as can be seen from Figures 22-25. While increasing the size of the MC sample from 10^5 to as much as

10^7 did improve the quality of the approximation, still even when the larger samples there was a noticeable mismatch between the exact equilibrium distribution and the approximation.

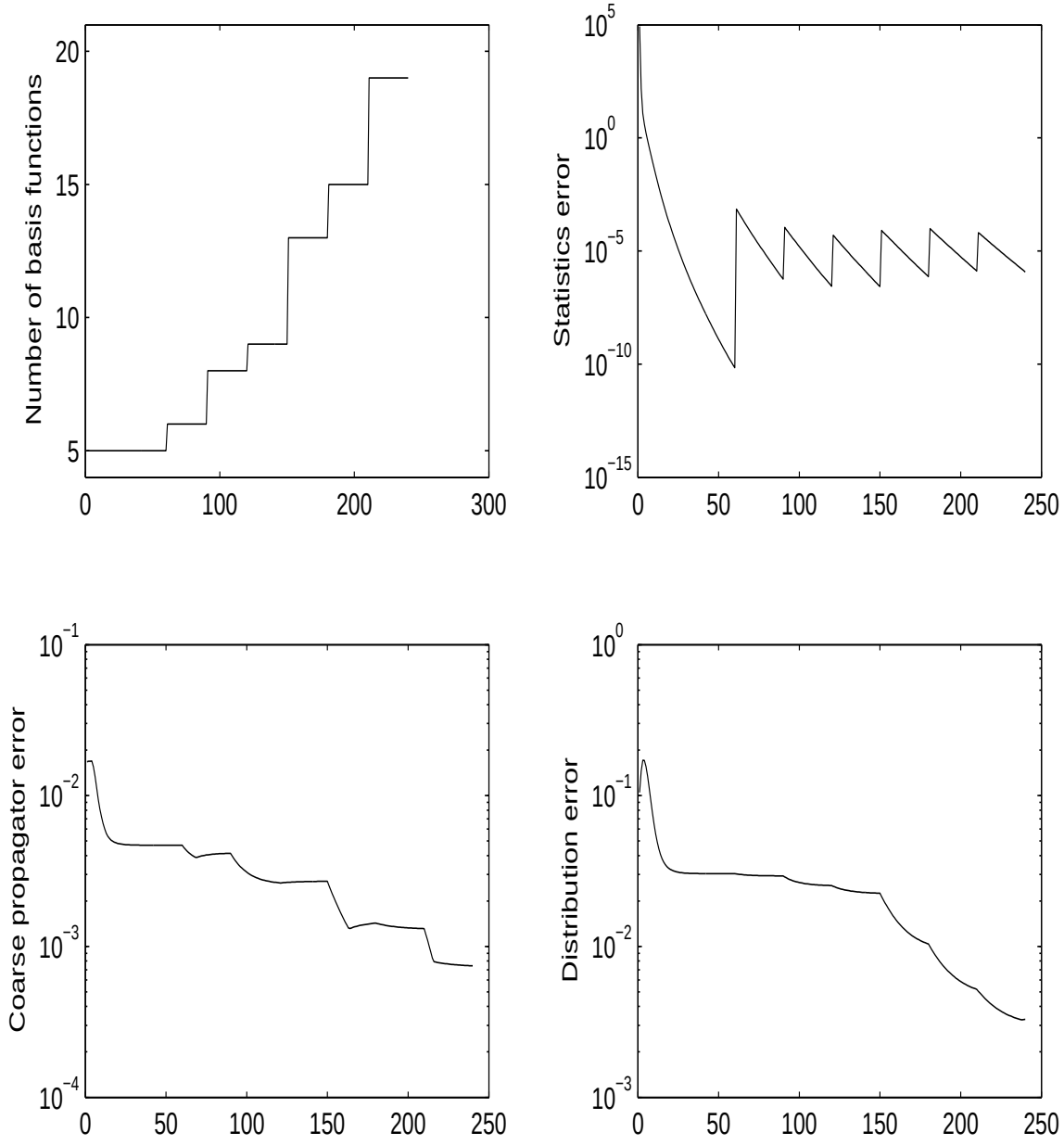

Figure 20: Double well - the progress of the algorithm with NMM estimates for the sixth coarsening level with 128 slices and $\beta = 10$: the number of the basis functions (top-left), the mean-squared error of the statistics in the basis set (top-right), maximal variation error in the coarse-level kernel (bottom-left) and maximal variation error in the estimated equilibrium distribution (bottom-right).

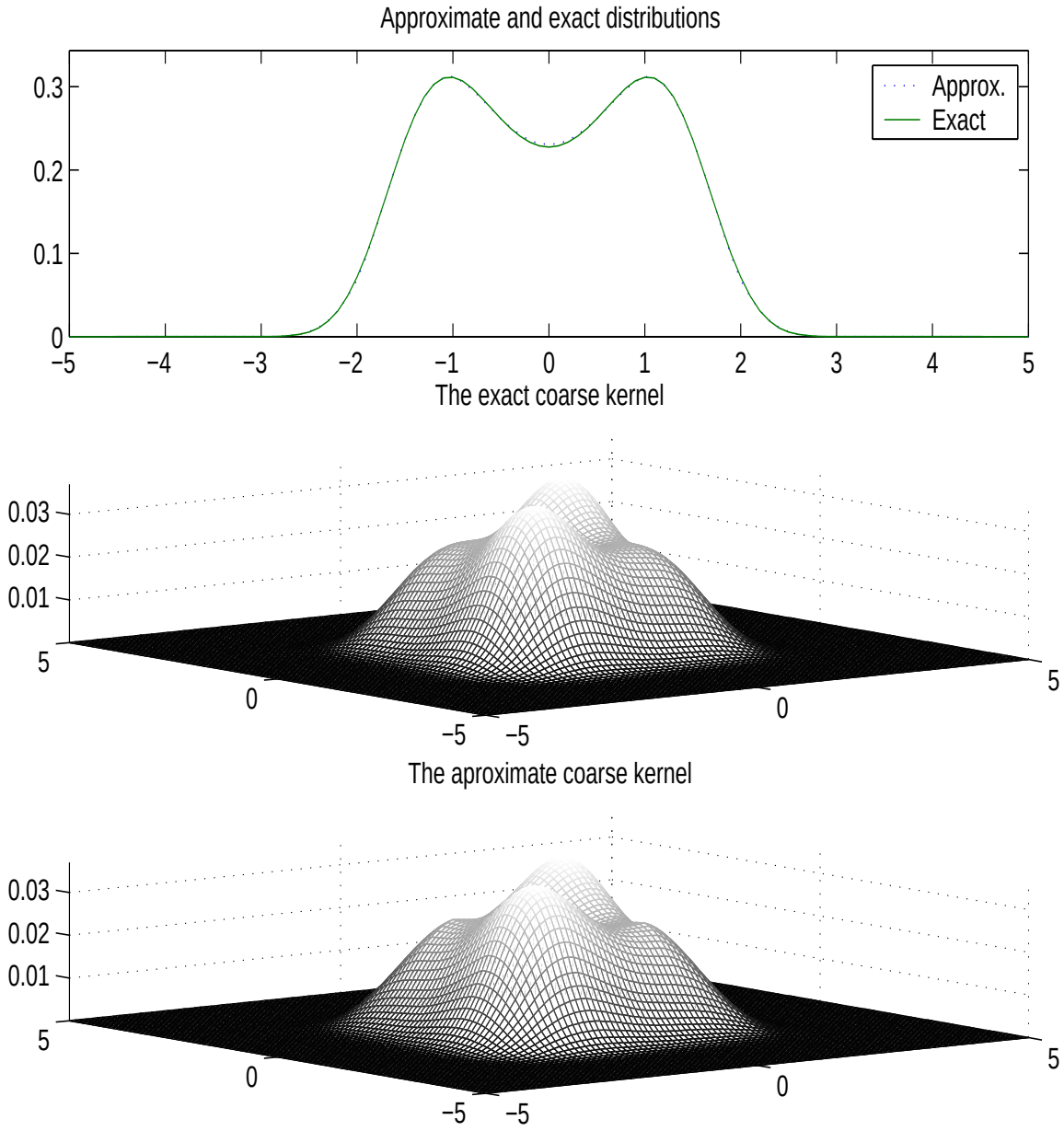

Figure 21: Double well - exact equilibrium distribution $d(x)$ versus its approximation $\hat{d}(x)$ (top), exact coarse level kernel $\rho_{coarse}(x, z)$ (middle) and approximate coarse level kernel $\hat{\rho}_{coarse}(x, z)$ (bottom) for the first coarsening level with $\beta = 10$ with NMM estimates.

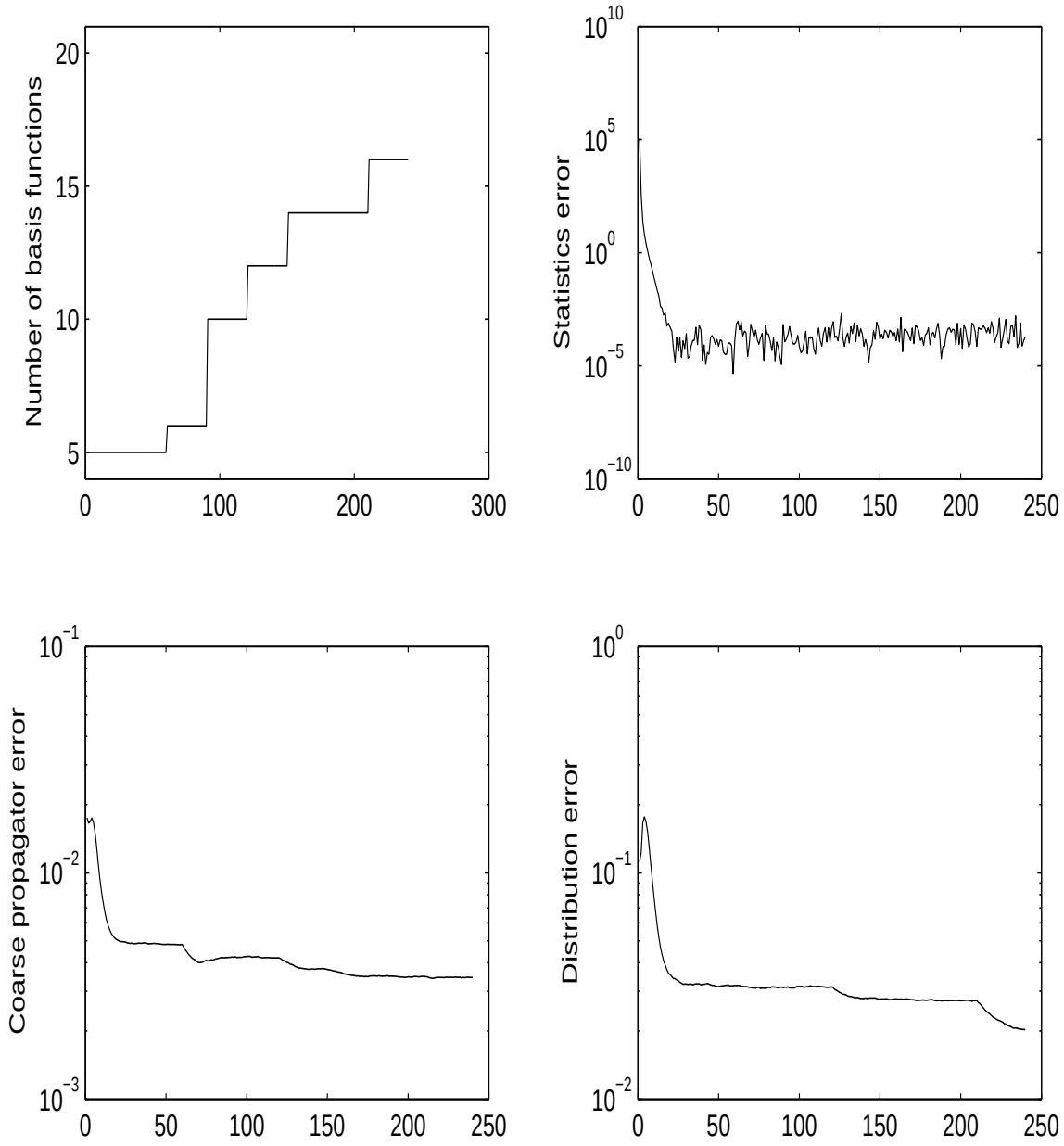

Figure 22: Double well - the progress of the algorithm with MC estimates (10^5 sample per iteration) for the sixth coarsening level with 128 slices and $\beta = 10$: the number of the basis functions (top-left), the mean-squared error of the statistics in the basis set (top-right), maximal variation error in the coarse-level kernel (bottom-left) and maximal variation error in the estimated equilibrium distribution (bottom-right).

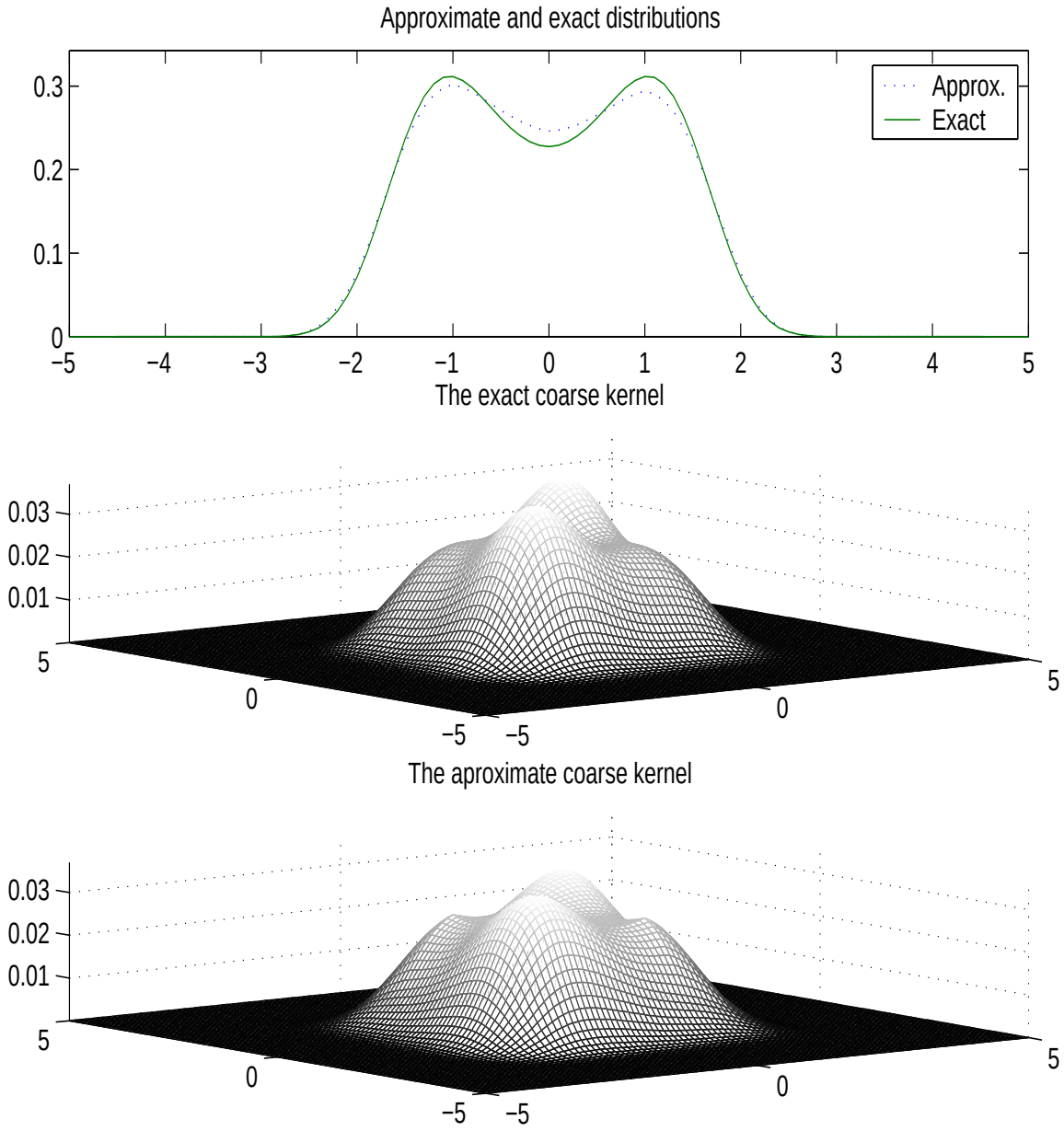

Figure 23: Double well - exact equilibrium distribution $d(x)$ versus its approximation $\hat{d}(x)$ (top), exact coarse level kernel $\rho_{coarse}(x, z)$ (middle) and approximate coarse level kernel $\hat{\rho}_{coarse}(x, z)$ (bottom) for the sixth coarsening level with $\beta = 10$ with MC estimates (10^5 sample per iteration).

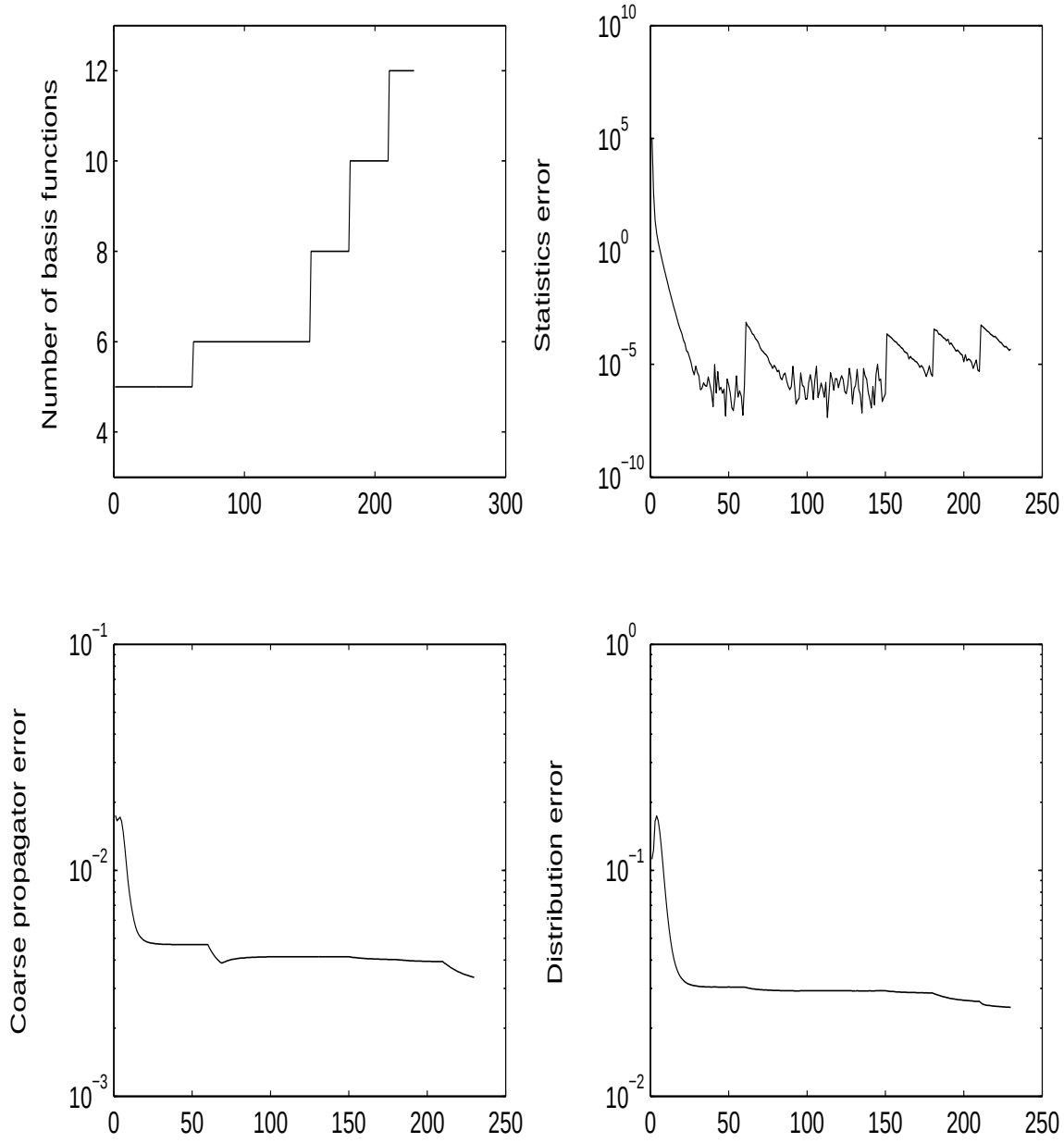

Figure 24: Double well - the progress of the algorithm with MC estimates (10^7 sample per iteration) for the sixth coarsening level with 128 slices and $\beta = 10$: the number of the basis functions (top-left), the mean-squared error of the statistics in the basis set (top-right), maximal variation error in the coarse-level kernel (bottom-left) and maximal variation error in the estimated equilibrium distribution (bottom-right).

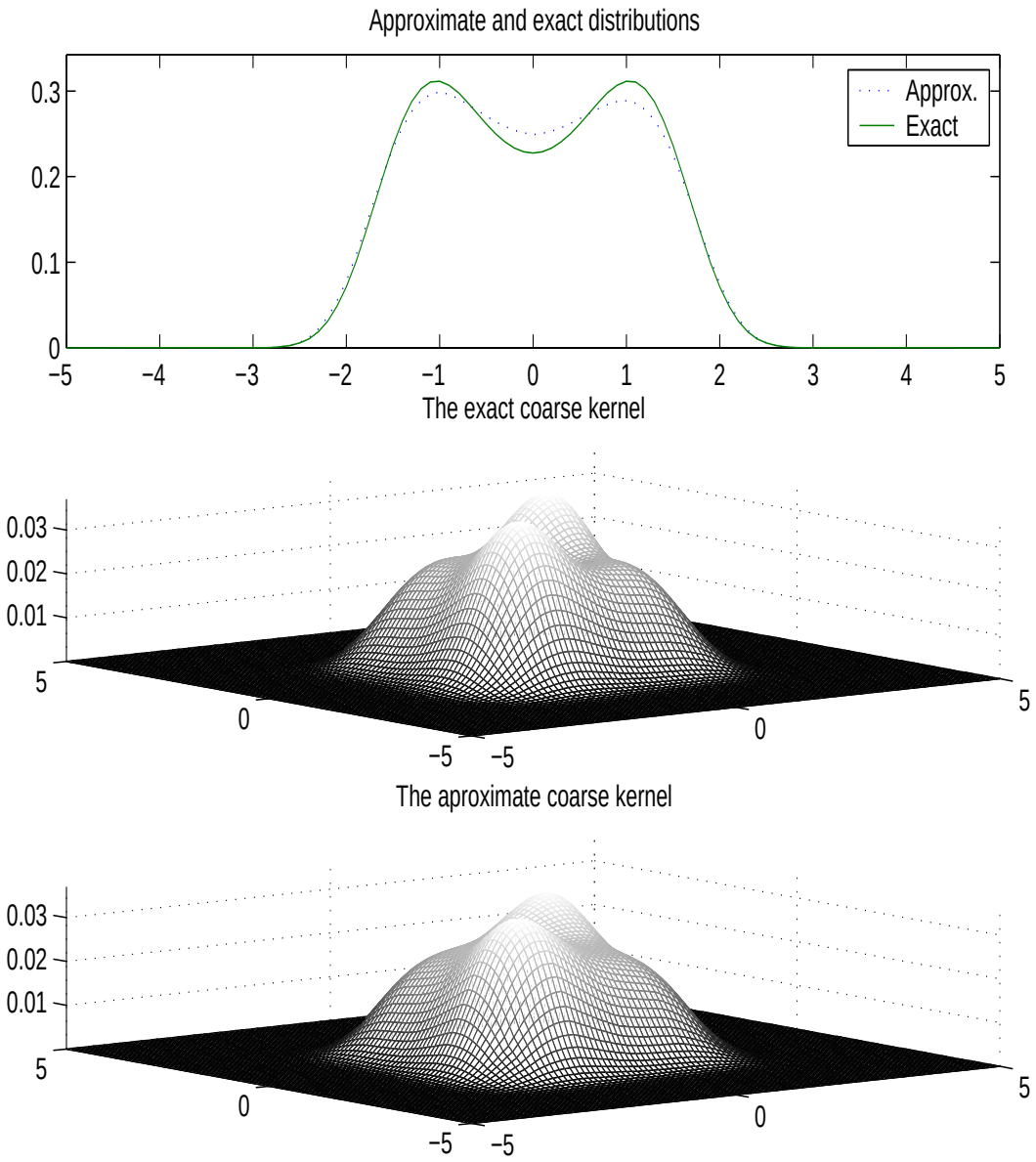

Figure 25: Double well - exact equilibrium distribution $d(x)$ versus its approximation $\hat{d}(x)$ (top), exact coarse level kernel $\rho_{coarse}(x, z)$ (middle) and the approximate coarse level kernel $\hat{\rho}_{coarse}(x, z)$ (bottom) for the sixth coarsening level with $\beta = 10$ with MC estimates (10^7 sample per iteration).

3.3.2 Complex-time calculations

As already mentioned in Section 1, path integrals can also be used to compute the thermally averaged quantum time correlation functions (Thirumalai and Bern, 1983) and thus information about dynamical effects at finite temperature. In the following experiments, we have explored the behavior of our algorithm for the complex time $t_c = t - ih\beta/2$, as employed in the calculations of the symmetrized time correlation function $G_{AB}(t)$ (Thirumalai and Bern, 1984):

$$G_{AB}(t) = Tr[A \exp(iHt_c^*/h) B \exp(-iHt_c/h)], \tag{27}$$

where A and B are quantum mechanical operators.

The complex time problems necessitate several important modifications in our algorithm. The first is the generalization of the basic Fisher scoring algorithm described in Section 3.2.3 above. Another is the utilization of the weighting measure μ already mentioned in Section 3.2.1. While for equilibrium calculations the uniform weighting via the choice of Lebesgue measure μ was adequate, for the complex-time problems such weighting lead to instabilities of the optimization process. Consequently, in the experiments to follow, we have used simple standard Gaussian weighting measure, which filtered out the remote phase-space regions, rendering both fine and coarse level averages calculation more tractable. It is important to note that in the limit of the zero approximation error the choice of the weighting measure is irrelevant, but it could be critical in the practical low-but-not-zero-error situations. Finally, it turned out that for complex-time problems, in the auxiliary quadratic programming problem:

$$\text{argmin}\Big\{\delta c^T I^* I \delta c - 2Re(\delta S^* I)\delta c\Big\}, \tag{28}$$

the matrix I is close to singular, hence the optimization algorithm was quite sensitive to numerical errors in its estimation. In order to further stabilize our algorithm we have used the well-known technique of regularization (Bertsekas, 1995), which in our case amounted to adding a penalty term $C\|\delta\|^2$ to the quadratic program (28). The coefficient C was modified adaptively to keep the approximation error and the penalty term on the same order of magnitude.

Few words are in order regarding the performance measures used in the following experiments. Similarly to equilibrium calculations, we considered the mean-squared statistics error, and, since a penalty term was added in these experiments, the penalized error was monitored as well. Next, due to the use of the weighting (damping) measure μ we have monitored the values of both damped and undamped coarse-level kernel. Finally, as already mentioned above, for the equilibrium calculations one eventually is interested in

the equilibrium distribution. Similarly, for the diagonal quantum operators, the thermally averaged quantum time correlation function can be written as:

$$G_{AB}(t) = Tr[A\exp(iHt_c^*/h)B\exp(-iHt_c/h)] = \int dx_0 dx_{end} A(x_0)B(x_{end})K(x_0, x_{end}), \qquad (29)$$

where $K(x_0, x_{end})$ is what we henceforth refer to as the "end-points-kernel", whose estimation error was monitored as well.

Similarly to the imaginary-time experiments, we have started with the harmonic oscillator. As a first test, we have considered the first coarsening level with $\beta = 1, t = 1$ and 8 slices. Figures 26-29 present the results of the simulation with NMM and MC coarse-level calculations. Similarly to the equilibrium case, we observe very high accuracy of approximation with NMM and slightly lower, but still very high, accuracy with MC estimation. It is interesting to mention that, starting with the basis $\{x^2, z^2, (x-z)^2, 1\}$, the algorithm added two refinements of $(x-z)^2$ as it went along - first with the support $[-4, 4] \times [-4, 4]$ and latter with the support $[-2, 2] \times [-2, 2]$. Apparently, this indicates that the main source of approximation errors was the complex correlative behavior in the central region.

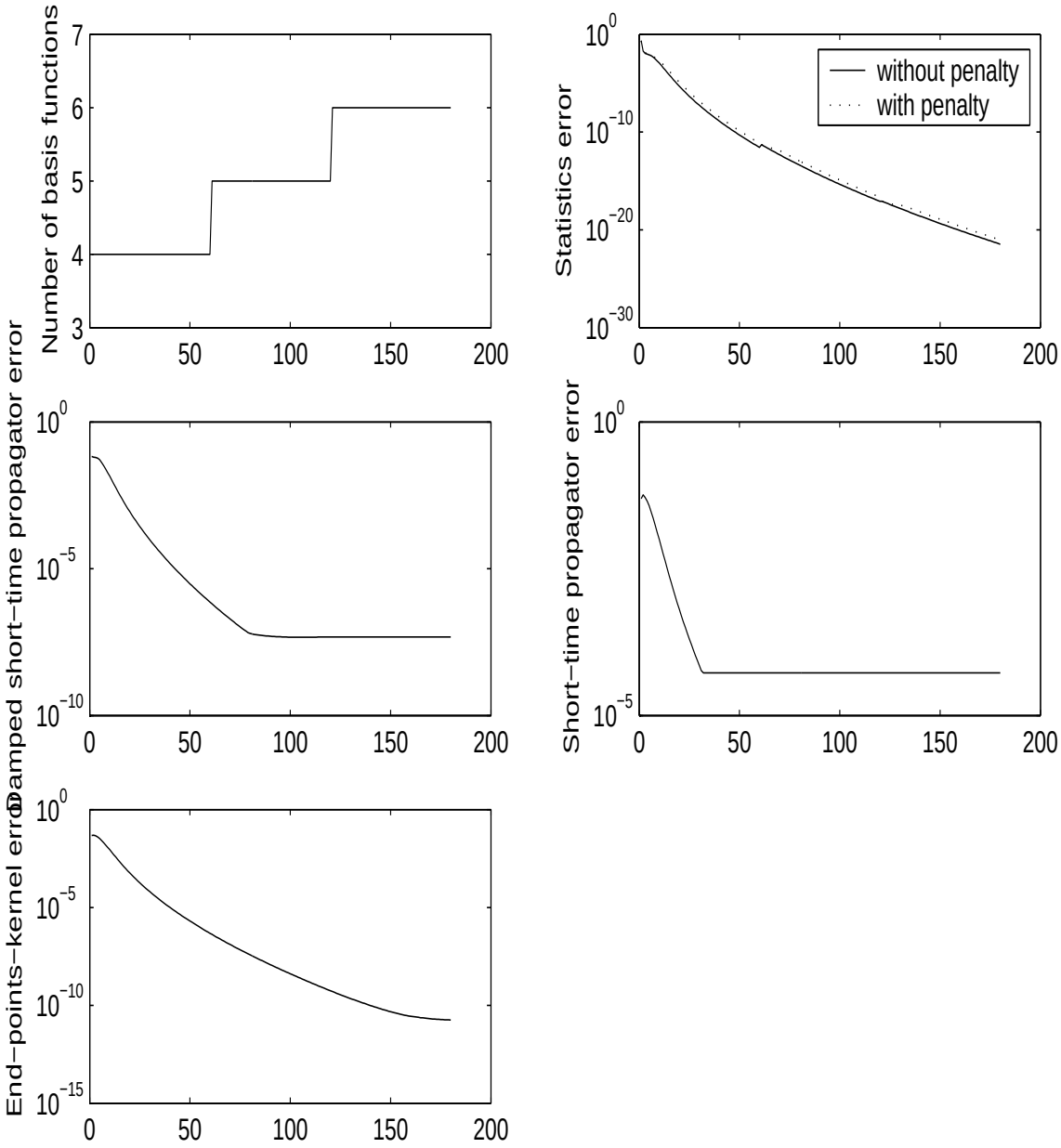

Figure 26: Harmonic oscillator - the progress of the algorithm with NMM estimates for the first coarsening level with 8 slices and $\beta = 1, t = 1$: the number of the basis functions (top-left), the mean-squared error of the statistics in the basis set with and without the penalty term (top-right), maximal variation error in the damped (middle-left) and undamped (middle-right) coarse-level kernel and maximal variation error in the estimated end-points-kernel (bottom).

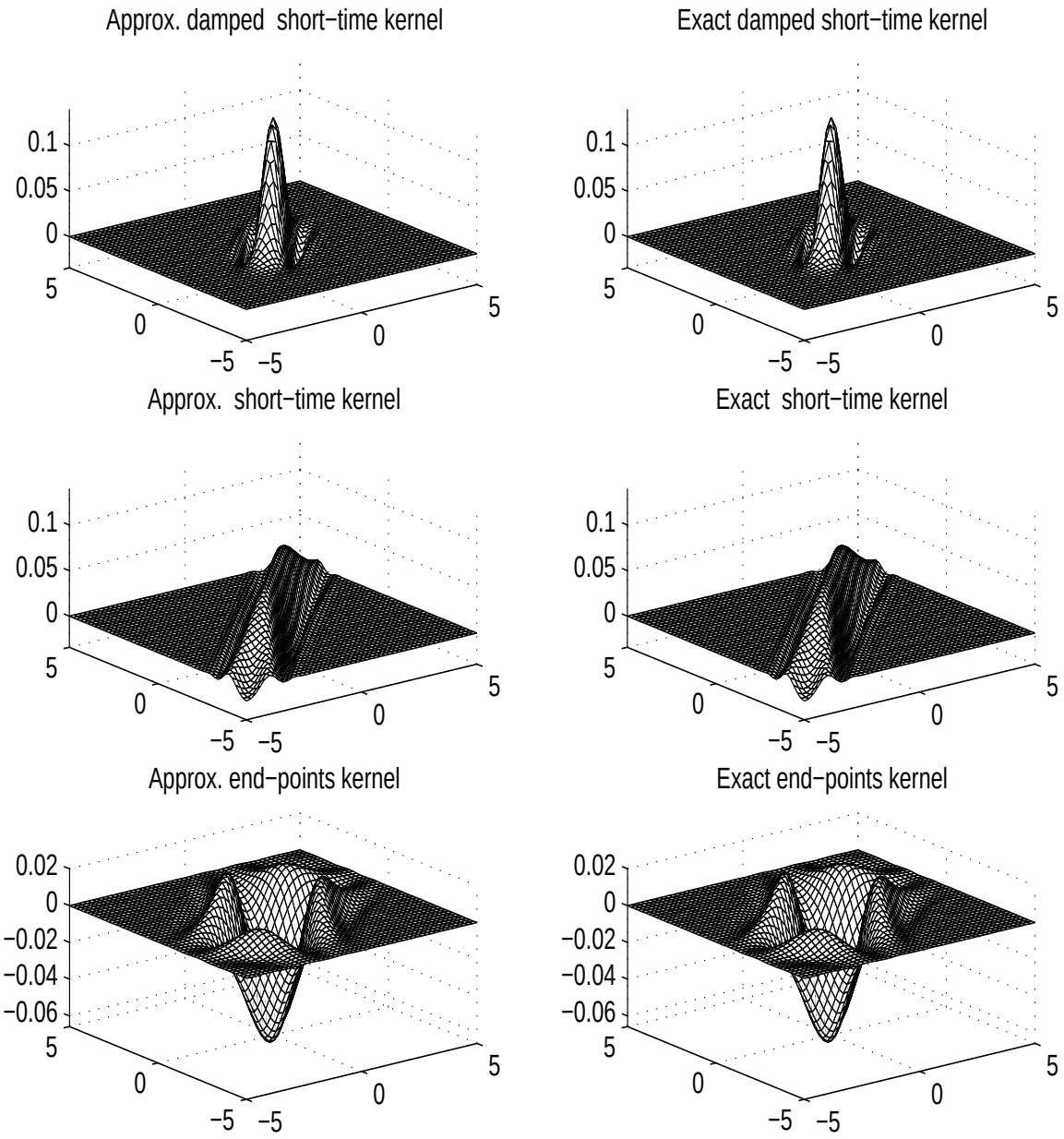

Figure 27: Harmonic oscillator - the real part of the exact and the approximated damped coarse-level kernel (top), undamped coarse-level kernel (middle) and the end-point-kernel (bottom) for the first coarsening level with $\beta = 1, t = 1$ with NMM estimates.

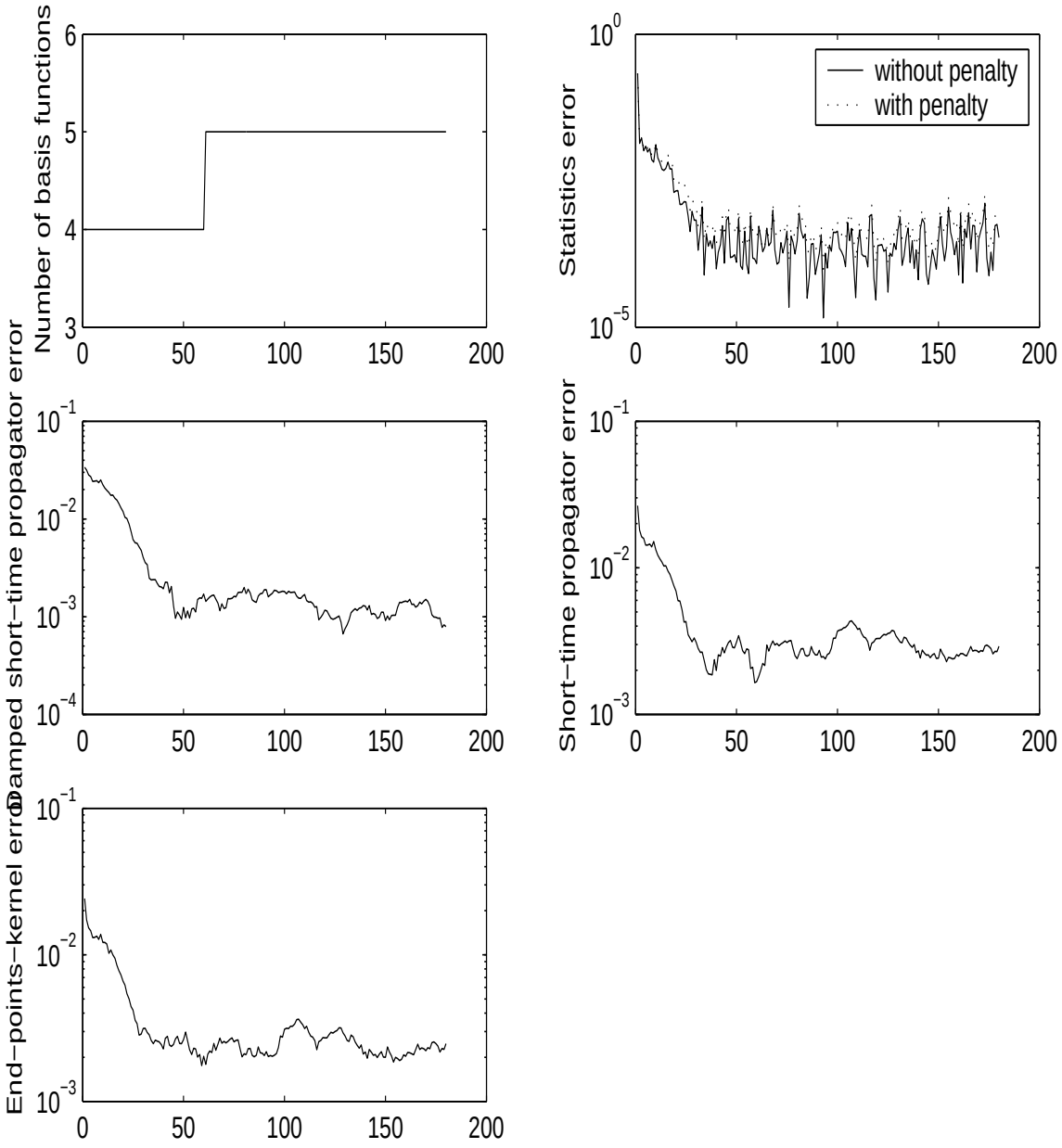

Figure 28: Harmonic oscillator - the progress of the algorithm with MC estimates (10^4 samples per iteration) for the first coarsening level with 8 slices and $\beta = 1, t = 1$: the number of the basis functions (top-left), the mean-squared error of the statistics in the basis set with and without the penalty term (top-right), maximal variation error in the damped (middle-left) and undamped (middle-right) coarse-level kernel and maximal variation error in the estimated end-points-kernel (bottom).

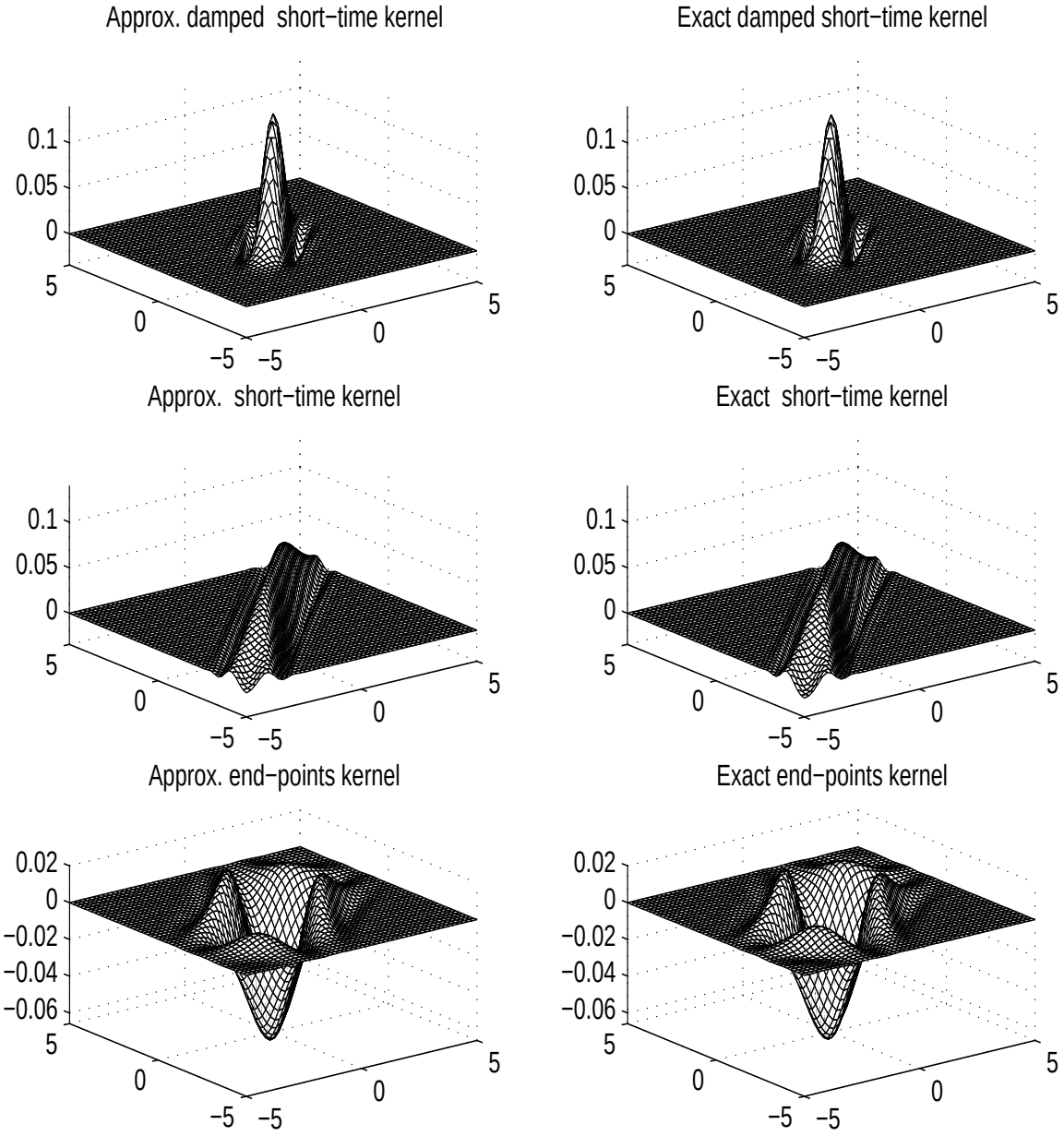

Figure 29: Harmonic oscillator - the real part of the exact and the approximated damped coarse-level kernel (top), undamped coarse-level kernel (middle) and the end-point-kernel (bottom) for the first coarsening level with $\beta = 1, t = 1$ with MC estimates (10^4 samples per iteration).

As a more stringent test we have considered the more difficult case of the third coarsening level with $\beta = 1, t = 10$ and 128 slices, presented in Figures 30-31. While the calculations involving NMM still produced very accurate results, the MC-based algorithm with as many as 10^7 samples per iteration failed to approach the optimal value of the parameters. Moreover, as can be seen from the upper-right chart in

Figure 31, after initial reduction of the error in the statistics, the algorithm apparently became dominated by a diffusion-like process, which increased the value of the parameters without significantly affecting the value of the statistics. This indicates that the algorithm still needs to be improved, in order to produce more robust results for the more realistic large scale problems.

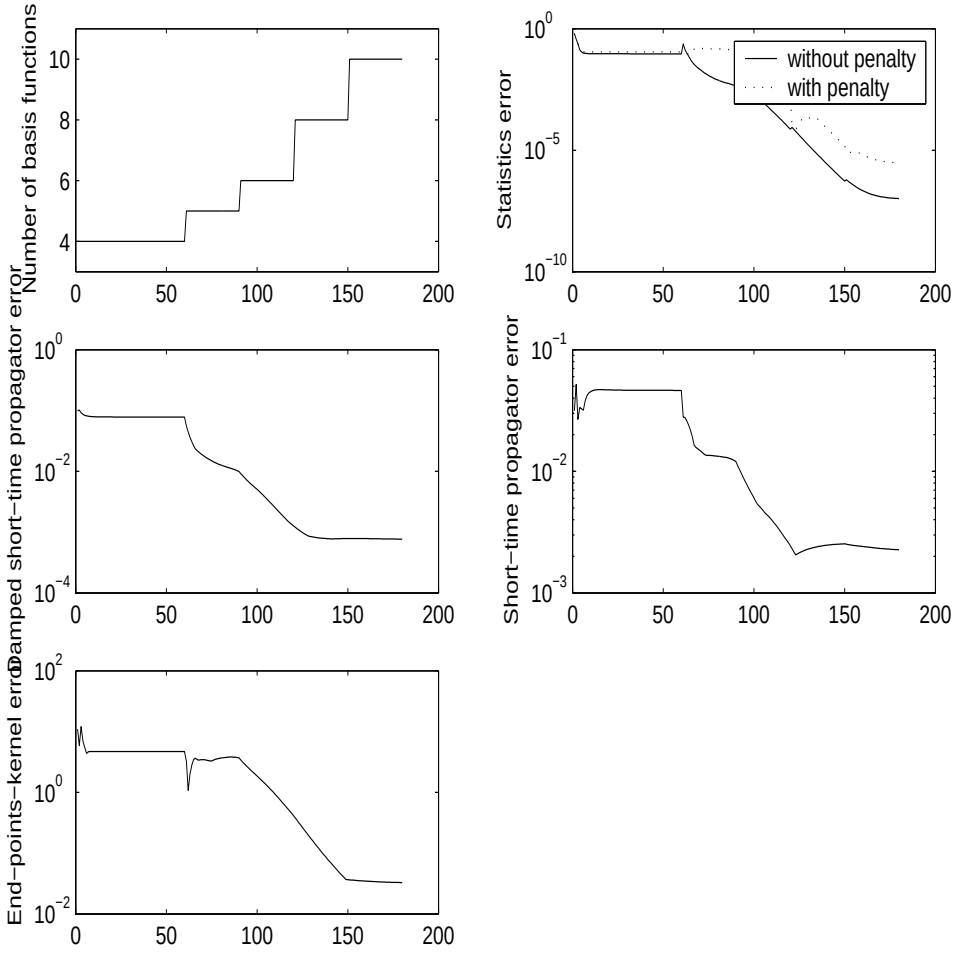

Figure 30: Harmonic oscillator - the progress of the algorithm with NMM estimates for the third coarsening level with 128 slices and $\beta = 1, t = 10$: the number of the basis functions (top-left), the mean-squared error of the statistics in the basis set with and without the penalty term (top-right), maximal variation error in the damped (middle-left) and undamped (middle-right) coarse-level kernel and maximal variation error in the estimated end-points-kernel (bottom).

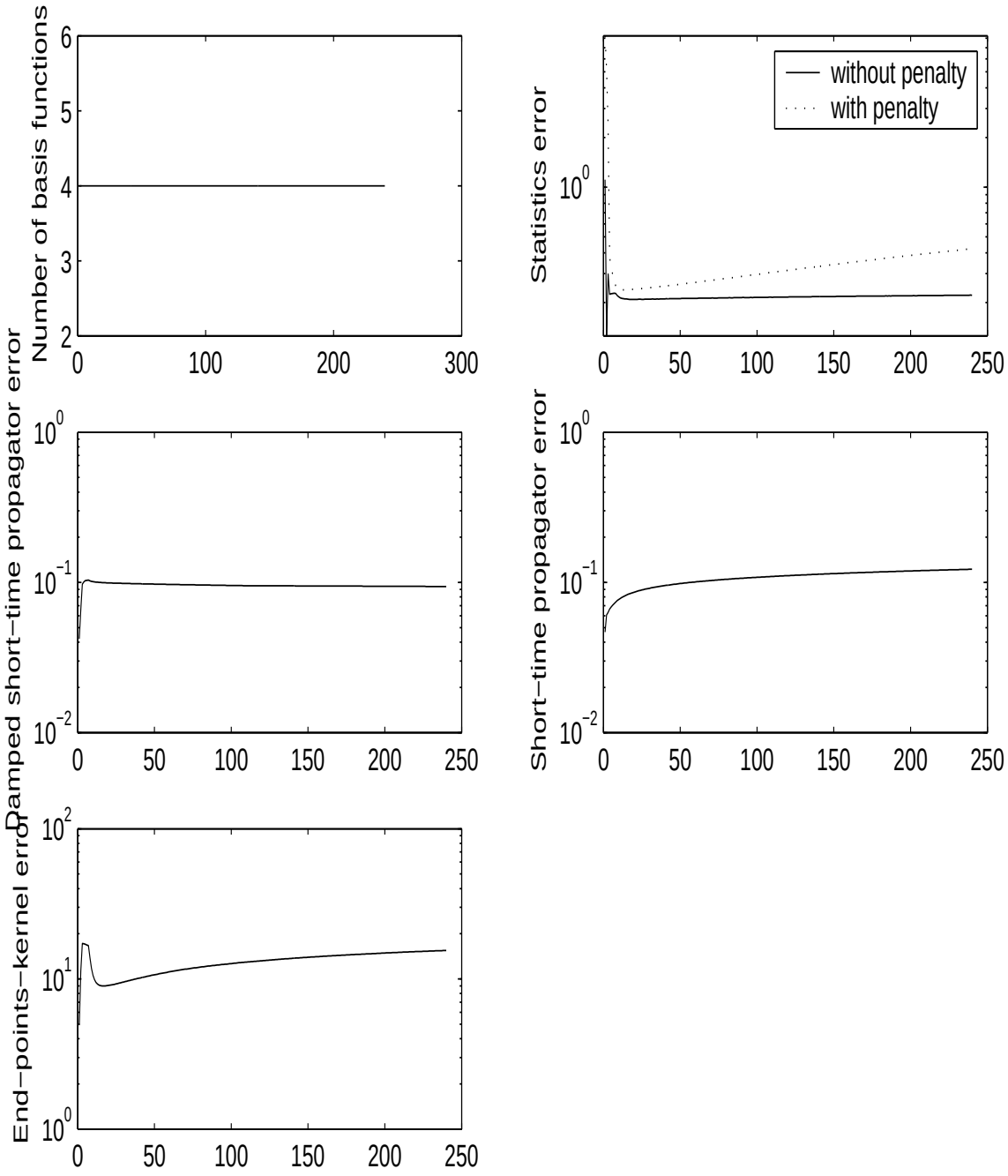

Figure 31: Harmonic oscillator - the progress of the algorithm with MC estimates (10^7 samples per iteration) for the third coarsening level with 128 slices and $\beta = 1, t = 10$: the number of the basis functions (top-left), the mean-squared error of the statistics in the basis set with and without the penalty term (top-right), maximal variation error in the damped (middle-left) and undamped (middle-right) coarse-level kernel and maximal variation error in the estimated end-points-kernel (bottom).

We have also repeated the experiments above for the double-well problem. The results were qualitatively the same, hence we do not present a detailed report here.

4 Conclusions and future research

The results presented in this report illustrate the plausibility of using the SU methodology in the context of Feynman path integrals. Still, there are some limitations of the present algorithm, which need to be considered in order to be able to tackle realistic PI problems.

First, as the double-well experiment have demonstrated, one needs to develop a more sophisticated stopping conditions for the optimization algorithm, since the mean-squared statistics error has been found to be unreliable estimate of the accuracy of the coarse-level Hamiltonian approximations.

Next, the optimization algorithms itself should be improved (i.e. using the extended Kalman filtering methodology (Bertsekas, 1995)) in order to allow more efficient use of the Monte Carlo sampling.

Finally, while producing some promising results for the simple test problems, it is not clear whether the present methods for the construction of the basis sets for approximating the coarse-level Hamiltonian is practical, when one considers more realistic problems. The more general methodology involving space-averaging and alternative basis construction methods described in Section 3 is expected to produce more stable and generic results. It is our feeling that one also needs more problem-specific knowledge in order to construct problem-tailored basis sets, which allow sparse and accurate approximations.

While still a lot of research work is needed in order to turn the SU methodology into practical algorithms, still the preliminary results presented in this report are promising. In particular, an important by-product of this research is a simple procedure for improving the accuracy of short-to-medium-time kernels by optimizing the coefficients of the default basis functions.

Acknowledgement

This research was supported by THE ISRAEL SCIENCE FOUNDATION (grant No. 295/01).

References

Bai, D. (2004). Multiscale computation of polypeptide backbone. arXiv.cond-mat/0312185, submitted to *Journal of Computational Chemistry*.

Bai, D. and Brandt, A. (2000). Multiscale computation of polymer models. In (Brandt et al., 2001), pages 250–266.

Bertsekas, D. P. (1995). Incremental Least Squares methods and the Extended Kalman Filter. *SIAM J. on Optimization*, 6:807–822.

Bertsekas, D. P. (1995). *Nonlinear Programming*. Athena Scientific, Belmont, MA.

Brandt, A. (1977). Multi-level adaptive solutions to boundary value problems. *Math. Comp.*, 31:333–390.

Brandt, A. (1982). Guide to multigrid development. In Hackbusch, W. and Trottenberg, U., editors, *Multigrid Methods*, pages 220–312. Springer-Verlag.

Brandt, A. (1986). Algebraic multigrid theory: The symmetric case. *Appl. Math. Comp.*, 19:23–56.

Brandt, A. (1991). Multilevel computations of integral transforms and particle interactions with oscillatory kernels. *Comp. Phys. Comm.*, 65:24–38.

Brandt, A. (1992). Multigrid methods in lattice field computations. *Nucl. Phys. B (Proc. Suppl.)*, 26:137–180.

Brandt, A. (2001). Multiscale scientific computation: review 2001. In Barth, T., Chan, T., and Haimes, R., editors, *Multiscale and Multiresolution Methods: Theory and Applications*, pages 1–96. Springer Verlag, Heidelberg. Available in www.wisdom.weizmann.ac.il/∼achi.

Brandt, A., Bernhole, J., and Binder, K., editors (2001). *Multiscale Computational Methods in Chemistry*. NATO Science Series, Computer and Systems Sciences, Vol. 177, IOS Press, Amstredam.

Brandt, A., Galun, M., and Ron, D. (1994). Optimal multigrid algorithms for calculating thermodynamic limits. *J. Stat. Phys.*, 74:313–348.

Brandt, A. and Ilyin, V. (2001). Multilevel approach in statistical physics of liquids. In (Brandt et al., 2001), pages 187–197.

Brandt, A. and Livshits, I. (1997). Wave-ray multigrid method for standing wave equations. *Electronic Trans. Num. An.*, 6:162–181.

Brandt, A. and Lubrecht, A. (1990). Multilevel matrix multiplication and fast solution of integral equations. *J. Comput. Phys.*, 90:348–370.

Brandt, A., McCormick, S., and Ruge, J. (1982). Algebraic multigrid (AMG) for automatic multigrid solution with application to geodetic computations. Technical report, Institute for Computational Studies, POB 1852, Fort Collins, Colorado.

Brandt, A. and Ron, D. (2001). Renormalization multigrid (RMG): Statistically optimal renormalization group flow and coarse-to-fine Monte Carlo acceleration. *J. Stat. Phys.*, 102:231–257.

Brandt, A., Ron, D., and Amit, D. (1986). Multi-level approaches to discrete-state and stochastic problems. In *Multigrid Methods, II*, pages 66–99. Springer-Verlag.

Briggs, W., Henson, V., and McCormick, S. (2000). *A Multigrid Tutorial*. SIAM, 2nd edition.

Ceperley, D. and Pollock, E. (1986). Path integral computation of the low temperature properties of liquid 4he. *Phys. Rev. Lett.*, 56:351.

Ceperley, D. M. (1995). Path integrals in the theory of condensed helium. *Reviews of Modern Physics*, 67:279.

Ceperley, D. M. (2003). Metropolis methods for quantum monte carlo simulations. In Gubernatis, J. E., editor, *The Monte Carlo Method in the Physical Sciences, AIP Conference Proceedings*, volume 690, pages 85–98.

Doll, J. and Frreeman, D. (1988). Stationary phase monte carlo methods. *Advances in Chemical Physics*, 73:289.

Feynman, R. P. (1948). Space-time approach to non-relativistic quantum mechanics. *Reviews of Modern Physics*, 20(2):367–387.

Feynman, R. P. (1972). *Statistical Mechanics*. Addison-Wesley, Redwood City.

Filinov, V. (1986). Construction of a monte carlo method for calculating feynmann integrals. *U.S.S.R. Comput. Maths. Math. Phys.*, 26:21–28.

Fisher, M. (1998). Renormalization group theory: Its basis and formulation in statistical physics. *Rev. Mod. Phys.*, 70(2):653–681.

Goodman, J. and Sokal, A. (1986). Multigrid monte carlo methods for lattice field theories. *Phys. Rev. Lett.*, 56:1015–1018.

Hackbusch, W. (1985). *Multigrid Methods and Applications*. Springer, Berlin.

Janke, W. and Sauer, T. (1993). Path integral monte carlo using multigrid techniques. *Chemical Physics Letters*, 201(5/6),:499–505.

Jaynes, E. (1957). Information theory and statistical mechanics. *Physical Review*, 106:620–620.

Klauder, J. R. and Skagerstam, B. S. (1985). *Coherent States — Applications in Physics and Mathematical Physics*. World Scientific, Singapore.

Kohn, W. and Sham, L. (1965). Self-consistent equations including exchange and correlation effects. *Physical Review*, 140:1133–1138.

Mack, g. and Pordt, a. (1985). Convergent perturbation expansions for euclidean quantum field theory. *Comm. Math. Phys.*, 87:267.

Mak, C. and Egger, R. (1999). A multilevel blocking approach to the sign problem in real-time quantum monte carlo simulations. *J. Chem. Phys.*, 110:12.

Makri, N. (1991). Feymann path integration in quantum dynamics. *Computer Physics Communications*, 63:389–414.

Makri, N. and Miller, W. H. (1987). Monte carlo integration with oscillatory integrands: implications for feymann path integration in real time. *Chemical Physics Letters*, 139:10–14.

Rao, R. (1948). The utilization of multiple measurement in problems of biological classifications. *Journal of the Royal Statistical Society B*, 10:159–193.

Ruge, J. and Stüben, K. (1987). Algebraic multigrid. In McCormick, S. F., editor, *Multigrid Methods*, pages 73–130. SIAM, Philadelphia.

Thirumalai, D. and Bern, B. J. (1983). On the calculation of time correlation functions in quantum systems: Path integral techniques. *Journal of Chemical Physics*, 79:5029–5033.

Thirumalai, D. and Bern, B. J. (1984). Time correlation functions in quantum systems. *Journal of Chemical Physics*, 81:2512–2513.

Thirumalai, D., Bruskin, E. J., and Berne, B. J. (1983). An iterative scheme for the evaluation of discretized path integrals. *Journal of Chemical Physics*, 79:5063–5069.

Trottenberg, U., Oosterlee, C., and Schüller, A. (2000). *Multigrid*. Academic Press, London.

Trotter, H. (1959). On the product of semi-groups of operators. *Proceedings of the American Mathematical Society*, 10:545–551.

Wilson, K. (1983). The renormalization group and critical phenomena. *Rev. Mod. Phys.*, 55:583–600. 1982 Nobel Prize Lecture.