Electronic Transactions on Numerical Analysis.
Volume 6, pp. 1-34, December 1997.
Copyright © 1997, Kent State University.
ISSN 1068-9613.

THE GAUSS CENTER RESEARCH IN MULTISCALE SCIENTIFIC COMPUTATION[*]

ACHI BRANDT[†]

Abstract. The recent research of the author and his collaborators on multiscale computational methods is reported, emphasizing main ideas and inter-relations between various fields, and listing the relevant bibliography. The reported areas include: top-efficiency multigrid methods in fluid dynamics; atmospheric data assimilation; PDE solvers on unbounded domains; wave/ray methods for highly indefinite equations; many-eigenfunction problems and ab-initio quantum chemistry; fast evaluation of integral transforms on adaptive grids; multigrid Dirac solvers; fast inverse-matrix and determinant updates; multiscale Monte-Carlo methods in statistical physics; molecular mechanics (including fast force summation, fast macromolecular energy minimization, Monte-Carlo methods at equilibrium and the combination of small-scale equilibrium with large-scale dynamics); image processing (edge detection and segmentation); and tomography.

Key words. scientific computation, multiscale, multi-resolution, multigrid, fluid dynamics, atmospheric flows, data assimilation, wave problems, Dirac equations, inverse matrix, Schrödinger operator, Monte-Carlo algorithms, critical slowing down, molecular mechanics, fast force summation, energy minimization, integro-differential equations, tomography, image processing, edge detection, segmentation.

AMS subject classifications. 34A50, 35A40, 44–04, 45–04, 65C05, 65F10, 65F15, 65F40, 65K10, 65M30, 65M50, 65M55,65N22, 65N25, 65N38, 65N55, 65R10, 65R20, 65Y05, 68U10, 70-08, 76-04, 76M20, 81-08, 81T80, 82-08, 82B80, 82C80, 92E99.

1. Introduction.

The Carl F. Gauss Center for Scientific Computation was established in 1993 jointly by the Minerva Stiftung Gesellschaft für die Forschung m.b.H., Germany, and by the Weizmann Institute of Science, Rehovot, Israel. Its mission is to develop new fundamental computational approaches in physics, chemistry, applied mathematics and engineering, focusing in particular on advanced *multiscale* (“multi-resolution”, “multilevel”, “multigrid”, etc.) methods.

1.1. Multiscale computation.

It is well known that some of the major bottlenecks in science and engineering are computational in nature. The detailed understanding and design of large molecules, condensed matter and chemical processes, for example, could in principle be achieved just by computation, since the underlying equations are fully known; except that our computing capabilities are inadequate for such tasks. The same is true for the calculation of elementary particle properties from first principles, or for the design of fusion reactors or airplane maneuvers, and for many other engineering and scientific endeavors. All would be greatly facilitated if unlimited computing power were available—or if much better algorithms could be devised.

Indeed, just building ever faster machines will not do. With current computational methods the needed amount of computer processing often increases too steeply with the rise in problem size, so that no conceivable computer will be adequate. Completely new mathematical approaches are needed.

Most computational super-problems in science and engineering share some common features. For example, all of them involve a multitude of variables located in a low dimensional

[*] Received May 17, 1997. Accepted October 17, 1997. Communicated by C. Douglas. This research was supported by grants No. G0289-065.07/93 from the German-Israeli Foundation for Research and Development (GIF), No. 379/93 from the Israeli Academy of Science and Humanities, and No. NAG 1 1823 from NASA; by grants from the U.S. Air Force Office of Scientific Research and the Material and Manufacturing Directorate of the Air Force Research Laboratory; and by the Carl F. Gauss Minerva Center for Scientific Computation at the Weizmann Institute of Science.

[†] Weizmann Institute of Science, Rehovot 76100, Israel

space (e.g., the four dimensional physical space-time). Closer examination reveals that the computational complexity of these problems results directly from this spatial nature, in several general ways that come up again and again, in different disciplines and in all kinds of guises. Past studies have demonstrated that such complexities can be effectively overcome, or drastically reduced, by multiscale algorithms.

Indeed, any many-variable problem defined in physical space can have an approximate description at any given length scale of that space: a continuum problem can be discretized at any given resolution; collective motions of a many-body system can be organized at any given characteristic length; etc. The multiscale algorithm recursively constructs a *sequence* of such descriptions at increasingly larger (coarser) scales, and combines local processing (relaxation of equations, simulation of statistical relations, etc.) at each scale with various inter-scale interactions. Typically, the evolving solution (or the simulated equilibrium) on each scale recursively dictates the *equations* (or the Hamiltonian) on coarser scales while modifying the *solution* (or configuration) on finer scales. In this way large-scale changes are effectively performed on coarse grids, based on information previously gathered from finer grids.

As a result of such multilevel interactions, the fine scales of the problem can be employed very sparingly, and sometimes only at special and/or representative small regions. Moreover, the inter-scale interactions can eliminate various kinds of difficulties, such as: slow convergence (in minimization processes, PDE solvers, etc.); critical slowing down (in statistical physics); ill-posedness (e.g., of inverse problems); large-scale attraction basin traps (in global optimization and statistical simulations); conflicts between small-scale and large-scale representations (e.g., in wave problems); numerousness of interactions (in many body problems or integral equations); the need to produce many fine-level solutions (e.g., in optimal control) or very many fine-level independent samples (in statistical physics); etc. Also, the multiscale interactions tend to bring out the large-scale dynamics, or the macroscopic equations, of the physical system, which is often the very objective of the entire calculation.

Since the local processing (relaxation, etc.) in each scale can be done in parallel at all parts of the domain (e.g., at all cells of a given lattice), the multiscale algorithms, based on such processing, are ideal for implementation on massively parallel computers. Indeed, many problems cannot be efficiently solved by such computers without employing a multiscale procedure.

1.2. Current research directions at the Gauss Center.

Over the last three years, the research at the Gauss Center has involved the following directions.

1. New multigrid methods for steady-state fluid dynamics at all Mach and Reynolds numbers, and other non-elliptic stationary PDE systems (see Sec. 2 below).
2. Multilevel approaches to time-dependent partial-differential equations, emphasizing applications to oceanic and atmospheric flows (see Sec. 2.4).
3. Direct multigrid solvers for inverse problems, including system identification (e.g., impedance tomography; see in Sec. 13) and data assimilation (in atmospheric simulations — Sec. 4).
4. Optimal control: Feedback control via very fast updating of open-loop solutions, based on their multiscale representations.
5. Optimal location of singularities of PDE systems (e.g., location of the nucleons in electronic structure calculations), integrated into the multigrid PDE solver (Sec. 6.1).
6. New multilevel algorithms for highly indefinite (e.g., standing wave) problems (Sec. 5).
7. Multigrid solvers for the Dirac equations arising in quantum field theory (Sec. 8).
8. Compact multiresolution representation of the inverse matrix of a discretized dif-

ferential operator; fast updating of the inverse matrix and of the value of the determinant upon changing an arbitrary term in the matrix itself; with application to the QCD fermionic interaction (Sec. 9).

9. Collective multiscale representation and fast calculation of many eigenfunctions of a differential operator, e.g., the Schrödinger operator in condensed-matter electronic-structures calculations (Sec. 6).
10. Multiscale Monte-Carlo algorithms for eliminating both the critical slowing down and the volume factor in increasingly advanced models of statistical physics (Sec. 10).
11. Multigrid Monte-Carlo approaches for solving the high-dimensional (several-particle) Schrödinger equation by real-time path integrals.
12. Introducing multiscale computations to many-particle calculations, including fast evaluation of forces, fast convergence to local and global ground states, fast equilibration, and large time steps, with application to molecular mechanics (Sec. 11); a new approach to molecular dynamics, based on stochastic implicit time steps (Sec. 11.6).
13. Multiscale approaches to molecular docking.
14. Multigrid methods for integro-differential equations, on adaptable grids, with applications to tribology (Sec. 7).
15. Multiscale methods for the fast evaluation and inversion of the Radon transform (Sec. 13); applications to X-ray tomography and airplane and satellite radar reconstruction.
16. Multiscale algorithms for early vision tasks such as surface reconstruction, edge and fiber detection, segmentation, and meaningful picture coarsening (Sec. 12).
17. Rigorous quantitative theory for predicting the performance of multigrid solvers (see [20]).

A survey of main ideas, current developments and future perspectives in these various directions is given in the following sections. The work in directions #4 and #13, which is in a preliminary stage, is *not* reported below.)

2. Computational Fluid Dynamics. *(with Dr. John Ruge (supported by NASA) and with Ph.D. student Boris Diskin)*

2.1. Background and objectives. An efficient multigrid algorithm for steady-state incompressible viscous flows in two dimensions appeared already in 1972 [7], a relatively efficient multigrid solver for a compressible inviscid transonic flow was already demonstrated in 1975 [80], and a fully efficient solver for a *system* of *several* coupled differential equations, characteristic to CFD, was presented already in 1978 [23]. However, in the decades that followed, the development in this area has not been really satisfactory. In particular, the efficiency of solvers for non-elliptic steady-state systems (such as Euler and high-Reynolds Navier-Stokes equations) has lagged several orders of magnitude behind the ideal efficiency that had been attained for general elliptic systems. Although the main reasons for this inefficiency have also been understood for a long time (see for example [11]), the recommended cures seemed complicated, and code developers opted for partial efficiency. The leading method has been based on multi-stage pseudo-time-stepping relaxation schemes [59], [60]. Although such schemes can be optimized to damp high-frequency errors [85], the resulting algorithms are still relatively slow, because some *intermediate* (neither high-frequency nor very smooth) “characteristic components” cannot adequately be reduced by coarse grids (cf. [11], [43]). Other multigrid solvers were based on incomplete LU decomposition (ILU) and related relaxation schemes [87], [83], [79]. While such schemes give excellent results in some cases, they cannot cure the aforementioned trouble of characteristic components in general transonic flows, especially in three dimensions. (Also, much of the efficiency of ILU schemes depends on their sequential marching, hence the performance on massively parallel machines will drastically diminish.) The same is true for other methods (e.g., based on defect corrections) which seem not even to identify that basic trouble.

More generally, all these attempted solution methods have failed to decompose the solution process into separate treatments of each factor of the PDE principal determinant, and therefore did not identify, let alone treat, the separate difficulties associated with each such factor. In fact, in CFD, each of these factors may have different ellipticity measures (some are uniformly elliptic, others are non-elliptic at some or all of the relevant scales) and/or different set of characteristic surfaces, requiring different relaxation/coarsening procedures.

The objective of our recent work has been to develop and demonstrate methods that solve non-elliptic *steady-state* problems in general, and high-Reynolds stationary flow problems in particular, at the same "textbook multigrid efficiency" typically attained for uniformly elliptic systems. The methods, again as in the elliptic case, will allow local refinements and high degree of parallel processing.

Solvers for *time-dependent* flow problems are in principle simpler to develop than their steady-state counterparts. Using semi-implicit or fully implicit discretizations, large and adaptable time steps can be used, and parallel processing across space *and time* is feasible [10], [16]. The resulting system of equations (i.e., the system to be solved at each time step) is much easier than the steady-state system because it has better ellipticity measures (due to the time term), it does not involve the difficulties associated with recirculation, and it comes with a good first approximation (from the previous time step). A simple multigrid "F cycle" at each time step should solve the equations much below the *incremental* discretization errors (the errors added in the current time step) [31]. It is thus believed that fully efficient multigrid methods for the steady-state equations will also easily yield fully efficient and parallelizable methods for time-accurate integrations.

2.2. Solution methods and current development. As shown in the past (see [13], [19] and [43]), to obtain the "textbook" multigrid efficiency for any discretized partial differential system of equations (PDE), it is necessary and usually (with proper boundary treatment) also sufficient to attain that efficiency for each factor of the PDE principal determinant. Each such factor is a scalar differential operator of first or second order, so its efficient solution is a vastly simplified task. The way for separating the factors is by a *distributed* (and possibly also *weighted*) relaxation scheme in which to each factor there corresponds a "*ghost*" discrete function. The latter can be directly relaxed for its corresponding factor, dictating a resulting pattern of changes to be distributed to the *actual* discrete functions (see details in Sec. 3.7 of [13] and also in [91]). To obtain the top efficiency, the relaxation of each ghost function should incorporate an essential part of an efficient multigrid solver for its corresponding operator: sometimes this is just the relaxation part of that solver, sometimes this may even be the entire solver (applied at some proper subdomain).

For the *incompressible* Euler and Navier-Stokes equations, the relevant factors are the Laplace and the convection (or convection-diffusion) operators. The former's multigrid solver is classical; the latter's can be based on downstream relaxation [43], with additional special procedures for recirculation flows [44], [92]. Indeed, we have shown that incorporating such procedures into the relaxation schemes for the appropriate ghost functions yields very efficient solvers for incompressible flows even at high Reynolds numbers and at second-order accuracy [43]. The same procedures will also yield efficient solvers for compressible flows at low Mach numbers, where the relevant factors are similar.

The only remaining factor of flow systems for which no general adequate multigrid solver has previously been developed is the "*full potential*" operator

$$(u\partial_x + v\partial_y + w\partial_z)^2 - a^2\Delta \ , \qquad (2.1)$$

where (u, v, w) is the flow velocity vector and a is the speed of sound. This operator appears as a factor in the principal determinant of the 3-D *compressible* Euler equations. Its *Mach number* is the ratio $M = (u^2 + v^2 + w^2)^{1/2}/a$.

In the deep *subsonic* case ($M \leq .7$, say) the operator (2.1) is uniformly elliptic, hence a usual multigrid V-cycle, employing red/black Gauss-Seidel relaxation at all levels, yields top-efficiency solvers. When M approaches 1, however, the operator becomes increasingly anisotropic, and classical multigrid algorithms severely degrade, due to the above-mentioned difficulty with characteristic components. (An exception is the case where the anisotropy directions are aligned with grid directions. For example, if $u^2 + v^2 \ll w^2$, full efficiency can still be obtained by employing z-plane block relaxation).

In the deep *supersonic* case (e.g., $M \geq 1.3$) the full potential operator is uniformly hyperbolic (with the stream direction serving as the time-like direction), and an efficient solver can be obtained using downstream relaxation, marching in the time-like direction. If the equations are of higher-order and/or not strictly upstream, a *predictor-corrector* marching can provide the same approximation order, hence fast convergence of smooth components; we have shown this by detailed experiments and mode analyses [51]. This procedure no longer works as M drops toward 1, since the Courant number associated with this time-like marching approaches infinity.

Thus, the most difficult situation for solving the full potential operator is the *near sonic* regime ($.7 \leq M \leq 1.3$, say), especially in the (usual) case of *non-alignment* (e.g., when the grid is Cartesian and no velocity component is consistently much larger than the others). No previous multigrid approach would attain good efficiency in this case.

We have developed a new approach for this case, based on a piecewise semi-coarsening and some rules for adding artificial dissipation at the coarser levels. To understand this, note first that in the general scheme for solving, e.g., the Euler equations, the solution of (2.1) is only a *relaxation* step, and it is enough to confine this step to one subdomain at a time (whose size, however, is not $O(h)$ but $O(1)$). Without loss of generality we can therefore limit the discussion to the case that throughout this subdomain the velocity is, e.g., *vertically-inclined* (i.e., $w^2 \geq .3(u^2 + v^2)$, say). In this case, the multigrid solver of (2.1) will use *horizontal semi-coarsening* (coarsening only in the x and y direction), possibly together with *vertical* line relaxation. (This z-line relaxation is actually not needed on the finest levels, but may be required after several levels of semi-coarsening.) With this semi coarsening, the inherent cross-characteristic numerical dissipation at the coarse level is *smaller* than at the fine one (opposite to their relation upon *full* coarsening); we can therefore stably add artificial dissipation terms at the coarse level so that its total cross-characteristic dissipation matches the local fine-level average.

The resulting algorithm can fully exploit massively parallel processing. It can be extended to other non-elliptic operators, including the convection operator. (The aforementioned approach for the convection operator, based on downstream relaxation, is not fully efficient on massively parallel machines.)

Extensive numerical tests have been performed with the linear full-potential equation: first in 2D, then in 3D, starting with constant-coefficients, then variable. Simple boundary conditions were chosen in a box: Dirichlet conditions on two opposite faces and periodic on the others. In 2D we have also carried out comprehensive half-space FMG mode analyses (cf. Sec. 7.5 in [13]). All the results (e.g., those already reported in [24], and in [51]) show that at any Mach number the algorithm always attains the "textbook" efficiency. (See additional details and related developments elsewhere in this volume [50].)

2.2.1. Comment on semi-coarsening schemes.

Instead of the *piecewise semi-coarsening* described above, another alternative is to use just one *global* semi-coarsening, but of one of the following two types (preferably the second).

A. Total semi-coarsening. By this we mean (e.g., in 2D) that each coarser grid is formed by omitting every other line from the next finer grid (every other vertical line *as well as* every other horizontal line), but on the remaining lines (the coarse-grid lines) leave *all* the fine-grid points (not just the intersections of the coarse-grid lines).

B. Variable-direction semi-coarsening. Here the coarser grid for each level is a *subset* of the total-semi-coarsening grid for that level. Simply omit from the latter all unnecessary points in regions where semi-coarsening at only one particular direction is needed (as in various anisotropic and non-elliptic cases, like those discussed above).

2.3. Future plans: The road map.

The first task ahead is to extend the solver from these linear model cases to a general solver for the nonlinear full-potential operator in the entire transonic regime. The second—to incorporate the latter multigrid solver as one of the *relaxation* steps (relaxing the ghost function corresponding to the full potential factor) in an *outer* multigrid solver for the entire Euler system. Then the next task would be to generalize to the Navier-Stokes equations.

This is an ambitious and expensive program, but we are not alone in it. A group at NASA/Langley last year has launched a multi-year program aimed at achieving “textbook” multigrid efficiency for flows at all Mach and Reynolds numbers, using the general approach described above, in cooperation with us and others. As a road map we are developing a detailed table called “Barriers to Achieving Textbook Multigrid Efficiency in CFD”. It lists every foreseen kind of computational difficulty for achieving that goal, together with the possible ways for resolving the difficulty, their current state of development, and references. A first draft is available [22].

Included in the table are staggered and nonstaggered, conservative and nonconservative discretizations of viscous and inviscid, incompressible and compressible flows at various Mach numbers, as well as a simple (algebraic) turbulence model and comments on chemically reacting flows. The listing of associated computational barriers involves: non-alignment of streamlines or sonic characteristics with the grids; recirculating flows; stagnation points; discretization and relaxation on and near shocks and boundaries; far-field artificial boundary conditions; small-scale singularities (meaning important features, such as the complete airplane, which are not visible on some of the coarse grids); large grid aspect ratios; boundary layer resolution; and grid adaption.

2.4. Atmospheric time-dependent flows.

In collaboration with Drs. J.R. Bates and L. Yong from NASA/Goddard we have finished developing multigrid solvers for the system of equations arising at each time step of shallow-water models of atmospheric flows on the entire globe [3], [72]. These solvers allow implicit discretization of nonlinear terms as well as linear, resulting in much more stable simulations. We are now working on solvers for the full three-dimensional flow on the sphere.

3. Atmospheric Data Assimilation. *(with post-doc fellow Leonid Zaslavsky)*

3.1. Background and objectives.

A major difficulty in weather prediction is the need to assimilate into the solution of the atmospheric flow equations a continuously incoming stream of data from measurements carried out around the globe by a variety of devices, with highly varying accuracy, frequency, and resolution. Current assimilation methods require much more computer resources than the direct solution of the atmospheric equations. The reason is the full 4-D coupling: Any measurement, at any place and time, should in principle affect the solution at any other place and time, thus creating a dense $N_sN_t \times N_sN_t$ matrix of influence, where N_s is the huge number of gridpoints representing the 3-D atmosphere and N_t is the large number of time steps spanning the full period over which large-scale atmospheric patterns are correlated. As a result, not only are current assimilation methods very slow, but they are also based on highly questionable compromises, such as: ignoring the all-important spatially or temporally *remote* correlations of *large-scale* averages; limiting control to only the initial value of the flow at some arbitrarily chosen initial time, instead of controlling the numerical equations at all times; and assimilating only the data from one time interval at a time, without fully correlating with other intervals.

Our objective is to develop multiscale methods that can avoid all of these compromises, and can assimilate the data into the multigrid solver of the direct flow equations at small extra cost, i.e., using extra computer time smaller than that required by the direct solver by itself.

We consider this to be possible because: (1) Large scale averages can inexpensively be assimilated on the correspondingly coarse levels of the multigrid solver (coarse in both space and time). (2) Deviations from any large-scale average must be assimilated on some finer scale, but their correlation on that scale is local. (3) The measurements (with their representativeness errors) are generally less accurate and in most regions less resolved than the numerical flow itself, hence their assimilation should not be done at the finest numerical level.

Multiscale methods can contribute to data assimilation processes in a variety of other ways, a survey of which is reported in Sec. 3.4 below.

3.2. Preliminary work: fast Kalman filtering. We have collaborated with a group headed by Dr. Steve Cohn of the Data Assimilation Office, NASA/Goddard Space Flight Center, in a preliminary work for demonstrating the potential of multiscale atmospheric data assimilation. The main result has been a fast multi-resolution algorithm to solve the dense-matrix equations arising at each time step in a Kalman filtering formulation of the assimilation problem [47], [45], [46]. The methods used are related to those in [17] and [33], but with an innovation demonstrating that such methods can deal with *scattered data*, having highly variable resolution.

3.3. Future plans: Multiscale 4D assimilation. The development will not be limited to the Kalman filtering formulation. We mainly intend to advance the multiscale capabilities with respect to the direct 4-D (space and time) best fitting of the scattered data. This problem involves full 4D couplings, both forward and backward in time. It is thus proposed to use one full-multigrid (FMG) algorithm for the entire 4D problem (but possibly with the storage-saving windowing described below). This algorithm would be like a usual FMG solver for the direct 4D atmospheric equations, except that at each stage, on each level excluding the finest ones, the relaxation of the solution variable will be accompanied by relaxation of the *control variables* $\sigma(x)$ at that level (see the nature of $\sigma(x)$ below). Thus, in essence, large-scale averages of the solution will be assimilated on correspondingly coarse grids (coarse in both space and time).

The levels at which $\sigma(x)$ will be adjusted will depend on the local density of the measurements, their accuracy and their distance from regions where details of the solution are of interest.

Windowing. Should the 4D solution require too much storage, it is possible to reorganize it in multiscale windows, marching in time, without much loss of efficiency. That is, only a certain window (time slice) of the finest grid need be kept in memory at a time. Having relaxed over it, residuals are then transferred from this window to the coarser grids. On returning from the coarser grids more relaxation is made on the finest grid, now in a somewhat advanced window (shifted forward in time, but partly overlapping its predecessor) and so on.

On increasingly coarser grids, increasingly wider (in real time, but poorer in gridpoints) windows are kept and advanced in a similar manner. The domain covered by each coarse-grid window always strictly contains all the finer ones. The coarsest windows extend very far in time, especially into the past; as far indeed as there exist data whose large-scale averages are still correlated to the solution at the time of the current finest window. At times where a coarse window exists while the next finer one has already been removed, the coarse-level equations can still retain the FAS-multigrid fine-to-coarse (τ) corrections (static or modified), thus still maintaining the fine-level accuracy of coarse-level features (cf. the "frozen τ" technique in [12, §15]).

Some of the finest windows may be local not only in time but also in space, effecting *local refinements* at regions of greater human interest and/or regions requiring higher resolution for physical reasons (sea straits, islands, mountains, etc.).

3.4. Multiple benefits of multiscale techniques. Multiscale computational methods can contribute to data assimilation problems in *several* different ways, listed below.

1. Implicit time steps. At the level of the underlying *direct* CFD equations, fast multigrid solvers make it possible to use implicit-time-step discretizations at full efficiency (see the general approach to time dependent problems in [31], and methods for the shallow water equations in [5], [4], [3] and [72]). This entails not only unconditional linear stability, but also avoidance of bad effects associated with *linearized* time steps (in which one would use fully implicit equations, but based on linearization around the previous-time-step solution) [3]. The unconditional stability is important for the multiscale data assimilation processes, enabling work on various temporal and spatial scales, unconstrained by various Courant numbers.

2. Local refinements are well known to be greatly facilitated by the multigrid algorithm, as also hinted in the algorithm description above. The multiscale environment simultaneously provides convenient flexible structures, refinement criteria and one-shot self-adaptive solvers; see [12, §9] and [2].

3. Space $+$ time parallel processing. Still at the level of the direct CFD equations (but similarly also at the level of the inverse (data assimilation) problem), multiscaling is a necessary vehicle to obtain parallel processing not only across space at each time step, but also across time. In other words, unnatural though it may seem, sequential marching in time can be avoided by using multiscale procedures. (This was first pointed out in [10, §3.10], and more appears in [16, §11] and [84].) This of course makes it possible to use efficiently (at a given arithmetic to communication ratio) a larger number of parallel processors.

4. One-shot solution of inverse problems. Normally, inverse problems are solved by a *sequence* of direct solutions (e.g., direct multigrid solutions), through which an iterative adjustment is made to the control parameters (the inverse-problem unknowns). For example, in the *adjoint method* for atmospheric data assimilation, a direct solver of the flow equations (marching forward in time) is followed by an adjoint solution (backward in time) that gauges the first derivatives of the data-fitness functional with respect to the initial values (the flow variables at the initial time). These derivatives then drive some adjustments of the initial values, from which another direct flow solution is next calculated, and so on. *Many* iterations are needed for this process to converge. In multigrid solvers, by contrast, one can integrate the adjustment of the inverse parameters into the appropriate stages of only *one* direct-problem solver (see Sec. 3.3 above. This general approach has been described in [24, §13], with more details in [16, §8.2] and full development in [82]).

5. One-shot continuation. The assimilation problem is highly nonlinear, hence a good starting guess for the solution is important. A general way to obtain such an initial guess is by continuation (embedding), in which the problem is embedded in a sequence of problems, each requiring another application of the solver. In multigrid solvers, however, the continuation can often be integrated into just one FMG solver. For example, at the coarser stages of the FMG algorithm more artificial viscosity (and/or more regularization, and/or a smaller coefficient of D_t in the continuity equation) can be used, then gradually be taken out as the algorithm proceeds to finer levels. This makes the solution much easier in the first stages, from which it is then continuously dragged into the desired neighborhood. Such FMG continuation devices are often natural. For example, larger artificial viscosity would quite naturally be introduced on coarse grids, even without aiming at continuation. A natural continuation is also supplied by the inverse covariance matrix S (see below), which would be smaller on coarser FMG levels due to larger discretization-error estimates.

6. Full flow control. In most data assimilation approaches (such as the adjoint method described above), the control parameters (the parameters that can be changed to obtain fitness of solution to observations) are only the initial values of the solution. This makes it impossible to benefit from the details (the oscillating components) of the observations at time far removed from the initial time, because those details at those times are ill-determined by the initial values. Instead of controlling just initial values, one should really control the entire numerical solution. Namely, the control parameters $\sigma(x)$ is a vector-valued grid function that at each point x gives the deviations in satisfying the set of flow equations. The objective function (the error functional that should be minimized) has the general form

$$E = \sigma^{\mathbf{T}}\mathbf{S}\sigma + \mathbf{d}^{\mathbf{T}}\mathbf{W}\mathbf{d},$$

where $\sigma = (\sigma(\mathbf{x}))$ is the vector of all control parameters, $\mathbf{d} = (\mathbf{d}(\mathbf{y}))$ is the vector of deviations of the solution u from the observation u^0 (i.e., $d(y) = (P^0u)(y) - u^0(y)$, where P^0 is a projection from the solution space (x) to the observation space (y)), and S and W are (positive-definite) weight matrices. In a crude approximation, one can take these matrices to be diagonal, where the diagonal inverse $S(x,x)^{-1}$ is (a very rough estimate of) the expected square error in the equation at x, which is the sum of the local discretization error (conveniently estimated by the "τ correction" of the FAS multigrid solver; see [13, §8.4]) and the local modeling errors (errors in the physical assumptions embodied in the equations). The diagonal inverse $W(y,y)^{-1}$ is (a very rough estimate of) the expected square error in the *measurement* $u^0(y)$, including in particular the "representativeness error" (accidental deviation at the point of measurement from the relevant local average). More precisely, S and W should be corresponding *general* (not necessarily diagonal) inverse covariance matrices (in which case the discussion at Item 8 below is relevant).

So extensive control parameters can only be handled by a multiscale treatment. Moreover, using the methods described above the solution is expected not to be expensive, especially since the control parameters $\sigma(x)$ need not be controlled at the finest computational levels; on such levels $\sigma(x)$ can simply be interpolated from the coarser levels and kept unchanged during the relaxation.

7. Unlimited correlation range. In conventional assimilation methods, each control value interacts with a limited range of measurements: measurements at a restricted (e.g., 6 hours) time interval and sometimes only at confined distances. However, it is clear that large-scale averages of the dynamic variables interact at much larger ranges. Multiscale data assimilation makes it possible to correlate solution and measurements at any desired distance in space and time, since correlations at increasingly larger distances are calculated on increasingly coarser grids.

8. Efficient representation of direct and inverse covariance. There are a number of ways to derive or estimate covariance matrices and various simplification assumptions are made. However, the real covariance matrices (especially the model error covariance) are actually dense (not sparse), and thus involve huge (8 dimensional, in principle) amounts of information. Even when the matrix is sparse, its inverse, used in the formulation of the objective function, is certainly dense. The only efficient way of *representing*, let alone computing, such huge dense matrices and their inverses is a multiscale representation, based on their asymptotic smoothness. This would be similar to the methods introduced in [12, §8.6], [15, App. A], [33], [17], [86], [41] and [42] for calculating integral transforms, many-body interactions and solutions to integro-differential equations, all involving $n \times n$ dense matrices whose complexity (the amount of computer operations required to perform a multiplication by either the matrix *or its inverse*) is reduced to $O(n)$ by multiscale techniques.

To achieve such a low complexity it is of course necessary to assume the covariance matrices to be *asymptotically smooth*. Namely, if the errors at two points, x and y, remote from each other, are correlated at all, their correlation is assumed to vary "smoothly" as function of x (similarly y). Smoothness here means that p-order derivatives are not larger than $O(|x - y|^{-p+q})$, q being a fixed small integer. (In fact, it may be enough to assume at each point x smoothness for variations in only some directions, although the complexity may then rise to $O(n \log n)$. The processing in such cases would be akin to those in [25] and [35].) Such assumptions seem very reasonable in practice, and are certainly more accurate than neglecting distant error correlation altogether. They can also be weakened in various ways and still benefit from multiscale processing.

9. Improved regularization. First, the multiscale solver described above is likely to require much less regularization than conventional solvers since the main ill-posedness in the problem is the long term and long range influence of fine-scale oscillations, while the multiscale large-scale interactions are mediated by coarse grids, omitting these oscillations. Secondly, attractive regularization devices are offered by the multiscale processing. For example, statistical theories of the atmospheric equations yield the relative expected energy at different scales. In a multiscale processing this can be used to properly penalize any excessive local energy at every scale, yielding an excellent regularization scheme (which could not even be formulated in uni-scale processing).

10. Fast assimilation of new data. Normally, new observation data keep arriving and need to be assimilated into an already partly existing approximate solution; i.e., the new data should usually both modify the previous solution and extend it into a new time interval. The multiscale solver is particularly suitable for the task: The new data normally does not affect the details of the solution in much older times; also, these details are normally no longer of interest. Hence, increasingly older times can participate in the new processing on increasingly coarser levels (still maintaining the fine-to-coarse τ corrections previously computed for them). This exactly fits into the windowing algorithm above (Sec. 3.3). The resulting ease of assimilating new pieces of data may well facilitate a *continuous assimilation policy*, with new data being assimilated much more often than today.

11. Multiscale organization of observation data. Either for the purposes of the multiscale assimilation procedure, or for a variety of other procedures, it is very useful to organize the observation data in a multiscale structure. This may simply mean pointers from a multiscale hierarchy of uniform grids into the set of data, with finer uniform levels introduced only where there are still more than a couple of observations per grid cell. Such data structures are commonly used to facilitate regional computations of all kinds. Beyond this, it is possible to replace many observations by their average at some larger scale, serving as a kind of *macro-observation*, its associated error estimate being of course reduced by standard rules of statistics. This can be repeated, to obtain still-larger-scale representations. Such structures may save much storage, and provide directly the needs of the multiscale assimilation algorithms.

4. PDE Solvers on Unbounded Domains. *(with post-doc fellow Jeffrey S. Danowitz)*

As pointed out already in [8, §7.1], problems in unbounded domains can be solved by a multigrid structure employing increasingly coarser grids on increasingly larger domains, using an FAS multigrid solver. We have embarked on a detailed study of how this should be done: At what rate should the domains increase with increased meshsize? What is the largest needed domain? What interpolation is needed at interior boundaries (boundaries of a grid h embedded in a larger domain covered by grid $2h$)? What multigrid algorithm should be applied?

For the Poisson equation $\Delta u = F$ we have developed theoretical answers to these questions, then tested them numerically. Using general grid optimization equations (see [8, §8.1] or [12, §9.5] or [13, §9.5]) one can find for example that if the domain of interest (outside which $F = 0$) has diameter d_0 and if the desired accuracy inside that domain would be obtained (had its boundary values been given) by a second-order discretization and a grid with meshsize h_0, then the diameter of each coarser grid h ($h = 2h_0, 4h_0, \ldots$) should satisfy $d(h) \geq d_0(h/h_0)^{2/3}$ and $d(h) \geq d(h/2) + Ch \log h_0$. Without significantly departing from the desired accuracy one can in this manner cover a domain (the coarsest-grid domain) with diameter R, spending only $O(\log R)$ gridpoints, so R can easily be taken so large as to admit small enough boundary-condition errors. Employing a suitable version of the λ-FMG algorithm [13, §9.6], it has been shown that the accuracy-to-work relation typical to multigrid solvers of the *bounded*-domain problem can in this way be obtained for the *unbounded* domain, where accuracy is in terms of approaching the *differential* solution. The same can be obtained for higher-order discretizations (with another exponent in the first $d(h)$ inequality).

The next plan would be to extend this study to non-elliptic equations, including high-Reynolds flows, in unbounded domains.

5. Wave/Ray Multigrid Methods. *(with post-doctoral fellow Ira Livshits)* The aim is to develop advanced and general numerical tools for computing wave propagation on scales much larger than the wavelength, when there may also exist interactions with special smaller-scale inhomogeneities where ray representations (geometrical optics) would break down. Such tools can revolutionize important computations, such as: radar cross sections; wave propagation through dispersive media; seismic wave characteristics resulting from various types of explosion zones; generation and control of acoustic noise; electronic waves in condensed matter; etc.

We have developed two basic approaches relevant to the problem. One is a general multiscale solver for *integral* equations with oscillatory kernels [17], which is a very efficient way to solve wave propagation in *homogeneous* (and some *piecewise* homogeneous) media (e.g., by replacing the differential equations with *boundary* integral equations). Multiscale ray representations first appeared in this work.

The other approach is a fast multigrid solver for the highly indefinite *differential* equations of stationary waves in a domain containing many wavelengths, with radiation boundary conditions. The basic idea of this work had been stated long ago (see, e.g., [9, §3.2], and more details in [93]),but important algorithmic aspects had still to be clarified or invented.

The model equation we used is the Helmholtz equation

$$\Delta u(x) + k^2 u(x) = f(x). \tag{5.1}$$

Traditional multigrid solvers are not effective for this problem, because some “characteristic” components (i.e., those with wavelength close to $2\pi/k$) are non-local (their size is determined by conditions many meshsizes away) exactly on all those grids fine enough to approximate such components.

On each of its levels, the new solver represents the solution as

$$u(x) = \sum_j A_j(x) \exp(i\varphi_j(x)). \tag{5.2}$$

At the highest (finest) level this sum includes just one term and $\varphi_j(x) \equiv 0$, so the representation includes just one function—the desired solution—and the equation for it is the usual five-point finite-difference discretization of (5.1). Increasingly lower levels of the solver employ on the one hand increasingly *coarser* grids of x to discretize each amplitude $A_j(x)$ and each eikonal $\varphi_j(x)$, and, on the other hand, correspondingly *finer* sets of "momenta" (i.e., more terms j in the above sum). The interaction between these levels has been shown to yield a solver (for the discrete equations given at the highest level) which is as efficient as the best traditional multigrid solvers for definite elliptic systems. The radiation boundary conditions are naturally enforced at the lowest level, where the representation essentially coincides with geometrical optics (ray representation, appropriate for scales much larger than the wavelength).

Details of the one-dimensional solver and a preliminary version of the two-dimensional solver were given in [67]. The current version of the two-dimensional solver, together with numerical results, is described in detail elsewhere in this volume [32].

An important feature of the solver is the alignment of the grid on which $A_j(x)$ is discretized with the propagation direction of the corresponding eikonal, (the direction of $\nabla\varphi_j(x)$), its meshsize growing faster in that direction than in the perpendicular directions. Specifically, if J is the number of terms taken by the summation (5.2) at a given multigrid level, then the propagation-direction meshsize for that level is $O(J^2k^{-1})$, while the perpendicular-direction one is $O(Jk^{-1})$. Incidentally, such oriented grids should have also been employed in [17], reducing the order of complexity stated there to the same one as in the non-oscillatory case (with an additional $O(\log n)$ factor in the case of integral transforms or integral equations defined on a curved manifold of codimension 1, e.g., a boundary).

A finite-element representation akin to (5.2) appears in [94]–[95], but only on one level, and without the above-mentioned grid alignment. That representation thus cannot be used to bridge the gap between the wave discretization needed at small subdomains and the ray discretization needed at the large outer regions, nor can it serve as a fast solver.

5.1. Variable coefficients, local refinements and diffraction. *The plan for the next years* is to develop the solver for the variable-coefficient case $k = k(x)$, and to advance a new setting where only geometrical optics is used in most of the domain, while the wave equations, as well as intermediate levels with representations of the type (5.2), are just introduced at special restricted subdomains where geometrical optics breaks down.

Geometrical optics can certainly be used throughout large regions where $k(x)$ is either a constant or has a small relative change per wavelength. Although in the latter case the rays are curved, they can still be followed by Snell's law, or more generally by marching solutions of the eikonal equation (see, e.g., [88]). Discontinuities in $k(x)$ can also be accommodated by geometrical optics, employing the usual rules of reflection and refraction, as long as the surfaces of discontinuity have curvature radii large compared with the wavelength (assuming the number of repeated reflections is not too large).

The pure geometrical optics approach will typically break down in smaller regions (e.g., neighborhood of fast changes in $k(x)$ or large-curvature surfaces of discontinuity). It is proposed to introduce in such regions nested local refinements, structured in the usual FAS-multigrid manner (where the coarser grids cover also the local-refinement subdomains, still playing the role of accelerating convergence in the finer grids, which are over-set in the desired subdomains; see [8, §7] or [12, §9] or [13, §9] or [2]). The finer levels will generally use representations of the type (5.2), the finer the level the smaller the number of terms in the summation, eventually yielding a direct discretization of (5.1) on sufficiently fine grids in small subdomains; see some more details in [32, §10].

Effectively this will produce ray dynamics in the large, with relations between rays modified by the finer grids in the small special regions (around an aperture, corners, edges, a radar target, etc.), yielding a general numerical tool for computing diffraction (the rays produced by small-scale disturbances; c.f. [80]).

6. Many Eigenfunction Problems: Ab-Initio Quantum Chemistry. *(with Ph.D. student Oren Livne and former Ph.D. student Ron Kaminsky)* Some important scientific problems involve the computation of a large number of eigenfunctions of a partial differential operator. In *ab-initio* condensed-matter calculations, for example, a large number of eigenfunctions of the Schrödinger operator $-\Delta + V$ should be calculated to determine the electron density function ρ. Moreover, this particular problem is in fact *nonlinear*, since the "self-consistent" potential function V depends on ρ, and is also *non-local*, since V in fact depends on *integrals* involving ρ.

Fast multigrid eigenproblem solvers have been developed long ago [37], but the ab-initio problem includes new traits and difficulties that call for new multiscale techniques, such as in the following list.

(1) *Singularities*. The nuclear potential energy harbors a singularity at each atomic nucleus (if pseudo-potential is not used). The multigrid solver (unlike Fourier methods) allows local refinements that would remove the global inaccuracies associated with such singularities [8], [24], [13], [2]. Because of the neighborhood of the singularity, *conservative discretization* is needed [2], which is especially tricky for *high-order* discretization at grid *interfaces* (the boundaries of any level of local refinement); see [6], where the FAS conservative discretization of [2] is extended to high-order schemes in three dimensions, and applications to quantum chemistry are discussed.

(2) *Unbounded or very-large-scale domains* can efficiently be treated by multigrid solvers which employ increasingly coarser grids at increasingly larger distances from the region(s) of interest (cf. Sec. 4 above).

(3) *Self-consistency*. The dependence of the potential function V on the total electronic charge distribution ρ introduces a nonlinearity into the problems, which usually requires many iterative applications of a linear solver. Multigrid procedures can directly solve nonlinear problems, as efficiently as solving their linear counterparts [13]. The development of such one-shot solvers for the Schrödinger operator depends on the ability to update the self-consistent potential as the solution changes on the coarse grids. This is also related to the following issue.

(4) *Multi-integrations* are required in calculating the potential (e.g., the Hartree potential). This can be performed fast by solving auxiliary Poisson equations. Solving them by multigrid would facilitate the needed interaction between the coarse-level moves of this Poisson solver and the coarse-grid updates to the self-consistent potential in the eigenproblem solver (see #3 above). Also, multigrid solvers (unlike the currently-used FFT solvers) will accommodate local grid refinements (see #1 above).

(5) *External optimization*. Whereas in solving the electronic problem the nuclei are assumed fixed (the Born-Oppenheimer approximation), one actually needs to find the nuclei positions for which the electronic-solution energy together with the inter-nucleus potential yield the minimal total energy. This external optimization would normally be done iteratively, requiring solving the electronic eigenproblem many times. Again, a one-shot multigrid solver + optimizer can and should be developed, incorporating suitable nucleus moves into each of the levels of the multigrid electronic solver. A model study reported below (Sec. 6.1) has shown the feasibility of this approach and the exact multigrid techniques required for its full efficiency.

(6) *Multitude of eigenfunctions.* Even with a multigrid solver, the cost of calculating a large number N of eigenfunctions (N being the number of electrons in the system) may grow

proportionally to N^3 (employing discretizations with $N_1 = O(N)$ degrees of freedom), since each eigenfunction is represented separately and may need to be orthogonalized with respect to all others to ensure their distinction. A theoretical study of a model problem indicates that it may be possible to reduce the complexity to $O(N_1 \log N \log \frac{1}{\epsilon})$, by employing a multiscale collective representation of the eigenmodes. Here ϵ is the desired accuracy and N_1 is just the number of grid points required for adequately resolving the various features of the potential function $V(x)$.

(7) Multiscale representations may also offer improved representations for the *exchange correlation* potential.

Of all the scaling difficulties listed above, several (those numbered 1,2,3,4, and partly also #5) have been dealt with in other contexts (similar difficulties in other fields). So, once multigrid solvers are introduced, the technique for treating these difficulties will already be at hand.

We therefore focus our current research mainly on #6: developing a new multiscale collective representation and collective calculation of many eigenfunctions. We have started with the easier, one-dimensional case with a linear and periodic potential function V without singularities. The eigenfunctions between two energies are represented by expressions similar to (5.2) above, with increasing separation between eigenfunctions described on increasingly coarser grids.

In the coming years we plan: (1) to complete the 1D multi-eigenfunction solver described above; (2) to move to the much more difficult two-dimensional case (similarly to the manner in which the work on standing waves (Sec. 5 above) has proceeded); (3) to extend the work on external optimization to multi-nucleus multi-eigenfunction cases (if suitable collaboration, a student or a researcher, join this effort); (4) to explore the possibilities offered by the multiscale representation of the eigenspace for efficient discretization of the exchange correlation potential (#7 above); (5) join forces with others to demonstrate the capability of multiscale techniques to overcome other obstacles (e.g., #1,2,3 and 4 above).

6.1. Model problem for the external optimization. A simplified model problem for the external optimization is the minimization of the two-dimensional two-atom total energy

$$\min_{z=(z_1,z_2)\in D} [E(z) + \lambda(z)], \tag{6.1}$$

where $E(z)$ models the ("external") repulsive energy between ions at $(0,0)$ and at (z_1, z_2), and $\lambda(z)$ is the corresponding electronic energy, modeled by the eigenvalue of the equation

$$(-\Delta + V(x,z))\psi(x) = \lambda\psi(x), \qquad x = (x_1, x_2) \in D. \tag{6.2}$$

We chose $V(x,z)$ that models the Coloumbic potential at x of the two-ion system, $D = [0,1] \times [0,1]$, and ψ was required to satisfy periodic boundary conditions on D (having chosen V and E also with this periodicity).

The Euler equations for minimizing (6.1) under the constraint (6.2) can be simplified (since the Lagrange multiplier coincides with ψ) to the system of equations (6.2)–6.4, where

$$\langle \psi, \psi \rangle = 1, \tag{6.3}$$

$$\frac{\partial E}{\partial z_i} + \langle \psi, \frac{\partial V}{\partial z_i}\psi \rangle = 0, \qquad (i = 1, 2). \tag{6.4}$$

The eigenproblem (6.2)–(6.3) was solved by a classical FAS multigrid eigen-solver [37]. The main point of the research was to find out how to include Eq. (6.4) and where to adjust

z in the course of this solver. Since (6.4) is a global equation and z is a "global" unknown (unlike $\psi(x)$ it cannot be *smoothed*, for example), it is enough to treat both of them at the coarsest level, where all the discrete equations can simply be solved simultaneously for all the unknowns, since their number is small. This would be fully efficient, provided a suitable "fine-to-coarse correction" for Eq. (6.4) is recursively calculated at each coarsening step, see [13,§5.6] or [12, §5.6], except that in the FAS scheme the residual transfer is replaced by the τ_h^{2h} fine-to-coarse correction; see [13, §8.2] or [24, §8.2]. Namely, the discretization of (6.4) on any grid h has the form

$$\frac{\partial E}{\partial z_i} + \langle \psi^h, \frac{\partial V^h}{\partial z_i} \psi^h \rangle_h = \tau_i^h, \qquad (i = 1, 2), \tag{6.5}$$

where $\langle\ ,\ \rangle_h$ is the discretization on grid h of the continuum L_2 inner product $\langle\ ,\ \rangle$, and $\tau_1^h = \tau_2^h = 0$ on the finest grids.

Usually in FAS schemes, the FAS fine-to-coarse corrections (τ) are fixed (i.e., they depend on the current fine-grid solution, but they do not change on the coarse level). The main finding of this research was, however, that in the above situation (and for similarly "localized" global unknowns, whose movements may not be resolved on some of the coarse grids), a linear dependence on the global unknowns should be introduced. Thus, on each coarser grid we introduce, for $i = 1, 2$,

$$\begin{aligned} \tau_i^{2h} \quad &= \tau_i^h + \left\langle \overline{\psi}^{2h}, \tfrac{\partial V(x,\tilde{z})}{\partial z_i} \overline{\psi}^{2h} \right\rangle_{2h} - \left\langle \tilde{\psi}^h, \tfrac{\partial V(x,\tilde{z})}{\partial z_i} \tilde{\psi}^h \right\rangle_h \\ &+ \textstyle\sum_{j=1}^2 (z_j - \tilde{z}_j) \left[\left\langle \overline{\psi}^{2h}, \tfrac{\partial^2 V(x,\tilde{z})}{\partial z_i \partial z_j} \overline{\psi}^{2h} \right\rangle_{2h} - \left\langle \tilde{\psi}^h, \tfrac{\partial^2 V(x,\tilde{z})}{\partial z_i \partial z_j} \tilde{\psi}^h \right\rangle_h \right], \end{aligned} \tag{6.6}$$

where $\tilde{\psi}^h$ and $\tilde{z}$ are the values of ψ^h and z^h at coarsening (i.e., when the grid-2h equations are derived), and $\overline{\psi}^{2h} = \hat{I}_h^{2h} \tilde{\psi}^h$ is the FAS fine-to-coarse transferred solution. (This means that the coarse-to-fine correction will be $\psi^{2h} - \overline{\psi}^{2h}$, where ψ^{2h} is the solution of Eq. (6.5) with $2h$ replacing h; see [13,§8.1] or [12, §8.1].)

The linear (in $z - \tilde{z}$) terms in (6.6) are important in the cases where the functions $\partial V / \partial z_i$ are not resolved well enough on the coarse level to yield there the correct dependence of $\langle \psi, (\partial V / \partial z_i) \psi \rangle$ on variations in z. This generally happens when V has a singularity (or a particularly large local variation on the scale of the grid h) which moves with z. Note however that, quite fortunately, exactly in such cases, it is enough to calculate the bracketed inner products in (6.6) over just a small neighborhood of the singularity.

With this simple change, the one-shot solver for the external optimization problem (6.2)–(6.4) attains essentially the same convergence factors as in solving Poisson equation, costing only a fraction more.

7. Fast Evaluation of Integral Transforms on Adaptive Grids. *(with post-doctoral fellow Kees Venner)* Multilevel algorithms previously developed for the fast evaluation of integral transforms, such as

$$Gu(x) = \int_\Omega G(x, y) u(y) dy$$

(and for the solution of corresponding integral and integro-differential equations; see e.g. [33], [17], [86]) rely for their efficiency on the (asymptotic) smoothness of the *discrete* kernel (the matrix) and thereby on *grid uniformity*. However, in actual applications, e.g., in contact mechanics [86], in many cases large solution gradients as well as singularities occur only locally, and consequently a substantial increase of efficiency can be obtained by using *nonuniform* grids.

A new discretization and evaluation algorithm has been developed which relies on the (asymptotic) smoothness of the *continuum* kernel only, independent of the grid configuration. (Asymptotic smoothness roughly means that $G(x, y)$ is smooth except possibly near $x = y$; cf. [17].) This will facilitate the introduction of local refinements, wherever needed. Also, the new algorithm is faster: for a d-dimensional problem only $O(s^{d+1})$ operations per gridpoint are needed, where s is the order of discretization and d is the dimension. (Multigrid algorithms with only $O(ds)$ operations per gridpoint are available for *potential-type* kernels, yielding faster evaluations at higher d and s; see §8 in [41].)

The algorithm has been tested using a one dimensional model problem with logarithmic kernel. For testing purposes, and to compare with results obtained with the "old" algorithms, uniform grids covering the entire domain were considered first, see [41]. Next the algorithm was implemented for the actual case of local grid refinements [42]. Numerical results were obtained for a model problem in which u has a singularity where its derivative is unbounded. First it is shown that on a *uniform* grid this singularity "pollutes" the entire approximation, dictating a much deteriorated relation between work and accuracy in comparison with the regular case (where accuracy is measured in terms of approximating the *continuum* transform, of course). Next we have demonstrated that with the new fast evaluation algorithm on a *non-uniform* grid one can restore the regular work to accuracy relation, i.e., obtain the same efficiency as for the case without a singularity.

In the next couple of years the plan is to develop a multigrid *solver* for integro-differential *equations* discretized on adaptive grid, based on the new discretization and evaluation algorithm. As explained in [33], it will again be attempted to show that the cost of the solver need not be more than a fraction above the cost of our fast evaluator of the involved integral transform. As previously developed for PDE systems [8], [2], [13], self-adaptation criteria based on the local fine-to-course defect corrections (τ) are planned, as well as full integration of the grid adaptation process into the solver (like the λ-FMG algorithm in [13]).

Future applications with our collaboration are expected in tribology and in ab-initio electronic structure calculations (where, however, another approach will be to use multigrid solvers; cf. #(4) in Sec. 6 above).

8. Dirac Solvers. *(with Ph.D. student Michael Rozantsev)* A major part of lattice quantum field calculations is invested in the inversion of the discretized Dirac operator M^h appearing in the fermionic action. Solutions of systems of the form

$$M^h \phi^h = f^h \tag{8.1}$$

are many times called for, either for calculating propagators or for the fast update of $\det M^h$ (see Sec. 9).

We used the Euclidean staggered lattice formulation [17], in which

$$(M^h\phi)(z) = \frac{1}{h}\sum_{\mu=1}^{d} \eta_\mu(z)[U(z+\frac{1}{2}e_\mu)\phi(z+e_\mu) - U^\dagger(z-\frac{1}{2}e_\mu)\phi(z-e_\mu)] + m_q\phi(z),$$

where h is the meshsize of the grid, $\phi = \phi^h$, d is the number of dimensions, m_q — the (given) quark mass, and e_μ — a vector of length h pointing in the μ-th coordinate direction. η_μ are complex numbers of modulus 1, and may be chosen as $\eta_1(z) = 1$, $\eta_2(z) = (-1)^{n_1}$, $\eta_3(z) = (-1)^{n_1+n_2}$ and $\eta_4(z) = (-1)^{n_1+n_2+n_3}$ for the gridpoint $z = h(n_1, \ldots, n_d)$, n_ν

being integers. $U(z+\frac{1}{2}e_\mu)$ is the gauge field value defined on the directed link $(z, z+e_\mu)$. The inversely directed link $(z, z-e_\mu)$ carries the gauge field $U^\dagger(z-\frac{1}{2}e_\mu)$, where † denotes the Hermitian conjugate of the matrix. Each $U(z+\frac{1}{2}e_\mu)$ is an element of the model's unitary gauge group.

We have investigated two such models: $U(1)$ and $SU(2)$. In the $U(1)$ model, the gauge group elements are complex numbers of modulus 1, and $\phi^h(z)$ and $f^h(z)$ are complex numbers. (In the case of a trivial gauge field $(U \equiv 1)$ in 2D, the $U(1)$ operator M^h reduces to the well known Cauchy-Riemann system.) In the $SU(N_c)$ model the gauge group elements are unitary complex $N_c \times N_c$ matrices whose determinant is 1, and $\phi^h(z)$ and $f^h(z)$ are complex N_c-vectors. See more about these models in [90], [64], [65], [66], [81], and about a multigrid approach to related, simplified models in [61] and [62].

These systems, despite their linearity and good ellipticity measures, are very challenging, due to their topology-induced singular (or nearly singular) eigenmodes and their disordered and non-commutative coefficients (the gauge field). The disorder results from the probabilistic physical rules by which the gauge field is determined, and from the "gauge freedom", i.e., the fact that those rules determine the field only up to arbitrary "gauge transformations". The latter are arbitrary multiplication of each $\phi^h(z)$ by an element of the gauge group and corresponding changes of the gauge field U so that (8.1) is still satisfied. Such changes do not change the physical content of the field.

Our approach, based on pre-coarsening gauge smoothing and on multiscale iterate recombination, had previously been applied to the two-dimensional $(d = 2)$ U(1) model (see general description in [18], and full account in [70]). More recently we have been working on the U(1) and SU(2) gauge models in 4D [71].

For the 4D-U(1) gauge model, general conditions have been formulated under which the gauge field can be smoothed *globally* by gauge transformations, hence a fully efficient multigrid solver can, and has been, constructed. These conditions are *not* satisfied, however, in two kinds of topological situations. In the first kind, the total topological charge over the domain does not vanish. In this case the field can be smoothed everywhere except for a certain local neighborhood which can easily be shifted away to any other place by gauge transformations, so that good intergrid transfers can be formulated *locally*. This is enough for obtaining nearly top multigrid efficiency.

The second topological case is more severe. It features gauge-field discontinuity along some non-local path (called "string") in the domain. This string can be shifted by gauge transformations, except for its ends. So we have at least two local discontinuities which cannot be shifted away. If not treated they lead to critical slowing down (CSD) of the solver (i.e., the larger the grid the more computational work per unknown is required). The number of slowly converging components introduced by the string is small, however, so they can be eliminated by recombining iterates (taking linear combinations of results of the latest multigrid cycles so as to minimize the residual L_2 norm; which can also be done on coarser levels of the multigrid hierarchy; see [70], [39]) together with local relaxation passes added around the local discontinuities. With these devices the convergence is still slower than in the absence of a string, but it seems free of CSD. We suspect that with wider regions of local relaxation the better efficiency may be restored; unfortunately, our domains were not wide enough for testing this.

Indeed, a severe problem in our entire work on these 4D models is the huge amount of computer time needed to produce reasonably sized, well equilibrated gauge fields on which to test our solvers: the Monte Carlo processes for producing these fields are far too slow. (This problem will presumably be alleviated when the work of the kind reported in Sec. 10 below is sufficiently advanced.)

In the case of the 4D-SU(2) model, it is not yet clear what the best gauge-smoothing gauge fixing is. We have found a fast multigrid algorithm for the so called "Landau fixing", but the resulting gauge field is not always smooth. Although the topological nature of the remaining discontinuities is not yet clear, it seems that the above techniques (iterate recombination and local relaxation passes) eliminate CSD here, too. The current state of development is summarized in [71].

CSD or not, the current solvers are too slow, (compared, e.g., to our solvers for trivial gauge fields, which are as efficient as 4D Poisson multigrid solvers). Our recent study indicates that this can be cured by adding *simultaneous-relaxation steps on every level*, relaxing simultaneously small local sets of strongly-coupled variables, which are randomly created by the random topological charges of the gauge field.

The plan is to continue working on the above models and to investigate also the 4D-SU(3) model.

9. Fast Inverse-Matrix and Determinant Updates. *(with programmer Ruth Golubev and Ph.D. student Michael Rozantsev)* In parallel to the development of the multigrid fast Dirac solvers (Sec. 8), work has been progressed on methods for using multigrid solvers for constructing an inexpensive structure of the *inverse* Dirac matrix, allowing fast self-updating upon each change in the matrix itself (each gauge update). This will allow fast updating of the fermion propagators and the associated determinant (repeatedly needed for calculating the action of the unquenched Monte Carlo simulations). The general approach was first described in Sec. 12 of [15].

For a large lattice with N sites and meshsize h, the storage of the Dirac inverse matrix $(M^h)^{-1}$ would require $O(N^2)$ memory and $O(N^2)$ calculations, even for fully efficient multigrid solvers. Using the following multigrid structure, both can be reduced to $O((l + \varepsilon^{-1/l})^d N)$, where ε is the relative error allowed in the calculations and l is the interpolation order below. More important, the structure will allow very fast self-updating.

Denoting the propagator from gridpoint x to gridpoint y by $((M^h)^{-1})_{x,y}$, for sufficiently smooth M^h the l-th "derivatives" (difference quotients) of this propagator, with respect to either x or y, decay as $O(|x-y|^{-1-l})$. Therefore, an l-order interpolation of the propagator from grid $2h$ to grid h will have at most $O(h^l(|x-y| - lh/2)^{-l})$ relative error, i.e., $O(h^l)$ times the relative magnitudes of the maximal l-th order derivative of the interpolated function (over the lh neighborhood spanned by the interpolation stencil) and the interpolated function itself. This relative error will be smaller than ε in the region

$$|x-y|/h \geq K \equiv C\varepsilon^{-1/l} + l/2,$$

where C is a (small) constant. Hence, propagators with $|x-y| \geq Kh$ need be stored on grid $2h$ only, except that, for a similar reason, those of them with $|x-y| \geq 2Kh$ need actually be stored only on grid $4h$; and so on.

This structure can be immediately updated upon changes in the gauge field. Indeed, each local change in the gauge field, if done in a properly distributive manner (i.e., so that some moments of the fields are kept unchanged) has only local effect on the propagators. Since the calculation of the latter can be regarded as solving by multigrid the system (8.1) with $f^h = \delta_{x,y}$, the effect of each local change can be calculated just by local relaxation sweeps around the change on some of the finest levels. More global changes can similarly be introduced (in a similar, distributive manner) at coarser levels. Changes in propagators described on grid $2h$ (associated with relaxing the *smooth* changes in the gauge field) affect those described on grid h through an FAS-like interpolation (it means correcting ϕ^h by $I^h_{2h}(\phi^{2h} - I^{2h}_h \phi^h)$, where I^H_h denote interpolation from grid h to grid H; except that here one interpolates both in x and in y). The cost per update is $O(1)$, i.e., independent of lattice size.

With $(M^h)^{-1}$ thus monitored, one can inexpensively calculate changes in $\log det\, M^h$. For a *small* change δM^h in the gauge field

$$\delta \log det\, M^h = Tr((M^h)^{-1}\delta M^h), \tag{9.1}$$

which can be computed locally, based on $((M^h)^{-1})(x,y)$ for neighboring (x,y). For *larger* changes one can locally integrate (9.1), since the local processing also gives the dependence of $(M^h)^{-1}$ on δM^h. Again, the amount of calculations per update does not depend on the lattice size.

Simplified model. The approach described above was first developed for model matrices with a simplified structure: matrices M^h arising from discretizing on a lattice with meshsize h the random diffusion equations $Lu = f$, where

$$Lu(x,y) = \frac{\partial}{\partial x}[a(x,y)\frac{\partial}{\partial x}u(x,y)] + \frac{\partial}{\partial y}[b(x,y)\frac{\partial}{\partial y}u(x,y)]$$

and the discrete analogs of the diffusion coefficients $a(x,y)$ and $b(x,y)$ have random values, uniformly distributed in $(0,1)$.

In the methods developed, “central” terms of the inverse matrix $(M^h)^{-1}$, together with similar central terms of $(M^{2h})^{-1}, (M^{4h})^{-1}, \ldots$ are calculated and stored; where a *central term* of the matrix $(M^H)^{-1}$ is a term $\left((M^H)^{-1}\right)_{i,j}$ for which i and j are neighboring sites on lattice H. It has been shown that this structure can update itself, upon changing one of the terms of M^h, in just $O(1)$ operations, i.e., amount of work independent of the lattice size.

The exact number of operations depends on the number of central terms (i.e., number of neighbors for each given site) kept in the system, which in turn depends on the *accuracy* at which one needs the most central (nearest neighbor) terms. Actually, for the purpose of fast determinant updates, important is the accuracy at which certain *differences* of the most central terms are calculated. The relation between this accuracy and the number of terms to be kept in the system is being thoroughly studied by us.

Having clarified these issues for the model problems, we are in the process of studying them for the 2D–$U(1)$ Dirac equations.

10. Monte Carlo Methods in Statistical Physics. *(with Ph.D. students Meirav Galun and Sergei Shmulyian and with part-time researcher Dr. Dorit Ron)* The general goal is the systematic development of advanced multigrid Monte-Carlo (MC) methods in statistical mechanics, quantum mechanics and quantum field theory. An initial stage research was done in collaboration with groups headed by Prof. Gerhard Mack (Hamburg University), Prof. Sorin Solomon (Hebrew University), Dr. Klaus Pinn (Münster University) and Dr. Martin Hasenbusch (CERN, Geneva). Our main objective is to overcome, for increasingly more interesting models, the basic complexity factors plaguing these fields: the critical slowing down (CSD) and the volume factor (as well as the factors arising from propagator calculations and fermionic interactions, a separate research for which is described in Secs. 8–9 above).

The leading idea is to use multigrid cycles in which coarser grids perform three different tasks:

(i) Acceleration of the Monte Carlo simulations on the finer levels (to eliminate the CSD). This is in general similar to the multigrid *convergence* acceleration in PDE solvers.

(ii) Gathering statistics: large-scale fluctuations can be cheaply averaged out through coarse-level MC, by cycling enough between these levels (much more than usual in multigrid PDE solvers) before returning to finer levels. Indeed, averaging out *fine-scale* fluctuations does not require *many* returns to the fine levels, since such fluctuations are largely averaged out in any *one* fine-level configuration.

(iii) Increasingly larger computational domains (decreasingly suffering from finite-size errors) can be simulated cheaply by using increasingly coarser levels: The finest grid covers only a relatively small domain, with periodic boundary conditions; a coarse level created from it can then switch to a larger periodicity (e.g., by doubling the period size); etc.

To enable the coarse levels to fulfill these tasks, we methodically learn how to construct coarse-to-fine interpolations and coarse-level Hamiltonians, with bounded complexity per coarse-level degree of freedom and such that allow the full physical mobility at the corresponding scales. If fully successful in eliminating the CSD *and* the volume factors, such Hamiltonians will also directly yield the macroscopic dynamics of the system.

The elimination of both the volume factor and the CSD factor means that a thermodynamic limit can be calculated to an accuracy ϵ in only

$$O(\sigma^2\epsilon^{-2})$$

computer operations (where σ is the standard deviation of the observable in question), independently of the lattice size ("the volume") required to obtain the ϵ accuracy. Such a performance we call "*statistically optimal*", or "*ideal MC performance*", since this is just the same order of complexity as needed to calculate, by statistical trials, any simple "*pointwise*" average, such as the frequency of "heads" in coin tossing.

Ideal performance was first demonstrated for *Gaussian models with constant coefficients* [53], [30]. It was shown there, for the one-dimensional Gaussian model, that the susceptibility thermodynamic limit can be calculated to accuracy ϵ in about $4\sigma^2\epsilon^{-2}$ random number generations, while the average energy per degree of freedom requires $3\sigma^2\epsilon^{-2}$ such generations for a similar accuracy. It was also found that the algorithmic flow (as determined by the multigrid *cycle index*) should generally depend on the observable being calculated. In the *two-dimensional* Gaussian model, the susceptibility limit can be measured to accuracy ϵ in about $20\sigma^2\epsilon^{-2}$ random number generations. In the one-dimensional *massive* Gaussian model, the susceptibility limit can be calculated in less than $8\sigma^2\epsilon^{-2}$ random generations, essentially independently of the mass size, although the algorithm flow may change with that size [27].

Next, the multigrid algorithms were extended, using new *analysis* methods, so that they would have the ability to eliminate the volume factor in more advanced models.

For the *variable-coupling Gaussian models*, we have shown that in order to reach ideal performance, the algorithm should employ during the multigrid cycle *weighted* interpolation and *variable* sampling (the Monte Carlo process should sample more frequently regions with smaller coupling values). Such algorithms have been implemented for *strongly* discontinuous cases in one and two dimensions. ("Strongly" means that the couplings may change by orders of magnitude between adjacent regions.) For the one dimensional variable-coupling Gaussian model, the susceptibility limit is calculated to accuracy ϵ in less than $8\sigma^2\epsilon^{-2}$ random number generations. In the two-dimensional variable-coupling Gaussian model, that limit can be measured in less than $20\sigma^2\epsilon^{-2}$ random generations [28]. These results are independent of the maximal ratio between strong and weak couplings, unlike the severe extra slowness that large such ratios can inflict on pointwise Monte Carlo.

The development of an optimal algorithm for the variable-coupling Gaussian model provides an important tool for general non-linear models, where non-constant couplings stochastically emerge at coarser levels of the multigrid Monte Carlo processing.

Doubts have however been raised whether ideal MC performance can really be obtained for non-linear models, where large-scale fluctuations are highly correlated with small-scale fluctuations. By applying the new analysis methods to the nonlinear *anharmonic crystal model* we have shown, and confirmed by actual simulations, that, down to a certain (small)

ϵ, performance similar to that of the Gaussian models can still be obtained, although it requires careful choice of the multigrid cycling parameters [54], [29]. Such a performance is realizable because the large-scale fluctuations depend only on some *averages* of the small-scale fluctuations, and these averages are approximated well enough at any *single* fine-level configuration used at coarsening.

For a sufficiently small ϵ, however, and for models sufficiently dominated by the anharmonic term, both the analysis and the numerical tests show that ideal performance can no longer be obtained by a multigrid process which employs our weighted linear interpolation. In fact, the analysis shows that no interpolation in the form of a linear operator can obtain ideal performance for all ϵ. We have therefore introduced another type of interpolation, the *minimization interpolation*.

This interpolation is best defined in terms of the Full Approximation Scheme (FAS; cf. Sec. 7.1 in [40]), where the coarse-grid variables represent the *full* current configuration (i.e., the sum of a coarsened representation of the current fine-grid configuration and the current coarse-grid correction) instead of just the current coarse-grid correction. To define a value u_0 at a fine-grid point based on coarse-grid values $(u_1, u_2, \ldots)$, the minimization interpolation method is first to calculate $U_0(u_1, u_2, \ldots)$, defined as the value of u_0 that would be obtained by some, exact or approximate, local Hamiltonian minimization with the values of $(u_1, u_2, \ldots)$ being held fixed. Then, to retain statistical detailed balance, the FAS minimization-interpolation value is defined by

$$u_0 = U_0(u_1, u_2, \ldots) + \tilde{u}_0 - U_0(\tilde{u}_1, \tilde{u}_2, \ldots). \tag{10.1}$$

where the $\tilde{u}_i$ are the values of the variables *at coarsening,* i.e., at the last transition from the fine level to the current coarse one.

Two-level unigrid experiments with the anharmonic crystal model have shown that the volume factor, along with the CSD, can be completely eliminated with an *exact* minimization interpolation. However, this interpolation creates a complicated coarse-level Hamiltonian, so we next designed simple *approximate* minimization interpolations, employing polynomial best fit. This yields simple (fourth-order polynomial) coarse level, allowing the recursive construction of still coarser levels and application of complete *multi*-level cycles, which do indeed demonstrate the desired ideal MC performance [29].

The situation is less convenient in more advanced physical models, where topological structures are present, because large-scale topologies may be correlated to specific fine-scale features, such as vortex centers. Also, linear-like interpolation of spinors is problematic.

A partial elimination of the volume factors in measuring susceptibility for *Ising models* was previously obtained by the three-spin coarsening technique [18], [30], as well as full elimination of that factor (namely, ideal MC performance) in determining that model's critical temperature [30].

Various types of attempts to attain ideal performance for two-dimensional non-linear σ *models* (several of which are described in [77]) have failed. Nevertheless, we have developed a variety of new *stochastic* coarsening procedures by which at least *partial* elimination of the volume factor can be achieved. These procedures include: a detailed-balance way to associate the introduction of linear (or linear-like) interpolation with a certain probability for reducing adjacent coupling strength; smart choice of the interpolation in a neighborhood depending on local features at coarsening; stochastic simplification of the derived coarse-grid Hamiltonian in ways which do not destroy the statistical detailed balance; and introduction of less restrictive stochastic interpolations [77]. Most of the developed schemes are applicable to specific cases of XY and Manton's models, while some are universal for any $O(N)$ model.

Specially devised two-grid numerical experiments have demonstrated that the designed techniques are capable of eliminating the volume factor almost *completely* at low temperatures of the XY and Manton's model, and *partially* in the $O(4)$ model as well as in the critical region of the XY model. The non-optimality of the latter results have been attributed to the insufficient accuracy in representing and sampling some of the statistically important features by means of currently employed interpolation and stochastic coarsening procedures.

This led us to an attempt to introduce the FAS minimization interpolation (10.1) also to the XY model. It yielded an improved, but not yet statistically optimal, performance. The reason for non-optimality has been shown to be the bias introduced by the FAS correction $\tilde{u}_0 - U_0(\tilde{u}_1, \tilde{u}_2, \ldots)$. For example, if the coarse configuration at coarsening $(\tilde{u}_1, \tilde{u}_2, \ldots)$ happens to be locally *non-smooth*, then the corresponding FAS correction is likely to be large, preventing the coarse-level system from efficiently sampling *smooth* configurations. A way around this difficulty is to replace (10.1) by

$$u_0 = U_0(u1, u_2, \ldots) + \frac{Q_0(u_1, u_2, \ldots)}{Q_0(\tilde{u}_1, \tilde{u}_2, \ldots)}[\tilde{u}_0 - U_0(\tilde{u}_1, \tilde{u}_2, \ldots)], \tag{10.2}$$

where $Q_o(u_1, u_2, \ldots)$ is a characteristic size of the likely fluctuations in u_0 given $(u_1, u_2, \ldots)$. More precisely, the interpolation (10.2), like (10.1), is suitable in case u_i are real variables; it has modified forms to suit other types of variables, such as XY.

In the next years we will not abandon our drive to attain ideal performance for advanced nonlinear models, since this seems to be a promising way to attain *macroscopic equations*, or more generally *macroscopic processing*, for real physical models. We will pursue various lessons learned from our past failures, including in particular the need to introduce new types of coarse-level variables and to have the coarse-grid Hamiltonian depending not only on the current fine-grid configuration, but also on certain statistics accumulated over all previous cycles. In particular, a new combination of group-renormalization and multigrid techniques is under development.

11. Molecular Mechanics. *(with Dr. Dov Bai, partly supported by the US Air Force)*

11.1. Background and objectives. Molecular mechanics (or dynamics) is becoming a major tool of theoretical chemistry, with immense practical potential in medicine, material design and biotechnology. The Born-Oppenheimer approximation to the potential energy $E(r)$ as function of the n atomic positions $r = (r_1, r_2, \ldots, r_n)$ can be imagined as the objective functional of these calculations, the electrons being implicit. Explicit approximations to $E(r)$ as a sum of various few-atom interactions are derived by accumulated computational experience, compared with finer-scale calculations (e.g., "ab-initio" quantum chemistry) and with molecular measurement data (crystal structure geometries, vibrational spectroscopy, heats of formation, etc.). The most common few-atom interactions are of the following two kinds: (1) The bond interactions between chemically-bonded atoms, including three types: length (distance) interaction between 2 atoms, angle interaction between 3 atoms and torsion interaction between 4 atoms. The first is much stronger than the second, which in turn is much stronger than the third. (2) Non-bond interactions, including the short-range Lennard-Jones and hydrogen-bond terms and the long-range Coulombic potential.

The aim of the calculations is usually either *statics* (finding the configuration r which minimizes E), or *dynamics* (calculating trajectories $r(t)$ which satisfy Newton's law $-\nabla E(r) = M\ddot{r}$, where M is the diagonal matrix of masses), or *equilibrium statistics* (average properties under the probability distribution $P(r) \sim \exp(-E(r)/k_B T)$), where k_B is the Boltzmann constant and T is the absolute temperature).

The computing cost of current molecular dynamics algorithms rises very steeply with problem size, restricting the modeling efforts to relatively small molecular ensembles and to time intervals many orders of magnitude smaller than needed. Preliminary model studies conducted by us have indicated that this steep rise in cost can be radically reduced by combining several types of multiscale approaches. Our research objective is to develop these approaches and demonstrate their ability to perform the above computational tasks in computing times that rise only *linearly* with the number n of atoms in the system. Moreover, the long term aim is to blend statistical approaches in the small (for the high-frequency molecular oscillations) with deterministic dynamics or statics in the large (see Sec. 11.6 below), and to derive macroscopic equations at increasingly larger scales, leading eventually to continuum-level processing.

11.2. Research steps. The computational-complexity factors arising in molecular mechanics, and the multiscale approach for dealing with each of them, were already specified in [21] (and partly mentioned also in [69] and [74]). The outlined multiscale techniques included: fast ($O(n)$) summation of all electrostatic interactions; increasingly coarser molecular motions (corresponding to collective atomic motions at increasingly larger scales), either for energy minimization or in Monte-Carlo simulations; multiscale annealing (similar to that developed in [40] for spin glasses); and multiscale eigen-bases for normal-mode analyses.

To investigate in detail each of these techniques, a systematic study of model problems has been undertaken. Unlike the common methodology of starting a research on molecular algorithms with small molecules and advancing to increasingly larger ones, the development of multiscale techniques necessarily employs at each stage molecules of *variable* size n, starting with very simple potential functionals and advancing to increasingly more complicated ones, progressing also from simple geometries (e.g., stretched homogeneous chains, then simple helics) to increasingly more realistic ones. At each stage just one new type of difficulty should be added, and the study objective is to still obtain the linear ($O(n)$) complexity. This research strategy is necessary since linear complexity and large-scale processing are indeed our ultimate aims, and since at small molecular systems the advantages of multiscaling cannot be observed.

11.3. Fast summation of forces. Direct summation of all the electrostatic interactions between n particles costs Cn^2 computer operations, where C is around 10. Instead, several methods exist to sum the forces in just $C_1 n$ operations (see, e.g., survey [55]), although note that in three dimensions $C_1 > 10^4$, so these methods become advantageous only for $n > 10^3$. A multiscale method for fast summation, suggested in [17] (based on an idea described earlier in [12, §8.6], [15, App. A] and [33], and related to the methods discussed in Sec. 7 above), is being used by us. It is based on a decomposition of the two-particle potential into a local part and a smooth part, the latter being evaluated at larger scales (interpolated from coarser grids), where a similar decomposition is being recursively used. An important advantage of this approach is that it gives the kind of multiscale description of the force fields which is needed for the efficient multiscaling of atomic *motions* — in statics, dynamics and equilibrium calculations (see for example the description of the coarse-level energy functional in Sec. 11.4 below).

Numerical experience with applying this approach to particle problems has been gathered by our Ph.D. student G. Hochman a couple of years ago, but to date, unfortunately, it has not been properly summarized. (Hochman left before finishing writing his thesis.)

11.4. Fast macromolecular energy minimization. Energy minimization serves here two somewhat different objectives: one in statics, the other in dynamics. In statics, the objective is to calculate the lowest energy $E(r)$, yielding the most stable conformations of the molecular structure. In dynamics, the objective is the solution of the system of equations arising at each time step of *implicit* dynamics simulations. “Implicit” refers to the method which evaluates the forces $-\nabla E(r)$ at each time step (partly or wholly) in terms of the particle *arrival* positions, i.e., positions r at the *end* of the step. This method ensures stability of very large time steps, but it does not yield the arrival positions explicitly. Instead, they should be calculated by solving a large system of equations. (Also, this method damps molecular vibrations at scales not resolved by the large time step; we return to this point below.) Solving the implicit system of equations is equivalent to minimizing an *augmented* energy functional, identical to $E(r)$ except for an additional quadratic *kinetic term* (cf., e.g., [68] or the more sophisticated functional H in Sec. 11.6 below). For large time steps this additional term is locally very small, but its large-scale effect is still profound.

The macro molecular energy minimization problem is somewhat similar to the minimization problem encountered in structural mechanics, for which very efficient multigrid solvers have been developed. Of these, the closest to the ones needed in molecular mechanics are the *algebraic multigrid* (AMG) solvers [12, §13.1], [36], [38], [14], [73], which do not assume that the problem arises from PDE or that the unknowns are really placed on a grid. The methods we have developed for molecular energy minimization follow the general AMG outline: coarser levels are constructed each by taking a suitable subset of the next-finer-level degrees of freedoms; a coarse-to-fine interpolation of displacements is defined based on the fine-level couplings and current configuration; the coarse-level set of equations (or rather, the coarse-level Hamiltonian) is derived based on this interpolation and on the current residual forces at the fine level; and the algorithm consists of relaxation (local minimization) sweeps at all levels with fine-to-coarse transfers of residual forces and coarse-to-fine interpolation of displacements. The molecular forces, however, are much more involved than those of structural mechanics, so very systematic development of all these algorithmic components was required.

Our first stage of developing multiscale molecular energy minimizers has been described in [21]. Starting with a two-dimensional model having only bond-length and bond-angle interactions, we constructed linear-time ($O(n)$) minimizers, assuming a reasonably good first approximation. We also showed that, by contrast, *conventional* atom-by-atom minimizers would instead require $O(n^3\epsilon^{-1})$ operations, where ϵ is the weak-to-strong ratio of the two kinds of interactions, properly scaled. (In real 3D models, the bond-torsion interactions are weaker by a factor $\epsilon \sim 10^{-3}$ compared with the bond-length.)

Some general rules have been learned at that stage that continue to guide our work with more advanced models. The most important is a general approach for deriving numerically the coarse-to-fine interpolation of displacements, based on a local set of atoms around each neighborhood where the interpolation is being defined. The interpolation expresses approximate relations satisfied by the minimal-energy configurations of this set, written as functions of the positions of the subset of atoms chosen to belong to the coarse level.

The model we are currently studying in detail is a three-dimensional helical chain of atoms featuring all the three (length, angle and torsion) bond interactions. From the intensive study of this particular model, several guidelines have emerged which are important for multiscale molecular dynamics in general. The most important of these guidelines are the following.

• **Relaxation** should be such that it converges fast *all* the stronger interactions. For example, at the *finest* level (the basic level of all atoms) all bond-length and bond-angle interactions should exhibit fast convergence. One can *test* this property by running cases where all other interactions are set to zero.

To *achieve* this property, simultaneous relaxation steps of several (e.g., 3) neighboring atoms at a time is required. Each such step can usually consist of just one simultaneous Newton iteration, but in some special cases may require slight employment of natural-temperature

Monte-Carlo (see below).

• **The coarse-to-fine interpolation** cannot simply be based on freezing internal coordinates (other than bond *lengths*), because of the strong nonlinearity of the interactions. Instead, they should be based on local energy minimization, done in a specific, fast and unique, way as follows.

The interpolation I from a given coarse-level configuration x^c to create a fine level configuration $x^f = I(x^c)$, where both x^c and x^f denote vectors of Cartesian positions of all involved atoms (the choice of the coarse-level atoms being described below), is performed using the following three steps.

(1) Let x_0^c and x_0^f denote the values of x^c and x^f at *the time of coarsening* (the stage at which the coarse-level Hamiltonian is defined; see its description below). A *first approximation* x_1^f to $x^f = I(x^c)$ is defined in terms of *internal-local* coarse-level coordinates (e.g., distances to neighboring coarse-level atoms), approximating in such coordinates the value of x_0^f. In this way one can ensure that any large nearly-rigid-body motion that has been performed (in the transition from x_0^c to x^c) by the neighboring coarse-level atoms is locally copied (in the transition from x_0^f to x_1^f) to the fine level.

(2) Keeping x^c fixed and a certain *local set* of x^f changing, and starting from $x^f = x_1^f$, several *local* energy minimization sweeps (i.e., relaxation sweeps confined to the local set) are performed. (The above-stated property of the (global) relaxation sweeps is required here too. Actually, instead of these local sweeps, one simultaneous Newton step for the entire local set may well be the best.) For a certain *interior subset* of atoms (a subset of the local set whose all neighboring atoms still belong to the local set), the obtained positions are then defined as their values in $\tilde{I}(x^c)$, our *tentative* interpolation. Then values of $\tilde{I}(x^c)$ are similarly obtained for another subset, interior to another local set, etc., until the tentative interpolation $\tilde{I}(x^c)$ is defined for all fine-level atoms. Note that the subsets must be disjoint while the local sets overlap each other, and that for each new subset the local relaxation must again start from x_1^f.

(3) The *final* interpolation is now defined by

$$I(x^c) = \tilde{I}(x^c) + x_0^f - \tilde{I}(x_0^c).$$

This ensures that the interpolation is essentially done on *changes* in the configuration (as in FAS multigrid algorithms; see [12, §8] or [13, §8]).

This 3-step interpolation is fast while still facilitating large global movements without destroying previous local convergence. These properties are essential for an efficient determination of the coarse-level Hamiltonian, described next.

• **The coarse-level energy functional** (or Hamiltonian) is $E(I(x^c))$. For efficient calculations on the coarse level (including recursion to still coarser levels), one needs to express this functional directly in terms of x^c, not through repeated interpolations. For this purpose $E(I(x^c))$ is Taylor-expanded around x_0^c. This expansion cannot be done directly in terms of Cartesian coordinates, because we need to get good approximations even for large $x^c - x_0^c$, e.g., for large near-rigid-body movements of local pieces of the molecule, because we want to perform such movements on the coarse levels. So the expansion should be done in *local* coordinates (e.g., distances between neighboring coarse-level atoms). This is true both for the bond interactions and for the *local part* of the long-range forces (see Sec. 11.3 above). On the other hand, the *smooth part* of the long-range forces, as well as any external field, should be computed directly in Cartesian coordinates.

We thus have developed a new method for approximating $E(I(x^c))$, based on:

(1) *Interpolating* the electrostatic charges from their fine-level (Cartesian) positions to the (Cartesian) positions of the coarse-level atoms (see [17]: this effectively retains on the

coarse level the full (non-linear!) *smooth-part* of the interactions between these charges, to any desired accuracy, even after arbitrary moves of the coarse-level atoms).

(2) Taylor-expanding (usually to quadratic order) all the *local* interactions in terms of local coordinates, approximating numerical derivatives of $E(I(x^c))$ by corresponding divided differences. This is straightforward and efficient to do due to the above local method for calculating the interpolation I.

(3) Denoting the overall *tentative* approximation to $E(I(x^c))$ thus calculated by $\tilde{E}^c(x^c)$, the *final* approximating E^c is defined by

$$E^c(x^c) = \tilde{E}^c(x^c) + [I^T R_0^f - \nabla_{x^c}\tilde{E}^c(x_0^c)] \cdot x^c \tag{11.1}$$

where I^T is the transpose of I, R_0^f is the vector of current residual forces on all fine-level atoms (in their Cartesian coordinates), and ∇_{x^c} is the gradient of $\tilde{E}^c$ (also Cartesian). The added inner product in (11.1) expresses a field-like correction (analogous to a similar term in FAS multigrid [12, §8]) which makes the first derivatives of E^c exactly correspond at coarsening to the current residual forces. The second derivatives in the expansion can be less accurate and still sustain fast overall convergence. In fact, most *mixed* derivatives (second derivatives with respect to two *different* local coordinates) are either zero (due to the local nature of the interpolation) or can be set to zero (being small compared to the relevant non-mixed second derivatives).

Since the local coordinates are in turn expressed *algebraically* (i.e., non-polynomially) in Cartesian coordinates (e.g., a distance or an angle between atoms is expressed algebraically in terms of the differences of their Cartesian coordinates), the obtained E^c has the same overall algebraic (non-polynomial) form as the fine-level energy E. This is in contrast to E^c which would result from direct Cartesian Taylor expansions.

The advantage of this $E^c(x^c)$ is that it remains a good approximation to $E(I(x^c))$ under large molecular movements. Only under extreme strains, and/or when new neighbor relations are formed, one needs to update E^c by consulting the fine level, and only locally.

• **Choice of coarse-level atoms** is material and level dependent.

In case of our polymer model, for example, we found it efficient in the first coarsening level to use a 2:5 coarsening ratio (i.e., a pair of 2 adjacent atoms out of 5 are taken to the coarse level). This yields simple interpolations and uniform interactions at the coarse level, except for the intra-pair length interaction, which can be frozen. Because of the inconvenience of working with such a frozen length, we also use a 1:4 ratio. Any well chosen (sufficiently small) ratio has the property that all the coarse-level interactions are of the order of the torsion interactions (although they include also interactions associated with angular variations which are smooth along the chain).

• **Role of temperature** is critical.

Even when all except for three atoms are fixed, there may exist several local energy minima. The stochasticity introduced by, e.g., room temperature, makes the transition between these minima trivial. This is one of the main motivations for a new approach, described below (Sec. 11.6).

11.5. Monte-Carlo methods at equilibrium. To calculate equilibrium statistics, an atom-by-atom Monte-Carlo process is usually performed. In this process, each atom in its turn changes position stochastically, according to the probability density distribution $P(r)$. Making repeated sweeps of this process, one can calculate the desired statistics on the sequence of produced configurations.

To calculate accurate averages of some observable, however, an extremely long sequence of configurations is needed. There are two basic causes for this complexity: (1) Due to the local nature of the Monte-Carlo process, only very slowly it affects large-scale conformational features, hence extremely many Monte-Carlo sweeps are needed to produce each new, statistically independent configuration. (2) Many such independent samples are needed to average out the deviation observed at each of them.

For some very simple model problems, multigrid Monte-Carlo algorithms were developed which overcome *both* these complexity causes (see Sec. 10 above, where these two causes, which multiply each other, are called the CSD factor and the volume factor, respectively). The algorithms are similar to the multiscale energy-minimization algorithms discussed above, with the following three modifications.

(a) The relaxation (local minimization) sweeps should be replaced by Monte-Carlo sweeps. Namely, instead of driving toward lower energy, energy differences associated with candidate changes are used to assign them relative probabilities, which are then applied in randomly selecting one of the candidates. For full efficiency, as in relaxation above, *simultaneous* changes of several atoms must be done in each MC step, so as to have fast equilibration of all the stronger interactions.

(b) The fine-to-coarse approximate-minimization interpolation described above (Sec. 11.4) should be modified to obtain statistical fidelity, or “detailed balance”. (Examples for such modification are Eqs. (10.1) and (10.2) above.) Any simplification of the resulting coarse-level Hamiltonian should in principle also be done in a way that retains the detailed balance (e.g., in stochastic manners; cf [18, §5.6] or [30, §4.3]). Methods to achieve this are highly nontrivial, and may require careful research and development. However, in view of the approximate nature of the molecular-mechanics Hamiltonian to begin with, and the non-critical nature of the involved temperatures, exact detailed balance may not be required, as long as statistical fidelity is retained in the limit of very smooth fluctuations. This can be achieved more easily.

(c) The multiscale cycle should switch many times back and forth between coarse levels, before returning to finer levels. In this way many samples of large-scale features can be averaged over. Not so many passes are needed at the finer scales, because many fine-scale features are already present, and hence averaged over, in any one configuration.

11.6. Small-scale statistics with large-scale dynamics. The multiscale structure allows the combination of temperature-accurate statistical simulations at small scales with time-accurate dynamics at large scales. For this purpose the multiscale minimizer discussed above should be modified in two ways.

First, the time-step discretization should be such that it gives accurate (non-damping, energy conserving) approximations for all scales whose time-accurate dynamics need be simulated.

Secondly, at all finer scales (finer levels of the multiscale solver), the local-minimization relaxation sweeps should be replaced with Monte-Carlo sweeps, for example in the following manner.

Stochastic implicit time stepping. A first-order implicit discretization to Newtonian dynamics, leading from old positions $r^0 = r(t)$ and old velocities $v^0 = v(t)$ to new positions $r^1 = r(t+\delta t)$ and new velocities $v^1 = v(t+\delta t)$, is given by $v^1 = (r^1 - r^0)/\delta t$ and $M(v^1 - v^0)/\delta t = -\nabla E(r^1)$. This set of equations in r^1 and v^1 is equivalent to the minimization of the functional

$$H(r^1, v^1) = E(r^1) + w^T M w + \frac{1}{4}(v^1 - v^0)^T M (v^1 - v^0),$$

where $w = (v^1+v^0)/2-(r^1-r^0)/\delta t$. In our *stochastic* dynamics, instead of minimizing H at each time step, we perform a Monte Carlo simulation with the probability density distribution

$$P(r^1, v^1) \sim e^{-\beta H(r^1, v^1)}.$$

On the finer scales of the multiscale cycle we take $\beta = (k_B T)^{-1}$, where T is the real temperature of the system. At increasingly coarser scales β increases, leading to practically deterministic large-scale dynamics.

This approach yields two benefits in performing very large time steps: first, it allows much easier handling of local minima; see Sec. 11.4. Secondly it avoids the killing of highly-oscillatory modes (the unresolved vibrations), which would occur if the implicit equations of a large time step were *imposed* at all scales. Instead, these modes assume stochastic amplitudes, nearly according to their equilibrium probability distribution. The desired temperature is introduced very directly in this way (the fast atomic vibrations serve as a natural heat bath), getting around the need for fabricating Langevin stochastic forces.

Tests with this scheme on model problems with quadratic potential have shown the expected behavior, except that the stochastic treatment at fine levels gradually introduces deviation from deterministic evolution also at large scales. This deviation seems generally to be of the order of the discretization error. We have nevertheless learned how to control this deviation by "distributive Monte Carlo" (similar to distributive relaxation [13]), forcing fine-scale moves to be as nearly orthogonal to large-scale moves as desired.

The testing and development of this technique has not been properly concluded, nor summarized, due to the premature leave of Leonid Zaslavsky, a postdoctoral fellow who has been working on this project.

12. Early-Vision Algorithms. *(with former Ph.D. student Jonathan Dym, Ph.D. student Eitan Sharon and in collaboration with Dr. Ronen Basri)*

12.1. Edge (or fiber) detection. Over the past several years, fast multiscale approaches for some early vision tasks, such as edge detection and surface reconstruction from sparse, noisy or blurred data, have been developed at the Weizmann Institute [52]. In particular, fast multiscale methods for enhancing and detecting *straight features* (straight edges or straight fibers) have been demonstrated [25], [26]: They detect *all* such features, of all widths and lengths, in just $O(N \log N)$ operations, where N is the number of pixels (picture elements) in the given picture.

For detecting smooth *curved* features (edges or fibers), a variety of approaches have been proposed. One good example is the *completion fields*. In this approach, the picture is described in terms of "edgels" (edge elements), i.e., short pieces of a straight edge (or fiber), defined at $N_1 = O(N)$ locations in the picture, at m different orientations in each location. The *original* value of edgel i is the response u_i to an elementary edge detector at i; that is, u is the result of a local integral transform which yields a higher value if the local picture elements do indicate existence of an edge at that particular location and orientation (and at the chosen scales of length and width, typically being, respectively, 3 and 1 times the pixel size). The *completion field* value v_j of edgel j can be built from the set of all elementary responses u_i in a variety of ways (see different approaches in [89] and [57]). As a representative example for our discussion here, we can take

$$v_j = \sum_{i=1}^{N_1 m} a_{ij} u_i, \qquad (j = 1, \dots, N_1 m), \tag{12.1}$$

where a_{ij} expresses the "affinity" of edgels i and j: it is large if edgel j is a direct continuation of edgel i, and it falls off with their distance and orientation difference. For a given i, its "induction field" a_{ij} is qualitatively similar to the field of a magnetic dipole. It is shown in [89] that such completion fields are biologically plausible, and give eye-pleasing curves. They are particularly powerful in completing curves partly occluded by large objects. The original method however has several severe shortcomings, which can be overcome by multiscaling.

Indeed, multiscale methods can contribute to the process in two fundamental ways. First, the method as described in [89] would require $O(N_1^2 m^2)$ computer operations; multiscale methods, resembling those of [17], will do the same job in $O(N_1 m)$ operations, while retaining the same (very high) degree of computational parallelism.

Secondly, and more important, still with this low cost, the multiscale processing can produce much better completion fields.

Indeed, a fundamental flaw in the uni-scale completion fields is their additivity, as in (12.1). In reality, the completion field of a long edge should be very *different* from (farther reaching and more orientation-specific than) the sum of the fields of the edgels composing it. In the multiscale approach, this flaw can be avoided, since completion fields can be constructed *separately* at each scale of length and width, with *scale-dependent* affinity parameters.

The multi-resolution *input* of straight-edge responses required for such multiscale completion fields is exactly the kind resulting from our earlier straight-feature algorithm [26], [25]. The multi-resolution of both input (straight responses) and output (completion fields) also involves further cost reductions. For example, for *short* edgels only low orientational resolution need be used, while for long edgels a low locational resolution is needed (in the lengthwise direction). Thus, the value of $N_1 m$ mentioned above can itself be radically reduced. Moreover, the multiscale *output* of the algorithm is a very desirable structure to interact with the higher vision processes of labeling and segmentation (cf. Sec. 12.2), whether or not the latter are themselves multiscaled.

A detailed study of multiscale completion fields, their parameterization and fast implementation is summarized in [75]. We further plan to investigate intriguing possibilities of combining the developed algorithms in a variety of ways, such as:

(1) Iterating a multiscale algorithm, with the output of the first iteration (e.g., the set of v_j) being used in forming the input (e.g., the set of u_i) for the next iteration. This can be done in various manners: linear, nonlinear, with or without thresholding.

(2) Using the output from one scale in forming the input for the next coarser scale.

(3) Thresholding after the previous iteration, one can use in the next iteration several different algorithms, due to the smaller set of data. Furthermore, one can afford at this stage more specialized algorithms, such as circle and corner detection. The latter can and should also be multiscaled.

12.2. Picture segmentation.

A basic task in pattern recognition is the decomposition of a given picture into meaningful segments. The criteria for blocking two picture elements into the same segment include similarity in color levels, absence of separating edges, etc. Quantitatively, these can be expressed in terms of *coupling* coefficients between neighboring pixels. It is not uniquely defined how to derive the segments once the coupling coefficients are given. Multiscale approaches can play several essential roles (somewhat analogous to their variety of roles in other areas; see for example Sec. 12.1 above).

Regarding the pixels as nodes of an electric network, and each coupling constant as the conductance (1/resistance) of a connecting wire, the approach to the segmentation problem is to define a picture segment as a block of nodes that will have approximately the same electric potential under whatever input currents applied to the network. The first possible role for a multiscale approach here is in terms of a fast solver for such networks. Since the network is highly disordered, *algebraic multigrid* (AMG) solvers best fit the task (see [36], [38], [14], [73]).

As pointed out by Sorin Solomon (in a private letter to us), there is in fact no need to solve the electric-network problem for any particular input currents: The coarse-level nodes defined by the AMG coarsening process can directly be identified with the desired picture segments.

More precisely, if all the couplings of a node at any coarse level are weak (compared with its own couplings to finer-level nodes), the node can be recognized as a picture segment, containing all the pixels (finest-level nodes) which are coupled to it (or to the *extent* of their dependence on it, through the AMG recursive coarse-to-fine interpolations).

The AMG hierarchical coarsening can indeed be viewed as a process of *recursive weighted aggregation*. In a recursive aggregation process, the elements (pixels) are blocked in small-scale aggregates, which are then blocked in larger-scale aggregates, then still larger, etc. In the *weighted* aggregation process, *fractions* of the same element can be sent into different small-scale aggregates, and similarly at all larger scales. This weighting is important in order to express the *likelihood* of elements to belong together; these likelihoods can then accumulate and reinforce each other at the higher levels of the process.

This process offers much more than simple segmentation. It in fact yields a *hierarchical segmentation*, where segments within segments can be recognized. It can also yield *scaled segmentation*, where the scale of the picture at which segmentation is desired can be specified (and be enforced, e.g., by adding a suitably-scaled “grounding” coupling to each node in the network).

More important, the multiscale weighted aggregation is free to apply new types of couplings at different levels. The coupling between larger-scale blocks (blocks which have been created by the smaller-scale aggregation, or alternatively, simple *geometric* blocks of $k \times k$ pixels), instead of (or in combination with) being induced by the fine-scale couplings (as in the AMG process), they can employ new criteria. Such criteria can include for example similarity in the *average* color level of the blocks. Or, more generally, similarity in any other intra-block statistics, including statistics on average sizes and directions of smaller-scale sub-blocks. This can result in segmentations according to texture, and various other properties.

Another criterion for blocking at each level can of coarse be the absence of separating edges on the scale of that level. This will directly benefit from the *multiscale* edge-detection algorithms, described above. Alternatively, though, it may be desired to detect the large-scale edges *from* the large-scale blocks by applying a suitable edge detector at that level (a suitable integral transform on a chosen block quantity, such as its average gray level or any other statistics).

13. Tomography: Medical Imaging and Radar. *(with post-doctoral fellow Jordan Mann and M.Sc. students Matthew Brodski, Rima Gandlin and Faina Shmulyian)* To develop multiscale computational methods for tomography, we have started by working on the two mathematically extreme cases: X-ray tomography, requiring the inversion of the sharp radon transform, and impedance tomography, requiring inversion of a very diffusive process.

Reconstruction of a function of two or three variables from its Radon transform has proven vital in computed tomography (CT), nuclear magnetic resonance (NMR) imaging, astronomy, geophysics, and a number of other fields [49]. One of the best known reconstruction algorithms is the convolution backprojection method (CB), which is widely used in commercial medical CT devices [49] (with “rebinning” for divergent-beam projection [58]). Recently, it has been applied to spotlight-mode synthetic aperture radar (SPSAR) image reconstruction [58]. While CB provides good reconstruction relatively efficiently, it is still too slow for some purposes, requiring large computational resources and limiting the ability of CT machines to produce real-time 3-D images or video. A faster technique, based on direct Fourier method yields images of much poorer quality.

For other medical imaging and radar problems, where the Radon transform is inapplicable, the performance of existing algorithms is still worse. This includes the Positron Emission Tomography (PET), impedance tomography, ultrasound and similar medical imaging techniques which suffer from high image blurring (see, e.g., [56]). The same is true for general types of Synthetic Aperture Radar (SAR) reconstructions.

A new multi-level approach to the inverse Radon transform (X-ray tomography) was developed by us several years ago. While the backprojection of the conventional CB raises the computational complexity of the method to $O(N^3)$ for an $N \times N$ images, we have developed a novel $O(N^2 \log N)$ backprojection algorithm, based on a multiscale approach, and an accompanying post-processing procedure [34], [35]. Empirical results for a number of phantoms, and measurements of point-spread functions, show that the combined method produce *better* images than those produced by classical CB, in far less time. Further improvements, including in particular an adjustment of the post-processing part to concrete CT machines, are planned.

On the other hand, the development of a similar fast method for general SAR reconstructions, especially for those where the antenna motion (in airplane or satellite) cannot be neglected (bringing Doppler shifts and nonlinearity into the problem), are far from trivial extensions, and require a new approach, which we have started to investigate.

Less advanced is our research program in multiscale approaches to diffuse tomography. The work, summarized in an M.Sc. thesis [76], is a first step in developing a fast multigrid solver to the *impedance tomography problem*.

This is an inverse partial differential problem, where the variable electrical conductivity in a body is to be found from a sequence of measurements on its surface (or on some part of it). Each measurement gives the potential over the surface generated by a given distribution of input currents. This inverse-conduction problem, first described by Calderon [48], is notoriously ill-posed, requiring some kind of regularization [1]. In [76], the problem, its regularization and its discretization are described in detail, together with a fast multigrid solver for the direct discrete problem and for some special cases of the inverse problem. It has been shown for these cases that the inverse problem can be solved in the same (high, multigrid) speed of the *direct* solver.

REFERENCES

[1] G. ALESSANDRINI,*Stable determination of conductivity by boundary measurements*, Appl. Anal., 27 (1988), pp. 153–172.

[2] D. BAI AND A. BRANDT, *Local mesh refinement multilevel techniques*, SIAM J. Sci. Stat. Comput., 8 (1987), pp. 109–134.

[3] J.R. BATES, Y. LI, A. BRANDT, S.F. MCCORMICK AND J. RUGE, *A global shallow water numerical model based on the semi-Lagrangian advection of potential vorticity*, Quart. J. Roy. Met. Soc., 121 (1995), pp. 1981–2005.

[4] J.R. BATES, S.F. MCCORMICK, J. RUGE, D.S. SHOLL AND I. YAVNEH, *A semi-Lagrangian approach to the shallow water equations*, 6^{th} Copper Mountain Conf. on Multigrid Methods, NASA Conference Publication 3224, Part 2, 1993, pp. 593–604.

[5] J.R. BATES, F.H.M. SEMAZZI, R.W. HIGGINS AND S.R.M. BARROS, *Integration of the shallow water equations on the sphere using a vector semi-Lagrangian scheme with a multigrid solver*, Mon Wea. Rev., 18 (1990), pp. 1615–1627.

[6] T.L. BECK, *Multigrid high order mesh refinement technique*, Preprint, Department of Chemistry, University of Cincinnati (1997).

[7] A. BRANDT, *Multi-level adaptive technique (MLAT) for fast numerical solutions to boundary value problems*, in Proc. 3^{rd} Int. Conf. on Numerical Methods in Fluid Mechanics, H. Cabannes, and R. Temam, eds., Lecture Notes in Physics 18, Springer-Verlag, 1973, pp. 82–89.

[8] ———, *Multi-level adaptive solutions to boundary value problems*, Math. Comp. **31** (1977), pp. 333–390.

[9] ———, *Stages in developing multigrid solutions*, in Proc. 2nd Int. Congr. on Numerical Methods for Engineers, E. Absi, R. Glowinski, P. Lascaux, and H. Veysseyre, eds., Dunod, Paris, 1980, pp. 23–43.

[10] ———, *Multi-grid solvers on parallel computers*, in Elliptic Problem Solvers, M. Schultz, ed.), Academic Press, New York, 1981, pp. 39–84.

[11] ———, *Multi-grid solvers for non-elliptic and singular-perturbation steady-state problems*, Weizmann Institute of Science, Rehovot, December 1981.

[12] ———, *Guide to multigrid development*, in Multigrid Methods, W. Hackbusch and U. Trottenberg, eds., Springer-Verlag, 1982, pp. 220–312.

[13] ———, *Multigrid Techniques: 1984 Guide, with Applications to Fluid Dynamics*, 191 pages, 1984, ISBN-3-88457-081-1, GMD Studien Nr. 85. Available from GMD-AIW, Postfach 1316, D-53731, St. Augustin 1, Germany, 1984.

[14] ———, *Algebraic multigrid theory: The symmetric case*, Appl. Math. Comput., 19 (1986), pp. 23–56.

[15] ———, *Multilevel computations: Review and recent developments*, in Multigrid Methods: Theory, Applications and Supercomputing, S. F. McCormick, ed., Marcel-Dekker, 1988, pp. 35–62.

[16] A. BRANDT, *The Weizmann Institute Research in Multilevel Computation: 1988 Report*, in Proc. 4^{th} Copper Mountain Conf. on Multigrid Methods, J. Mandel, et al., eds., SIAM, 1989, pp. 13–53.

[17] ———,*Multilevel computations of integral transforms and particle interactions with oscillatory kernels*, Comput. Phys. Comm., 65 (1991), pp. 24–38.

[18] ———, *Multigrid methods in lattice field computations*, Nuclear Phys. B Proc. Suppl., 26 (1992), pp. 137–180.

[19] ———, *Rigorous local mode analysis of multigrid*, in Prel. Proc. 4^{th} Copper Mountain Conf. on Multigrid Methods, Copper Mountain, Colorado, April 1989. The first part has appeared as: Rigorous quantitative analysis of multigrid: I. Constant coefficients two level cycle with L_2 norm, SIAM J. Numer. Anal., 31 (1994), pp. 1695–1730.

[20] ———, *Rigorous quantitative analysis of multigrid: I. Constant coefficients two level cycle with L_2 norm*, SIAM J. Numer. Anal., 31 (1994), pp. 1695–1730.

[21] ———, *Multiscale Research in Molecular Dynamics*, Gauss Center Report WI/GC-3, April, 1995.

[22] A. BRANDT, ET AL., *Barriers to Achieving Textbook Multigrid Efficiency in CFD* (draft), 1997.

[23] A. BRANDT AND N. DINAR, *Multi-grid solutions to elliptic flow problems*, in Numerical Methods for Partial Differential Equations, S. Parter, ed., Academic Press, New York, 1979, pp. 53–147.

[24] A. BRANDT AND B. DISKIN, *Multigrid solvers for the non-aligned sonic flow: the constant coefficient case.* Report WI/GC–8, The Gauss Center for Scientific Computation at the Weizmann Institute of Science, Rehovot, Israel (1997). Comput. & Fluids, to appear.

[25] A. BRANDT AND J. DYM, *Fast computation of multiple line integrals*, SIAM J. Sci. Comput., to appear.

[26] ———, *Fast robust discontinuity detection using multiple scales*, a preprint.

[27] A. BRANDT AND M. GALUN, *Optimal multigrid algorithms for the massive Gaussian model and path integrals*, J. Statist. Phys., 82 (1996), pp. 1503–1518.

[28] A. BRANDT AND M. GALUN, *Optimal multigrid algorithms for variable-coupling isotropic Gaussian models*, J. Statist. Phys., 88 (Issue 3/4, August, 1997).

[29] ———, *Statistically optimal multigrid algorithm for the anharmonic crystal model*, a preprint (1997).

[30] A. BRANDT, M. GALUN AND D. RON, *Optimal multigrid algorithms for calculating thermodynamic limits*, J. Statist. Phys., 74 (1994), pp. 313–348.

[31] A. BRANDT AND J. GREENWALD, *Parabolic multigrid revisited*, in Multigrid Methods III, W. Hackbusch and U. Trottenberg, eds., Birkhäuser Verlag, Basel, 1991, pp. 143–154.

[32] A. BRANDT AND I. LIVSHITS, *Wave-ray multigrid methods for standing wave equations*, Electron. Trans. Numer. Anal., 6 (1997), pp. 162-181.

[33] A. BRANDT, AND A.A. LUBRECHT, *Multilevel matrix multiplication and the fast solution of integral equations*, J. Comput. Phys., 90, 2 (1990), pp. 348–370.

[34] A. BRANDT, J. MANN AND M. BRODSKI, *A Fast and Accurate Radon Transform Inversion Scheme*. A patent application, Assigned to Yeda Research and Development Co. Ltd., August 1995. U.S. Patent 08/659,595, filed June 6, 1996, allowed September 10, 1997. European Patent 97108722.6-2305, filed May 30, 1997.

[35] ———, *An $O(N^2 \log N)$ Multilevel Backprojection Method*, Gauss Center Report WI/GC–6, January 1997, SIAM J. Appl. Math., to appear.

[36] A. BRANDT, S. MCCORMICK AND J. RUGE, *Algebraic multigrid (AMG) for automatic multigrid solution with application to geodetic computations*, Institute for Computational Studies, POB 1852, Fort Collins, Colorado, 1982.

[37] ———, *Multi-grid methods for differential eigenproblems*, SIAM J. Sci. Statist. Comput., 4 (1983), pp. 244–260.

[38] ———, *Algebraic multigrid (AMG) for sparse matrix equations*, in Sparsity and its Applications, D.J. Evans, ed., Cambridge University Press, Cambridge, 1984, pp. 257–284.

[39] A. BRANDT AND V. MIKULINSKY, *Recombining iterants in multigrid algorithms and problems with small islands*, SIAM J. Sci. Comput., 16 (1995), pp. 20–28.

[40] A. BRANDT, D. RON AND D.J. AMIT, *Multi-level approaches to discrete-state and stochastic problems*, in Multigrid Methods, II, W. Hackbusch and U. Trottenberg, eds., Springer-Verlag, 1986, pp. 66–99.

[41] A. BRANDT AND C.H. VENNER, *Fast evaluation of integral transforms with asymptotically smooth kernels*, Gauss Center Report WI/GC-2 (1995), SIAM J. of Sci. Comput., in press.

[42] ———, *Multilevel evaluation of integral transforms on adaptive grids*, Gauss Center Report WI/GC-5 (1996).

[43] A. BRANDT AND I. YAVNEH, *On multigrid solution of high-Reynolds incompressible entering flows*, J. Comput. Phys., 101 (1992), pp. 151–164.

[44] ———, *Accelerated multigrid convergence and high-Reynolds recirculating flows*, SIAM J. Sci. Comput., 14 (1993), pp. 607–626.

[45] A. BRANDT AND L. YU. ZASLAVSKY, *Multilevel algorithm for atmospheric data assimilation*, Proc. of the 7th Copper Mountain Conference on Multigrid Methods. Copper Mountain, Colorado, April 2–7, 1995. NASA Publication, 1995.

[46] ———, *Multiscale algorithm for atmospheric data assimilation. Part I. Multiscale iterative process*, Technical Report CS95–09, The Weizmann Institute of Science, Rehovot, Israel, 1995.

[47] ———, *Multiscale algorithm for atmospheric data assimilation*, SIAM J. Sci. Comput., 18 (1997), pp. 949–956.

[48] A. CALDERON, *On an inverse boundary value problem*, in Seminar on Numerical Analysis and its Applications, W. Meyer and M. Raupp, eds., Brazilian Math. Soc., Rio de Janeiro, 1980, pp. 1–7.

[49] S.R. DEANS, *The Radon Transform and Some of its Applications*, John Wiley and Sons, New York, 1983.

[50] B. DISKIN, *Multigrid algorithm with conditional coarsening for the non-aligned sonic flow*, Electron. Trans. Numer. Anal., 6 (1997).

[51] ———, *Efficient Multigrid Solvers for the Transonic Full Potential Equation*, Final Report for the Feinberg Graduate School, Weizmann Institute of Science, Rehovot, Israel, 1997.

[52] J. DYM, *Multilevel Methods for Early Vision*, Ph.D. Thesis, Weizmann Institute of Science, Rehovot 1994.

[53] M. GALUN, *Optimal Multigrid Algorithms for Model Problems in Statistical Mechanics*, M.Sc. Thesis, Weizmann Institute of Science, (1992).

[54] ———, *Optimal multiscale methods in statistical physics*, Final Ph.D. Report, Weizmann Institute of Science, September 1996.

[55] L. GREENGARD, *Fast algorithms for classical physics*, Science, 265 (1994), pp. 909–914.

[56] F.A. GRÜNBAUM, *Diffuse tomography: the isotropic case*, Inverse Problems, 8 (1992), pp. 409–419.

[57] G. GUY AND G. MEDIONI, *Inferring global perceptual contours from local features*, Proc. DARPA Image Understanding Workshop, (1993), pp. 881–893.

[58] G. HERMAN, *Image Reconstruction from Projections: The Fundamentals of Computerized Tomography*, Academic Press, New York, 1980.

[59] A. JAMESON, W. SCHMIDT AND E. TURKEL, *Numerical solution of the Euler equations by finite volume methods using Runge-Kutta time stepping schemes*, AIAA Paper 81-1259, 1981.

[60] A. JAMESON, *Solution of the Euler equations by a multigrid method*, Appl. Math. Comput., 13 (1983), pp. 327–356.

[61] T. KALKREUTER, *Multigrid Methods for the Computation of Propagators in Gauge Fields*, Ph.D. Thesis, Hamburg University, Report DESY92–158 (1992).

[62] ———, *Idealized multigrid algorithm for staggered fermions*, Phys. Rev. D, 48 (1993), pp. 1926–1930.

[63] J.B. KELLER, *Geometrical theory of diffraction*, Journal of the Optical Society of America, 52 (1962).

[64] J.B. KOGUT, *An introduction to lattice gauge theory and spin systems*, Rev. Modern Phys., 51 (1979), pp. 659–713.

[65] ———, *The lattice gauge theory approach to quantum chromodynamics*, Rev. Modern Phys., 55 (1983), pp. 775–836.

[66] J.B. KOGUT AND L. SUSSKIND, *Hamiltonian formulation of Wilson's lattice gauge theory*, Phys. Rev. D, 11 (1975), pp. 395–408.

[67] I. LIVSHITS, Ph.D. Thesis (supervised by A. Brandt), Bar-Ilan University, November, 1995.

[68] C.S. PESKIN AND T. SCHLICK, *Molecular dynamics by backward-Euler method*, Comm. Pure Appl. Math., 42 (1989), pp. 1001.

[69] A. BRANDT, ed., Proceedings of the Multigrid Tutorial, with Applications to Molecular Dynamics. Weizmann Institute of Science, October 10–12, 1995.

[70] M. ROZANTSEV, *Methods for Multigrid Solution of Quantum Electrodynamics Equations*, M.Sc. Thesis, Weizmann Institute, Rehovot, 1993.

[71] ———, *Multigrid Solver for 4D Quantum Chromodynamics Equations*, Progress Report, Feinberg Graduate School, Weizmann Institute of Science, Rehovot, Israel (August, 1997).

[72] J.W. RUGE, Y. LI, S. MCCORMICK, A. BRANDT AND J.R. BATES, *A nonlinear multigrid solver for a semi-Lagrangian potential vorticity based shallow water model on the sphere*, SIAM J. Sci. Comput., to appear.

[73] J. RUGE AND K. STÜBEN, *Algebraic multigrid*, in Multigrid Methods, S.F. McCormick, ed., SIAM, Philadelphia, 1987, pp. 73–130.

[74] T. SCHLICK AND A. BRANDT, *A multigrid tutorial with application to molecular dynamics (a report on a workshop)*, IEEE Comput. Sci. and Engrg., in press.

[75] E. SHARON, A. BRANDT AND R. BASRI, *Completion Energies and Scale, IEEE Conference on Computer Vision and Pattern Recognition*, (CVPR–97), Puerto Rico, 1997. Report CS97–19, Weizmann Institute of Science, Rehovot, Israel. Submitted to IEEE Trans. on Pattern Anal. and Machine Intelligence.

[76] F. SHMULYIAN, *Multigrid Algorithms for an Inverse Problem in Impedance Tomography*, M.Sc. Thesis, Weizmann Institute of Science, August 1995.

[77] S. SHMULYIAN, *Toward optimal multigrid Monte Carlo computations in two-dimensional $O(N)$ non-linear σ-models*. Final report for a research leading to Ph.D. thesis, Feinberg Graduate School, Weizmann Institute of Science, Rehovot, Israel, September 1997.

[78] G.D. SMITH, W. PAUL, D.Y. YOON AND D. RICHTER, *Microscopic Dynamics in a Long-chain Alkane Melt: Computer Simulation and Neutron Scattering*, Report, Department of Chemical Engineering, University of Missouri at Columbia, July 1996.

[79] P. SONNEVELD, P. WESSELING AND P.M. DE ZEEUW, *Multigrid and conjugate gradient methods as convergence acceleration techniques*, Proc. Multigrid Conf., H. Holstein and D. Paddon, eds., Bristol, 1983.

[80] J.C. SOUTH AND A. BRANDT, *Application of a multi-level grid method to transonic flow calculations*, in Transonic Flow Problems in Turbo Machinery, T. C. Adam and M. F. Platzer, eds., Hemisphere, Washington, 1977, pp. 180–207.

[81] L. SUSSKIND, *Lattice fermions*, Phys. Rev. D, 16 (1977) pp. 3031–3039.

[82] S. TA'ASAN, *One-shot methods for optimal control of distributed parameter systems, I. Finite dimensional control* ICASE Report 91–2.

[83] A.J. VAN DER WEES, J. VAN DER VOOREN AND J.H. MEELKER, *Robust calculation of 3D transonic potential flow based on the nonlinear FAS multigrid method and incomplete LU-decomposition*, AIAA Paper 83-1950, 1983.

[84] S. VANDEWALLE AND G. HORTON, *Fourier mode analysis of the waveform relaxation and time-parallel multigrid methods*, SIAM J. Sci. Comput., 1995.

[85] B. VAN LEER, C.H. TAI AND K.G. POWELL, *Design of optimally-smoothing multi-stage schemes for the Euler equations*, AIAA Paper 89-1933, June 1989.

[86] C.H. VENNER AND A.A. LUBRECHT, *Multilevel solvers for integral and integro-differential equations in contact mechanics and lubrication*, in Multigrid Methods IV: Proc. 4th European Multigrid Conference, Amsterdam 1993, P. Hemker and P. Wesseling, eds., Birkhäuser Verlag, Germany, 1994.

[87] P. WESSELING AND P. SONNEVELD, *Numerical experiments with a multiple grid and a preconditioned Lanczos type method*, in Approximation Methods for Navier-Stokes Problems, R. Rautmann, ed., Lecture Notes in Math., 771, Springer-Verlag, 1980, pp. 543–562.

[88] G.B. WHITHAM, *Linear and Nonlinear Waves*, Wiley-Interscience, New York, 1974.

[89] L.R. WILLIAMS AND D.W. JACOBS, *Stochastic completion fields: A neural model of illusory contour shape and salience*, ICCV-95, (1995), pp. 408–415.

[90] K.G. WILSON, *Confinement of quarks*, Phys. Rev. D, 10 (1974), pp. 2445–2459.

[91] I. YAVNEH, *A method for devising efficient multigrid smoothers for complicated PDE systems*, SIAM J. Sci. Comput., 14 (1993), pp. 1437–1463.

[92] I. YAVNEH, C.H. VENNER AND A. BRANDT, *Fast multigrid solution of the advection problem with closed characteristics*, SIAM J. Sci. Comput., to appear (January 1998).

[93] A. BRANDT, *Multiscale computational methods: research activities*, in Proc. 1991 Hang Zhou International Conf. on Scientific Computation, T. chan and Z.-C. Shi, eds., World Scientific Publishing Co., Singapore, 1992.

[94] I. BABUŠKA AND J.M. MELENK, *The partition of unity method*, International J. for Num. Methods in Engineering 40 (1997), pp. 727–758.

[95] J.M. MELENK, *On generalized finite element method*, Ph.D. thesis, University of Maryland, 1995.