Multiscale Scientific Computation: Six Year Research Summary

Achi Brandt

Abstract. The Gauss Center research 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 flows and data assimilation; feedback optimal control; PDE solvers on unbounded domains; wave/ray methods for highly indefinite equations; rigorous quantitative analysis of multigrid; 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, both for macromolecules and for large ensembles of small molecules, and the combination of small-scale equilibrium with large-scale dynamics); image processing (edge detection and segmentation); and tomography (medical imaging and radar reconstruction).

Key words. Scientific computation, multiscale, multi-resolution, multigrid, fluid dynamics, atmospheric flows, data assimilation, optimal control, 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, medical imaging, radar, image processing, edge detection, segmentation, algebraic multigrid.

AMS subject classification. 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

Table of Content

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.

Just building ever faster machines will not do, in fact. 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. In particular, all of them involve a multitude of variables located in a low dimensional 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; average motions of a many-particle system can be represented 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 *solutions* (or configurations) 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. For example, to fully parallelize a time-dependent calculation (i.e., to compute for earlier and later times simultaneously), a multiscale (multigrid) algorithm must be used.

Often, a combination of several multiscale approaches can benefit one particular problem in *many* different ways (see examples in Secs. 4.4, 9 and 14.2 below. Also the different multiscale algorithms discussed in Secs. 11, 12 and 13 are all parts of solving the same "grand challenge" problem of elementary particles).

1.2 Research directions at the Gauss Center

The work at the Gauss Center treats multilevel computation as a discipline by itself, having its own internal development, gradually increasing our understanding of the many types of multiscale interaction, their modes of operation and domains of application. Various underlying relations and algorithmic ideas are carried back and forth between widely varying types of problems. We develop the multiscale methodology in a very systematic way, through sequences of model problems in which we first isolate each type of computational difficulty, study it in separation from others, and then combine the learned techniques to treat increasingly more complicated problems.

Over its five years or so of operation, 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. 3).
3. Direct multigrid solvers for inverse problems, including system identification (e.g., impedance tomography; see in Sec. 16.2) 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 (Sec. 5).
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. 9.1).
6. New multilevel algorithms for highly indefinite (e.g., standing wave) problems (Sec. 7).
7. Multigrid solvers for the Dirac equations arising in quantum field theory (Sec. 11).
8. Compact multiresolution representation of the inverse matrix of a discretized differential 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. 12).
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. 9.2).
10. Multiscale Monte-Carlo algorithms for eliminating both the critical slowing down and the volume factor in increasingly advanced models of statistical physics (Sec. 13).
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 (macromolecule or many-small-molecule) 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. 14); a new approach to molecular dynamics, based on stochastic implicit time steps (Sec. 14.8).
13. Multiscale approaches to molecular docking.
14. Multigrid methods for integral transforms and for integro-differential equations, on adaptable grids, with applications to tribology (Sec. 10).
15. Multiscale methods for the fast evaluation and inversion of the Radon transform (Sec. 16.1); applications to medical tomography and airplane and satel-

lite radar reconstruction.

16. Multiscale algorithms for early vision tasks such as surface reconstruction, edge and fiber detection, segmentation, and meaningful picture coarsening (Sec. 15).
17. Rigorous quantitative theory for predicting the performance of multigrid solvers (Sec. 8).
18. New general and accurate approaches for coarsening linear and nonlinear algebraic systems of equations (Sec. 17).

1.3 The present report

The present report has been written as a thorough updating and substantial extension of [31], with several additional sections. It surveys the main ideas, current developments and future perspectives in the various research directions listed above. Some basic algorithms and concepts are described in more details than others, mainly because they have not yet appeared elsewhere. Some topics, being still in preliminary stages, are not reported.

2. Steady-State Fluid Dynamics

2.1 Background and objectives

An efficient multigrid algorithm for steady-state incompressible viscous flows in two dimensions appeared already in 1972 [14], a relatively efficient multigrid solver for a compressible inviscid transonic flow was already demonstrated in 1975 [112], and a fully efficient solver for a *system* of *several* coupled differential equations, characteristic to CFD, was presented already in 1978 [34]. 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 [18]), 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 [80], [79]. Although such schemes can be optimized to damp high-frequency errors [120], 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. [18], [56]). Other multigrid solvers were based on incomplete LU decomposition (ILU) and related relaxation schemes [122], [118],

[111]. 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. The fact is that, 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 in principle 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 [17], [22]. 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) [43]. It is thus believed that fully efficient multigrid methods for the steady-state equations will also directly yield fully efficient and parallelizable methods for time-accurate integrations. (See, however, some specific works on time-dependent problems in Sec. 3 below.)

2.2 Solution methods and recent developments

(with Ph.D. student Boris Diskin and with Dr. John Ruge)

As shown in the past (see [20], [24] and [56]), 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 [20] and also in [126], and examples in Secs. 17–20 of [20]). 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 [56], with additional special procedures for recirculation flows [57], [130]. 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 [56]. (the work at the Gauss Center of M.Sc. student Vadim Ogranovich [96] also contributed to these studies.) The same procedures will also yield efficient solvers for compressible flows at low Mach numbers, where the relevant factors are similar.

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

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

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 () 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 [66]. 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 ($.7 \le M \le 1.3$, 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 () 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., $w^2 \ge .3(u^2 + v^2)$, say). In this case, the multigrid solver of () 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 [20]), achieving full agreement with the numerical tests. All the results (e.g., those already reported in [35], [36], [65] and [66]) show that at any Mach number the algorithm always attains the "textbook" efficiency.

2.2.1. Comment on semi-coarsening schemes. Instead of the *piecewise semi-coarsening* described above, another concept developed by us 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, where the full potential factor is replaced by the more complicated "core operator", whose general treatment is described in [32].

This is an ambitious and expensive program, but we are not alone in it. A group at NASA/Langley 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 have developed 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 [32].

Included in the table are staggered and nonstaggered, conservative and non-conservative 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.

3. Time-Dependent Problems: Geophysical Flows

For basic multigrid approaches to time-dependent problems, see [43], and more in [74].

As explained in Sec. 2.1, the main CFD effort at the Gauss Center has naturally been directed toward steady-state flows. However, the Center, especially in its first years, has also been involved in a couple of developments related to time-dependent flows, mainly with atmospheric and/or oceanic applications. They are briefly reported next. (Atmospheric *data assimilation* is separately discussed in Sec. 4 below.)

3.1 New multigrid solvers with reduced temporal order

Large scale oceanic and atmospheric flows are often modeled by systems of partial differential equations of reduced temporal order, such as the *quasi-geostrophic equations* and various *balance models*. The resulting nonlinear systems to be solved at each time step are typically of a mixed type—effectively the product of an advection operator and an operator that is elliptic in the relevant regime of parameters. For such systems multigrid techniques are particularly advantageous. Our research in this direction has been in collaboration with Dr. Irad Yavneh (the Technion, Haifa; formerly a student here) and Dr. James C. McWilliams (National Center for Atmospheric Research, Boulder, Colorado), with most of the actual development being made by the former.

Fully implicit multigrid solvers have been developed for the *shallow-water equations* (SWE) and for the *shallow-water balance equations* (SWBE). These solvers exhibit top multigrid efficiency, even at large advective Courant numbers [129]. The solvers enables researchers [128] to investigate the range of parameters at which the slow manifold breaks down, the form of this breakdown, and how well the SWBE approximate the SWE.

A fully efficient multigrid solver has also been developed for the *three-dimensional quasi-geostrophic equations*, a fundamental system in geophysical modeling. Implementation in parallel processing made on the Cray C-90 enabled researchers [94] to employ a 320^3 resolution and to discover several new features of stably stratified turbulent flows. The work has also motivated a new insight and new analytical formulae for the smoothing rates of a large class of relaxation algorithms for anisotropic equations [127].

In the last several years this work has been continued by Drs. McWilliams and Yavneh, but without further participation of the Gauss Center.

3.2 Semi-Lagrangian atmospheric solvers

Several solvers for atmospheric equations on the sphere using semi-Lagrangian discretizations have been developed by us in a collaboration with Drs. J. Ray Bates and Yong Li at NASA/Goddard Laboratory for Atmospheres (Greenbelt, Maryland), and with Prof. Steve McCormick and Dr. John Ruge at the University of Colorado (Boulder).

The first project, reported in [7], [6], [90], [101] used the shallow-water equations in a formulation that used the evolution equation of the *potential vorticity* (PV) as one of the governing primitive equations, supplementing it with an equation for the flow divergence. The direct semi-implicit semi-Lagrangian (SLSI) discretization of the PV equation offers the prospect of a more faithful simulation of the PV, which is one of the most important dynamical quantities determining the evolution of atmospheric flow patterns, including extra tropical cyclones. A two-time-level SLSI discretization was used in which even the background Lagrangian velocity has been treated implicitly. This approach was made feasible by using a fully-efficient multigrid method (using latitude-line relaxation near poles) for solving the nonlinear implicit finite-difference equations arising at each time step.

Numerical experiments showed this PV model to give phase accuracy comparable to that of a spectral Eulerian model. On the other hand it exhibits much less noise at large wind speeds and longer range of maintaining stable Rossby-Haurwitz waves than either the spectral Eulerian model or a more conventional semi-Lagrangian model that discretizes the usual *velocity* primitive equations. It has been realized that these advantages are largely due to the use of the fully nonlinear multigrid solver (and not to the particular PV formulation).

In the next project, preliminarily reported in [91], the above approach has been extended to a full three dimensional general circulation model. A hybrid vertical coordinate has been introduced, which is isentropic at high levels, thus retaining the conservation property of potential vorticity there, but becomes terrain following at low levels, thus avoiding a possible ill-definition of potential vorticity in atmospheric boundary layers. Multigrid convergence troubles have called for modified formulations in the polar regions. Numerical experiments with developing baroclinic waves have given realistic results, although the solver development is not yet completed.

3.3 Multilevel turbulence simulations (MTS)

In the early years of the Gauss Center, Drs. Eli Tziperman, Irad Yavneh and Shlomo Ta'asan have been developing a novel method for the simulation of turbulent flows [117], motivated by and based on multigrid formalism. The MTS method is potentially more efficient and more accurate than the conventional large eddy simulations, in which only the largest turbulent scales are explicitly simulated while smaller scales are modeled by some closure scheme. MTS takes advantage of the fact that one is usually interested in the effects of the small scales on the larger ones, but not in the detailed time history of each small scale feature.

In the future we hope to develop a multiscale ab-initio turbulence modeling based on the RMG approach (see Sec. 13.2).

3.4 Multigrid for long time integration

A work on this topic, started by S. Ta'asan, has been continued by the PhD student Zigo Haras. One part of the research was aimed at obtaining *better discretizations* which will provide accurate approximate solutions for longer integration times; it is summarized in [77]. A second part was aimed at obtaining a multilevel algorithm with *infrequent use of finer grids.* The latter are used only to initialize, once in many time steps, the finite-to-coarse correction τ_h^{2h}. In between such initializations, τ_h^{2h} is advanced by an evolution equation of its own, derived from the primary discretized system.

4. Atmospheric Data Assimilation

4.1 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 $N_s N_t \times N_s N_t$ 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 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. 4.4 below.

4.2 Past work: fast Kalman filtering

(with Post-doc fellow Leonid Zaslavsky)

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 of optimal statistical interpolation arising at each time step in a Kalman filtering formulation of the assimilation problem [60], [58], [59]. The problem is to calculate the vector of the "analyzed" (best estimated) atmosphere flow field w^a by the formula

$$w^a = w^f + P^f H^T y,$$

where the quantity y is defined by the equation

$$(HP^f H^T + R)y = w^o - Hw^f, \tag{4.1}$$

using the given model forecast first guess w^f and the vector of observations w^o. H is an interpolation operator from the regular grid to the observation network, P^f is the (dense) forecast error covariance matrix, and R is the observation error covariance matrix. We treated the simplified case of univariate analysis of single-level radiosonde height data. The matrix R was assumed to be diagonal, and the matrix P^f to be given in the form $P^f_{ij} = \sigma^f_i \mu_{ij} \sigma^f_j$, where μ_{ij} is a smooth decreasing function of the distance between the ith and the jth points.

Two different multiscale processes participate in the efficient solution of this problem: a technique for fast evaluation of the discrete integral transform $\sum_i P^f_{ij} v_j$, related to those in [25] and [45] (and in Secs. 10 and 14.3 below); and a fast multiscale iterative process for solving (4.1), featuring a multiresolution simultaneous displacement technique and a localized variational calculation of iteration parameters.

The resulting algorithm has linear complexity (work proportional to the number of gridpoints plus the number of observations), demonstrating full efficiency even in the case of observation sets having highly variable resolution.

4.3 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 $\sigma(x)$ in Sec. 4.4 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. At the coarser levels, 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 [19, §15], in [74] and in Sec. 3.4 above).

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.).

The research is closely related to other projects on inverse problems, such as those described in Secs. 5, 9.1 and 16.2 below, and will be carried out by the same researchers (in particular Rima Gandlin, now a Ph.D. student). It will of course deal with a sequence of model problems. A work on data assimilation for a simple convection-diffusion problem is already under way.

4.4 Multiple benefits of multiscale techniques

Our studies have uncovered *many* different ways in which multiscale computational methods can contribute to data assimilation problems (and similarly to other inverse problems). The following is the full list — brought as an example of what a "full multiscaling" of a problem may involve.

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 [43], and methods for the shallow water equations in [9], [8], [7] and [101]). 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) [7]. 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 [19, §9] and [5].

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 [17, §3.10], and more appears in [23, §11] and [119].) 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. 4.3 above and Secs. 5, 9.1 and

16.2 below. This general approach has been described in [19,§13] and [20,§13], with more details in [23, §8.2] and full development in [116],[2],[3]).

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 (see Sec. 8.3.2 in [19] or in [20]). 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

$$E = \sigma^{\mathbf{T}}\mathbf{S}\sigma + \mathbf{d}^{\mathbf{T}}\mathbf{W}\mathbf{d}, \tag{4.2}$$

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 [20, §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 ($N_s^2 N_t^2$, in principle) amount of information. Even when the matrix is sparse, its inverse, used in (4.2), 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 [19, §8.6], [22, App. A], [45], [25], [121], [54], [55] and [60] for calculating integral transforms, many-body interactions, solutions to integro-differential equations, and Kalmen filtering, 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 *reasonably 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 like $g_1(x)g_2(y)G(x,y)$, where $G(x,y)$ is *asymptotically smooth* (meaning that up to a certain order, p-order derivatives of $G(x,y)$ 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 [37] and [47].) 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). Generally, the multiscale data assimilation need not be governed by one all-embracing objective function, but can employ a collection of different directives at different scales. (Cf. Item 12 below).

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. 4.3). 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.

12. Scale-dependent data types. Instead of simple averaging, the macro-observations just mentioned can be formed from the fine-scale data in a variety of other, often more meaningful, ways. In particular, fine-scale waves should be represented on coarse scales by their slowly-varying *amplitude*. Indeed, at large distances the wave phase is ill-posed, while its amplitude is still meaningful. (Cf. the techniques in Sec. 7 below).

5. Feedback Optimal Control

We consider a dynamical system that involves a vector x of *state functions* and a vector u of *control functions*, both being functions of time (and possibly also of pace), governed by the initial-value ODE (or PDE)

$$\frac{dx}{dt} = F(x,u), \quad B_0 x(t=0) = b_0, \tag{5.1}$$

where F and B_0 are known matrices of functions (or spatial operators). The *optimal control* problem is to find the control u for which this dynamical system minimizes a given *objective functional* $J(x,u)$ under various constraints, such as target-time (T) conditions of the type $B_1 x(t=T) = b_1$. In the *feedback* optimal control problem it is assumed that new initial conditions $B_o x(t)$ are continuously fed from the controlled device at all times $0 \le t < T$, requiring continuous updating of the control u. Fast numerical updates are required for *real-time* control.

In a usual approach to the feedback problem, the dynamical system is approximated by a linear-quadratic regulator (LQR), in which $F(x,u) = \mathcal{A}x + u$ and $J(x,u) = \int_0^\infty (|\mathcal{C}x(t)|^2 + |u(t)|^2)dt$, where $\mathcal{A}$ and $\mathcal{C}$ are linear operators on a suitable Hilbert space $\mathcal{H}$, $x \in \mathcal{H}$ and $u \in \mathcal{U} \subset \mathcal{H}$. Provided the system is stabilizable and $\mathcal{C}$ renders it detectable, there exists a unique nonnegative solution $\mathcal{K}$ to the *Ricatti equation*

$$(\mathcal{A}^*\mathcal{K} + \mathcal{K}\mathcal{A} - \mathcal{K}^2 + \mathcal{C}^*\mathcal{C})x = 0 \qquad \forall x \in \mathcal{H}, \tag{5.2}$$

and $u(t) = -\mathcal{K}x(t)$ yields the desired feedback. In most cases this approach is very inefficient, either because the LQR approximation should be iterated many times, and/or because of the non-sparseness of the (discretized) operator $\mathcal{K}$, and the resulting high dimensionality of the Ricatti equation.

Our approach, by contrast, is based on a fast multigrid solver for the *open-loop* (i.e., not feedback) optimal control problem, installed in a multiscale way that allows super-fast (essentially local) updates upon feedbacks.

The multigrid open-loop solver is very efficient by itself, a one-shot solver for the nonlinear (non-LQR) problem. In fact, for various actual problems, it can share many of the potential benefits listed in Sec. 4.4 above. The super-fast *updates* are based on the observation that, upon changing the initial conditions, the change in the solution is smooth (and increasingly smoother at times increasingly far from the initial. In various actual problems, the sense of this smoothness has to be carefully understood.) This makes it possible for the multigrid *re*-solving algorithm to re-process its *fine* grids only at the very early times, while at later times only coarse level are re-processed, with FAS fine-to-coarse defect corrections being frozen there (cf. [19,§15]). More precisely, at increasingly later times, any given multigrid level (a given timestep and, when relevant, spatial meshsize) need be re-processed increasingly more rarely. As a result, the computational cost of re-resolving is equivalent to only *local* re-processing (essentially just few steps near the initial time) of the full solver. This will usually be far less expensive than applying $\mathcal{K}$ (even just once, and even assuming the Riccati equations has already been solved).

We have tested this approach on several toy F-16 maneuvering problems (given to us by Dr. Meir Pachter of the Air Force Institute of Technology at Wright Patterson Air Force Base.) The linear dynamics includes 3 state and one control functions. Both quadratic and non-quadratic objectives were tested, including the L_∞ norm (the maximum absolute value) of one of the state functions.

The open-loop optimal problem, which in this case is a two-point boundary value ODE system with 7 unknown functions, has been discretized by second-order finite differences on a staggered grid and solved by an FMG algorithm. Just two $V(1,1)$ multigrid cycles per grid proved enough to produce a solution with algebraic errors much smaller than (only few percent of) the discretization errors. In the case of the L_∞ objective, a continuation process has been integrated into this FMG solver, approaching L_∞ by L_p, with $p = 2^m$ at the m-th FMG level. Each of the relaxation sweeps included one red/black pass for each of the 7 ODEs, some of the passes being of the Gauss-Siedel type, the others — Kacmarz type. For some of the toy problems the principal part of the ODE system was scale dependent, producing boundary layers and requiring two different discretization schemes, one at fine levels the other at coarse levels, each with its own corresponding relaxation scheme.

The feasibility of the super-fast solution updates upon feedbacks has been established in our tests by monitoring the fine-to-coarse (τ) corrections. When τ is appropriately scaled (divided by proper solution values available to the coarse-level re-solver) its values (excluding a couple of them near the initial time) turn out to change very little upon changing the initial conditions. This shows that τ can be frozen, so that re-solving can be restricted to coarse levels, as expected.

In the future we hope to find suitable collaborations with which to demonstrate the potential of this approach on other, more difficult feedback control problems.

6. PDE Solvers on Unbounded Domains

(with post-doc fellow Jeffrey S. Danowitz)

As pointed out in [15, §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 [15, §8.1] or [19, §9.5] or [20, §9.5]) and the known smoothness properties of the solution, one can calculate how far out one must use a certain meshsize to maintain a certain accuracy. For example, one can find 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 only 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 [20, §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 work on extending these techniques to other equations has been discontinued (upon the unexpectedly early leave of Danowitz).

7. 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 [25], 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., [16, §3.2], and more details in [26]), but important algorithmic aspects had still to be worked out.

The model equation we used is the Helmholtz equation

$$\Delta u(x) + k^2 u(x) = f(x). \tag{7.1}$$

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

$$u(x) = \sum_j A_j(x) \exp(i\varphi_j(x)). \tag{7.2}$$

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 (7.1). 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 [93]. The current version of the two-dimensional solver, together with numerical results, is described in detail in [44].

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 (upon coarsening) faster in that direction than in the perpendicular directions. Specifically, if J is the number of terms in the summation (7.2) 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 [25], 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 (7.2) appears in [4]–[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.

7.1 Future plans: variable coefficients, local refinements and diffraction

The plan 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 (7.2), 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., [123]). 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 [15, §7] or [19, §9] or [20, §9] or [5]). The finer levels will generally use representations of the type (7.2), the finer the level the smaller the number of terms in the summation, eventually yielding a direct discretization of (7.1) on sufficiently fine grids in small subdomains; see some more details in [44, §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. [85]).

8. Rigorous Quantitative Analysis of Multigrid

The work on developing rigorous justification to the local mode analysis (LMA)—the chief practical analytical tool in designing multigrid computations—has been continued, especially during the first years of the Gauss Center. The first part of the work appeared as the Gauss Center Report, WI/GC–1, and was also published [28].

For general linear elliptic PDE systems with piecewise smooth coefficients in general domains discretized by uniform grids, it has been proved that, in the limit of small meshsizes, the convergence factors predicted by LMA are indeed obtained, provided the cycle is supplemented with a proper processing at and near the boundaries. That processing, it is proved, costs negligible extra computer work. Apart from mode analysis, a Coarse Grid Approximation (CGA) condition has been introduced which is both necessary and sufficient for the multigrid algorithm to work properly.

The first part [28] studied the L_2 convergence in one cycle, for equations with constant coefficients. In the second part, which has not been published in a final form (but most of it appeared in [24]), extensions is discussed to many cycles (asymptotic convergence), to more levels with arbitrary cycle types (V, W, etc.), and to FMG algorithms. Various error norms and their relations to the orders of the inter-grid transfer operators are analyzed. Global mode analysis, required to supplement the local analysis in various border cases, is developed, and partial relaxation sweeps are systematically introduced into both analysis and practice.

9. 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 [49], 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 [15], [19], [20], [5]. Because of the neighborhood of the singularity, *conservative discretization* is needed [5], which is especially tricky for *high-order* discretization at grid *interfaces* (the boundaries of any level of local refinement); see [10], where the FAS conservative discretization of [5] 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. 6 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 [15], [19], [20]. 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. 9.1) 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 and tests with 1D model problems, reported in Sec. 9.2 below, indicate that for periodic problems (the usual type in condensed matter calculations), 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 per periodicity cell required for adequately resolving the various features of the potential function $V(x)$.

(7) *Highly oscillatory functions*, such as the orbitals of high-energy electrons, would normally require dense grids, and would not allow effective multigrid solvers, because of the corresponding indefiniteness of their equation. The multiscale approach described in Sec. 9.2 below effectively deals with this difficulty too.

(8) 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 have therefore focused our attention mainly on #5 (see Sec. 9.1) and #6 and #7 (see Sec. 9.2).

In the future: The main task will be to develop two (and later three) dimensional solvers along the lines described in Sec. 9.2 below, following a current Ph.D. work [92] on the one dimensional case. In addition, we hope (1) to extend the work on external optimization (Sec. 9.1) to multi-nucleus multi-eigenfunction cases (if a suitable collaboration, a student or a researcher, joins this effort); (2) to explore the possibilities offered by the multiscale representation of the eigenspace for efficient discretization of the exchange correlation potential (#8 above); (3) join forces with others to demonstrate the capability of multiscale techniques to overcome other obstacles (e.g., #1,2,3 and 4 above), and proceed to real-world molecular structures.

9.1 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

$$\min_{z=(z_1,z_2)\in D}[E(z)+\lambda(z)], \tag{9.1}$$

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

$$(-\Delta+V(x,z))\psi(x)=\lambda\psi(x), \qquad x=(x_1,x_2)\in D. \tag{9.2}$$

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 (9.1) under the constraint (9.2) can be simplified (since the Lagrange multiplier coincides with ψ) to the system of equations (6.2)–(6.4), where

$$\langle\psi,\psi\rangle=1, \tag{9.3}$$

$$\frac{\partial E}{\partial z_i}+\langle\psi,\frac{\partial V}{\partial z_i}\psi\rangle=0, \qquad (i=1,2). \tag{9.4}$$

The eigenproblem (9.2)–(9.3) was solved by a classical FAS multigrid eigensolver [49]. The main point of the research was to find out how to include Eq. (9.4) and where to adjust z in the course of this solver. Since (9.4) 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. (9.4) is recursively calculated at each coarsening step, see [20,§5.6] or [19, §5.6], except that in the FAS scheme the residual transfer is replaced by the τ_h^{2h} fine-to-coarse correction; see [20, §8.2] or [19, §8.2]. Namely, the discretization of (9.4) on any grid h has the form

$$\frac{\partial E}{\partial z_i} + \langle \psi^h, \frac{\partial V^h}{\partial z_i} \psi^h \rangle_h = \tau_i^h, \qquad (i = 1, 2), \tag{9.5}$$

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 $i = 1, 2$,

$$\begin{aligned} \tau_i^{2h} = \tau_i^h &+ \left\langle \overline{\psi}^{2h}, \frac{\partial V(x,\tilde z)}{\partial z_i} \overline{\psi}^{2h} \right\rangle_{2h} - \left\langle \tilde\psi^h, \frac{\partial V(x,\tilde z)}{\partial z_i} \tilde\psi^h \right\rangle_h \\ &+ \sum_{j=1}^{2} (z_j - \tilde z_j) \left[\left\langle \overline{\psi}^{2h}, \frac{\partial^2 V(x,\tilde z)}{\partial z_i \partial z_j} \overline{\psi}^{2h} \right\rangle_{2h} - \left\langle \tilde\psi^h, \frac{\partial^2 V(x,\tilde z)}{\partial z_i \partial z_j} \tilde\psi^h \right\rangle_h \right], \end{aligned} \tag{9.6}$$

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. (9.5) with $2h$ replacing h; see [20,§8.1] or [19, §8.1].)

The linear (in $z - \tilde z$) terms in (9.6) 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 (9.6) over just a small neighborhood of the singularity.

With this simple change, the one-shot solver for the external optimization problem (9.2)–(9.4) attains essentially the same convergence factors as in solving Poisson equation, costing only a fraction more.

9.2 Collective multiscale organization of eigenbasis

In condensed-matter ab-initio quantum calculation of electronic structures, the "molecular orbitals" ψ_n are the eigenfunctions of the Kohn-Sham equation, written briefly as

$$-\Delta\psi_n(\boldsymbol{r}) + V(\boldsymbol{r})\psi_n(\boldsymbol{r}) = \epsilon_n\psi_n(\boldsymbol{r}) \tag{9.7}$$

where $V(\boldsymbol{r})$ is a periodic function of $r = (x, y, z)$ and $n = 1, \ldots, N$. Actually, V depends on the electronic density $\rho(\boldsymbol{r}) = \sum_{n=1}^{N} |\psi_n(\boldsymbol{r})|^2$, but this nonlinear aspect (which can be incorporated into the multigrid solver) can be skipped in the present discussion. Here we concentrate on the most difficult aspect: the number of electrons N being very large

Our method is *not* based on a localization constraint on the electronic orbitals or the electronic density matrix, like most of the methods proposed recently (for a review, see [68]). Instead, it is based on the observation that "neighboring" molecular orbitals (neighboring in some parameter space $\boldsymbol{\beta}$, clarified below) are distinguishable from each other only at large enough scales, and hence, in suitable representations, one can use a common description of their details at finer scales, and progressively separate them out only on increasingly coarser grids. The core procedure is similar in structure to multigrid algorithms developed for *wave* equations (see Sec. 7 above), utilizing in addition the efficiency at which the FAS multigrid generally solves nonlinear problems and performs continuation processes (cf. e.g., [19,§8.3.2]).

To give an idea of the basic approach, we describe first the representation and computation of the *global* orbitals, i.e., those ψ_n which cannot be approximated on just a subdomain of the periodic supercell. Each such orbital $\psi(x)$ we represent in the Bloch-like form

$$\psi(\boldsymbol{r}) = \chi(\boldsymbol{r})e^{i\alpha(\boldsymbol{r})+i\boldsymbol{\beta}\cdot\boldsymbol{r}+\boldsymbol{\gamma}\cdot\boldsymbol{r}}$$

where $\chi(\boldsymbol{r})$ and $\alpha(\boldsymbol{r})$ are two real periodic functions, $\boldsymbol{\beta} = (\beta_1, \beta_2, \beta_3)$ and $\boldsymbol{\gamma} = (\gamma_1, \gamma_2, \gamma_3)$ are six real constants, $\boldsymbol{\beta}\cdot\boldsymbol{r} = \beta_1 x + \beta_2 y + \beta_3 z$, $\boldsymbol{\gamma}\cdot\boldsymbol{r} = \gamma_1 x + \gamma_2 y + \gamma_3 z$ and $\boldsymbol{\gamma} = 0$ for physical periodic orbitals. (Transition through non-physical regions, with $\gamma \neq 0$, will however be needed for both the analytical and the computational continuations mentioned below.) The values of $\boldsymbol{\beta}$ could be restricted to one "Brillouin zone" (e.g., $0 \leq \beta_i \leq 2\pi/a_i$, where (a_1, a_2, a_3) are the sizes of a periodicity cube), as in the usual reduced zone Bloch representation. However, we will *not* use such values. Instead, the values of $\boldsymbol{\beta}$ are chosen so that both $\alpha(\boldsymbol{r})$ and $\boldsymbol{\beta}$ (as well as $\boldsymbol{\gamma}$, in the non-physical regions) vary continuously as functions of the eigenvalue ϵ, in an effective "extended zone" representation. We can then parameterize ψ_n and ϵ_n by $\boldsymbol{\beta}$ only (instead of by $\boldsymbol{\beta}$ and a large number of n's; this is why n is omitted in the above representation): $\psi(\boldsymbol{r}) = \psi(\boldsymbol{r};\boldsymbol{\beta})$ and $\epsilon = \epsilon(\boldsymbol{\beta})$, and also $\boldsymbol{\gamma} = \boldsymbol{\gamma}(\boldsymbol{\beta})$.

(Parenthetically, an important technical comment: the three-dimensional parameter space is not everywhere well represented by the 3 parameters $\beta_1, \beta_2, \beta_3$.

In some regions (e.g., in non-physical ones) various other 3 parameters should be used, such as $(\epsilon, \beta_2, \beta_3)$ or $(\beta_1, \epsilon + \beta_2, \beta_3)$, etc. A suitable local choice of parameters can be based on the well developed methods of computational continuation [62], [63]. For simplicity, though, the description below assumes the parameters to always be $(\beta_1, \beta_2, \beta_3)$.)

Computations can be performed either in terms of the pair of real functions χ, α, or in terms of the complex function $\varphi = \chi \exp(i\alpha)$. The latter gives a simpler (complex linear) equation. However, the former may often present substantially smoother functions (φ is not smooth when α is large), thus allowing the use of coarser grids for the same accuracy. The equations in this case will be a pair of *nonlinear* real PDEs; but nonlinearity is not much of an issue for the FAS multigrid solver (aided with a proper continuation); what matters is that the PDE system has good ellipticity measures at all scales, which is the case here.

The parameter domain over which $\varphi(\boldsymbol{r}; \boldsymbol{\beta})$ should be calculated is $B = \{\boldsymbol{\beta}\colon \epsilon(\boldsymbol{\beta}) \le \epsilon_{\max}\}$, where $\epsilon_{\max}$ is determined by $(2\pi)^{-3} a_1 a_2 a_3 \int_B 1 d\boldsymbol{\beta} = N$, the number of computed electrons. The electronic density is given by $\rho(\boldsymbol{r}) = (2\pi)^{-3} a_1 a_2 a_3 \int_B |\varphi(\boldsymbol{r}, \boldsymbol{\beta})|^2 d\boldsymbol{\beta}$. Thus, B is roughly determined by the number of electrons *per unit volume*, not by the size of (or the number of atoms in) the periodicity cell. The smoothness of $\varphi(\boldsymbol{r}, \boldsymbol{\beta})$ as function of $\boldsymbol{\beta}$ at any given $\boldsymbol{r}$ may depend on that size; however, $\psi(\boldsymbol{r}; \boldsymbol{\beta})$ has complementary smoothness in $\boldsymbol{r} - \boldsymbol{\beta}$, i.e., $\psi(\boldsymbol{r}; \boldsymbol{\beta}) / \psi(\boldsymbol{r}; \boldsymbol{\beta}')$ is smooth in $\boldsymbol{r}$ on the scale $|\boldsymbol{\beta} - \boldsymbol{\beta}'|^{-1}$. Hence the number of values of $\psi(\boldsymbol{r}; \boldsymbol{\beta})$ needed per gridpoint value of ρ can grow only logarithmically with the size of the periodicity cell.

It may seem as if the above parameterization can relieve us from the need to orthogonalize, or otherwise separate, the eigenfunctions with respect to each other. This is not quite the case. Note that for any $\boldsymbol{\beta}^0$ there exists the whole surface (topologically a sphere) $S^0 = \{\boldsymbol{\beta}\colon \epsilon(\boldsymbol{\beta}) = \epsilon(\boldsymbol{\beta}^0)\}$, such that for each $\boldsymbol{\beta} \in S^0$ the differential equation at $\boldsymbol{\beta}^0$ (the PDE derived for $\varphi(\boldsymbol{r}, \boldsymbol{\beta}^0)$) has the spurious formal solution $\varphi^0_{\boldsymbol{\beta}}(\boldsymbol{r}) \underset{\text{def}}{=} \varphi(\boldsymbol{r}, \boldsymbol{\beta}) \exp(i\boldsymbol{\beta} \cdot \boldsymbol{r} - i\boldsymbol{\beta}^0 \cdot \boldsymbol{r})$, although the latter does not necessarily have the required periodicity. Likewise, for any $\boldsymbol{\beta}$ on or even just *near* S^0, the *discrete* equation at $\boldsymbol{\beta}^0$ will be *approximately* satisfied by a discrete approximation to $\varphi^0_{\boldsymbol{\beta}}$. In a large system (large N_{atoms}), many of these formal approximate solutions will in addition have the required periodicity, hence yielding many *genuine* approximate solutions. The solution process will have to weed all of them out.

Fortunately, employing a proper discretization, the multigrid solver of the discrete equations can inexpensively expel all these spurious (approximate) solutions. To see that, note first that for $\boldsymbol{\beta}$ sufficiently far from $\boldsymbol{\beta}^0$, the function $\varphi^0_{\boldsymbol{\beta}}(\boldsymbol{r})$ is so oscillatory that it is not at all visible on the target grid, hence pose no problem. For $\boldsymbol{\beta}$ somewhat closer to $\boldsymbol{\beta}^0$, the function $\varphi^0_{\boldsymbol{\beta}}(\boldsymbol{r})$ oscillates with a wavelength comparable to the meshsize of the target grid, hence (with a proper discretization) it can be efficiently weeded out by the target-grid relaxation process.

The main problem are $\boldsymbol{\beta}$ values (near S^0) still closer to $\boldsymbol{\beta}^0$, for which multiples of $\varphi^0_{\boldsymbol{\beta}}$ constitute smooth error components that on one hand cannot efficiently be treated by relaxation, and on the other hand are ill approximated by the coarse grid (because, being near solutions to the homogeneous differential equation, their residuals on the fine grid are much smaller than on the coarse). Each "neighborhood" of such components should be approximated on a differently "shifted" coarse grid: the residuals (from the discrete equation at $\boldsymbol{\beta}^0$) should be multiplied by $\exp(i\boldsymbol{\beta}^0 \cdot r - i\boldsymbol{\beta} \cdot r)$ before being transferred to the coarse grid, and the correction should be multiplied by $\exp(i\boldsymbol{\beta} \cdot r - i\boldsymbol{\beta}^0 \cdot r)$ after being interpolated from the coarse grid. The neighborhood (near S^0) that can be well approximated by each such shifted coarse grid is quite large; but when solving for each of these coarse grids, the multigrid transition to still-coarser grids requires *further* separation to more-accurately-shifted grids (because the coarser the grid the smaller the $\boldsymbol{\beta}$ neighborhood it can approximate well). Indeed, the total number of shifted grids required at each multigrid coarsening stage (doubling the meshsize) is multiplied by 4 (because S^0 is a two-dimensional surface). However, since the number of gridpoints in each of these coarser grids is 1/8 their number in each of the finer grids (since the physical domain is three dimensional) the total number of points on all the grids of any coarse level is just *half* that number at the next-finer level. Hence the total work (similarly storage) of the multigrid cycle adds up to less than twice the work at the target (finest) grid, which is $O(A_1 N_{atoms})$. By using progressively high-order discretizations, A_1 would grow only logarithmically with the desired accuracy.

This described procedure to calculate the global orbitals should of course interact with local procedures for computing *localized orbitals* (including outer core orbitals not included in the pseudopotentials). However, this interaction is rather weak, involving just local-to-global updates of the local contribution to $\rho(\boldsymbol{r})$, and global-to-local (FAS-type) corrections to $V(\boldsymbol{r})$. (This is quite analogous to the usual multigrid interaction between fine and coarse levels, since a change in ρ entails a much smoother change in V.) The local procedures can inexpensively employ classical multigrid eigensolvers [49]. They can include fine-to-coarse (τ) corrections from finer localized grids that resolve inner core electrons; since those electrons are unaffected by the outer electronic structure, the τ corrections should be calculated once-for-all for any *kind* of atom, so the finer localized grid need not actually appear at all in our procedures. This approach may be used to replace pseudopotentials.

A current PhD work [92] has been used to realize and develop the basic concept described here, in the framework of one-dimensional problems with fixed V. The multidimensional problem is of course much harder: in one dimension S^0 contains only two points, hence only one "shifted coarse grid" is needed, and no further separations at still coarser levels are necessary.

10. Fast Evaluation of Integral Transforms on Adaptive Grids

(with post-doctoral fellow Kees Venner)

Multilevel algorithms have previously been developed for the fast evaluation of integral transforms, such as

$$Gu(x) = \int_\Omega G(x,y)u(y)dy, \qquad \Omega \subseteq \mathbb{R}^n,\ x \in \Omega' \subseteq \mathbb{R}^n, \tag{10.1}$$

and for the solution of corresponding integral and integro-differential equations. They exploit in various ways the smoothness properties of the kernel $G(x,y)$. For an $M \times N$ evaluation (i.e., x being discretized with M gridpoints, y with N) of either the transform *or its inverse* to a certain accuracy ϵ, these algorithms require $O((M+N)\log\frac{1}{\epsilon})$ operations in the case of the Gaussian ($G(x,y) = \exp(-(x-y)^2/r^2)$) or the potential-type ($G(x,y) = |x-y|^{-\alpha}$ or $G(x,y) = \log|x-y|$) transforms, and $O((M+N)\log(\min(M,N))\log\frac{1}{\epsilon})$ operations in the case of the Fourier ($G(x,y) = \exp(ixy)$) or the Laplace ($G(x,y) = \exp(-xy)$) transforms, for example. More generally, the algorithms yield fast matrix multiplication rules for matrices that include large low-rank submatrices. They can also be used for the fast ($O(N\log\frac{1}{\epsilon})$) summation of all the forces that N particles exerts on each other. See [45],[25],[121] and references in [121].

In their original form, the fast algorithms for evaluating (10.1) for potential-type kernels relied 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 [121], 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 more recently at the Gauss Center 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. [25].) 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.

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 [54]. Next the algorithm was implemented for the actual case of local grid refinements [55]. 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 years the plan is to develop a multigrid *solver* for integro-differential *equations* discretized on adaptive grids, based on the new discretization and evaluation algorithm. As explained in [45], 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 [15], [5], [20], 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 [20]).

11. 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

$$M^h \phi^h = f^h \tag{11.1}$$

are many times called for, either for calculating propagators or for the fast update of $\det M^h$ (see Sec. 12).

We used the Euclidean staggered lattice formulation [105], 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 $\dagger$ 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 [125], [86], [87], [88], [114], and about a multigrid approach to related, simplified models in [83] and [84].

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 (11.1) is still satisfied. Such changes do not change the physical content of the field.

Our approach, based on red/black Kacmarz relaxation (since all equations in the Dirac system are first order), 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 [27], and full account in [99]). More recently we have been working on the U(1) and SU(2) gauge models in 4D [100].

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. An important concept in this multigrid solver (as in any other multigrid Dirac solver, for any model in any dimension) is to distinguish between different *species of unknowns* and between different *species of equations*. They can best be distinguished at the limit of low-temperature (well-ordered) gauge fields, for which each species of unknowns forms a grid function that must be constant for the homogeneous equations to be satisfied, and each species of equations forms a grid function which changes smoothly upon a smooth change of any one species of unknowns. The multigrid fine-to-coarse transition must transfer each fine residual to a coarse equation of the same species. Similarly, the coarse-to-fine transition must interpolate a correction to a fine unknown from course unknowns of the same species. It is also important, in the fine-to-coarse gauge averaging, to distinguish different species of gauge links: Two links are in the same species if they join the same species of unknowns.

The conditions for global gauge smoothing 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 still be smooth *semi-globally*, i.e., it 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, featuring a *local* topological object, i.e. gauge-field discontinuities which cannot be shifted away. In such cases, and in many other cases (e.g., non-abelian models), it has been found that global or semi global gauge smoothing is not feasible. A general procedure has then been developed by which just *local* gauge smoothing at a time (over just 5 gridpoints in each direction) allows local constructions on each grid of the fine-to-coarse residual transfers and the coarse-to-fine correction interpolations required for the multigrid cycle. The local smoothing, which can be applied in any model, is done in a unique way, so that the resulting inter-grid transfers come out gauge invariant. Also, a general gauge-invariant procedure for the fine-to-coarse averaging of the gauge field itself has been constructed, based on transport averaging similar to that of [13, [11], [12]. This averaging has the advantage of reproducing on the coarse level various local topological objects, facilitating good coarse-grid approximations.

The local topological objects in the 4D-U(1) model are "*monopol loops*", and they can also appear in a U(1) component of a 4D-SU(2) configuration. In the SU(2) case these loops would persist for only few Monte-Carlo passes in gauge simulations at the critical temperature, and it is believed that in both U(1) and SU(2) only short loops are physically probable at critical or larger temperatures. If not treated, these loops lead to critical slowing down (CSD) of the multigrid solver (i.e., the larger the grid the more computational work per unknown is required). The number of slowly converging components introduced by the loops is small, however, so they can be eliminated by recombining iterants (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 [99], [51]) together with local relaxation passes added around the local discontinuities. With these devices, and with the local-gauge intergrid transfers and transport gauge averaging mentioned above, the multigrid convergence is still slower than in the absence of loops, but it seems free of CSD (at least when applied to gauge fields which are physically probable at critical or larger temperatures) [100]. 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 was 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 were far too slow. A general method to overcome this problem has only recently been devised (see Sec. 13.2 below). We hope soon to obtain by such a method larger 4D gauge fields for testing our Dirac solvers.

In the future we plan to produce larger (up to 64^4) SU(2), and perhaps also SU(3), gauge fields equilibrated at temperature around critical, using the "half-V" cycle (see Sec. 13.2). With these we will try to clarify the nature and size of probable topological objects, and devise improved multigrid cycles, including in particular wider local relaxation and *simultaneous* relaxation steps around loops (especially large ones. We have found that pointwise relaxation near loops causes residuals to *increase* at some points, rather than decrease). We also plan **completely new approaches** based on new accurate algebraic coarsening schemes (discussed in Sec. 17 below).

12. 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. 11), 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 (needed at each step of the Monte Carlo simulations of the unquenched gauge field). The general approach was first described in Sec. 12 of [22]; the substantially improved version presented here exploits the development described in Sec. 17 below.

For a large d-dimensional lattice with $N = L^d$ sites and meshsize $h = O(L^{-1})$, 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 special multigrid structure, they can be reduced to $O(N(\log \varepsilon^{-1})^{q_1})$ and $O(N(\log \varepsilon^{-1})^{q_2})$, respectively, where ε is the relative error allowed in the calculations and q_1 and q_2 depend only on d. More important, the structure will allow very fast *self-updating*.

The proposed multigrid structure first calculates and stores $O(\log L)$ *accurate algebraic coarsening* levels of the operator M^H, from the given finest $(H = h)$ to the coarsest possible $(H = O(1))$. Each M^H is constructed from the next finer one in the manner of [33] (see Sec. 17 below): for accuracy ε, the stencil of M^H at each point involves some $n_c = O((\log \varepsilon^{-1})^{q_2})$ neighbors (so each row of M^H has n_c non-zero terms).

Secondly, the proposed structure calculates and stores at each level enough "central terms" of $(M^H)^{-1}$. The central terms in each column of $(M^H)^{-1}$ are terms corresponding to variables neighboring to the equation associated with that column (e.g., the n_c neighbors participating in the equation will suffice). The central terms of $(M^H)^{-1}$ can easily be constructed from those of the next coarser level, since the latter are all one needs in a two-level multigrid cycle for solving the relevant systems (at the finest level, for example, each of these systems has the form (11.1), with f^h being the delta function corresponding to the calculated column of $(M^H)^{-1}$).

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 (11.1) 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 will similarly be introduced (in a similar, distributive manner) at coarser levels of the gauge field Monte Carlo simulations. 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

$$\delta \log det\, M^h = Tr((M^h)^{-1}\delta M^h), \tag{12.1}$$

which can be computed locally, based on the central terms of $(M^h)^{-1}$. For *larger* changes one can locally integrate (12.1), 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 is first being 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 $(0,1)$.

13. Monte Carlo Methods in Statistical Physics

(with Dr. Dorit Ron and Ph.D. students Meirav Galun and Sergei Shmulyian)

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. 11–12 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. Also, and more important, statistical optimality implies that on sufficiently coarse level the derived Hamiltonian directly yields the *macroscopic dynamics* of the system.

Ideal MC performance was first demonstrated in calculating various thermodynamic limits in Gaussian models with constant coefficients, and also in calculating the critical temperature of the Ising model]30], [69]. This led to several years of efforts to develop interpolation-based multigrid algorithms, with cycles similar to those of multigrid PDE solvers. These are reported in Sec. 13.1 below. The increasing complicated nature of the interpolation rules and the coarse-level Hamiltonians required to treat advanced non-linear models has later brought us to favor renormalization-type methods, which are described in Sec. 13.2.

13.1 Interpolation-based methods

These methods imitate multigrid solvers of discretized differential minimization problems, where the *Hamiltonian* (or the *energy* functional which should be minimized) is automatically defined on increasingly coarser grids by recursively specifying, level after level, coarse-to-fine interpolation rules. They also use the same type of cycles, except that the relaxation sweeps (each composed of a sequence of local minimization steps) are replaced by Monte Carlo sweeps (local steps of simulating the probability distributions induced by the Hamiltonian).

The *cycle index* (specifying how many times one switches from a given multigrid level to the next coarser level per each switch to the next finer level) in statistical multigrid algorithms for computing large-scale observables will be larger than usual in multigrid solvers, to allow cheap averaging of large-scale fluctuations.

To obtain statistically optimal algorithms, as defined above, it is necessary to choose the coarse-to-fine interpolation so as to obtain full physical mobility at the coarse level: poor interpolation would not allow accessing mutually independent samples at the coarse level without in-between visits to the fine level for Hamiltonian updating. In addition, for statistical optimality the resulting coarse-level Hamiltonians need to be computable in a bounded complexity per coarse-level degree of freedom; they cannot be left just expressed in terms of finer-level Hamiltonians.

This kind of statistical optimality was first demonstrated for *Gaussian models with constant coefficients* [69], [42]. 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 [39].

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 [40]. 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 [70], [41]. 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 [53]), 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

$$u_0 = U_0(u_1, u_2, \ldots) + \tilde{u}_0 - U_0(\tilde{u}_1, \tilde{u}_2, \ldots). \tag{13.1}$$

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 [70], [41].

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 [27], [42], as well as full elimination of that factor (namely, ideal MC performance) in determining that model's critical temperature [42].

Various types of attempts to attain ideal performance for two-dimensional non-linear σ *models* (several of which are described in [109]) 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 [109]. 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 (13.1) 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 (13.1) by

$$u_0 = U_0(u_1, u_2, \ldots) + \frac{Q_0(u_1, u_2, \ldots)}{Q_0(\tilde{u}_1, \tilde{u}_2, \ldots)}[\tilde{u}_0 - U_0(\tilde{u}_1, \tilde{u}_2, \ldots)], \qquad (13.2)$$

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 (13.2), like (13.1), is suitable in case u_i are real variables; it has modified forms to suit other types of variables, such as XY.

13.2 Renormalization multigrid (RMG)

The increasing complexity of the coarse Hamiltonians produced by the interpolation-based techniques has led us more recently to a new type of algorithms. They combine renormalization-like derivation of increasingly coarser actions with multigrid-like Monte-Carlo accelerations, with the coarse actions and the accelerated MC iteratively employing and serving each other.

This RMG approach has already yielded optimal performances (eliminating both the CSD and the volume factors) for the Ising and XY models, and it can in principle be extended to arbitrary models, since it is not based on cluster-type (such as Swendsen-Wang and Wolff) algorithms. In fact, we have already preliminarily used this approach in two models of molecular dynamics: a model of polymers (see Sec. 14.6) and a model of Lennard-Jones fluids (see Sec. 14.7). Moreover, the approach has inspired a similar, equally-general procedure for the coarsening of *deterministic* equations (see Sec. 17).

For simplicity, the RMG techniques are surveyed here in terms of the 2D Ising model with the majority-rule coarsening, where they were first developed (see details in [52]). Each coarse level here consists again of a 2D array of Ising spins (± 1 variables), each of which being a "block spin", i.e., its sign representing the sign of the sum of the four spins in a corresponding 2×2 block of the next finer level. (The sign of zero is taken to be $+$ or $-$, each with probability $1/2$).

Coarsening. The action for each coarse level is derived by running an efficient MC simulation at the next finer level, during which statistics is gathered for estimating $P_+(u_1, u_2, \ldots)$, the probability for any block spin to be $+1$ given that its (ordered set of) neighboring block spins are having the values $(u_1, u_2, \ldots, u_m)$.

The size m of the considered neighborhood is *variable*: If only a small amount of statistics is gathered, only the four nearest neighbors (u_1, u_2, u_3, u_4) are considered. (In this case, due to the symmetries of the model, only two "equivalence classes" of neighborhoods need to be distinguished: all the neighborhoods within the same class must have the same P_+, hence only their joint statistics needs be gathered.) With more statistics, all *eight* closest (nearest and next-nearest) neighbors $(u_1, \ldots, u_8)$ are considered (or rather their 24 equivalence classes). Further on, when the amount of statistics *for a particular neighborhood* $(u_1, \ldots, u_8)$ (or for an equivalence class of such neighborhoods) is sufficiently large, that neighborhood is *split*, i.e., statistics is gathered for its "*child neighborhoods*": These are neighborhoods $(u_1, \ldots, u_8, u_9, \ldots, u_{12})$ with the same $(u_1, \ldots, u_8)$ as in the "parent" neighborhood, and with $(u_9, \ldots u_{12})$ representing values of the four *subsequent* neighbors (those which are exactly two meshsizes away). Children with enough statistics may further be split into grand-children, and so on.

The general rule is to split a neighborhood when (and only when) some of its children have enough statistics to make the difference between their P_+ values significant (i.e., larger than their standard deviations). Between several candidate splits of a neighborhood, the one with the largest *spread* (average child deviation) should be adopted. (Statistics for *several* candidates can simultaneously be accumulated in the same MC run.)

The P_+ probabilities represent the coarse-level action. Indeed, it is all we need, and exactly what we need, to run an MC simulation on that level (the level of blocks). Also, due to the adaptability in the size of the neighborhoods, the coarse action calculation is *statistically optimal*, in the sense that it will yield $O(\varepsilon)$ relative accuracy in calculating large-scale averages when the amount of statistics (the total number of random-number generations in producing the P_+ tables) has been $O(\varepsilon^{-2})$. This claim has been confirmed in a sequence of numerical tests. It may depend on having a fully efficient Monte-Carlo simulation at the fine level. Which is our next topic.

Monte-Carlo Acceleration. For a given lattice with a given action (possibly in the form of P_+ tables), suppose now that the actions for all its coarser levels (the level of blocks, the level of block of blocks, etc.) are also given (e.g., in P_+ tables). Then a new equilibrium of the given action can efficiently be produced using a "*half-V cycle*" (a name derived from classical multigrid nomenclature), defined as follows.

First an equilibrium is easily obtained at the coarsest level, by few MC passes with the corresponding action. From this, an equilibrium in the next level will be derived, and so on, until the target level (the given lattice) will be reached. To obtain an equilibrium in any level of spins given an equilibrium of its blocks, we use "stochastic interpolation", i.e., a number of "compatible Monte-Carlo" (CMC) passes. By this we mean Monte-Carlo passes at the spin level which keep the values of the blocks unchanged. The CMC has a very short decorrelation time: actually smaller than 1 in all our tests. (More generally, for any model: if the CMC decorrelation time is not short, than the definition chosen for the block variables is inadequate.) So only few (typically at most 4 or 5) CMC passes are really needed. If the coarse-level (the block) action has not been fully accurate, the CMC passes should be followed by a small number of regular MC passes, a process we call "post relaxation" (following again classical multigrid nomenclature).

Fast iterations. Since the derivation of a coarse action depends on efficient simulations at the finer levels, which in turn depends on employing the coarse action, iterating back and forth between these processes is in principle needed. However, these iterations converge very fast, since these processes only *weakly* depend on each other. Indeed, a very good first approximation to the P_+ block-spin tables is already obtained by just *local* equilibration at the spin level, produced by just a limited number (independent of the lattice size) of regular Monte Carlo passes (even starting from a completely random configuration). And, on the other hand, a very good approximation to critical-temperature equilibrium at the fine level can be obtained from a half-V cycle even with crude approximations to the coarse level actions (as long as they are kept critical: see below), provided a small number of post-relaxation sweeps is added.

In fact, an extremely simple way to obtain an equilibrium on a given lattice with a critical action is by a half-V cycle employing *this same action* at all levels, plus post relaxation at each level. It can be shown that the required number of post relaxation sweeps is small whenever the convergence to a fixed point of the renormalization flow is fast. For the Ising model we have confirmed, in a sequence of numerical tests, the excellent quality of equilibria obtained in this simple way [52].

Errors. The principal sources of errors in the RMG processes are the finite statistics, the truncated size of the neighborhoods for which P_+ is calculated, and the finite size of the lattice employed at each level.

The latter type of error is easily removed; arbitrarily large lattices can be used due to the fast equilibration, while the P_+ calculations have nearly-local nature at all levels; and the simulation at any coarse level can be extended to its own desired domain size, since it is done directly, not through simulations at a finer level.

The finite-statistics errors are well controlled so as to keep all of them, at all levels, at the same optimal order ε, where the amount of statistics is $O(\varepsilon^{-2})$. The truncation errors are also kept at $O(\varepsilon)$, by adjusting the neighborhood sizes; it is estimated that the size of the considered neighborhoods should grow proportionately to $\log(\varepsilon^{-1})$. The only trouble is the error enhancement from level to level, which is discussed next.

Back to criticality. In critical-temperature calculations of the P_+ tables, errors introduced at any level are magnified in the level derived from it (the next coarser level), and so on, due to the strong divergence of the renormalization flow away from the critical surface. To check this magnification, a mechanism should be added at each level to project the action (i.e., the P_+ tables) back onto the critical surface. Such a "criticalization" mechanism also facilitates calculating renormalization flows toward a fixed point when the critical temperature of the initial (finest-level) Hamiltonian is not known in advance.

The criticalization of a given action is done by multiplying the temperature by a suitable factor $1/\theta$. In terms of the P_+ tables, this means to raise each probability to the power θ, then normalize; i.e., to replace each value of P_+ by $P_+^\theta/[(1-P_+)^\theta + P_+^\theta]$. The criticalization factor θ can be estimated in a number of ways. In our fixed-point calculations we found it convenient to derive θ from quantities we were calculating anyway, such as the next-level P_+ values and the estimated value of the correlation-length critical exponent (whose derivation is discussed below).

More exactly, in the same MC runs (with the given action) in which P_+ statistics are gathered for the 2×2 block spins, similar P_+ statistics are also gathered for the 4×4 blocks (meaning, more precisely, 2×2 block of 2×2 blocks), and for the 8×8 blocks, etc. At criticality, these different statistics for different block sizes would coincide (at least for sufficiently large blocks; but close to the fixed point even for small blocks; the use of the latter is preferred since their P_+ tables are based on more statistics and faster equilibration). The differences between P_+ at two different block sizes, together with an (even rough) knowledge of the critical exponent, easily yields an estimate for the needed criticalization factor θ.

This criticalization process may be repeated several times, until those differences between the P_+ values at different block sizes become comparable to the statistical noise. Actually, however, such a repetition is not needed: applying the process just once at each level (each renormalization stage) is enough to drive the actions at subsequent levels ever closer to the critical surface. Even better is to apply the criticalization factor directly to the *next* action (the one that has currently been calculated for the 2×2 blocks). The return-to-criticality cost is then really negligible.

Fixed point algorithm. The fixed point of the renormalization group is quickly approached by a sequence of coarsening steps, as described above, with a criticalization factor applied to each new action in the sequence. At each step the accuracy is raised in every respect: The amount of statistics is much increased (e.g., by a factor 64) with a corresponding (slower) increase in the lattice linear size (e.g., by a factor 4) and a (logarithmically slow) increase in the size of the neighborhoods. For each new action, equilibration is obtained by a half-V cycle which employs this same action at all its levels.

Critical exponent. An early estimation of the correlation-length critical exponent ν (needed for calculating criticalization factors) may be done at any early stage of the fixed point algorithm. At advanced stages of the algorithm, increasingly accurate calculations of the exponent and the associated eigenvector can be done as follows.

The calculation is done in terms of the vector $P_+(u)$, where $u = (u_1, u_2, \ldots)$ runs in principle over all possible value at all neighbors. (Neighborhoods u for which P_+ has not been explicitly calculated are assumed to have the P_+ computed for their parent.) More directly, the computation is performed in terms of the vector $S(u)$ defined by $S(u) = \log\big[P_+(u)/(1 - P_+(u))\big]$. (Notice that the criticalization step is equivalent to multiplying the vector $S(u)$ by θ.) For any perturbation $q(u)$ to the vector $S(u)$ we define the norm

$$\| q \| = \left(\sum_u w(u) |q(u)|^2 \right)^{1/2} ,$$

where $w(u)$ is a non-negative number proportional to the amount of statistics gathered for $P_+(u)$, and $\sum_u w(u) = 1$. A vector q is called normalized if $\| q \| = 1$.

Each stage of the fixed point algorithm can be regarded as a transformation T, transforming $S(u)$ into $TS(u)$. The algorithm converges to the fixed point $S_*(u) = TS_*(u)$. The desired critical exponent can immediately be derived from the largest eigenvalue λ_* of the equation

$$T(S_* + q) = S_* + \lambda q \ , \qquad \| q \| \ll 1 \ .$$

Denoting by q_* the normalized eigenvector associated with λ_*, the task is, at each stage of the fixed point algorithm, to calculate (increasingly more accurate) values for λ_* and q_*.

Let q_0 be a (normalized) approximation to q_* obtained at the previous stage of the algorithm, together with the approximate action S_0. The current stage of the algorithm has computed a new action $S_1 = TS_0$. We then also calculate $\widetilde{q} = \big[T(S_0 + c\varepsilon^{1/2}q_0) - S_1\big]/(c\varepsilon^{1/2})$, where ε is the level of accuracy at the current stage (i.e., its amount of statistics is $O(\varepsilon^{-2})$). The new approximation to q_* is then defined as $q_1 = \widetilde{q}/ \| \widetilde{q} \|$, and λ_* is estimated by

$$\lambda_1 = \sum_u w(u)\widetilde{q}(u)q_0(u) \ .$$

RMG for the XY model. Initial steps of applying the above coarsening and acceleration techniques to the XY model are reported in [109]. Each 2×2 block spin is here defined to be the *average* of its four constituent spins, *without normalization* (whereby the original XY group of length-1 vectors is *not* preserved at the coarse levels). Compared with the ± 1 majority spins discussed above, each coarse spin here contains much more information; as a result, much smaller neighborhoods are needed in the probability tables to attain a given truncation accuracy. Still, these tables are more complicated than the above P_+ tables, since they should describe a *continuous* distribution, conditioned on *continuous* neighboring values.

To accumulate continuous-variable statistics, one of course partitions the range of this variable into *bins*: Counting the number of MC hits in each bin gives an estimate for the *integral* of the continuous variable over that bin. From those integrals, the value of the variable at any particular point can be *interpolated* (by a polynomial whose integrals over several adjacent bins fit the estimates). The same is true for a *vectorial* variable, such as the one representing the entire (truncated) neighborhood, whose bins may each be a tensor product of *elementary* bins, one elementary bin per each real variable participating in describing the neighborhood. More generally, the bins of the neighborhood are constructed *adaptively*, similar to the adaptive neighborhoods in the Ising case above, except that here a bin can be split into several bins in *two* ways: either by adding another variable to the description of that particular neighborhood, or by refining the current partition of one of the existing variables.

The set of tests with the XY model reported in [109], though still limited (for personal reasons), clearly indicates that ideal MC performance is obtained in calculating various thermodynamic limits, such as the two-point correlation and the scaled susceptibility.

Future plans. After further program improvements (more accurate and automatic implementation of the rules described above) and further testing of potential optimality for various observable, we plan to extend the RMG techniques to more advanced physical problems, including gauge field models such as U(1), SU(2) and SU(3). Together with the methods of Secs. 11–12 above, we hope ultimately to obtain (with proper collaboration) ideal MC performance also with unquenched fermionic interactions.

14. Molecular Mechanics

(with Ph.D. student Gershon Hochman, post doc fellow Billie Sandak and Drs. Dov Bai and Valery Ilyn)

14.1 Background and objectives

Molecular mechanics (or dynamics) is 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. 14.8 below), and to derive macroscopic equations at increasingly larger scales, leading eventually to continuum-level processing.

14.2 Research steps

The computational-complexity factors arising in molecular mechanics, and a multiscale approach for dealing with each of them, were specified in [29] (and partly mentioned also in [30] and [103]). 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), for accelerating either energy minimization or Monte-Carlo simulations; multiscale annealing (similar to that developed in [53] 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.

14.3 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 [73]), 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 [25] (based on an idea described earlier in [19, §8.6], [22, App. A] and [45], 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. 14.4 below, and also in Secs. 14.6 and 14.7).

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.)

14.4 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., [98] or the more sophisticated functional H in Sec. 14.8 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 [19, §13.1], [48], [50], [21], [102], 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 [29]. 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 have next studied 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 and angles between neighboring coarse-level atoms), approximating the value of x_0^f in such coordinates. 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 coarse-level molecule 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 fast convergence for all the stronger interactions is required in this local relaxation too. Actually, instead of the 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 [19, §8] or [20, §8]).

This 3-step interpolation is quite 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 it is exactly such movements that we want to perform on the coarse levels. So the expansion should be done in terms of the internal-local coarse-level coordinates mentioned above. 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; their smoothness facilitates doing it without any linearization (see (1) below).

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

(1) *Anterpolating* the electrostatic charges from their fine-level (Cartesian) positions to the (Cartesian) positions of the coarse-level atoms (see [25]: 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. 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). The first derivatives need not be very accurate either, provided they are consistently used in the correction described next.

(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

$$E^c(x^c) = \tilde{E}^c(x^c) + [I^T R_0^f - \nabla_{x^c} \tilde{E}^c(x_0^c)] \cdot x^c \tag{14.1}$$

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 (14.1) expresses a field-like correction (analogous to a similar term in FAS multigrid [19, §8]) which makes the first derivatives of E^c exactly correspond at coarsening to the current residual forces.

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. In fact, strict (zero-temperature) energy minimization is not of real practical interest. For this reason, *our interest has more recently shifted to finite-temperature calculations*, discussed next.

14.5 Monte-Carlo methods at equilibrium: General

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 lattice problems, multigrid Monte-Carlo algorithms were developed which overcome *both* these complexity causes (see Sec. 13 above, where these two causes, which multiply each other, are called the CSD factor and the volume factor, respectively). Based on the RMG approach (Sec. 13.2), we have developed two complementary types of multiscale Monte-Carlo methods for the fast simulation of atomistic systems: one type for macromolecules, the other for very large, flowing collections of small molecules (gases, liquids). The methods are described in the following two subsections.

In the future the intention is to combine those two types of methods to treat macromolecules in solutions.

14.6 Multiscale Monte Carlo for macromolecules

The approach is based on the type of coarse-level internal coordinates described above (Sec. 14.5), but the coarse-level "atoms" themselves are defined differently. Each coarse-level "atom" stands for the average location of several, say m, next-finer level atoms. The coarse-level Hamiltonian is developed by extensive, but only local, Monte-Carlo simulations, iteratively adjusting coarse-level distribution functions and correlation coefficients so as to fit coarse-level with fine-level simulations. This kind of calculation needs to be done only once for all similarly-structured molecular neighborhoods. Due to the employment of internal coordinates, and with suitable coarsening ratios 1:m, it turns out that most inter-coordinate coarse-level correlations can be neglected, yielding quite simple coarse-level Hamiltonians.

Numerical experiment have been conducted with the united-atom polymer model of [97], which involves all chemical bonding and Lennard-Jones interactions. We have used fixed 1:4 coarsening ratio. The coarse-level Hamiltonian includes, first of all, modified Lennard-Jones-type interactions in terms of cartesian coordinates. The exact formula for this interaction has been derived once-for-all (for this coarsening ratio), by averaging during MC simulations over all vectorial sums of fine-level Lennard-Jones forces involved in the interaction between two coarse atoms in a given distance bin. The rest of the coarse Hamiltonian is in terms of local coordinates: distances, angles and dihedral angles (between the *coarse* atoms, of course, which also form a chain). The distribution of each of these coarse coordinates, and correlation coefficients for neighboring coordinates (e.g., correlation between the angle ABC and the distance AB) where determined in local MC simulations, iterating a couple of times between fine-level and coarse level simulations.

The resulting coarse Hamiltonian did turn out to fulfill our expectations: With vastly-reduced number of degrees of freedom and allowing much larger simulation steps, it very accurately reproduced large-scale statistics: Averages and standard deviations of distances between any two atoms, for example, were reproduced with at most 3% error.

A particular difficulty we had to overcome in this development is the *triple local minima of the dihedral potential* given at the original (fine) level. In ordinary (one-scale) molecular dynamics simulations, many thousands of MC passes are required for one transition between these local minima to occur, since each such transition is not local: it calls for a large-scale molecular change, involving in fact simultaneous such transitions at several dihedrals. Our *coarse*-level simulations are in fact already free from this particular slowness. However, the slow dihedral transitions did threaten to slow down our simulations of medium-size chains at the *fine* level, needed in our *derivation* of the coarse Hamiltonian. (Even though this derivation needs only be done once for all, similar ones will be needed for each new local molecular configuration of future interest.) To eliminate this difficulty, we took advantage of the fact that, for the purpose of that derivation, the simulation is indeed needed only for a medium-size chain, of at most several dozen atoms. At this size one can simulate the molecule with global movements, which allows, at each MC step, changing a couple of internal coordinates while keeping all others fixed. Except that changing a centrally-located dihedral from one of its local minima to another is likely to cause the molecule to intersect itself, creating an unacceptable MC move. We have therefore searched for several combinations of *simultaneous dihedral changes* which are particularly likely to get accepted in such MC runs of medium-size molecules. We found such combinations by preliminary inexpensive MC runs that measured the changes in certain molecular distances caused by each possible combination, and singled out those combinations tending to cause small such changes.

In the future we first plan to test polymer models that also involve electrostatic interactions. Similarly to the approach described above (in Sec. 14.4) the *local part* of these interactions (see Sec. 14.3) will participate in the local MC determination of the internal-coordinate terms of the coarse Hamiltonian, while the *smooth parts* will be computed directly in Cartesian coordinates of the coarse atoms, to which the electrostatic charges (and/or dipoles) will be anterpolated.

Next we plan to extend our two-level one-polymer scheme to a *multi-level* one, deriving increasingly coarser levels, and to a *multi-polymer* one, enabling fast simulations of many interacting long chains.

In principle, a multiscale structure should be constructed *individually* for each given material or macromolecule. Our main objective will be to demonstrate the *feasibility* of such constructions, by working on a sequence of model cases. This will hopefully lead more and more specialists to join forces, building eventually, level after level, from small scales to increasingly large ones, *libraries of multiscale structures*. We believe that much of the future chemistry-related research and development can efficiently be computerized in terms of such libraries.

14.7 Multiscale Monte Carlo for fluids

The efficient equilibrium simulation of gases and liquids at the atomic level, needed for the derivation of their large-scale behavior and macroscopic equations, is a central problem in scientific computation. Direct fine-scale MC simulations tend to be extremely inefficient due to the very slow change of various kinds of *clusters* at various scales. Especially difficult are the calculation at critical conditions, where all scales interact. Of particular interest of course is the simulation for *water*, by itself or as a solvent.

The main two kinds of clustering difficulties associated with water and other fluids are *positional* clustering and *electrostatic* ones. We have decided to start out by studying in parallel the following two simple models, in each of which only one of the difficulties is present. The first model will deal with the molecular motions, the second — with the molecular dipole rotations.

A. Lennard-Jones fluid. In this model, describing noble gases, the single-atom molecules move solely under their mutual two-atom Lennard-Jones interaction. Clusters of atoms that move together, and clusters of "*holes*" (i.e., absence of atoms in a larger-scale atomic cluster) are stochastically formed, their likely scales depending on the physical conditions, such as temperature and density. At critical conditions all scales are likely. The larger the scale of clusters, the slower their changes in MC simulations.

Generally speaking, the multiscale approach here is again of the RMG type (cf. Sec. 13.2), but unlike the former cases (statistical mechanics in Secs. 13.2 and macromolecules in Sec. 14.6), here the coarse levels all *differ* from the finest one in their nature: at the finest level we have atoms at arbitrary locations, while each coarse level is defined on a uniform *lattice*. The value defined at each lattice point stands for the average fluid density in a lattice box around that point. The probability distribution of this density depend on neighboring-point densities, as specified in detail by a suitable "*conditional-probability (CP) table*". The CP table for each level is calculated by extensive local simulations at the next finer level (the level of a *lattice* with half the meshsize or, eventually, the finest level of *atoms*). As before (Sec. 13.2), this multiscale structure is used both for accelerating the finest-level Monte Carlo, and for inexpensive (statistically optimal) calculations of large-scale statistics.

The development of this multiscale structure is now in progress. Its ability to predict meaningful averages at criticality by fixed-point calculations is already being clearly indicated.

B. Fixed-location rotating dipoles. This model features a large set of electrostatic dipoles, of given strengths and fixed locations, rotating in their mutual fields in thermal equilibrium. Clusters of aligned dipoles tend to form, their sizes depending on the given temperature. Again, these clusters are very slow to change in ordinary MC simulations, making large-scale fluctuations extremely slow to average out.

As before (cf. Sec. 14.3), at any spatial scale the electrostatic interactions can be decomposed into the sum of their *smooth* part and a *local* part. In addition to using this decomposition for the fast summation of the dipole field, here it will also be used for accelerating the MC simulations and for cheap (coarse-level!) averaging over many large-scale fluctuations.

Similar to the Lennard-Jones fluid above, each coarse level here is defined on a grid, the mesh size being doubled at each coarsening. The vector defined at each lattice point stands for the dipole anterpolated from the next-fine-level dipoles. With this type of coarsening, the RMG methodology is again applied: a CP table at each level is derived by local MC simulations at the next finer level. Then this structure can be employed both for MC acceleration (e.g., by "half-V cycles"; see Sec. 13.2), and for calculation of the large-scale average electrostatic properties.

This system is currently under development. It is too early to report results.

Future plans include transition from our current 2D models to 3D, and simulations of real water models, such as TIP4P [81],[82], in which both molecular translations and rotations, in both Lennard-Jones and dipole interactions, are considered.

14.8 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. 14.4. Secondly it avoids the killing of highly-oscillatory modes (those vibrations that are not resolved by the time step), 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 (with the fast atomic vibrations serving as a natural heat bath), thus 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 [20]), 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.

15. Early-Vision Algorithms

(with former Ph.D. student Jonathan Dym, Ph.D. student Eitan Sharon and in collaboration with Dr. Ronen Basri)

15.1 Edge (or fiber) detection

In the early years of the Gauss Center (and before), 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 [67]. In particular, fast multiscale methods for enhancing and detecting *straight features* (straight edges or straight fibers) have been demonstrated [37], [38]: They detect *all* such features, of all widths lengths, positions and orientations, in just $O(N \log N)$ operations, where N is the number of pixels (picture elements) in the given picture. This has been achieved by constructing a hierarchical collection of numerical integrals of grey levels along straight segments of the pictures, with the lengths, positions and orientations of the segments chosen in such a way that: (1) the collection is *rich enough*, in the sense that any other straight-segment integral over the picture can readily be interpolated from the collection integrals; and (2) the collection is *fast to construct*, by using shorter segment integrals to calculate the next-longer ones, scale after scale.

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 [124] and [76]). As a representative example for our discussion here, we can take

$$v_j = \sum_{i=1}^{N_1 m} a_{ij} u_i, \qquad (j = 1, \ldots, N_1 m), \tag{15.1}$$

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 [124] 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 [124] would require $O(N_1^2 m^2)$ computer operations; multiscale methods, resembling those of [25], would 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 (15.1). 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 [38], [37]. The multi-resolution of both the input (straight responses) and the 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. 15.2), whether or not the latter are themselves multiscaled.

A detailed study of multiscale completion fields, their parameterization and fast implementation is summarized in [106].

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.

(4) Combining in various ways edge detection with picture segmentation (see next).

15.2 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. 15.1 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 [48], [50], [21], [102]).

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 *iterated weighted aggregation*. In an iterated 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 will then accumulate at the higher levels of the process, automatically reinforcing each other where appropriate.

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.

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 levels of the blocks. More generally, all kinds of other intra-block "*observables*" can be used: the block's center of mass, its diameter, principal orientation, texture measures (being, e.g., statistics on average sizes and directions of smaller-scale sub-blocks), etc., with the number of observables per block increasing at coarser levels. For example, strong couplings can be assigned between two (not necessarily neighboring) aggregates whose principal orientations align with the direction of the line connecting their centers of mass.

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 observable).

A detailed account of our current multiscale algorithm for image segmentation is given in [107], demonstrating its properties on several line drawings and real images. As explained there, the cost of the algorithm is only several dozen operations per pixel.

The future plan is to enhance the algorithm in several ways indicated above: adding more coarse-level observables; introducing various interactions with our multiscale edge detection processes; etc. Multiscale approaches to other aspects of image processing are also envisioned.

16. Tomography: Medical Imaging and Radar

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.

16.1 Inverting the Radon transform and related problems

(with post-doctoral fellow Jordan Mann, Ph.D. student Meirav Galun and M.Sc. students Matthew Brodski and Eyal Shai)

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 [64]. One of the best known reconstruction algorithms is the convolution backprojection method (CB), which is widely used in commercial medical CT devices [64] (with "rebinning" for divergent-beam projection [78]). It has also been applied to spotlight-mode synthetic aperture radar (SPSAR) image reconstruction [78]. 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 sometimes used, 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), the Single Photon Emission Computed Tomography (SPECT), impedance tomography, ultrasound and similar medical imaging techniques (see, e.g., [75]), as well as 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, even less expensive, post-processing procedure [46], [47]. Empirical results for a number of phantoms, and measurements of point-spread functions, show that the combined method produces at least as good images as 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.

Future work. Fast algorithms for other fields with line-integral transforms are at the initial stage of development. Direct and inverse computation of line integrals of the two-dimensional SPECT can be done by multiscale computation in $O(N^2 \log N)$. (The direct algorithm is an extension of the line integral computation for the direct Radon transform [5].) Another important problem that can be solved by fast multiscale computation is *restricted-sector* view data. In CT, for example, data is usually taken over 180° degrees, but there are cases when the X-ray tomograph scans only part of the full view. In such cases a fast multiscale algorithm can still be developed, by constructing a new type of multiscale transform to replace in the classical convolution. This construction can be done once for all, in off-line iterations. *Three dimensional Cone-Beam* reconstruction can be achieved by fast multiscale solver in $O(N^3 \log N)$ complexity, using a similar off-line construction of a suitable multiscale transform together with a 3D version of our backprojection algorithm. A more complicated problem is the *three dimensional* PET reconstruction. In this case, a typical situation is that the number of emission events is much smaller than the number of possible rays. Multiscale processes can be applied here to efficiently perform *three* types of tasks: the gathering and averaging of the event data; the backprojection; and, again, a multiscale transform to replace the convolution, constructed off-line.

Even more complicated is the general SAR reconstruction, especially when the antenna motion (in airplane or satellite) cannot be neglected (bringing Doppler shifts and nonlinearity into the problem). This problem will require an iterative solution, employing the multiscale algorithms both for the direct and the inverse transform, using also the methods we plan to develop for the restricted sector case.

16.2 Impedance tomography: inverse diffusion problem

(with M.Sc. student Faina Shmulyian and former M.Sc., now Ph.D. student Rima Gandlin)

An EIT (Electrical Impedance Tomography) device for medical use consists of a set of N electrodes attached to the chest of a patient. A small *known* current is passed between two driver electrodes. In each measurement the current is passed through a different electrode pair, while the voltage drops at *all* the electrodes is recorded. The collected data are used in order to calculate the conductivity distribution in a part of the patient's chest and then to display it on a screen, in order to detect anomalies, such as tumors.

The electrical potential satisfies the equation $\nabla(\sigma \nabla u) = 0$, where σ is the electrical conductivity. The set of measurements gives ideally (in the limit of many small electrodes and as many measurements) the *Neumann to Dirichlet* mapping: the Dirichlet (u) boundary condition resulting from any Neumann ($\partial u/\partial n$) condition. The inverse EIT problem is to calculate σ from this mapping.

The first description of the inverse EIT problem was given by Calderon [61]. Kohn and Vogelius [89] showed that under certain assumptions the conductivity of a medium is uniquely determined by the Neumann-to-Dirichlet mapping. Then Sylvester and Uhlmann [115] provided a general framework for proving uniqueness of the solution of the inverse problem. Alessandrini [1] gave a mathematical explanation for the blurriness of conductivity images and proved that the conductivity depends on the EIT data in a very weak way. Therefore the inverse problem of

EIT is ill-posed, and a regularization is necessary if conductivity is to be obtained stably from data.

There exist some works on numerical methods for the relevant problems, but their number is rather sparse and even those papers do not consider the question of numerical efficiency, despite its importance for applications.

The main purpose of our work on this problem has been to demonstrate two general methodological points. First, an ill-posed problem is not necessarily difficult or expensive to solve. On the contrary: once the nature of the ill-posedness has been generally understood, the solution may even be much less expensive than solving the direct problem. For example, in the inverse EIT problem, employing local Fourier decompositions one can show that all components of wavelength λ are ill-posed at distances $r \gg \lambda$ from the boundary. Hence there is no need to use at such distances fine solution grids: all we can know about the solution can be calculated with grids whose meshsizes increase proportionality to r. Moreover, one can recombine the different measurements into N new ones, such that in the k-th measurement the electric current enters the k-th electrode and leaves *uniformly* through all other electrodes. Then it can be shown that in solving the k-th (direct) problem one needs a fine grid only near the k-th electrode, with increasingly coarser grids away from it.

The second general point is that such a careful choice of grids, in a suitable multigrid algorithm, can *replace the need for explicit regularization of the problem.*

Our first pass at the problem, though, employed the well known Tikhonov regularization method, reformulating the inverse problem as a variational minimization problem. The resulting Euler equations form a PDE system (for u, σ and a Lagrange-multiplier function), which make the problem suitable in principle for an effective numerical solution by multigrid methods. The FMG solvers were designed with large and then with progressively smaller regularization. Special attention has been paid to properly adapting many features of classical multigrid to the case of the problem under consideration (including intergrid communications, boundary condition treatment and coarse grid solution). In the case of large regularization, numerical experiments have demonstrated a good convergence of the developed solver, but the obtained solution is too smeared and doesn't approximate the real conductivity function too well.

At *small regularization values* the final approximation is much better, especially near the boundary. In this case, however, the system is no longer elliptic, and much more sophisticated relaxation methods are necessary, featuring a DGS scheme [20, §3.7], which effectively decomposes the system into its scalar factors. With this approach, although the multigrid cycles asymptotically slow down, the final approximation *to the conductivity* is practically obtained by just *one* multigrid cycle per grid refinement, even when approaching the smallest regularization for which solution still exists.

It took two M.Sc. Theses [108], [71] to complete this part of the program.

Then, the solution method *without regularization* has been developed. Preliminary results [72] show it to give better approximations to σ than the regularized method with its many artificial parameters (the regularization coefficients, which should change over the domain), for less work (no Lagrange multipliers). However, it also turns out that without regularization the solver requires a control parameter p to be gradually decreased through the FMG algorithm, allowing in each stage only those σ changes whose "profit" (in terms of improving the approximations to the Dirichlet data) per unit change is at least p.

17. Algebraic Multigrid (AMG): New Approaches

Algebraic multigrid (AMG) algorithms are solvers of linear systems of equations which are based on multigrid principles but do not explicitly use the geometry of grids; see [48], [50], [21], [102], [113]. the emphasis in AMG is on automatic procedures for *coarsening* the set of equations, relying exclusively on its algebraic relations. AMG is widely employed for solving discretized partial differential equations (PDEs) on unstructured grids, or even on structured grids when the coarse grid can no longer be structured, or when the PDE has highly disordered coefficients. AMG can also be used (as in [48]) for certain discrete systems not arising from differential equations.

The scope of AMG solvers has been rather limited, though. Its coarsening procedures have been inadequate for general non-scalar, or high-order, or non-elliptic and anisotropic PDE systems, and also for non-variational discretizations. The purpose of our recent work [33] has been to delineate general algebraic coarsening techniques that can be employed in all those cases and also in nonlinear systems. It has been inspired by the RMG method (see Sec. 13.2), which employs an analogous approach for non-deterministic problems.

Two types of devices have been developed in [33]. The first is a general criterion for gauging, and a method to control, the quality of the *set* of coarse-level variables, prior to deriving the coarse-level equations. The second includes general approaches for deriving the coarse-level *equations* once the coarse variables are given.

Two such approaches have been advanced. The first one is based on the premise that although in principle each coarse variable depends on *all* others, this dependence decays exponentially with distance. (For 2D discrete Poison equations, for example, the dependence size tends (after enough coarsening levels) exactly to $\exp(-\pi r^2/2)$, where r is the distance measured in meshsizes of the coarse level [104], [131].) Hence a highly accurate coarse equation can be constructed locally. This is done by solving a certain local optimization problem. We call this approach *direct coarsening*.

The second approach is based on the traditional *Galerkin coarsening*, where the interpolation and restriction operators can again be very accurately derived by solving local optimization problems.

A detailed comparison of the two approaches has not yet been done. The Galerkin approach seems to enjoy less expensive setup. In the cases where this is not the overriding issue, and in particular in highly repetitive systems (as most natural systems tend to be, provided they are not linearized or carelessly discretized), the direct approach is most widely applicable: it can be extended to nonlinear equations (as outlined in [33, §8]). In statistical physics, the Galerkin coarsening corresponds to the interpolation-based methods (Sec. 13.1 above), while the direct coarsening corresponds to the RMG method (Sec. 13.2).

In both these approaches one can control the level of coarsening accuracy, and the corresponding amount of computational work per coarse equation, by choosing the size of certain stencils: The error in approximating sufficiently "smooth" components (i.e., those slow to converge in relaxation) decreases *exponentially* with the size of those stencils, while the work per coarse equation increases proportionally to some *power* of that size.

For the purpose of multi-level (multigrid) *cycles*, a moderate coarsening accuracy would usually suffice. For example, a coarse grid equation with at most 10% error for all "smooth" components (i.e., those slow to converge in relaxation) can yield a multilevel cycle with a convergence factor close to 0.1. By performing successfully any number of such cycles, any desired accuracy can rapidly be obtained. This will usually be more cost effective than deriving higher accuracy coarsening.

On unstructured grids and for other disordered algebraic systems and even on structured grids in the case of non-elliptic systems, even that moderate accuracy may be difficult to obtain by standard approaches, so the use of the new coarsening techniques can be very useful.

Moreover, in many other cases, the still higher degrees of coarsening accuracy obtainable through those techniques (by investing more work) are really needed. *Examples*:

(i) Problems with a large number of almost-zero modes (AZMs), i.e., eigenvectors with unusually close to zero eigenvalues. Such modes often reflect some ill defined global moves, such as rigid-body motions of the entire system in problems of elasticity, or a gliding motion of two rigid bodies along their contact surface. Such AZMs also plague various disordered problems, such as Dirac equations on critical gauge fields (cf. Sec. 11). For problems with *many* AZMs, a general cure is to increase the coarsening accuracy. A small number m of AZMs (such as those associated with global rigid body motions) may still persist even at higher accuracies, but they can be eliminated by recombining $m+1$ iterants (each being, for example, the approximate solution obtained after another multi-level cycle) so as to minimize the ℓ_2 residual norm; see, e.g., [51].

(ii) Once-for-all coarsening, for the purpose, e.g., of deriving the *macroscopic* *equations* of the given system, or *homogenizing* it.

(iii) Cases in which one needs to solve many linear systems of the form $Ax = b$, where large parts of A and b do not change from problem to problem, so recomputing at fine levels of those parts can be avoided by having accurately coarsened them before. One important such case is the calculation of many desired terms (e.g., the main diagonal) of A^{-1}. Moreover, in important cases (e.g., see Sec. 12 above), those terms of A^{-1} need be repeatedly updated upon each change in A occuring during certain Monte-Carlo simulations.

(iv) A computing environment which makes it preferable to use as few multigrid cycles as possible, such as massive parallel processing with poor inter-processor communications, or a computer with a very high-speed cache memory.

References

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

[2] E. Arian and S. Ta'asan, *Multigrid one shot methods for optimal control problems: Infinite dimensional control*, ICASE Report 94–52 (1994.)

[3] E. Arian and S. Ta'asan, *Shape optimization in one shot*, In Optimal Design and Control (J. Boggard, J. Burkardt, M. Gunzburger and J. Peterson, eds.), Birkhauser Boston Inc., 1995.

[4] I. Babuška and J.M. Melenk, *The partition of unity method*, International J. for Num. Methods in Engineering 40 (1997), pp. 727–758.

[5] D. Bai and A. Brandt, *Local mesh refinement multilevel techniques*, SIAM J. Sci. Stat. Comput., 8 (1987), pp. 109–134.

[6] J.R. Bates, Y. Li, A. Brandt, S. McCormick and J. Ruge, *Numerical integration of the primitive equations based on the semi-Lagrangian advection of potential vorticity.* In Proc. of the Symposium on Cyclones, Bergen, Norway, June 27–July 1, 1994.

[7] 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.

[8] 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.

[9] 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.

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

[11] R. Ben-Av, *Acceleration Methods in MC Simulations*, Ph.D. Thesis, Weizmann Institute of Science, 1991.

[12] R. Ben-Av, A. Brandt, M. Harmatz, E. Katznelson, P.G. Lauwers, S. Solomon and K. Wolowesky, *Fermion simulation using parallel transported multigrid*, Phys. Lett B, 253 (1991) 185.

[13] R. Ben-Av, A. Brandt and S. Solomon, *The fermiomic matrix, instantons, zero modes and multigrid*, Nucl. Phys. B 329 (1990) pp. 193.

[14] 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.

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

[16] A. Brandt, *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.

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

[18] A. Brandt, *Multi-grid solvers for non-elliptic and singular-perturbation steady-state problems*, Weizmann Institute of Science, Rehovot, November 1981.

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

[20] A. Brandt, *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.

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

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

[23] 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.

[24] A. Brandt, *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 [28].

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

[26] 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.

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

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

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

[30] A. Brandt, *Proceedings of the Multigrid Tutorial, with Applications to Molecular Dynamics*, Weizmann Institute of Science, October 10–12, 1995.

[31] A. Brandt, *The Gauss Center research in multiscale scientific computation*, Elect. Trans. Numer. Anal. 6 (1997), pp. 1–34.

[32] A. Brandt, *Barriers to Achieving Textbook Multigrid Efficiency in CFD*, ICASE Interim Report No. 32, NASA/CR-198-207647. Gauss-Center Report WI/GC–10, June, 1998; updated December, 1998.

[33] A. Brandt, *General highly accurate algebraic coarsening schemes*, submitted to Electronic Trans. Num. Anal., May, 1999.

[34] 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.

[35] A. Brandt and B. Diskin, *Multigrid solvers for the non-aligned sonic flow: the constant coefficient case*. Gauss Center Report WI/GC–8 (1997). Computer & Fluids, 28 (1999), pp. 511–549.

[36] A. Brandt and B. Diskin, *Multigrid solvers for non-aligned sonic flows*, SIAM J. Sci. Comput., in press.

[37] A. Brandt and J. Dym, *Fast computation of multiple line integrals*, SIAM J. Sci. Comput., in press.

[38] A. Brandt and J. Dym, *Fast robust discontinuity detection using multiple scales*, a preprint.

[39] A. Brandt and M. Galun, *Optimal multigrid algorithms for the massive

Gaussian model and path integrals, J. Statist. Phys., 82 (1996), pp. 1503–1518.

[40] A. Brandt and M. Galun, *Optimal multigrid algorithms for variable-coupling isotropic Gaussian models*, J. Statist. Phys., 88 (1997), pp. 637–664.

[41] A. Brandt and M. Galun, *Statistically optimal multigrid algorithm for the anharmonic crystal model*, Gauss-Center Report WI/GC–9, March, 1998.

[42] A. Brandt, M. Galun and D. Ron, *Optimal multigrid algorithms for calculating thermodynamic limits*, J. Statist. Phys., 74 (1994), pp. 313–348.

[43] 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.

[44] A. Brandt and I. Livshits, *Wave-ray multigrid methods for standing wave equations*, Electron. Trans. Numer. Anal., 6 (1997), pp. 162–181.

[45] 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.

[46] A. Brandt, J. Mann and M. Brodski, *A Fast and Accurate Radon Transform Inversion Scheme*, Assigned to Yeda Research and Development Co. Ltd., August 1995. U.S. Patent and Trademark Office Application No. 08/659,595, filed 06/06/96. U.S. Patent No. 5,778,038, granted July 7, 1998 under the title "Computerized Tomography Scanner and Method of Performing Computerized Tomography". European Patent Office Application No. 97l08722.6-2305, filed 05/30/97.

[47] A. Brandt, J. Mann, M. Brodski and M. Galun, *A fast and accurate multilevel inversion of the Radon transform*, SIAM J. Appl. Math., to appear.

[48] 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.

[49] A. Brandt, S. McCormick and J. Ruge, *Multi-grid methods for differential eigenproblems*, SIAM J. Sci. Statist. Comput., 4 (1983), pp. 244–260.

[50] A. Brandt, S. McCormick and J. Ruge, *Algebraic multigrid (AMG) for sparse matrix equations*, in Sparsity and its Applications, D.J. Evans, ed., Cambridge University Press, Cambridge, 1984, pp. 257–284.

[51] A. Brandt and V. Mikulinsky, *Recombining iterants in multigrid algo-*

rithms and problems with small islands, SIAM J. Sci. Comput., 16 (1995), pp. 20–28.

[52] A. Brandt and D. Ron, *Statistically optimal renormalization group flow and coarse-to-fine Monte Carlo acceleration*, Gauss Center Report WI/GC-11, June, 1999.

[53] 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.

[54] 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., 19 (1998), pp. 468–492.

[55] A. Brandt and C.H. Venner, *Multilevel evaluation of integral transforms on adaptive grids*, Gauss Center Report WI/GC-5 (1996). Proc. European Multigrid Conference 1996, in press.

[56] A. Brandt and I. Yavneh, *On multigrid solution of high-Reynolds incompressible entering flows*, J. Comput. Phys., 101 (1992), pp. 151–164.

[57] A. Brandt and I. Yavneh, *Accelerated multigrid convergence and high-Reynolds recirculating flows*, SIAM J. Sci. Comput., 14 (1993), pp. 607–626.

[58] 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.

[59] A. Brandt and L. Yu. Zaslavsky, *Multiscale algorithm for atmospheric data assimilation. Part I. Multiscale iterative process*, Technical Report CS95–09, The Weizmann Institute of Science, Rehovot, Israel, 1995.

[60] A. Brandt and L. Yu. Zaslavsky, *Multiscale algorithm for atmospheric data assimilation*, SIAM J. Sci. Comput., 18 (1997), pp. 949–956.

[61] 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.

[62] T.F. Chan and H.B. Keller, *Arc-length continuation and multi-grid techniques for nonlinear elliptic eigenvalue problems*, SIAM J. Sci. Stat. Comput., 3 (1982) pp. 173–194.

[63] T.F. Chan and R.E. Bank, *PLTMGC: A multi-grid continuation program for parametrized nonlinear elliptic systems*, SIAM J. Sci. Stat. Comput., 7 (1986), pp. 540–559.

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

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

[66] B. Diskin, *Efficient Multigrid Solvers for the Linearized Transonic Full Potential Equation*, Ph.D. Thesis, Feinberg Graduate School, Weizmann Institute of Science, Rehovot, Israel, June 1998.

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

[68] G. Galli, *Linear scaling method for electronic-structure calculations and quantum molecular dynamics simulations*, Current opinion in solid state and material science, 1(6) (1996), pp. 864–874.

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

[70] M. Galun, *Multigrid Algorithms for Optimal Computations in Statistical Physics*, Ph.D. Thesis, Weizmann Institute of Science, April, 1998.

[71] R. Gandlin, Multigrid Algorithms for an Inverse Problem in Impedance Tomography, M.Sc. Thesis, Weizmann Institute of Science, June 1998.

[72] R. Gandlin, Proposal for Ph.D. Thesis, 1999.

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

[74] J.C. Greenwald, *Multigrid Techniques for Parabolic Problems*, Ph.D. Thesis, Weizmann Institute of Science, June 1992.

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

[76] G. Guy and G. Medioni, *Inferring global perceptual contours from local features*, Proc. DARPA Image Understanding Workshop, (1993), pp. 881–893.

[77] Z. Haras and S. Ta'asan, *Finite difference schemes for long time integration*, J. Comp. Phys., 114(2) (1994).

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

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

[80] 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.

[81] W.L. Jorgensen, J. Chandrasekhar, J.D. Madura, R.W. Impey and M.L. Klein, *Comparison of potential functions for simulating liquid water*, J. Chem. Phys., 79 (1983) pp. 926–935.

[82] W.L. Jorgensen and C. Jenson, *Temperature dependence of TIP3P, SPC and TIP4P water from NPT Monte Carlo simulations: seeking temperatures of maximum density*, J. Comp. Chem., 19 (1998) pp. 1179–1186.

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

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

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

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

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

[88] J.B. Kogut and L. Susskind, *Hamiltonian formulation of Wilson's lattice gauge theory*, Phys. Rev. D, 11 (1975), pp. 395–408.

[89] P. Kohn and M. Vogelius, *Determining conductivity by boundary measurements*, Comm. Pure Appl. Math., 37 (1984) pp. 289–298.

[90] Y. Li, and J.R. Bates, *A study of the behavior of semi-Lagrangian models in the presence of orography*, Quart. J. R. Met. Soc, 122 (1996) pp. 1675–1700.

[91] Y. Li, J. Ruge, J.R. Bates and A. Brandt, *A proposed adiabatic formulation of three-dimensional global atmospheric models based on potential vorticity*. Submitted to Tellus A, December, 1998.

[92] O. Livne, *Multilevel Solvers for Eigenbasis computations of Periodic Schrödinger Opertor*, Final Report to Feinberg Graduate School, Weizmann Institute of Science, March 1999.

[93] I. Livshits, Multigrid Solvers for Wave Equations, Ph.D. Thesis (supervised by A. Brandt), Bar-Ilan University, November, 1995.

[94] J.C. McWilliams, J.B. Weiss and I. Yavneh, *Anisotropy and coherent vortex structures in planetary turbulence*, Science, 264(5157) (1994) pp. 410–413.

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

[96] V. Ogranovich, *Investigation of multigrid algorithms for nonelliptic problems*, Weizmann Institute M.Sc.. Thesis, Oct. 1994.

[97] W. Paul, D.Y. Yoon and G.D. Smith, *An optimized united atom model for simulations of polymethylene melts*, J. Chem. Phys. 103 (1995) pp.

1702–1709.

[98] C.S. Peskin and T. Schlick, *Molecular dynamics by backward-Euler method*, Comm. Pure Appl. Math., 42 (1989), pp. 1001.

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

[100] M. Rozantsev, *Multigrid Solver for 4D Quantum Chromodynamics Equations*, Progress Report, Feinberg Graduate School, Weizmann Institute of Science, Rehovot, Israel, August, 1997; Final Report, January, 1999.

[101] 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.

[102] J. Ruge and K. Stüben, *Algebraic multigrid*, in Multigrid Methods, S.F. McCormick, ed., SIAM, Philadelphia, 1987, pp. 73–130.

[103] 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.

[104] J. Schröder, U. Trottenberg and H. Reuterssberg, *Reuktionsverfahren fuer differenzengleichungen bei randwertaufgaben II*, Num. Math., 26 (1976), pp. 429–459.

[105] H. Sharatchandra, H. Thun and P. Weisz, *Susskind fermions on a Euclidean lattice*, Nuc. Phys. B192 (1981) 205.

[106] 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. To appear in IEEE Trans. on Pattern Anal. and Machine Intelligence.

[107] E. Sharon, A. Brandt and R. Basri, *Fast multiscale image segmentation*, Submitted to IEEE International Conference in Computer Vision, Greece, Sept. 1999 (ICCV–99).

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

[109] S. Shmulyian, *Toward optimal multigrid Monte Carlo computations in two-dimensional $O(N)$ non-linear σ-models*. Ph.D. thesis, Feinberg Graduate School, Weizmann Institute of Science, Rehovot, Israel, March 1999.

[110] 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.

[111] 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.

[112] 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.

[113] K. Stüben, *Algebraic multigrid (AMG): An introduction with applications*, GMD Report 53, March 1999.

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

[115] J. Sylvester and G. Uhlmann, *A global uniqueness theorem for an inverse boundary value problem*, Annals of Math, 125 (1987) pp. 153–169.

[116] S. Ta'asan, *One-shot methods for optimal control of distributed parameter systems, I. Finite dimensional control* ICASE Report 91–2, (1991).

[117] E. Tziperman, I. Yavneh and S. Ta'asan, *Multilevel turbulence simulations*, Europhysics Let. 24(4), (1993) 239.

[118] 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.

[119] S. Vandewalle and G. Horton, *Fourier mode analysis of the waveform relaxation and time-parallel multigrid methods*, SIAM J. Sci. Comput., 1995.

[120] 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.

[121] 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.

[122] 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.

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

[124] 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.

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

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

[127] I. YAVNEH, *Multigrid smoothing factors of red-black Gauss-Seidel applied to a class of elliptic operators*, SIAM J. Num. Anal., 32 (1995), pp. 1126–1138.

[128] I. YAVNEH AND J.C. MCWILLIAMS, *Breakdown of the slow manifold in the shallow-water equations*, Geo. and Astr. Fluid Dyn., 75 (1994), pp. 131–161.

[129] I. YAVNEH AND J.C. MCWILLIAMS, *Robust multigrid solution of the shallow-water balance equations*, J. Comput. Phys., 119 (1995) pp. 1–25.

[130] I. YAVNEH, C.H. VENNER AND A. BRANDT, *Fast multigrid solution of the advection problem with closed characteristics*, SIAM J. Sci. Comput., 19 (1998), pp. 111–125.

[131] H. ZIMARE, *Beweis der sternenkonvergenz und untersuchung der stabilitaet beim verfahren der totalen reduktion*, Ph.D. Thesis, University of Cologne, 1980.