Parallel Implementation of the Regional Spectral Atmospheric Model

By

Masao Kanamitsu

and

Hideki Kanamaru

Climate Research Division
Scripps Institution of Oceanography
University of California, San Diego

Yifeng Cui

San Diego Supercomputer Center
University of California, San Diego

and

Henry Juang

National Centers for Environmental Prediction
National Oceanic and Atmospheric Administration

Introduction

The first generation regional spectral models utilized Fourier series as base functions (Machenhauer and Haugen 1993; Gustafsson and McDonald 1996). This approach required cyclic boundary conditions in X- and Y-directions, which was accommodated by adding buffer zones in the east-west and north-south directions. Despite the cleanness of the model formulation and its good performance in selected cases, the model has not been tested for a large number of cases as required in operational implementation, and therefore, no specific merit/demerit of the model is available. There is a possibility that an artificial zone tended to degrade the regional integrations by erroneous propagation of signal from the eastern boundary to the western boundary (as well as from the northern to southern boundary) through cyclic boundary condition, but previous studies have not examined this in any detail.

A major breakthrough in this situation occurred when the first successful operational regional spectral model was developed by Tatsumi (1986). He applied sine and cosine expansion (not the full Fourier series) to the *difference* between the full and an idealized base field. Spectral representation of differences, not the total field, was the key to success since the difference satisfies either the rigid or symmetric lateral boundary conditions without artificial buffer zones. Hoyer (1987) at the European Centre for Medium Range Weather Forecasts (ECMWF) and Juang and Kanamitsu (1994) at the United States National Meteorological Center (NMC) further extended the Tatsumi method by utilizing the perturbation from a global analysis/forecast base field, rather than the difference from an idealized base field. This procedure added physical meaning and reduced the amplitude of perturbations, dramatically improving the regional model performance. This model has been extensively used by the Japan Meteorological Agency (JMA), ECMWF and NMC as an operational model as well as for research. Further refinements of the NMC and JMA models, such as the reduction of the effect of discontinuities of topography at the lateral boundaries, and improvement in lateral boundary relaxation, are described in Juang et al. (1997) and in Segami et al., (1989).

There are several advantages of the RSM over the grid point models, but three major ones are noted here. The first is the high accuracy and efficiency of the spectral calculations. Compared to grid point methods, the spectral method has negligible truncation error and no phase error. It also satisfies important conservation properties of the equations without requiring any special treatment. The spectral method is also efficient in inversing the Laplacian operator, which appears in the semi-implicit integration methods. The second advantage is that the use of a global base field for the computation of perturbation allows much longer integration without significant deterioration, making the model best suited for climate applications. This method also enables the use of special filtering, an attribute that will be discussed in a separate report. The third, very significant benefit is that the model can be formulated in a manner very similar to the global spectral model (GSM). This allows almost complete integration of the regional and global models into one package, allowing easier maintenance and development. It also allows a simple and physically consistent downscaling of existing large scale simulations by GSM. In particular, the NCEP/NCAR Reanalysis (Kalnay et al. 1996) of the last 5 decades of

model-analyzed "observations" of the 3-dimensional atmosphere have been produced from a version of GSM, and are thus available as the global "base field" from which the regional simulations herein are derived.

The forerunner of the regional spectral model (RSM) employed here was transferred from the National Meteorological Center (NMC) to Scripps Institution of Oceanography (SIO) in the early 1990's. The model has been maintained and developed jointly with NMC since then. This model is now used widely for research at SIO as well as at universities and operational forecast centers around the world (Roads, 2000). The model was also implemented into an operational forecast suite at NCEP in 2000 (Kanamitsu et al., 2003).

The model is designed to run on different computer systems, including single processor PCs, high power workstations, Cray vector processor super-computers, massively parallel computers, and massively parallel vector processor machines. The model can be run by researchers in developing nations using older computing systems, as well as by those using state-of the-art supercomputers. This extreme portability was accomplished by the basic design of the model and an extended use of C pre-processor. Recently, consolidation of the global and regional models into one system was completed at Scripps. With this integration, the code shares all the physical parameterization schemes as well as many of the i/o's, including diagnostic outputs between global and regional models, making the maintenance and the development of regional and global model extremely simple. We named this integrated model G-RSM. Additionally, we use GSM to refer to the global part and RSM to refer to the regional part of the integrated G-RSM system in this report.

In this report, we will describe in some detail the design, evaluation and performance of the "parallel" version of the regional part of the G-RSM. The parallelization of the global part of G-RSM was completed in 2000, in order to provide model code that is compatible with, and efficient in running on simultaneous, multiple processor (parallel) computing platforms. To take advantage of multi-processor platforms in running the RSM, it was necessary to extend the parallelization to the regional part. Such a capability is particularly important in order to conduct extensive climate model simulations, which often require massive amounts of calculations that are only available on parallel computing platforms. Parallelization of the RSM required fairly straightforward modification of the Legendre transform in the GSM to sine/cosine transform, but it additionally requires parallelization of computations for lateral boundary conditions and perturbation calculations, which do not exist in the GSM. The parallelization was first conducted on the IBM SP3 machine, and later adapted to the Earth Simulator machine. The latter machine requires an extended optimization effort that is not needed for SP3. Details of this optimization are also described in this report.

2. Regional Spectral Model

The basic concept of RSM (Juang and Kanamitsu, 1994) is to apply sine and cosine series to the deviation of the full forecast field from global base field (referred to as

perturbations). The basic formulation is written in terms of perturbations, but the actual calculation of tendencies is made using the full field, since it is not possible to write the entire prediction equations using perturbation as dependent variables. Strictly speaking, since the model uses full field, it is not a perturbation model; rather it should be regarded as an optimum perturbation filtering model. This becomes more apparent later in this section.

The computational procedure is summarized as follows: First, the forcing terms of the full field tendencies (right hand side of tendency equations) are computed as a sum of perturbation forcing and base field forcing, which are computed independently. Base field quantities are provided from the large scale model or analysis, while perturbations are deviations from this large scale structure that are imposed by the more detailed regional domain. Note that perturbation and base fields are both expressed in sign/cosine and spherical harmonics, respectively, so that the space derivatives are computed analytically without any space truncation error. Then, perturbation tendencies are calculated as a difference between the computed tendencies and the base field tendencies. This perturbation tendency is expanded to sine and cosine series, and finally the spectral perturbation of the next time step is calculated by advancing the time scheme by a step. As mentioned earlier, strictly speaking, this method is not a perturbation prediction, but is rather more like an optimal space filtering technique, since perturbation tendencies/perturbation field are computed as the difference between full-field-tendency/full field and base field-tendency/base field.

The basic formulation in RSM is the primitive equation system, consisting of the momentum equation, hydrostatic equation, thermo-dynamic equation and mass continuity equation. The dependent variables are the zonal and meridional component of winds, virtual temperature, specific humidity and log of surface pressure. The model utilizes a terrain following sigma coordinate system. The primitive equation system assumes that the horizontal scale is less than the vertical scale (which leads to hydrostatic assumption). This limits the refinement of the horizontal resolution of the model to about 10km. It should also be noted that many of the physical parameterizations used in the model are formulated for primitive equation system, which also place the limit of the horizontal resolution. Recently, non-hydrostatic models are becoming more popular. The horizontal resolution is not dynamically restricted in this type of model. However, it is important that the non-hydrostatic model has proper physical parameterizations (or explicit predictions) to cope with the high horizontal resolution. The development of many such physical parameterizations is still in process.

The basic equations and coordinate system of RSM are the same as the GSM, except that the RSM uses zonal and meridional components of momentum as dependent variables, while GSM uses divergence and vorticity. There is no fundamental difference in the choice of these variables; the choices are due to the convenience resulting from map projections used in RSM.

3. Computational detail

In order to make regional simulations, the model starts from grid point values of dependent variables over the regional domain, supplied from an existing global model (GSM) run. The model also needs global base field at specified time intervals (normally 6-hours), which is either global analysis or forecast. In the case of RSM, the global fields are given as spectral coefficients of divergence, vorticity, virtual temperature and log of surface pressure. *As a first step*, the global spectral coefficients are converted to grid point values of RSM dependent variables on the regional grid as base field.. Determination of the regional domain is described later in this report. The difference of the RSM field and base field provides grid point perturbations. *As a second step*, the grid point perturbations are transformed to sine and cosine series. Note that this process acts as spatial smoothing but also forces the perturbation field to satisfy zero and symmetric lateral boundary conditions. *The third step* is to calculate the right hand side of the dynamical part of the prediction equations, which includes values of dependent variables, their space derivatives and their multiples. The horizontal space derivatives are performed analytically using the spectral formulations, while vertical derivatives are evaluated using finite difference methods. The vertical derivative calculations are designed to conserve various quantities for vertical transport, as well as to conserve energy through conversion from potential to kinetic energy. *The fourth step* is to compute, on the left hand side of the dynamical equations, the difference between the computed tendency and the tendency of the base field at each regional grid point, which is the perturbation tendency. *In the fifth step*, the perturbation tendency is converted to sine and cosine series, thus providing the spectral representation of the tendency of perturbations. *In the sixth step*, a semi-implicit time integration scheme is performed, yielding new predicted spectral perturbation coefficients due only to the dynamical forcing. *In the seventh step*, the new coefficients are converted to regional grid point values and added with the base field to construct a full field of dependent variables forced only by the dynamics. *The eighth step* is to calculate tendencies due to additional physical processes, such as vertical diffusion, convection, radiation, boundary layer physics, hydrological processes, and others. These computations are done in 'adjustment' mode, in which all of the dependent variables are adjusted to new values by each physical process. These new grid point values are used as 'initial condition' for the next time step. Then, the above nine steps are repeated until the target forecast time is reached.

The model physics packages include short- and long-wave radiation (Chou 1992; Chou and Lee 1996; Chou and Suarez 1996) with diurnal variation and diagnostic cloud (Slingo, 1987), the Monin-Obkhov similarity theory surface layer, non-local vertical diffusion (Hong and Pan, 1996), gravity-wave drag (Alpert, 1988), Relaxed Arakawa Schubert cumulus convection (Moorthi and Suarez, 1992), shallow convection, large-scale precipitation, Oregon State University hydrological model (Pan and Mahrt, 1987), and mean smoothed topography. The physical parameterizations are executed every time step except for the radiation routine, which is computed every hour to save computer time.

The spectral representation of perturbation is a two-dimensional cosine series for pressure, divergence, temperature, and water vapor mixing ratio, and a two-dimensional

sine series for the perturbation of vorticity. In the vertical, the RSM uses the same finite-difference formulation as in the global model.

Computationally, a one dimensional Fast Fourier Transform is applied in the X-direction while a Fourier summation is performed in the Y-direction, intentionally avoiding the use of the two-dimensional Fast Fourier Transform. This approach has the advantage of reducing the computational memory requirement, and is best suited to the distributed machine architecture. This method is also consistent with the Legendre transform employed in the GSM, thus having an additional advantage of keeping the program structure as similar as possible to the global model.

4. Implementation of RSM on parallel computer

The basic strategy underlying the implementation of RSM to a parallel computer platform is to provide the flexibility to allow the same code to run on sequential, shared memory parallel as well as on distributed memory parallel machines. This is achieved by preprocessing the code before it is compiled on different machines. We also designed the system very carefully such that the results obtained on the same machine but with different numbers of processors are bit-to-bit exact, and reproducible. The single program multiple data (SPMD) programming paradigm is used for this purpose. In this paradigm, each processor performs the same computations but with different data. The adaptation of the codes to SPMD machines requires the data to be distributed onto multiple processors with a minimum amount of communications (data moves). For the spectral model, the data distribution is based on the ease of computations without communication. Looking into each step of the spectral conversions and model tendency calculations, it becomes clear that the conversion requires the entire array in one of the three dimensions (either X, Y or Z) to reside in one processor to perform computations without communication. For example, fast Fourier Transform requires all the arrays in the X-direction to reside in one processor, but arrays in Y- and Z-directions can be separated into different processors. Similarly, Fourier summation in the Y-direction requires that the entire array in Y-direction must reside in one processor, but arrays in X- and Z-directions can be in separate processors. The physical process calculations require all the variables at all levels for each grid point, so that arrays in the Z-direction need to be in the same processor but arrays in the X- and Y- directions can be separated onto different processors. This array distribution requires that entire arrays be rearranged into different configurations before the computational operations. This transpose method is named 2-Dimensional decomposition because one of the dimensions is fixed but the other two are distributed. It has been studied by many authors (e.g. Foster etc., 1997, Barros etc., 1995, Skalin and Bjorge, 1997), and has been widely used in many global spectral models. Since the RSM code structure is very similar to the GSM which uses the transpose method for parallelization, the same method was adopted for RSM parallelization.

To enable the code to run on various platforms (which is necessary to produce model simulations and conduct research under varying resource availability), the original code is preserved and the parallelization is added as a new option in the preprocessing stage. Thus the code can run efficiently on serial computer systems as well as on parallel

computer systems. This procedure also makes debugging the parallel code relatively easy by step-by-step examination of the intermediate output.   This approach limits somewhat the optimization of the code for cache based machines, since the original GSM/RSM was written for a Cray vector processor machine, and some part of the code remains vectorized, but no particular attempts have been made to take advantage of the cache based processors at this time.   In fact, this approach made it easier to optimize the code on the Earth Simulator.   Fortran 90 data structure is used to simplify the data sharing among all routines through few passing arguments.   To avoid unnecessary memory requirements, all the work arrays required for distributed memory computation are dynamically allocated first and deallocated after the computation. This is achieved by using C-preprocessing ifdef directives.

Unlike 1-D decomposition, the 2-D decomposition is flexible in the choice of number of processors.   There are strict limitations to the number of processors in the case of 1-D decomposition (such as, the number should be a multiple of the number of model levels), but the limitations for 2-D decomposition are much less restrictive, and almost any number can be chosen, in practice. As long as the number of processors is not a prime number, 2-D decomposition works, but the efficiency may be affected for multi-way node machines, where inter-node communication tends to be faster than intra-node communication. (depending on the computer architecture, "node" contains multiple-processors. For the IBM-SP machine, a node contains 4 to 8 processors). In this situation, it is better to choose the number of processors as a multiple of the number of processors in a node, but our extended test showed very little improvement in efficiency, due to the unavoidable intra-node communication in various transposes. We also note that 1-D decomposition is a special case of 2-D decomposition in our system, such that even a prime number of processors can be utilized in our computation, although the efficiency drops.

Figure 1 illustrates how the transpose method is applied to the RSM.   The computational flow starts from the left top configuration. At this stage, all the dependent variables are expressed as double sine/cosine series.   These coefficients are distributed to each processor with a configuration of values at full levels, but only a portion in Y- and X-wavenumbers (say, n and m).   In this configuration, computation in sine/cosine space that requires values at all levels is performed. An example is the semi-implicit integration after the dynamical tendency calculations (sixth step mentioned earlier). The first transpose (named NN2NK in the figure) transposes from top left configuration to full array in Y-direction but only a portion in X- and vertical directions. Since full array in Y- direction is available, it is possible to perform X- Fourier transform sum in each processor without communication during the computation (top right configuration after the Y- Fourier transform).   The next transpose, named NL2NY, is to rearrange the arrays in such a way that there is full array in X- direction but only a portion in Y- and vertical directions. This configuration allows the X- Fast Fourier transform in each processor without communication between the processors. These steps are shown as right middle, bottom right, and center bottom configurations. Note that at the center bottom configuration, all the sin/cosine coefficients were converted to grid point values.   The final step is to rearrange the arrays in such a way that a processor contains full levels but only a portion of

the X- and Y- arrays. In this configuration, physical processes, such as radiation, convection, etc. are evaluated. After the computation of these physical processes, the entire scheme needs to be reversed to obtain the sine/cosine coefficients. In this scheme, three transposes are needed to perform one forward Fourier transform. The important point here is that the transpose process does not require communication among all the processors, but only among the slices (in the case shown in Figure 1, so communication is necessary between only 3 processors). This design minimizes communication and maximizes efficiency of parallel execution.
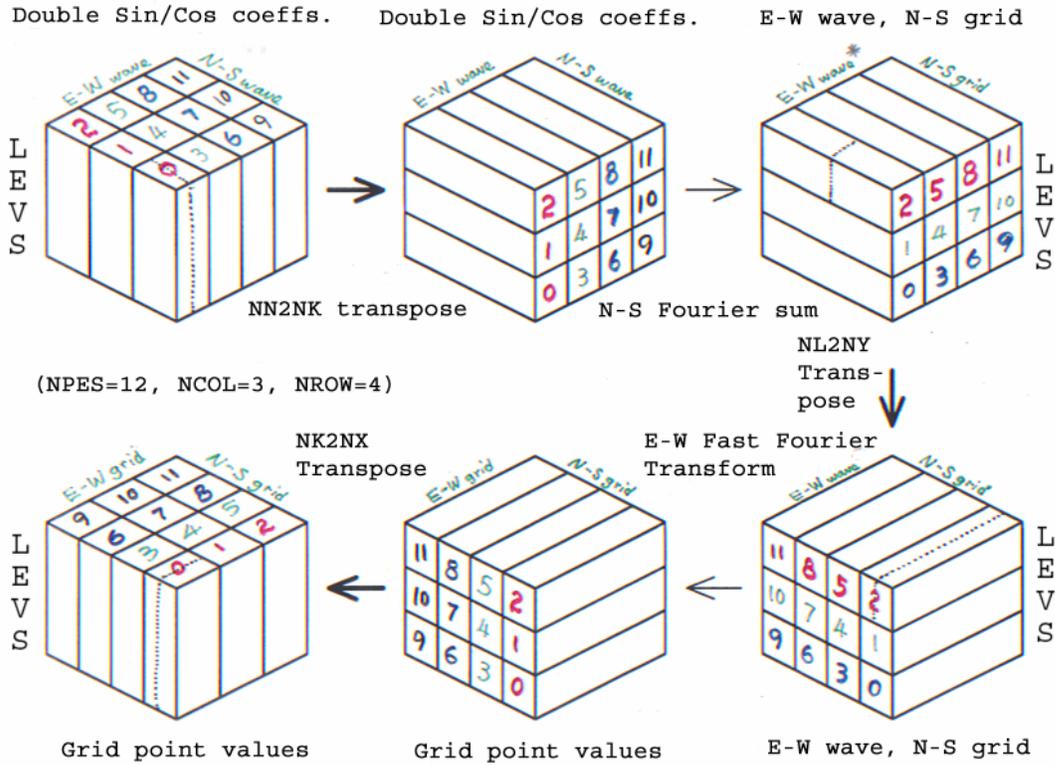
Figure 1. Diagram of transpose method used in the parallelization of RSM.

The unique part of the parallelization of the RSM code is the computation of the perturbation quantities. This computation is performed in grid point space (lower left figure in Figure 1). The coarse resolution global field over the regional sub-domain (domain covered by single processor) covering a slightly larger area than the sub-domain is distributed to each processor at the start of the time integration. This global field is spatially interpolated to regional model grid at each processor and is added to the perturbation to get the full field values in the sub-domain. Note that the conversion of global spectral coefficients to regional coarse grid is performed every nesting interval (of 6 hours). Since this computation occupies a very small portion of the regional model integration (once per nesting period), it is performed in a single processor to avoid excessive complication for parallelizing the spherical transform.

In order to further improve the efficiency of the model, two major sub-codes, lower boundary condition processing and post processing, are integrated into the forecast model.

This process allows parallelization of the sub-codes, as well as improving the efficiency by reducing the I/O, global field conversions (which is done in serial mode), and allowing the model to execute without exiting frequently for diagnostics and base field I/O.

Bit-to-bit reproducibility of the computation is crucial for debugging and maintaining codes, and performing experiments. The Fourier conversion sum is designed in such a way that the order of the computation is always maintained. This assures that the numerical results are reproducible and are exactly the same for any number of processors used. This bit-by-bit reproducibility made it possible to debug the code without extensive knowledge of the entire code, and made the process relatively easy.

5. Performance

As described above, the RSM code makes for straightforward adaptation to different platforms. So far, the present version of RSM has been successfully run on an IBM-SP, Linux cluster, Mac cluster, NEC SX-6 and the Earth Simulator machines. For purposes here, we present the performance of the model on the IBM-SP and the preliminary result on the Earth Simulator.

5.1 Performance on IBM-SP

We tested RSM performance using a horizontal domain of 128 x 85, on the San Diego Supercomputer Center's IBM SP3 and on IBM SP4 (Cui et al. 2004). The global base field has a resolution of T62 (~200km) with 28 levels, which is the same as the RSM. Although the regional model on the Earth Simulator will ultimately run over a much larger domain, this test on the IBM SP was performed on a smaller domain due to limited computer resource availability.

Blue Horizon is an IBM tera-scale machine, having 144 SMP nodes with 8 processors per node. Each SMP node has 4 gigabytes of memory shared among its eight Power3 processors. The Power3 processors are capable of executing four floating-point operations per cycle. The application processors run at 375 MHz and are capable of a peak performance of 1.5 GFLOPS.

The Data Star has 176 (8-way) P655+ compute node with 16GB of memory each. The Power4 processors are super-scalar (implying simultaneous execution of multiple instructions) pipelined 64-bit RISC chips with two Floating-point Units, 2 Fixed Point (Integer) Units, Branch Execution Unit, and a Conditional Register Unit. These processors feature out-of-order execution capabilities. They are capable of executing up to 8 instructions per clock cycle and up to four floating point operations (two fused multiply-adds) per cycle. Each Power4 CPU has a two-way L1 (32 KB) cache, and a L2 (0.75 MB) cache which is four-way set associative. There is also an 8-way L3 cache on each node (16 MB per processor). The application processors run at 1.5 GHz and are capable of a peak performance of 6.0 GFLOPS (sited from NPACI web site http://www.npaci.edu/DataStar).

As an indication of performance, we tested the model using a simulation period of 6 hours with 360 sec time step using 60km grid length. For the 128x85x28 domain, we observed efficient scaling for the range of processors up to 128. For the IBM SP3, the wall clock time, speedup factor and efficiency of the parallel code on different numbers of processors are shown in Figure 2. In Figure 2, the solid line represents the speedup factor and the dashed line reflects the efficiency on various numbers of processors. We have restricted the test to 2-D decomposition, but an additional test shows 1-D is significantly slower than 2-D (not shown). The smoothness of the curve is affected at certain points by a mismatch of the domain decomposition with the number of mesh points (Wehner etc., 1995), as well as by a transition from using 1 node (each node with 8 processors) for 1, 4, 8 processors, using 4 nodes for 16 and 32 processors, and using 16 processors for 64 and 128 processors. Note that there is a significant falloff in performance from 16 to 32 processors. The reason for the falloff was diagnosed as the result of inefficiencies in communication. Using 128 processors, the efficiency is reduced to 22%, due to lack of effective scaling by the serial part of the code in the model. However, this experiment was performed before the merging and parallelization of two sub-programs, and the latest code shows higher efficiency.
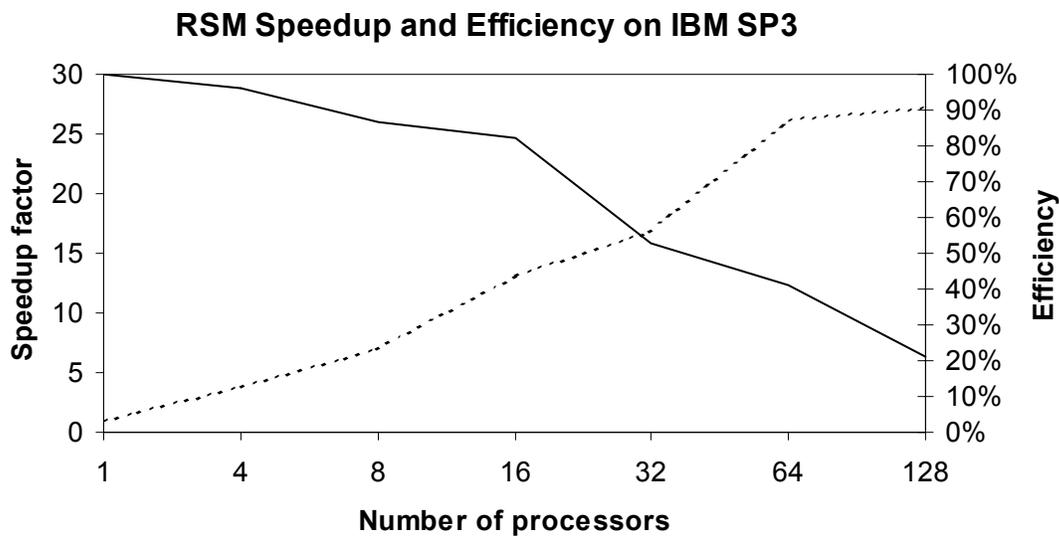
Figure 2. Speed-up and efficiency of RSM on IBM-SP machine. Domain size is 128x85.

The total performance of the IBM SP single processor is about 106 Mflops as measured by the utilities HPM (High Performance Monitor). The average performance using 64 processors is 31 Mflops per processor, which is fairly typical when compared with other applications. The per-processor performance achieved on microprocessor-based parallel computers is often disappointing: a small fraction of peak.

The measured communication overhead in parallel RSM shows 12% at the testing problem size, referring to the 16 processors run. With an increasing number of processors the load imbalance becomes noticeable. The load balance is about 0.92, referring to the 16 processor run. Here load balance is calculated as the division of execution time at a processor averaged over all processors and maximum execution time at a processor. The

load balance of 1.0 means all processors take exactly the same amount of time. In general, the dynamics computations are uniformly distributed across processors at all time steps, while the physics computations may have significant spatial and temporal variations of the computational load per grid column depending upon meteorological conditions in that column.

As indicated in previous work by Juang and Kanamitsu (1997), the performance of the MPP's is not better than that of the vector-parallel platforms. The experiments with the global T62L28 model on a Fujitsu vector machine indicated that the 1-D decomposition is faster than 2-D decomposition. The MPP computers lack direct interconnection between processors and the entire set of memory chips. These architectures require substantially more overhead than shared memory architectures to access any data not contained in that local memory. In such cases, software and hardware restrictions impede the efficiency of the machine by spending valuable time communicating data rather than calculating it. This communication overhead makes it difficult to utilize more than 128 processors for the 128 x 85 model size. For a much larger domain, the overhead becomes relatively small compared to the computation. However, limit in memory size and resources prevent us from performing such an efficiency test on the IBM SP.

5.2 Performance on Earth Simulator

The Earth Simulator machine in Yokohama, Japan is currently the world's fastest computer (www.top500.org). It consists of 640 nodes with eight vector processors each with 16-GB memory. The vector processor has 6 different types of vector pipelines, 72 vector registers and 17 mask registers, and its peak speed is 8 Gflops. In total, the machine has 5120 vector processors with 10 TB of memory and theoretical peak performance of 40Tflops (sited from http://www.es.jamstec.go.jp/esc/eng/ES/index.html).

Optimization of the RSM for the Earth Simulator machine required extensive vectorization, which is not necessary for a cache based machine (such as IBM SP). The work has been accomplished with extensive modification of the code, both for Fourier conversions, perturbation calculations and physical processes. The Earth Simulator Center placed a rather strict requirement to the efficiency of the model when utilizing their machine. The number of processors allowed for use is determined by the vectorization and parallelization ratio. With the original code, the vectorization ratio was middle 90's% and we were only permitted to use 10 nodes (80 processors) with very small domain of 54x55 (December 2003) at the start. In the spring of 2004, the parallelization reached 99.1%, but a 1-hour simulation with the target domain took 541 seconds to complete. In July-August 2004, the parallelization ratio was further increased to 99.84% with a vectorization ratio of 97.23%, and the same computation now completes in 73 seconds using 32 nodes. On the basis of improved efficiency, we obtained permission to utilize 74 nodes (592 processors). The overall efficiency of 1.9 Gflops is recorded, which is about 25% of the peak performance. With this model, one-day simulation with 10km grid over the entire continental US covering part of the Pacific and Atlantic oceans takes about 20 minutes. Further optimization is in progress in the fall of 1994, with a goal to reduce this time to less than 10 min.

6. Model system availability

Currently, users of the G-RSM are spread around the world, from mainland China, Korea, Hong Kong, Taiwan, Israel, Spain, Italy, Germany and several locations in the U.S. One of the essential considerations when providing the model to such a wide community is to make the system user-friendly. This is not an easy task, particularly considering that the model undergoes continuous development and upgrading. We accomplished this task by utilizing Concurrent Versions System (CVS), together with common Unix interfaces, such as make and c-processor.   The CVS maintains all past-version histories, and is capable of retrieving the model components of any versions, or any dates. The CVS's log function is also valuable in monitoring the development status of the modeling system. The system allows simultaneous development of the code, which significantly speeds up the development.

The G-RSM is available to the public. Detailed documentation and a user guide is available from the http://ecpc.ucsd.edu web page.   Following is a brief summary of procedures to obtain and execute the codes:

1. Make sure that your system has CVS installed. This can be checked by simply typing 'cvs' (without quotes). If it is not found in your system, you need to install CVS from http://www.cvshome.org. Detailed installation instructions are found in the web page.
2. Define environmental variable CVSROOT to:
:pserver:anoncvs@rokka.ucsd.edu:/rokka1/kana/cvs-server-root/cpscvs.
3. Type 'cvs logon'. You should get a password prompt. Simply return. There is no password for the user 'anoncvs'.
4. Type 'cvs co install'.
5. Type './install' and follow instruction.

These five simple steps will download the libraries, source codes and scripts. Compilation and configuration of scripts are also made and finally generate a simple script to make a test execution of RSM or GSM.

7. Preliminary results of downscaling

As described in this report, the computer performance of the parallelized version of the RSM has been tested extensively on several platforms.   Continuing efforts are underway to further accelerate the code on vector machines.   The model computer performance is important for making mass production of downscaling analysis, but it will not in any way assure the quality of the product itself.   Longer runs and careful monitoring, diagnostics and comparison with station observations will be carried out for this purpose. Thus far, we have performed five and a half years of downscaling integrations for a preliminary checkup of the system.

The horizontal resolution of the model is 10km with the domain size of 128 x 199, covering the State of California and surrounding ocean and land. The choice of the domain was made somewhat empirically using our past experience that the effect of the lateral

boundary disappears about 5-6 grid points from the boundary. Further testing of the size of the domain and a new method to nudge the regional field to global analysis field is in progress and will be reported as a separate report. Fig. 3 shows the precipitation climatology of the RSM downscaling runs. A clear maximum over the Sierra Nevada and a secondary peak along the coast are found in the winter precipitation. In summer, the peak that was stationed over the Sierras during winter is shifted to the east and the maxima along the coast disappear.
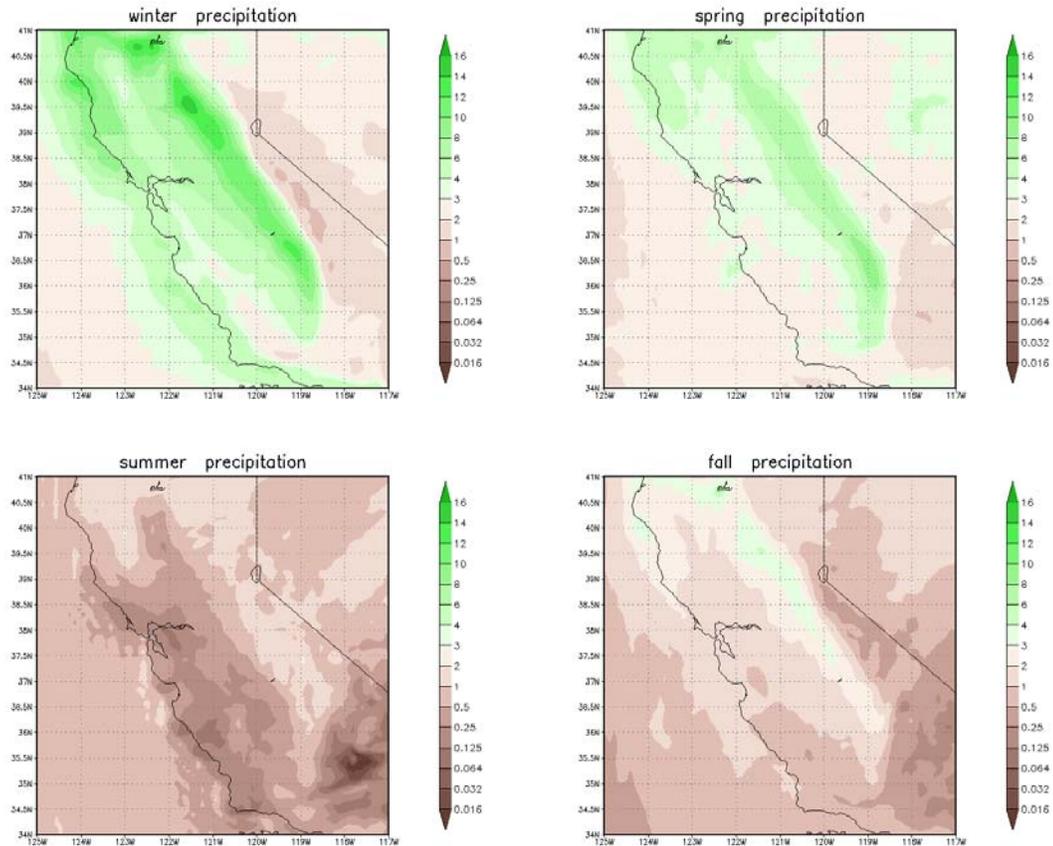
Figure 3. Seasonal precipitation climatology from 5.5 year downscaling simulations.

During winter, the two meter temperature climatology in Fig. 4 shows cold over and east of the Sierras while more uniform in the Central Valley to the coast. During summer, temperature is highest over the Central Valley and south east (Death Valley area). It is interesting to see peaks of high temperature to the north and to the south with slight minimum in between over the Central Valley.

The 10-meter wind climatology (Figure 5) shows a clear low level jet along the Sierras in winter. During the summer, northerly wind over the ocean penetrates into the Central Valley through the Bay Area.
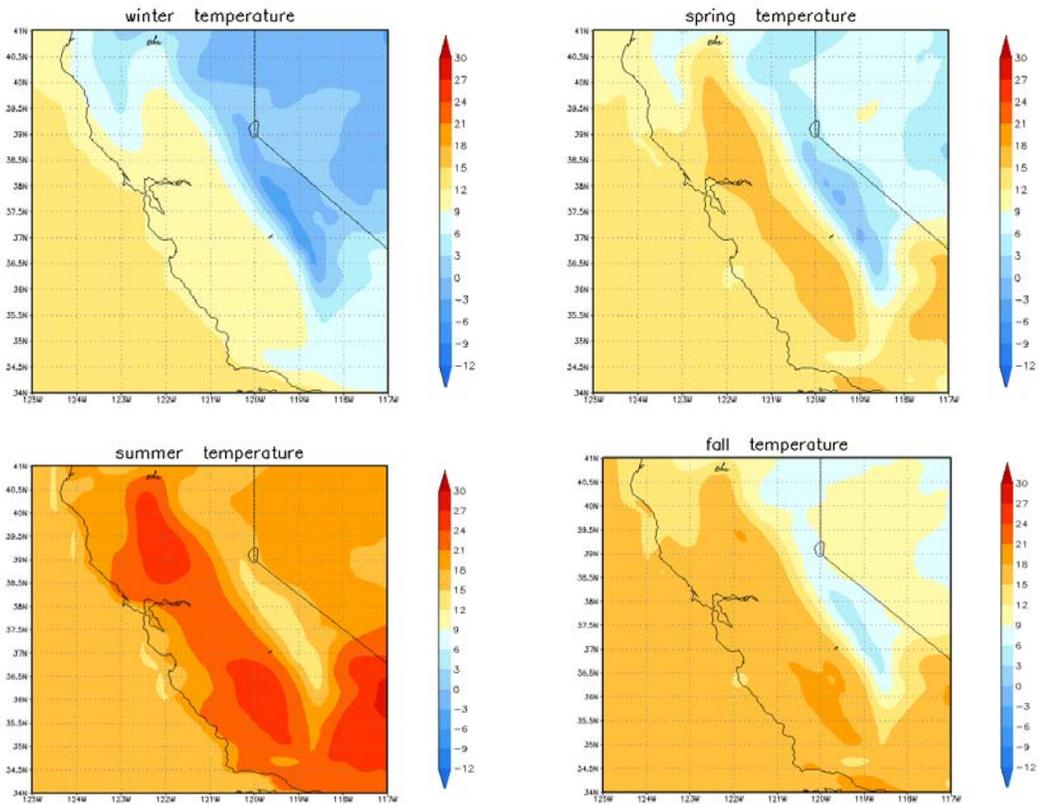
Figure 4 Seasonal climatology of 2-meter temperature obtained from 5.5 years of downscaling simulations.

The diurnal variability of temperature, wind and precipitation over coast, valley and mountain regions are examined. Here, we show only the results for temperature and precipitation during summer in Figs 6 and 7.   The temperature shows gradual cooling from midnight to morning with a much faster warming in the morning. The peak cold hour is 4 am (local time) over all the regions. The hour of maximum temperature is different for different regions, ranging from 10 am over the coast, 12 pm over the mountains and 16 pm over the valley.

A not very well described behavior in summer precipitation is indicated by the RSM climatology, which contains a marked diurnal variation of precipitation over the Sierra Nevada mountains, peaking at 15 pm local time. Over coast and valley regions, diurnal variations are much smaller, but still quite evident. Curiously, there seems to be a double peak in the precipitation at the coast at 4 am and 12pm. Over the valley, there are triple peaks at 4am, 10am and 18pm.

Although these simulations and details look realistic, they still need to be carefully verified against station observations. This effort is now in progress and will be reported in the future.
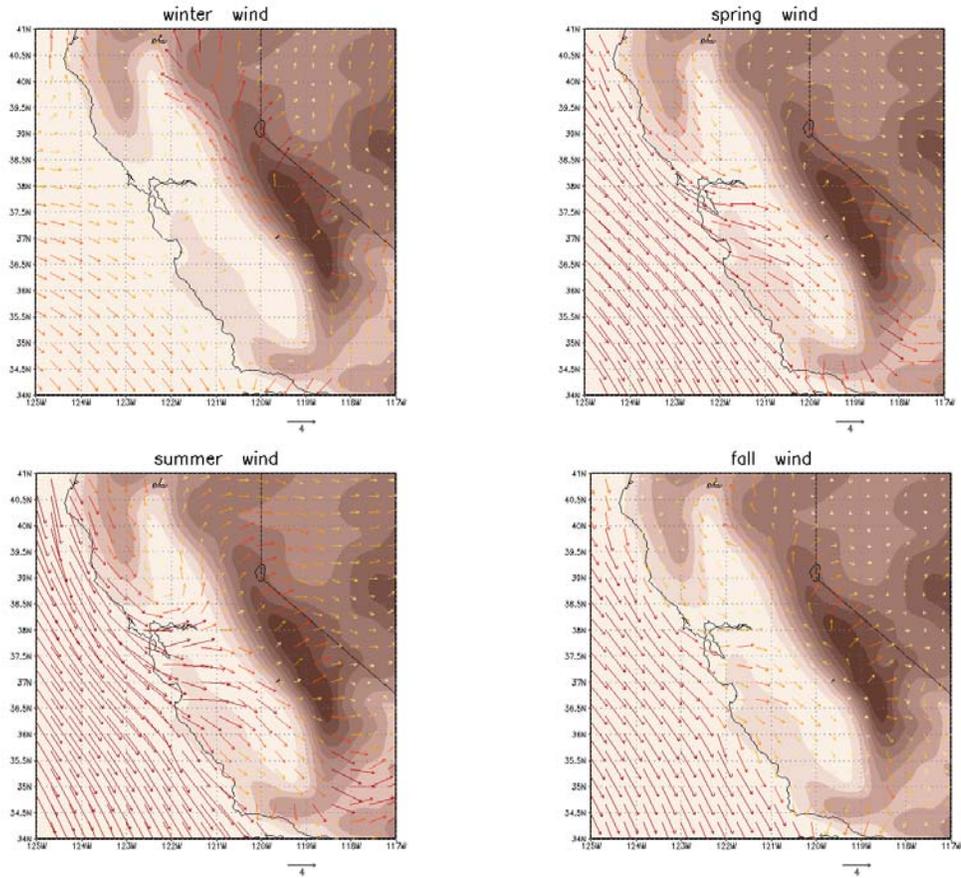
Figure 5. Seasonal climatology of 10 meter wind obtained from 5.5 years of downscaling simulations.
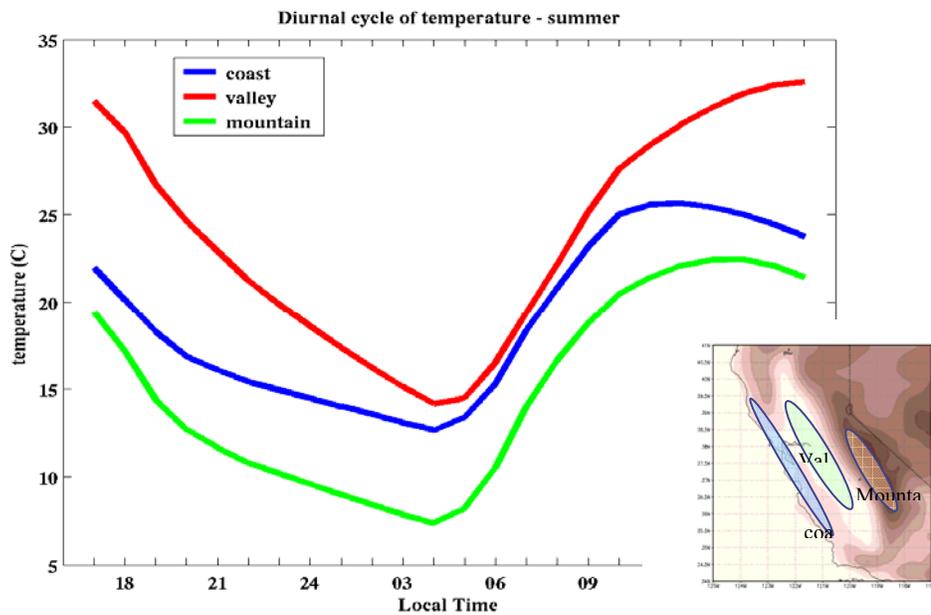
Figure 6. Climatological diurnal variation of near surface temperature during summer obtained from 5.5 years of downscaling simulations.

14
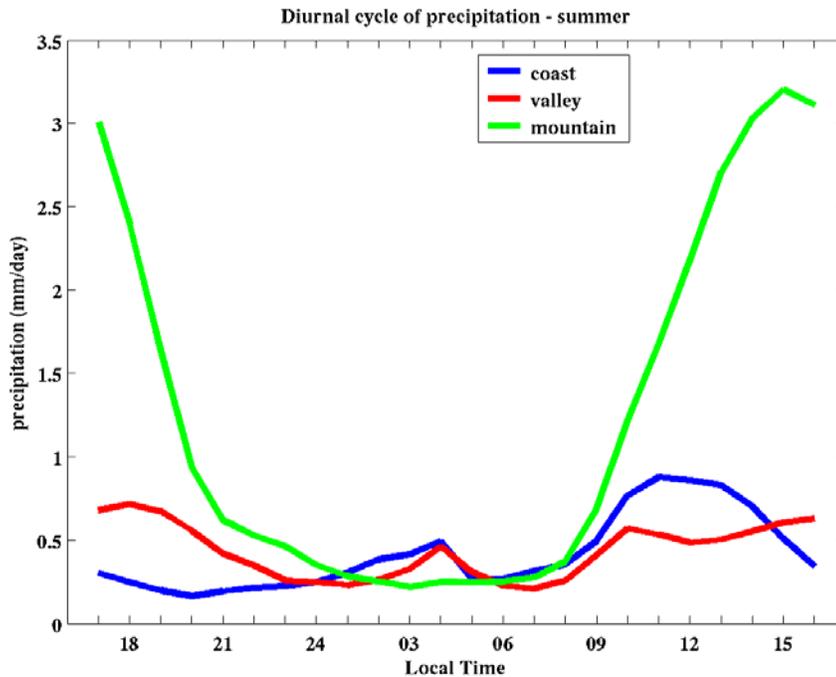
Figure 7

Climatological diurnal variation of precipitation during summer obtained from 5.5 years of downscaling simulations.

8. Concluding remarks

The Regional Spectral Model was successfully parallelized for massively parallel computers. The model code has been designed in order to be configurable for a variety of machine architectures, while still retaining its original serial model structure. This approach is very useful in debugging the parallel code, since step-by-step comparison of the computational results was possible, and made locating the source of any errors in the parallel code relatively easy. Parallelization design followed the global model, utilizing a 2-Dimensional decomposition with 3-step transpose strategy. Additional steps to compute perturbation quantities from the global base field in RSM were successfully parallelized.

The computational performance of the model on parallel computer was found to be reasonable. For the domain of 128 x 199, the code scales well up to 128 processors. For a larger domain, calculations indicated that the code should scale with more processors. On an IBM-SP machine, the performance of the model was approximately 10% of the peak performance.

The code has been optimized for the Earth Simulator machine with extensive vectorization and parallelization of peripheral codes. The latest performance of the model is 1.9GFlops, which is about 25% of the peak. Work is in progress to further accelerate the code on the Earth Simulator.

Preliminary evaluation of the 5-year 10km downscaling simulation over California yields results that appear to be very promising, but further verification is necessary.

The production of downscaling analysis is dependent on the speed of execution, the I/O, and the availability of the computer at a particular computer center where the resource is allocated. Our trial runs on IBM Datastar at the San Diego Supercomputer Center, and on Earth Simulator machines showed the following: For 1-month simulation over California domain (covering, 126.5W - 113.2W, 29.5N - 45.7N, the area larger than shown in Figure 3), IBM DataStar required 5-6 hours of computer time (wall clock time including waiting time for execution) using 64 processors and highest priority. This implies that for this domain, it will take about 4-5 months to complete the 50 year downscaling. This is a feasible and reasonable project. On the other hand, if we expand the area to cover the contiguous US that extends to both the Pacific and Atlantic Oceans, the amount of computer time will increase by a factor of 15, or 5-6 years to complete. This is apparently not a feasible option. If we use the Earth Simulator machine, our current, somewhat optimistic estimate is 7.5 hours of computer time for one-day integration of the contiguous US domain, using about 800 processors. The entire 50-years of downscaling will take about 7-10 months, which is reasonable, but the success is heavily dependent on how readily the computer time will be available on the Earth Simulator machine. We will be working further to optimize the code over the next couple of months to make the collaboration with the Earth Simulator Center a success.

References

Alpert, J.C., M. Kanamitsu, P.M. Caplan, J.G. Sela, G.H. White, and E. Kalnay, 1988: Mountain induced gravity wave drag parameterization in the NMC medium-range model. Preprints of the Eighth Conference on Numerical Weather Prediction, Baltimore, MD, American Meteorological Society, 726-733.

Barros, S. R. M., D. Dent, L. Isaksen, G. Robinson, G. Mozdynski and F. Wollenweber, 1995: The IFS model: A parallel production weather code. *Parallel Computing*, **21**, 1621-1638.

Bourke, W., 1974: A multi-level spectral model. Part I: formulation and hemispheric integration. *Mon. Wea. Rev.* **102**, 687-701.

Chou, M.-D., 1992: A solar radiation model for use in climate studies. *J. Atmos. Sci.*, **49**, 762-772.

_____, and M. J. Suarez, 1994: An efficient thermal infrared radiation parameterization for use in General Circulation Models. *Technical Report Series on Global Modeling and Data Assimilation*, National Aeronautical and Space Administration/TM-1994-104606, 3, 85 pp.

Cui, Y. F., Juang, H.-M. H., Chukkapalli, G. and M. Kanamitsu, 2004: Development and performance of a massively parallel regional spectral model. Proceeding of the High Performance Computing & Simulation Conference, June 13-16, Magdeburg, Germany, 4, 83-89.

Drake, J. and I. Foster, 1995: Introduction to the special issue on parallel computing in climate and weather modeling. *Parallel Computing*, **21**, 1539-1544.

Foster, I. and P.H. Worley, 1997: Parallel algorithms for the spectral transform method, *SIAM J. Sci. Comput*. **18(2)** , 806-837.

Gustafsoon, N, and A, McDonald, 1996: A comparison of the HIRLAM gridpoint and spectral semi-Lagrangian models. *Mon. Wea. Rev.,* **124**, 2008-2022.

Hack, J. J., J.M. Rosinski, D.L. Williamson, B.A. Boville and J.E. Truesdale, 1995: Computational design of the NCAR community climate model, *Parallel Computing*, **21**, 1545-1569.

Hong, S.-Y., H.-L. Pan, 1996: Nonlocal Boundary Layer Vertical Diffusion in a Medium-Range Forecast Model. *Mon. Wea. Rev.*, **124**, 2322-2339.

Hoyer, J. M., 1987: The ECMWF spectral limited area model. *ECMWF workshop Proc. on Techniques doe Horizontal Discretization in Numerical Weather Predition Models*. 343-359.

Juang, H.-M. H. and M. Kanamitsu, 1994: The NMC Nested Regional Spectral Model. *Monthly Weather Review,* **122**, 3-26.

Juang, H.-M. H., S.-Y. Hong and M. Kanamitsu, 1997: The NCEP Regional Spectral Model: An Update, *Bul.Amer. Met. Soc.* **78**, 2125-2143.

Juang, H.-M. H. and M. Kanamitsu, The Computational Performance of the NCEP Seasonal Forecast Model on Fujitsu VPP5000 at ECMWF. *ECMWF report.*

Kalnay, E., **M. Kanamitsu**, R. E. Kistler, W. Collins, D. Deaven, L. Gandin, M. Iredell, S. Saha, G. White, J. Woollen, Y. Zhu, M. Chelliah, W. Ebisuzaki, W. Higgins, J. Janowiak, K. C. Mo, C. Ropelewski, J. Wang, A. Leetmaa, R. Reynolds, Roy Jenne and D. Joseph, 1996: The NCEP/NCAR 40-year reanalysis project. *Bull. Amer. Met. Soc.*, **77**, 437-471.

Machenhauer, B. and J. E. Haugen, 1993: A spectral limited-area model formulation with time-dependent boundary conditions applied to shallow-water equations. *Mon. Wea. Rev.*, **121**, 2618-2630.

Moorthi, S., and M. J. Suarez, 1992: Relaxed Arakawa-Schubert: A parameterization of moist convection for general circulation models. *Mon. Wea. Rev.*, **120**, 978-1002.

Pan, H.-L. and L. Mahrt, 1987: Interaction between soil hydrology and boundary layer developments. *Boundary Layer Meteor.,* **38**, 185-202.

Segami, A., K. Kurihara, H. Nakamura, M. Ueno and T. Tatsumi, 1989:   Operational mesosale weather prediction with Japan spectral Model. *J. Meteor. Soc. Japan,* **67**, 907-923.

Skalin, R. and D. Bjorge, 1997: Implementation and performance of a parallel version of the HIRLAM limited area atmospheric model. Parallel Computing, **23**, 2161-2172.

Slingo, J.M., 1987: The development and verification of a cloud prediction model for the ECMWF model. *Quart. J. Roy. Meteor. Soc.*, **113**, 899-927.

Tasumi, Y., 1986: A spectral limited-area model with time dependent lateral boundary conditions and its application to a multi-level primitive equation model. *J. Meteor. Soc. Japan*, **64**, 6367-663.