Usability of parallel processing computers in numerical weather prediction

B. K. Basu

Computers based on vector processors have been dominating the super computing scene till the last decade. However, with the advent of restricted instruction set computers based processor, it is now possible to design parallel processing computers with large (>64) or massive (>1024) number of inexpensive processors to achieve a peak speed comparable to or exceeding that of a vector computer. The usability of such a parallel processing system depends critically on the nature of the scientific application, especially on the amount of parallelism available.

NUMERICAL weather prediction is a compute intensive problem and scientists working in the field have always been keen to take advantage of the latest developments in the computer technology. The advent of the very large scale integration (VLSI) technology has now made it possible to transfer most of the complex operations to the hardware instead of software, allowing the development of restricted instruction set computers (RISC). This along with other developments in the field has made it possible to manufacture micro-processors of individual peak computing speed exceeding 100 Mflops at a cost much less than that of a shared memory vector processor. Because of the low expenses involved, it became possible to add numerous processors to a machine and try to extract more computing power than machines based on a few vector processors. This is possible by using parallel processing method in which different parts of the application code are computed at the same time by different processors working in parallel. However, the low cost technology has certain disadvantages associated with it and the RISC-based computers do not have a quick access to its memory due to limited memory access bandwidth. The cache memory, used to enhance the speed of memory access, is not very effective due to its small amount. Inter processor communication, required to exchange data between distributed memory resident on different processors, also adds to the wall-clock time required to solve an application program. The success of the low cost parallel processing computers depends on the proper exploitation of the parallelism in the application code by developing a suitable strategy for the numerical computations.

The leading operational centres along with the research centres working on numerical weather prediction are already experimenting with various parallel processing system (PPS) computers to develop suitable strategies[1–5] for their application codes. Since the prediction model takes most of the wall-clock time in an operational centre at present, this code has been most frequently used for porting and experimentation on PPS machines. Again as the global spectral model is the most common formulation used in the operational centres, most of the experience gained on implementation of weather models in PPS computers is in respect of this type of models.

In India several centres, working on various aspects of computation, have designed and successfully fabricated PPS machines. Some of these machines have produced sustained speed exceeding 100 Mflops for specific application codes. In November 1992, several of these centres were invited to test the suitability of the NCMRWF weather prediction codes for parallelization. Four of these centres accepted the invitation and have experimented with the porting of and parallelization strategies[6–9] for the global prediction model of NCMRWF between end of 1993 to early 1996. All numbers mentioned in this work refer to the status as on 31 March 1996. It is understood that some of the centres have since then upgraded their machines with more powerful processors and higher bandwidth communication and have recently obtained timings which are comparable and even superior to the operational timing of NCMRWF (15 minutes for 1-day forecast by the model) as on 31 March 1996. These results are not mentioned here as the model output could not be verified at NCMRWF.

Global spectral model – Inherent parallelism

The global spectral model is based on the assumption that a finite number of orthogonal functions can represent any arbitrary continuous field to a reasonable accuracy and can be used to derive analytically the derivatives of the approximate field exactly. Since most of the inaccuracies in numerically solving the nonlinear set of differential equations, which govern the atmos-

---

B. K. Basu is in the National Centre for Medium Range Weather Forecasting, Mausam Bhavan Complex, Lodi Road, New Delhi 110 003, India.

pheric circulation, arise from the computation of derivatives, the spectral method is expected to provide superior predictions from the same initial field. The nonlinear terms are, however, not easy to compute in the spectral (functional) space and a transform method is used to convert the spectral quantities to the grid point values in the physical space. After computation, the nonlinear terms are transformed back to the spectral space.

Since the geometry of the earth is spherical, the natural choice for the orthogonal set of functions for expansion of the global fields is the spherical harmonics. These functions depend on two parameters, n and m, and are analytic functions of latitude (ϕ) and longitude (λ).

$$Y_n^m(\phi,\lambda) = P_n^m(\phi)\exp(im\lambda). \tag{1}$$

The associated Legendre polynomials are orthogonal to each other for the same value of zonal wave number m and different total wave number n.

$$\int_0^\pi P_n^m(\phi)P_{n'}^m(\phi)\sin\phi\,d\phi = \delta_{nn'}. \tag{2}$$

Any global field can be expressed in terms of the spherical harmonics as

$$F(\phi,\lambda,t) = \sum_m\sum_n f(m,n,t)P_n^m(\phi)\exp(im\lambda). \tag{3}$$

The coefficients of expansion, called the spectral coefficients, can be evaluated from the known values of the function in the physical space through Fourier and Legendre transforms:

$$f(m,n,t) = \int_0^\pi F^m(\phi,t)P_n^m(\phi)\sin\phi\,d\phi, \tag{4}$$

where, the Fourier coefficients are defined as

$$\begin{aligned} F^m(\phi,t) &= \frac{1}{2\pi}\int_0^{2\pi} F(\phi,\lambda,t)\exp(-im\lambda)d\lambda \\ &= \sum_n f(m,n,t)P_n^m(\phi). \end{aligned} \tag{5}$$

The present day operational models are multilayered and one set of spectral coefficients are required for each dependent variable at each layer.

In a spectral model, computations are done in three distinct spaces, namely, the spectral space, the physical space and the half spectral space of Fourier coefficients. In the NCMRWF model, Fourier coefficients for only one latitude are computed at a time to save memory and after completion of all computations in the Fourier space, resultant coefficients are converted into grid point values through the inverse of the discrete fast Fourier transform (DFFT). After completion of all computations in the physical space, the prognostic variables are transformed back to the Fourier space by direct DFFT and then to the spectral space by direct Legendre transform (Figure 1). Thus the present spectral model is designed to hold in memory the complete set of spectral coefficients but only one latitude of Fourier coefficients or grid point values. This technique of saving memory has implication in evolving a suitable strategy for parallelization of the model code.

All weather models have inherent parallelism embedded in their structure as atmosphere itself evolves in parallel. In a spectral model, the easiest and most obvi-

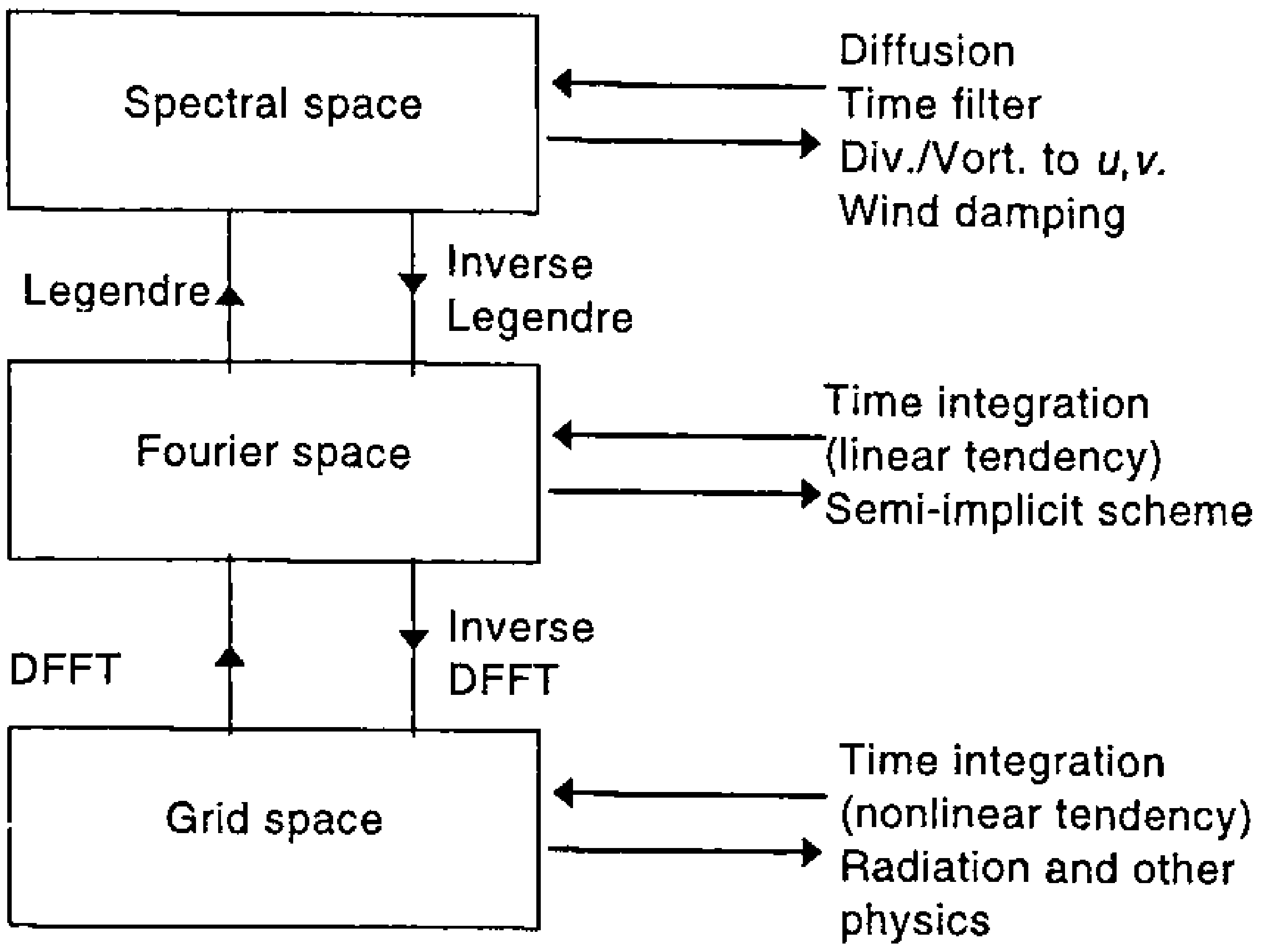


Figure 1. Distribution of model forecast computations in different spaces.

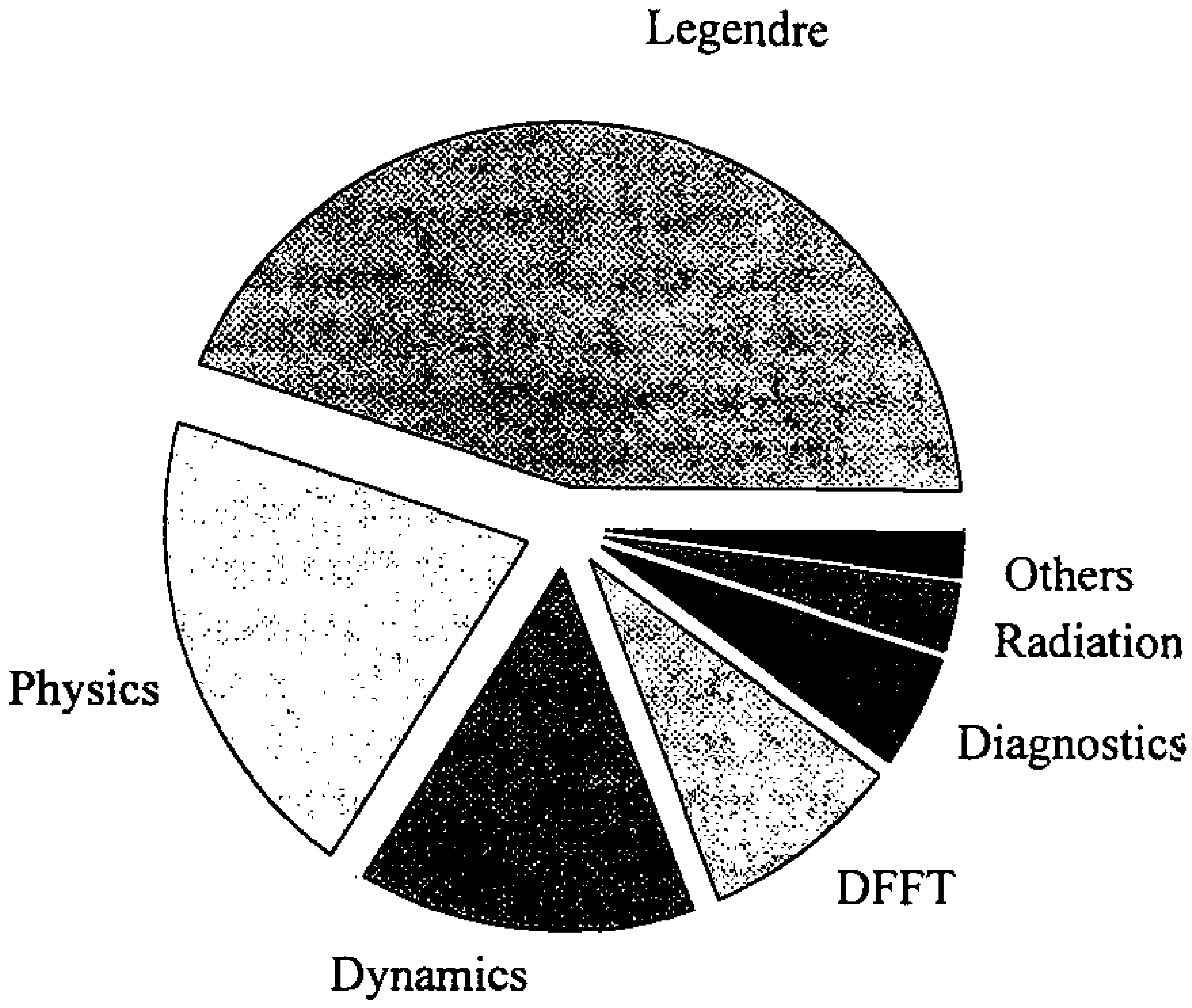


Figure 2. Distribution of computation load between various components of the NCMRWF operation (T80L18) model on CRAY-XMP/216 machine with one call to radiation in 12 h. Legendre transform = 45.0%; Physics = 21.0%; Dynamics = 15.0%; DFFT = 9.0%; Diagnostics = 5.0%; Radiation = 3.0%; Others = 2.0%.

ous strategy of parallelization is to split the global domain into sets of latitudes and carry out computation for one set of latitudes in one processor of the parallel machine. In this strategy of 1-D domain decomposition, the application code requires least modification but there is no saving in memory as each of the processors must have the same memory as the sequential machine itself.

In spectral models, a major amount of computation time (Figure 2) is spent in the computation of the Fourier and Legendre transforms. For each time step of integration, the set of direct and inverse transforms are to be carried out at least once and in some cases (like the NCEP model) twice – once each for dynamics and physics part of the code. Hence, it is desirable that the complete data to carry out the transforms are available in the same part of the memory and no communication is required. In a shared memory multitasking system, data for both the transforms are available at a central location and no communication overhead is accrued. In a distributed memory parallel processing computer, data resident in one processor are not readily available to others and computation with data from several processors becomes costly timewise. In the latitudewise parallelization strategy, the data relevant to one latitude are always resident in the memory and the Fourier transform does not require any inter-processor communication. The Legendre transforms, however, will require data which are resident in the memory of different processors. In the longitudewise parallelization, data for one longitude are available in one processor while data belonging to same latitude have to be collected from various processors. In this strategy the memory requirement is about half of that in the latitudewise parallelization (as the number of latitudes in a spectral model is about half the number of longitudes) but the code requires some rewriting as the latitude loops have to be replaced by longitude loops. In the longitudewise parallelization, the Legendre transforms become inexpensive as the data for the same are resident in the memory of a single processor, while the Fourier transforms are costly as data have to be collected from different processors. From this point of view, the best strategy is to distribute data for one layer to one of the processors such that complete input for both Legendre and Fourier transforms are available at the same memory location and communication overhead is minimized. However, since the number of vertical layers in a global model is rather small (usually less than 40), load balancing may become a serious problem. In addition, this strategy is not suitable for the computations involving physical processes which require data along a vertical column. A compromise between the requirement of minimal inter-processor communication and load balancing on massively parallel processing systems is the transposition strategy in which the data ordering is transposed in the memory so that data required for both the Fourier and Legendre transforms are available to the same processor at the time when the transforms are to be computed.

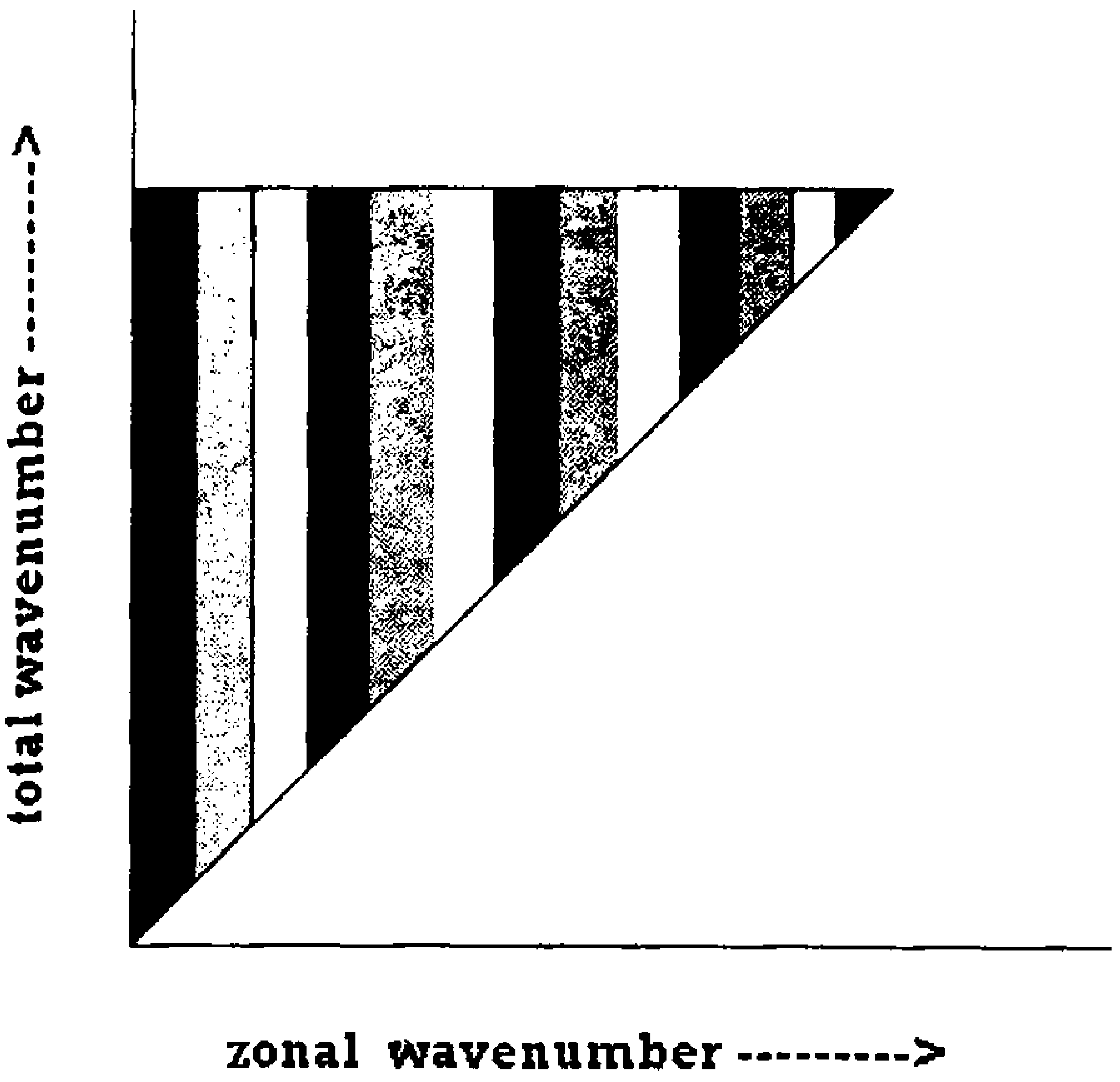


Figure 3. Distribution of computations in the spectral space as per zonal wavenumber for a spectral model with triangular truncation.

All of the above strategies are, however, coarse-grained ones and also do not consider parallelization of computations in the spectral space. The simplest way to parallelize in the spectral space is to distribute computations for different values of zonal wave number (or total wave number) to different processors. The triangular truncation method used in most of the operational spectral prediction models (Figure 3) is not really suitable for parallelization as load balancing between several processors becomes difficult due to variable data length for different wave numbers. However, since each function in the spectral expansion series is orthogonal to the other, it may be possible to distribute data successively to available processors and carry out the computations in the single instruction multiple data (SIMD) mode.

Speed up limitations in parallel processing computer

It is well-known from Amdahl's law that the wall-clock time taken to complete a user application does not reduce in proportion to the number of processors used to do the computations but reaches a theoretical limit depending on the fraction of the application code that can be parallelized. For a n-processor machine the speed up factor over a single processor can be defined as the ratio of the time taken by the single processor to that by n-

processors for solving the same application problem. Following Amdahl, this can be expressed as

$$S_n = \frac{1}{1-\alpha(n-1)/n}. \tag{6}$$

This sets a theoretical limit of speed up of any application code to

$$S_\infty = \frac{1}{1-\alpha}. \tag{7}$$

Hence it is obvious that if a parallel machine is to be designed to achieve a sustained speed of X Mflops, then each processing element must have a sustained speed exceeding $X(1-\alpha)$ Mflops. Thus the minimum sustained speed of each processor required to produce a targeted sustained speed depends also on the parallelizability characteristics of the application code. For a targeted sustained speed of 100 Mflops, the minimum sustained speed per processor for different values of parallelizability (α) is listed in Table 1.

It is evident from the table that for application codes with parallelizability less than 95%, it may be advisable to use few processors of relatively high sustained speed. A massively parallel computer with numerous (>256) low speed processors will be efficient only for application codes with $\alpha \geq 0.95$. It may be necessary to rewrite old application codes partly or fully to take full advantage of the massively parallel machines.

In the above estimates, the communication time required to distribute the initial data set to different processors and to collect back the computational result are not included. This, however, is a major issue in parallel processing computers which normally have low interprocessor communication speed. The communication overhead depends very much on the nature of the computational scheme used in solving the application problem. When a scheme can be adopted such that the memory requirement for each processor becomes small and computations require only variables available in the cache memory resident in the processor, a sustained speed exceeding 50% of the peak speed of the processors may be achieved. However, for applications with large memory requirement, the sustained to peak speed ratio will decrease by a large amount. If the interprocessor communication time is assumed to be independent of the number of processors used (as in the latitude-wise strategy) then the speed up of a n-processor machine can be expressed in the modified form as

$$S_n = \frac{1+\beta}{1-\alpha(n-1)/n+\beta}, \tag{8}$$

where β is fraction of one processor wall-clock time taken to complete the job. Such constant communications overhead essentially imply a reduction in the parallelizability (α) of the code by a factor $(1+\beta)$, which in turn imply loss of scalability.

Table 1. Sustained speed for processors

	Number of processors				
α	$n = 4$	$n = 8$	$n = 16$	$n = 32$	$n \to \infty$
0.85	36.3	25.6	20.3	17.7	15.0
0.90	32.5	21.3	15.6	12.8	10.0
0.95	28.8	16.9	10.9	8.0	5.0

Minimum sustained speed per processor for target machine speed of 100 Mflops.

Indian experience of using PPS for numerical weather prediction modelling

The four centres in India, who participated in the implementation of the different components of NCMRWF operational weather prediction suite on their PPS machines, are listed in Table 2 along with some of the important experiences gained during the exercise. The aim of this project, which was financed by DST and DoT and technical expertise for which was provided by the NCMRWF, was to demonstrate amongst the scientific community that the PPS systems developed in India are capable of handling large application code with reasonable ease and also to benchmark the different PPS machines by running the same application code (spectral forecast model at T80 resolution) with the same input

Table 2. Indian experience of NWP on PPS

	C-DoT	NAL/IISc	C-DAC/IIT(D)	BARC
No. of processors	128	8	1 + 13	1 + 8
Type of processors	Transputer	Intel i860	Intel i860	Intel i860
Brand name	Chipps-192	Flosolver Mark-3	Param-8600	Anupam
Peak Mflops	192	640	924	720
Sustained Mflops	10.5	36.0	42.5	25.0
Length of integration	One day	Five days	Five days	Five days
Man month invested	27	36	84	20
Parallelization by	Data-parallel implementation	SPMD	SPMD	SPMD

Sustained speed computed by comparison with time (15 minutes) taken by the forecast code on CRAY-XMP/216 computer for which a sustained speed of 150 Mflops is indicated by the performance monitoring tool.

SPMD, Single program multiple data.

Status as on 31 March 1996.

data provided by a common agency. An extensive evaluation of the scientific accuracy of the output from the PPS machines, against that from the CRAY, was carried out by a group of experts to determine the extent to which parallelization can lead to difference in output. The other issues which had been explored during this exercise are the scalability of the PPS machines developed by the participating centres and the developmental/parallel tools available with each system. The latter is an important feature in selection of computing system as users would like to spend only minimum time for porting of available software from sequential to parallel machines. The experiences gained in this exercise, spanning over a period of about three years, can be listed as follows:

Reproducibility. Even a simple application program, run in a sequential mode, may not reproduce results exactly when the computing system is changed. This may be due to the differences in the compilers or in mathematical library functions. In a parallel implementation, the non-associativity of mathematical operations on computers (due to truncated binary representation of floating point numbers)' add another source of error. However, at the cost of additional wall-clock time, the order of operations in the parallel implementation can be ensured to be exactly the same as that in the sequential mode. A small but definite difference between the CRAY output and the PPS output is still noticed. Even a double precision run of the whole of the forecast model code failed to reduce this difference to less than half of its original value for the selected double precision run. The differences are similar in magnitude and geographical location for all the PPS machines and even for the sequential forecast on VAX front end used at NCMRWF. Global maps of the differences between the CRAY and VAX runs, noticed after five-day forecast integration, are presented in Figure 4. It can be seen that there are only minor differences between the two forecasts over most of the globe and the significant differences are confirmed to a small area close to the edges of high topography where large gradients in meteorological fields occur. The exact origin of these differences is yet to be determined, but the possibility of the application code taking a different branching (depending on the different outcome of a logical 'if' statement in the two types of computers) in one of the physical parameterization routines is being explored.

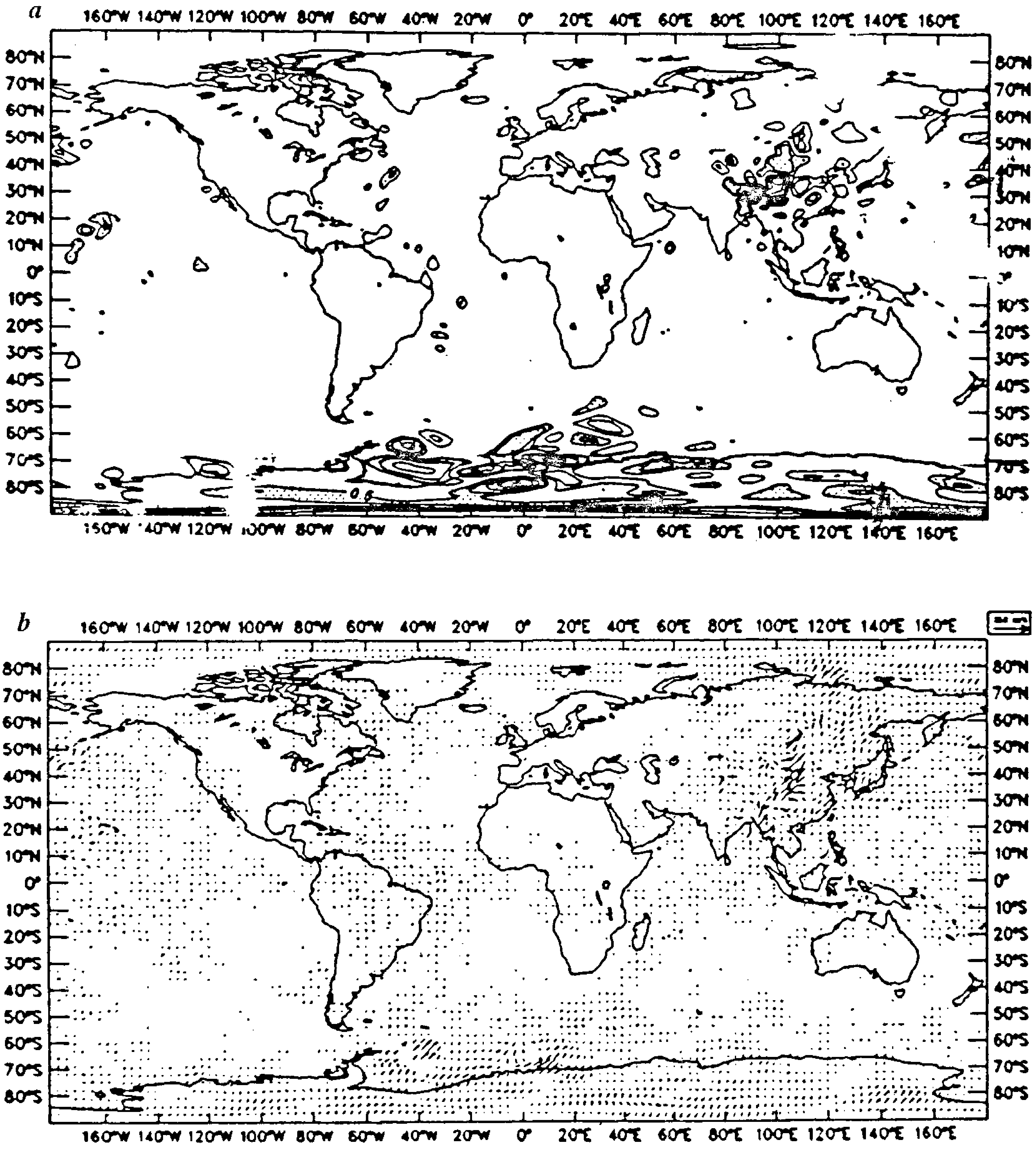
It may be mentioned here that the PPS computations referred above are IEEE complaint unlike the CRAY one. Recent experiences in porting of NCMRWF model code to the SUN-ULTRA-1 at SAC, Ahmedabad and the IBM machine at IISc, Bangalore indicate that the problem may be more related to the initial condition than the machine differences.

Speed. Most of the PPS machines used in this project were based on various versions of the Intel i860 processors except the one developed by C-DoT which used the transputers. The peak speeds of the machines and their sustained speed, for the NCMRWF forecast model code, are listed in Table 2. It appears that the machines, based on the slow processors and narrow communication bandwidth, as available in India in 1994, could not deliver the computing power required to meet the then operational schedule of NCMRWF. Since the computing requirements of NWP centres increase by a factor of about two in every three years, sustained effort will be required by PPS groups to keep pace with the growth of user demand.

From the extensive international experience of porting of many global spectral weather forecast models to a number of PPS platforms, a consensus for the value of 10% for the sustained to peak speed ratio seem to be emerging for this application. It has been suggested that this value may be increased to 15% by extensive rewriting of the application code to take advantage of the additional parallelization present in it. This, however, will require substantial investment of manpower. The experience of parallelizing the global spectral forecast model operational at NCMRWF showed that the PPS computers designed and fabricated in India during 1994 could attain a sustained to peak performance ratio close to 6%. Since this value is significantly less than the internationally accepted figure, it is possible that the basic design of the processor boards used in the machines was not suitable for the spectral forecast model.

The two processors of the CRAY-XMP/216 machine at NCMRWF have a 8.5 ns clock and two arithmetic units for each processor which can carry out addition/subtraction and multiplication/reciprocation operations independently. This implies a peak theoretical speed of 470 Mflops. Thus, for the present forecast model, the ratio of the sustanied (150 Mflops) to peak speed of the CRAY works out to be more than 30% for the two-processor machine at NCMRWF.

Scalability. The demand for enhanced computing power for NWP work is a continuous phenomenon and one or more upgradations of a computer system, during its normal life span of five to seven years, is common at operational centres. Since operations cannot be interrupted, the usual method of enhancement of computing power is by adding extra processors at site. The additional power will be useful only if the application code has almost linear speedup with the increase in the number of processors. To ensure scalability of an application code is not a trivial task even for multitasking, shared memory, vector processing computer. Distribution of data and optimization of inter-processor communication make it even more difficult[10] for a distributed memory PPS machine. Internationally, the spectral model has



gure 4. *a*, Difference between VAX and CRAY run (850 hPa temperature), *b*, Difference between VAX and CRAY run (200 hPa wind).

been installed on various PPS computers and the code has been found to scale with a number of processors reasonably well up to more than thousand processors. The Indian machines, however, have not demonstrated scalability clearly and some more effort in this direction is required. The experience at the European Centre for Medium Range Forecasting (ECMWF) has shown that considerable manpower investment is required to rewrite the model code extensively to make it scalable. In fact the availability of such a code was one of the facto behind ECMWF's decision to move to a distribut memory system. It should, however, be mentioned he that the inter-processor communication rate of Indi machines was of the order of 20 Mb per second for th project (up to 31 March 1996) while international m chines already used communication speed of the ord of 1.2 Gb per second. Improvement in inter-process communication speed may remove the hardware restri tions on scalability and the only constraint will be the scalar component of the application code.

User friendliness. Most of the present day atmospheric model codes were developed for sequential scalar computers. Migration of programs written for scalar computers to a multi-processor shared memory computer is a relatively easy task as the issues involved are simple and frequently utility programs are available which can perform most of the code conversion. Similar utility programs to help the user to optimize his code for distributed memory PPS machine need to be developed. Also, it is required to develop parallel libraries for various mathematical operations. For example, availability of suitable parallel Legendre Transform or FFT may become a crucial factor in deciding in favour of 1-D static domain decomposition along the latitude or the longitude. It is also necessary to prepare extensive documentation and training material to help the user to pick up the techniques of parallel programming with ease.

The PPS computers developed in India have been able to demonstrate their ability to handle the computations of weather prediction models but the wall-clock time was behind the operational requirement at the end of the experiments. In 1996 most of the operational NWP centres had sustained computing power of about 1 Gflops for their application codes and more than one Gb of memory. This requirement is likely to double every three years and any new developments in computing have to deal with this rate of expansion to be attractive to the NWP community. Another very important requirement of an operational NWP centre is the round the clock availability of its computer system. Normally a minimum uptime of 95%, guaranteed by a maintenance agency, is expected by the operational centres.

Table 3. International experience of parallelizing atmospheric forecast models

Name of the centre	Type of the model	Machine used	Number of processors
ECMWF, Europe	Spectral T213L31	T3D, Fujitsu VPP500, IBM RS/6000, SP1, SP2, Intel i860/Paragon, Meiko-CS2, TMC CM-5	1024, 16
EMC, USA	Spectral, T84L32, T170L32	TMC CM-2, CM-5, CRAY-3D	256, 512, 1024
DWD, Germany	Grid point		
NMI, Norway	Grid point HIRLAM 12.5 km, L16	Intel Paragon	
SMHI, Sweden	Spectral HIRLAM	MasPar	16384
NCAR, USA	Spectral CCM2	CM-5	32, 64, 128, 256 & 512
	T42L18,	SP2	32, 64, 128
	T170L18	Intel Paragon	32 to 1024
UCLA, USA	Grid point 2° × 2.5°, L9	IMB-SP2, Meico-CS2	160 (SP2), 256 (CS2)
GFDL, USA	Grid point 1.2° × 1°, L40	CM-5	256, 512

T, Triangular truncation; L, Layer; HIRLAM, High Resolution Limited Area Model; CCM2, Community Climate Model (cycle 2); TMC, Thinking Machines Corporation.

International experience

Internationally, porting of weather and climate prediction models to PPS machines is being carried out for more than ten years now and experiences gained in many such efforts were reported at the workshops organized by the ECMWF. A description of some of the models that have been ported to various PPS machines is listed in Table 3. At the moment the status is that some of the model codes (e.g. those of ECMWF and NCAR) are already suitably modified to run on many PPS machines with scalability up to thousands of processors. Tailor-made codes for atmospheric circulation models (e.g. SKYHI of GFDL) have been developed which are approximately 99.9% parallelizable. The manpower invested by ECMWF in parallelization of their operational Integrated Forecast Suite (IFS) has helped them to switch to a distributed memory Fujitsu VPP500 vector parallel processor machine. Inspite of such successes and continued investment in the development of parallel processing, the opinion of the numerical weather forecasting centres as expressed through the WMO remains that of cautious optimism. In the recent statement on computing requirements for 3-D atmospheric models[11], it has been stated that '...model algorithms and codes contain deeply-rooted assumptions and structures that were not favourable to distributed memory approach. ...performance depends on sustainable (as opposed to peak) processor performance, on the effective bandwidth for communication between processors and on the speed of the input/output...'

None of the operational weather prediction centres has so far opted for loosely coupled massively parallel computers based on inexpensive scalar processors. Out of the leading NWP centres only DWD, Germany has reported computational speed obtained, for their regional model, on a moderately parallel computer to be comparable to their vector processing supercomputer. The ECMWF has recently (from 1 October 1996) switched to a 45-processor VPP500 parallel machine only after extensive parallelization efforts spanning past 12 years.

The complete numerical weather prediction system

The forecast model is only one component of the complete NWP system. Other major components are shown in Figure 5. For a PPS machine to be accepted as the main computing system of an operational NWP centre, it is required that each of the components can be completed within a reasonable time interval such that the final product can be disseminated within the predetermined time schedule.

The NWP procedure begins with decoding of *in situ* or remote-sensed observations collected globally through the WMO communication network. Since the data collected from different platforms like surface observatories, balloon-borne instruments, satellites, etc. are coded differently, each type of observation can be directed to one set of processors. Though this may lead to severe load imbalance, specially since the daily volume of data in each type are highly variable, the operational schedule may not be much affected as data reception and decoding is normally a continuous process up to an operational cutoff time.

Flow chart of a data assimilation and forecast system

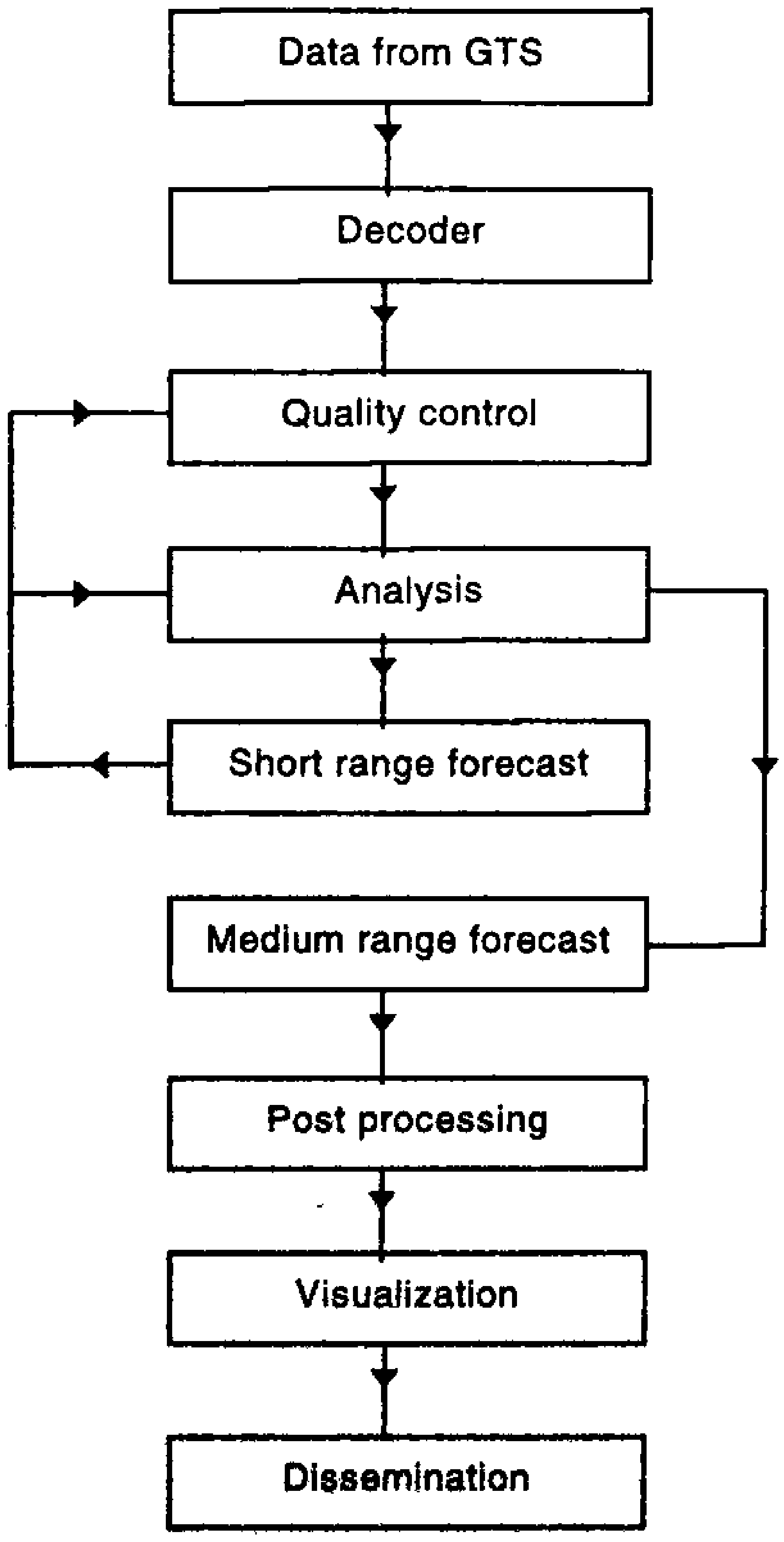


Figure 5. Components of a typical operational NWP system.

The quality control procedure for decoded observations cannot be done fully in parallel as atleast the horizontal consistency check will require neighbouring observations at the same vertical level and quality of one observation may influence that of succeeding ones through the shape of the local analysis against which the quality of observation is checked. Still parts of the procedure, like the hydrostatic check which require data from the same upper air observation, can be completed in continuation with decoding and before the data cutoff time. With judicious planning it should be possible to complete the quality control of data within the constraint of operational schedule.

The analysis of observations is the process of interpolating information to the specific 3-D grid of the forecast model. Most of the analyses procedures, operational at present, involve inversion of a large dimensional matrix which represent the cost function to be minimized. Since the matrix is sparse in nature, it may be possible to parallelize the computations involved to determine its inverse. For the NCMRWF operational analysis code (spectral statistical interpolation), the possibility of parallelization was examined by the group at BARC who ported the code to their parallel computer and ran it on a 9-processor machine by using static 1-D domain decomposition in the vertical. Though it could be ascertained that the standard methods of parallelization of sparse matrix inversion can be applied in this case, actual implementation was not attempted as large investment of manpower was estimated. This confirms the requirement of extensive preparations before the operational NWP work can be transferred to PPS machines.

The post-processing of model forecast is a relatively inexpensive component of the operational system and visualization can be distributed to multiple graphic workstations to meet the demands of operational time schedule. Thus, in principle, there is no insurmountable difficulty in porting the complete operational NWP work to a parallel processing computer. However, in practice, this is a formidable task which has not been attempted so far by any of their operational centres – even those who have spent considerable resources on parallelization of their codes. Much more experimentation and development of user-friendly tools are required before such a goal can be realized.

Concluding remarks

Parallel computing in numerical weather prediction problems has attracted considerable interest for more than a decade now. Several forecast models have been ported to various distributed memory parallel machines and suitable strategies for optimizing the computation of Legendre Transform (most expensive in a spectral formulation of the forecast model) and FFT has been developed. Even parallel versions (e.g. the community climate model PCCM2) have been developed for implementation on distributed memory computers to get scalability. However, till the end of 1996, no operational centre had adopted a massively parallel machine with distributed memory as their main computing system. The trend appears to be multiprocessor computers with few (not more than 64) individually powerful (order of Gflops) processors. Even for such a computer, all processors may not be utilized for a single job but may be used to run a set of similar jobs like the ensemble of forecasts. Recently the UK Met Office has acquired a CRAY-T3E system with 880 processing elements (PE) for their operational work. This machine has a shared memory simulator which makes the distribution of memory transparent to the user. At present the weather forecast and climate versions of the UKMO global model are run using 36 and 72 PE respectively while a new high resolution version is expected to use 144 PE. Several forecasts using slightly different initial conditions are run in parallel to utilize the full set of 840 PE available to the user.

The weather prediction codes have been able to utilize only 10% of the peak power of the individual processors in a parallel machine. This small value for the sustained to peak performance achieved on a single processor is mainly due to the smallness of the cache for memory access, but also depends on the ability of the compiler to optimize memory access and computations. The speed in Mflops for a single processor is also influenced by the communication costs and load imbalances. Both of these tend to increase, relative to the computation cost, as the number of processors increase.

The Indian efforts have been able to produce a sustained to peak speed ratio for computations close to 6%. This indicates that more work needs to be done on the parallelization strategy and also possibly towards the hardware design regarding the cache utilization. The scalability has also suffered due to low inter-processor communication rate. Compared to international standard of 1 Gb per second, the Indian machines participating in the DST project used values of the order of 20 Mb per second.

Regarding the complete NWP work on distributed memory system, even internationally the experience is not sufficient to arrive at a firm conclusion. At the moment, the share of the components other than forecast model, is only a small part of the total operational time. Even after the adoption of the 4-D variational procedure, which appears to be the future trend in assimilation of data, most of the operational time will be taken up by the model integration. The data decoding, quality control, post-processing, visualization, etc. are not expensive time wise and may be suitable for distributed computing on multiple workstations.

As stated earlier, the meteorological community as a whole, has taken a cautious approach towards the parallel processing computers. This is because of the facts that a) many centres have not yet tested their models on any distributed memory machine and lack in experience, b) the performance of an application code can vary largely with the parallelization strategy and hence may require large manpower investment for optimization, and c) the distributed memory MPP machines are yet to firmly demonstrate the advantage in cost of unit sustained computing power over the vector processing machines.

Thus the distributed memory PPS machines, specially the MPP ones, may still have to wait for further technological improvement such that the benefit in the unit cost of computing can more than compensate the large manpower investment required to rewrite the large application codes like the forecast model.

---

1. Dent, D., Proceedings of 5th ECMWF Workshop on the Use of Parallel Processors in Meteorology (eds Hoffmann, G. R. and Kauranne, T.), World Scientific, New Jersey, 1993.
2. Drake, J., Foster, I., Michalakes, J., Toonen, B. and Worley, P., *J. Parallel Computing*, 1995, **21**, 1571–1591.
3. Hack, J. J., Rosinski, J. M., Williamson, D. L., Boville, B. A. and Truesdale, J. E., *J. Parallel Computing*, 1995, **21**, 1545–1569.
4. Jones, P. W., Kerr, C. L. and Hemler, R. S., *J. Parallel Computing*, 1995, **21**, 1677–1694.
5. Sela, J. G., Anderson, P. B., Norton, D. W. and Norton, M. A., *J. Parallel Distrib. Computing*, 1994, **21**, 140–149.
6. IIT(D) and C-DAC (Pune), Final Project Report on Development and Implementation of Parallel Numerical Weather Prediction Algorithm on PARAM, 1996.
7. Jagadeesh, B. S., Phoolchand, Makkar, G. and Ramanandham, M., Parallelization and implementation of medium range weather forecasting codes on Anupam, BARC Project completion report, 1995.
8. Sinha, U. N., Sarasamma, V. R., Rajalakshmy, S., Subramannian, K. R., Bharadwaj, P. V. R., Chandrasekhar, C. S., Venkatesh, T. N., Sunder, R., Basu, B. K., Gadgil, Sulochana and Raju, A., *Curr. Sci.*, 1994, **67**, 178–184.
9. Selvakumar, S., Padmini, U., Padmalatha, A. and Periasamy, M., Parallelization of T80 weather model on CHiPPS, project completion report of C-DoT, 1994.
10. Foster, I., Gropp, W. and Stevens, R, *Mon. Weather Rev.*, 1992, **120**, 835–850.
11. WMO report No. 23, Research Activities in Atmospheric and Oceanic Modelling (ed. Staniforth, A.), 1996, WMO/TD No. 734.

Received 4 August 1997; revised accepted 18 February 1998.