NMRF/TR/07/2023

Scalability Analysis of the NCMRWF Unified Model (NCUM): July-August 2023

Shivali Gangwar, Dr. D Preveen Kumar, Dr. B Athiyaman

TECHNICAL REPORT

National Centre for Medium Range Weather Forecasting
Ministry of Earth Sciences, Government of India
A-50, Sector-62, NOIDA-201 309, INDIA

Scalability Analysis of the NCMRWF Unified Model (NCUM)

Shivali Gangwar, Dr. D Preveen Kumar, Dr. B. Athiyaman

July 2023

National Centre for Medium Range Weather Forecasting
Ministry of Earth Sciences, Government of India
A-50, Sector 62, NOIDA-201309, INDIA www.ncmrwf.gov.in

Ministry of Earth Sciences

National Centre for Medium Range Weather Forecasting
Document Control Data Sheet

1	Name of the Institute	National Centre for Medium Range Weather Forecasting
2	Document Number	
3	Month of publication	July 2023
4	Title of the document	Scalability Analysis of the NCMRWF Unified Model (NCUM)
5	Type of Document	Technical Report
6	No of pages, Figures and Tables	15 pages, 3 Figures, 6 Tables
7	Number of References	5
8	Author (S)	Shivali Gangwar, Dr. D Preveen Kumar, Dr. B. Athiyaman
9	Originating Unit	NCMRWF
10	Abstract	NCMRWF Unified model (NCUM), the seamless modeling framework used as the operational numerical weather prediction model at NCMRWF across a range of timescales. This report outlines the analysis of NCUM's scalability, specifically version 11.1 (UM vn11.1) across different horizontal resolutions 130 km (n96e), 12 km (n1024e) and 10 km (n1280e). These simulations are conducted on NCMRWF (National Centre for Medium Range Weather Forecasting) supercomputers, namely MIHIR (Cray XC40 system) and BHASKARA (IBM Idataplex system). The objective of this study is to compare the performance of NCUM model across systems at NCMRWF, IBM Idataplex and the Cray XC40. The performance scaling of 12 km (n1024e) resolution has been compared on MIHIR and BHASKARA system and it is observed that the cost of simulating a model in a day for the 12 km (n1024e) has increased by 34% with the increase in number of cores on MIHIR and by 10.1% on BHASKARA. Comparing cost of simulating a NCUM model for a day for 10 km and 12 km resolution on MIHIR resulted in increased cost for 10 km resolution compared to 12 km. The general trend in observations suggests that BHASKARA is cost efficient compared to MIHIR, but MIHIR demonstrates better scaling capabilities compared to BHASKARA system. Upon examining execution timings of MIHIR and BHASKARA system, with similar core counts (3132 and

		3136 cores respectively, considering the nearest core numbers), the execution timings on MIHIR are approximately 1.57 times higher than that on BHASKARA. A comparison of the speedup achieved by executing jobs of 12 km (n1024e) and 10 km (n1280e) resolution on the MIHIR system resulted in speedup of 86.6% and 78.5% respectively. The performance results of enabling Hyper-Threading on MIHIR resulted in slowing down of NCUM model. As a result, the Hyper-threading has been disabled for NCUM operational executions and NCUM model is running with 12 MPI tasks and 3 threads per node configuration. The observations obtained from execution of NCUM model on both MIHIR and BHASKARA systems yielded valuable insights, aided in understanding the behavior of application on different architectures.
11	Security classification	
12	Distribution	Unrestricted Distribution
13	Key Words	NCUM, CrayPAT, XC40

Table of Contents

सारांश

एनसीएमआरडब्ल्यूएफ एकीकृत मॉडल (एनसीयूएम), निर्बाध मॉडलिंग ढांचा जिसका उपयोग एनसीएमआरडब्ल्यूएफ में कई समय-सीमाओं में परिचालन संख्यात्मक मौसम पूर्वानुमान मॉडल के रूप में किया जाता है। यह रिपोर्ट विभिन्न क्षैतिज रिज़ॉल्यूशन 130 किमी (एन96ई), 12 किमी (एन1024ई) और 10 किमी (एन1280ई) पर एनसीयूएम की स्केलेबिलिटी, विशेष रूप से संस्करण 11.1 (UM संस्करण11.1) के विश्लेषण की रूपरेखा तैयार करती है। यह सिमुलेशन एनसीएमआरडब्ल्यूएफ (राष्ट्रीय मध्यम अवधि मौसम पूर्वानुमान केन्द्र) सुपर कंप्यूटर, अर्थात् मिहिर (क्रे एक्ससी 40 सिस्टम) और भास्कर (आईबीएम इडाटाप्लेक्स सिस्टम) पर आयोजित किए गए हैं।

इस अध्ययन का उद्देश्य एनसीएमआरडब्ल्यूएफ, आईबीएम इडाटाप्लेक्स और क्रे एक्ससी40 में सभी प्रणालियों में एनसीयूएम मॉडल के प्रदर्शन की तुलना करना है। 12 किमी (एन1024ई) रिज़ॉल्यूशन के प्रदर्शन स्केलिंग की तुलना मिहिर और भास्कर प्रणालियों पर की गई है। यह देखा गया है कि 12 किमी (एन1024ई) के लिए एक दिन में एक मॉडल का अनुकरण करने की लागत मिहिर पर कोर की संख्या में वृद्धि के साथ 34% और भास्कर पर 10.1% की वृद्धि के साथ बढ़ी है। मिहिर पर 10 किमी और 12 किमी रिज़ॉल्यूशन के साथ एक दिन के लिए एनसीयूएम मॉडल को अनुकरण करने की लागत की तुलना करने पर 12 किमी रिज़ॉल्यूशन की तुलना में, 10 किमी रिज़ॉल्यूशन की बढ़ी हुई लागत का पता चलता है। सामान्य अवलोकन में देखा गया है कि अवलोकनों की प्रमुख प्रवृत्ति सुझाव करती है कि भास्करा मिहिर की तुलना में लागत कुशल है, लेकिन मिहिर भास्करा सिस्टम की तुलना में बेहतर स्केलिंग क्षमताओं को प्रदर्शित करता है।

मिहिर और भास्करा सिस्टम के निष्पादन समयों की जांच करते हुए, समान कोर संख्या (3132 और 3136 कोरों को उपयुक्त कोर संख्याएँ मानते हुए), मिहिर पर निष्पादन समयों का आंकलन करने पर पाया गया है कि मिहिर पर निष्पादन समयों में लगभग 1.57 गुना की वृद्धि है जो कि भास्करा पर नहीं है।

मिहिर प्रणाली पर 12 किमी (एन1024ई) और 10 किमी (एन1280ई) रिज़ॉल्यूशन के कार्यों को निष्पादित करके प्राप्त स्पीडअप की तुलना के परिणामस्वरूप क्रमशः 86.6% और 78.5% देखी गयी है।

मिहिर पर हाइपर-थ्रेडिंग के प्रदर्शन परिणामों के परिणामस्वरूप एनसीयूएम मॉडल धीमा हो गया। जिसके परिणामस्वरूप, एनसीयूएम परिचालन निष्पादन के लिए हाइपर-थ्रेडिंग को अक्षम कर दिया गया है और एनसीयूएम मॉडल 12 एमपीआई कार्यों और 3 थ्रेड प्रति नोड कॉन्फ़िगरेशन के साथ चल रहा है।

मिहिर और भास्कर दोनों प्रणालियों पर एनसीयूएम मॉडल के निष्पादन से प्राप्त टिप्पणियों से मूल्यवान अंतर्दृष्टि प्राप्त हुई, जिससे विभिन्न आर्किटेक्चर पर मॉडल (एनसीयूएम), के व्यवहार को समझने में सहायता मिली। यह अध्ययन ओपन एमपी और एमपीआई के साथ-साथ स्रोत कोड में संभावित भविष्य के संवर्द्धन को शामिल करते हुए एनसीयूएम के प्रदर्शन विश्लेषण अध्ययन में योगदान देगा।

Abstract

NCMRWF Unified model (NCUM), the seamless modeling framework used as the operational numerical weather prediction model at NCMRWF across a range of timescales. This report outlines the analysis of NCUM's scalability, specifically version 11.1 (UM vn11.1) across different horizontal resolutions 130 km (n96e), 12 km (n1024e) and 10 km (n1280e). These simulations are conducted on NCMRWF (National Centre for Medium Range Weather Forecasting) supercomputers, namely MIHIR (Cray XC40 system) and BHASKARA (IBM Idataplex system).

The objective of this study is to compare the performance of NCUM model across systems at NCMRWF, IBM Idataplex and the Cray XC40. The performance scaling of 12 km (n1024e) resolution has been compared on MIHIR and BHASKARA system and it is observed that the cost of simulating a model in a day for the 12 km (n1024e) has increased by 34% with the increase in number of cores on MIHIR and by 10.1% on BHASKARA. Comparing cost of simulating a NCUM model for a day for 10 km and 12 km resolution on MIHIR resulted in increased cost for 10 km resolution compared to 12 km. The general trend in observations suggests that BHASKARA is cost efficient compared to MIHIR, but MIHIR demonstrates better scaling capabilities compared to BHASKARA system.

Upon examining execution timings of MIHIR and BHASKARA system, with similar core counts (3132 and 3136 cores respectively, considering the nearest core numbers), the execution timings on MIHIR are approximately 1.57 times higher than that on BHASKARA.

A comparison of the speedup achieved by executing jobs of 12 km (n1024e) and 10 km (n1280e) resolution on the MIHIR system resulted in speedup of 86.6% and 78.5% respectively.

The performance results of enabling Hyper-Threading on MIHIR resulted in slowing down of NCUM model. As a result, the Hyper-threading has been disabled for NCUM operational executions and NCUM model is running with 12 MPI tasks and 3 threads per node configuration.

The observations obtained from execution of NCUM model on both MIHIR and BHASKARA systems yielded valuable insights, aided in understanding the behavior of application on different architectures. This study will contribute to the performance analysis study of NCUM involving Open MP and MPI as well as potential future enhancements in the source code.

1. Introduction

The National Centre for Medium Range Weather Forecasting (NCMRWF) is operationally using NCUM global NWP system since 2012(Rajagopal et al., 2012). This system has been adapted from Unified Model (UM) seamless prediction system of "UM Partnership" and is being upgraded periodically to adapt new scientific and technological advancements. The major components of the NCUM global NWP system include components for data processing, data assimilation and forecast model (Brown et al., 2012, George et al., 2016).

The NCMRWF Unified Model (NCUM) is a numerical modelling software which has been designed to support prediction across range of timescales from weather forecasting to climate change. The version of the NCUM discussed here is (UM Vn11.1) global atmosphere configuration.

The presented results constitute the analysis of the computational performance regarding the scalability of the weather forecast configuration. The NCUM, has parallelization code that allow it run on parallel computers. The basic principle used is that of horizontal domain decomposition. Grid point space is divided into sub-domains each containing complete set of vertical levels. Each processor is responsible for its sub-domain and contains all the data in its local memory. Regular inter-processor communication is required for information transfer between neighboring subdomains.

A shared memory parallelization is available based on OpenMP compiler directive. This works underneath the parallel decomposition layer with each processor being able to utilize a number of threads to share work available.

The code is mainly in FORTRAN with a few calls to ANSI C routines. The MIHIR system, situated at NCMRWF as the national computing facility for weather forecast and research, constitutes a Cray XC40 infrastructure. The system has Broadwell compute nodes and relies on an Aries interconnect.

In this study we examine accessing the scalability of the NCUM on the MIHIR system. The examination covers three different resolutions: n96e, n1024e and n1280e, which correspond to horizontal resolutions of approximately 130 km, 12 km and 10 km, respectively. It focuses on observing how application behaves with increased MPI ranks and nodes. How a combination of improving a range of MPI settings and incorporating

Open MP directives can lead to speed improvements on the Cray XC40 system (MIHIR) for certain NCUM configurations (Bermous, Ilia & Steinle, Peter et al.,2015).

Additionally, a similar NCUM execution has been performed on BHASKARA for 12 km (n1024e) for comparative study with MIHIR system providing insights into their respective efficiency and resource utilization.

2. Methodology

NCMRWF Unified Model is a numerical weather prediction system that has been compiled using the Cray CCE 8.5.7 and Intel compiler version 16 on MIHIR and BHASKARA respectively and performance and scalability of both the systems has been compared. Table I provides an overview of the hardware specifications for the BHASKARA and MIHIR clusters.

NCUM obtains its initial condition analysis data from the global deterministic Hybrid 4DVar atmospheric data assimilation (DA) system operated at NCMRWF. The model's operations are managed and executed within a Python-based Rose/Cylc environment.

3. Problem Measurement

The NCMRWF NCUM model performance is measured in number of core-hours needed to simulate a day in the model (Sivalingam, Karthee et al.,2015).

T_{model} represents the time taken by the model to simulate a 5-day forecast, measured in seconds. This interval denoted as T_{model}, is calculated by considering the total wallclock time ($T_{wallclock}$ in seconds) and the initial setup time ($T_{initial}$ in seconds) necessary for the 5-day forecast.

$$T_{model} = T_{wallclock} - T_{initial} \qquad (1)$$

$$T_{hour} = T_{model}/3600 \qquad (2)$$

T_{hour} specifies the duration it takes to complete model execution, measured in hours.

The computation of the cost in core-hours for simulating a day within the model (C) can be evaluated using the formula:

$$C = (n_{cores} * T_{hour}) / 5 \qquad (3)$$

The resulting cost is expressed in kilo core-hours.

4. Experimental Setup

Description of the HPC Clusters and software used:

A. Hardware: Specifications of HPC clusters

Numerical results have been obtained through computations conducted on two distinct High-Performance Computing (HPC) clusters. These clusters are equipped with IBM InfiniBand interconnect and Intel Sandybridge processors on BHASKARA system and Intel Xeon Broadwell processors and Cray XC40 Aries Interconnect on MIHIR system at NCMRWF. Table I provides comprehensive overview of the hardware specifications of both systems.

TABLE I. MIHIR AND BHASKARA HARDWARE SPECIFICATIONS

	BHASKARA	**MIHIR**
Processor Type	E5-2670 SandyBridge-EP	Intel Xeon Broadwell E5-2695
Number of Compute Nodes	1052	2322
Total Number of Cores	16832	83592
Interconnect	FDR14 InfiniBand	Cray Aries with Dragon Fly
Node Processor Cores	2 x (2.6 GHz, 16-Core)	2 x (2.1 GHz, 36-Core)
Memory Size per Node	64 GB	128 GB
Memory Type	DDR3-1600 DIMMs	DDR4
Node Cache Size	2 x 20 MB	2 x 24 MB
Turbo Boost	OFF	OFF
Usage of Hyper Threading	ON	ON
Peak Compute Power	350 TF	2806 TF
Effective Distributed Memory	67 TB	297 TB

B. Software: Compiler and MPI Library

NCUM version 11.1 has been compiled with different compilers. On BHASKARA system, the Intel compiler version 16 has been used for compilation and on MIHIR system cray compilers CRAY CCE 8.5.7 has been used to compile NCUM. This compiler selection has been made to ensure compatibility and optimization for the respective hardware systems.

The following Cray FORTRAN compiler flags are used to enforce bit reproducibility:

-e m -s default64 -hflex_mp=intolerant -h omp

- '-e m': Specifies the generation of code that run on specified target hardware
- '-s default64': Sets the default pointer size to 64 bits.
- '-h flex_mp=intolerant': Enforces bit reproducibility, which aims to achieve consistent results across different runs by avoiding floating point variability.
- '-h omp': Enables support for OpenMP parallelization.

These compiler flags have been chosen to ensure specific performance and behavior characteristics specifically in terms of bit reproducibility and parallelization.

Enforcing bit reproducibility though results in performance penalty but have been adopted to ensure the numerical results and facilitate future code level optimization. (Sivalingam, Karthee et al.,2015).

For optimizing the execution results on MIHIR cluster, following compiler options have been used:

-e m -s default64 -hflex_mp=default -hfp4 -h omp

- '-e m': Specifies the generation of code that run on specified target hardware
- '-s default64': Sets the default pointer size to 64 bits.
- '-h flex_mp=default': Instructs the compiler to use the Cray's default memory model
- '-h omp': Enables support for OpenMP parallelization.
- '-h fp4': Optimizes the floating-point operations for performance.

Compilation of Fortran and C sources of NCUM on Intel's Sandy Bridge system of BHASKARA, following Intel compiler options have been used:

-O3 -no-vec -fp-model precise -qopenmp -i8 -r8 - mcmodel=medium -xavx

- '-O3': Specifies the highest level of optimization.
- '-no-vec': Disables vectorization
- '-fp-model-precise': Ensures precise floating-point calculations.
- '-qopenmp': Enables OpenMP parallelization.
- '-i8 -r8': Makes integer, logical, real and complex variables 8 bytes long.

- '-mcmodel=medium': Specifies medium memory model.
- '-xavx': Generates codes for advanced vector extensions supporting up to 256-bit vector data.

The General COMmunications (GCOM) library, responsible for interfacing with MPI communication libraries is supplied with NCUM sources. GCOM version 6.6, is used with NCUM vn11.1, also compiled with '-O3' optimization level.

5. Performance Analysis

A. *Hyperthreading (HT)*

In the NCUM (NCMRWF Unified Model), parallelization is achieved through a combination of message passing and threads. Both the MIHIR and BHASKARA systems supports Hyper-Threading (HT). On the MIHIR system, Hyper-Threading (HT) can be enabled using the'-j' option with the aprun command. For instance, using'-j 2' enables 2 hardware threads per processing elements (PE) as demonstrated below based on M. J. Glover et al.'s work in 2016.

aprun -n 72 -j 2 NCUM.exe

Enabling Hyper-Threading (HT) increases the number of PE's available per node, permitting NCUM to run with twice the number of MPI tasks or threads.

Table II presents the performance results of Hyper-Threading (HT) on MIHIR for a job conducted with 130 km (n96e) resolution. The performance is measured using 1, 2, 3, 6 threads. The count of MPI task per node is doubled when Hyper-Threading (HT) is enabled resulting in 72 cores per node, which include 1 physical core and 1 virtual core. However, the speedup expected from Hyper-Threading has not been achieved in NCUM on MIHIR. In fact, its observed that for high resolution jobs that spans thousands of cores, enabled Hyper-Threading, slows the NCUM.

T_{model} ($_{\text{n96e single node HT ON}}$) = 2786.54 secs

T_{model} ($_{\text{n96e single node HT OFF}}$) = 2741.265 secs

Speedup = $T_{\text{model (n96e single node)}}$ / $T_{\text{model(n1024e)}}$

Speedup $_{\text{(HT-ON)}}$ = 2786.54/2399.99 = 1.16

Speedup $_{\text{(HT-OFF)}}$ = 2741.265/1999.99 = 1.37

TABLE II. HT ON AND OFF SETTING ON NCUM N96E RESOLUTION

Cores	Threads	HT ON	Cores	Threads	HT OFF
MIHIR: n96e			***MIHIR: n96e***		
72	1	2976.75	36	1	2848.36
72	2	2833.16	36	2	2755.69
72	3	2786.54	36	3	**2741.265**
72	6	2794.26	36	6	2864.786

As a result of these findings, the Hyper-threading has been disabled for NCUM operational executions and NCUM model is running with 12 MPI tasks and 3 threads per node configuration.

B. NCUM Performance Scaling

Table III and IV, provides insight into the performance scaling of the NCUM jobs for 12 km (n1024e) resolution on both the MIHIR and BHASKARA systems. IO servers have been used for the runs conducted.

TABLE III. PERFORMANCE SCALING OF NCUM JOBS ON MIHIR FOR 12 KM (N1024E) RESOLUTION

EW	NS	OMP	NODE	T_{model}	T_{hour}	n_{core}	Cost in kilo core-hours(C)
n1024e							
12	30	3	33	14921.61	4.144	1188	0.984
42	12	3	45	11087.38	3.08	1620	0.997
24	42	3	87	6021.23	1.68	3132	1.047
54	24	3	111	4915.73	1.37	3996	1.091
36	54	3	165	3729.41	1.034	5940	1.230
30	48	6	246	2593.92	0.720	8856	1.276
36	54	6	330	1999.99	0.555	11880	1.319

TABLE IV. PERFORMANCE SCALING OF NCUM JOBS ON BHASKARA FOR 12 KM (N1024E) RESOLUTION

EW	NS	OMP	NODE	T_{model}	T_{hour}	n_{core}	Cost in kilo core-hours(C)
n1024e							
16	32	2	68	25887.27	7.2	1088	1.564
16	48	2	100	18214.49	5.06	1600	1.619
24	40	2	124	14924.367	4.15	1984	1.644
32	48	2	196	9486.58	2.64	3136	1.652
48	56	2	340	5480.64	1.52	5440	1.656
48	60	2	364	5200.12	1.45	5824	1.682
48	64	2	388	4995.24	1.39	6208	1.722

Performance is measured in terms of the number of kilo core-hours needed to simulate a day within the model, denoted as “C”.

On the MIHIR system, the cost of simulating a day in a model (C) increases by approximately 34% as the number of cores escalates from 1188 cores to 118800 cores. In contrast, on the BHASKARA system, the cost of simulating a day in model (C) increases by approximately 11.8% with increase in core count from 1088 cores to 6208 cores.

Using approximately two times more cores on MIHIR (11880) compared to those on BHASKARA (6208), MIHIR exhibits the peak performance (T_{model}) that is approximately 2.5 times higher than BHASKARA which approximately 60% increased performance on MIHIR.

In terms of execution timings, when comparing MIHIR and BHASKARA with nearly equivalent core counts (3132 and 3136 respectively), MIHIR’s performance surpasses that of BHASKARA by approximately ~1.57 times.

TABLE V. PERFORMANCE SCALING OF NCUM JOBS ON MIHIR FOR N1280E (10 KM) RESOLUTION

EW	**NS**	**OMP**	**NODE**	T_{model}	T_{hour}	n_{core}	**Cost in kilo core-hours(C)**
n1280e							
12	30	2	33	24305.73	6.75	1188	1.604
18	24	2	39	20751.26	5.76	1404	1.618
12	42	2	45	18156.89	5.04	1620	1.634
24	42	2	87	9634.09	2.68	3132	1.676
24	54	2	111	7586.86	2.11	3996	1.684
36	54	2	165	5436.49	1.51	5940	1.794

Table V presents the scaling analysis of the 10 km (n1280e job) on MIHIR system. Upon comparing the runs of 12 km (n1024e) and 10 km (n1280e) on MIHIR system, it is observed that the cost of simulating a model in a day for 12 km (n1024e) has increased by 34% as the number of cores increased from 1180 to 11880. However, on transitioning to the higher resolution of 12 km (n1280e) there is only 11.8 % increase in cost (C).

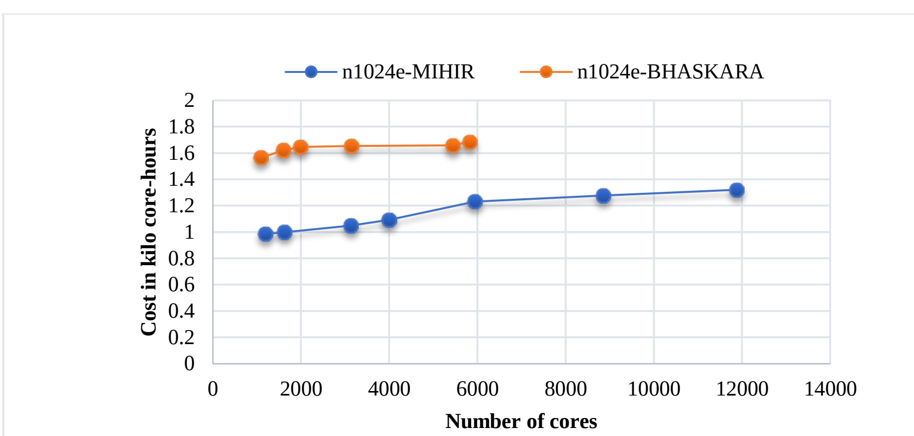

Fig. 1. Performance scaling of NCUM Job on MIHIR and BHASKARA comparing cost in simulating a model for 5 days to number of cores used

Figure 1 compares the cost of simulating a NCUM model in a day (C) on both MIHIR and BHASKARA systems to the number of physical cores (n_{core}). In this figure a horizontal line represents perfect scaling depicting a constant cost of simulating a NCUM model (C) as number of physical cores ($n_{core)}$ increases. Longer line represents better performance scaling. As a result, though the BHASKARA appears more cost efficient when compared to MIHIR system but MIHIR demonstrates better scaling capabilities in comparison to BHASKARA.

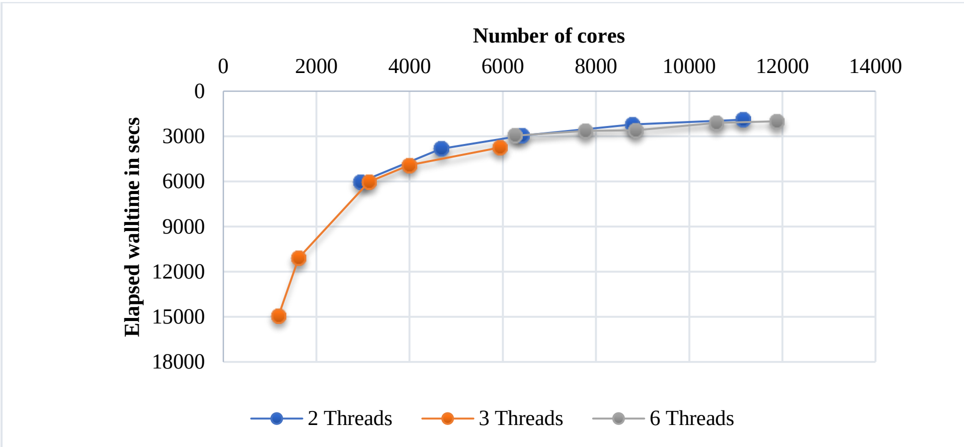

Fig. 2. Performance scaling of NCUM 12 km (n1024e) job on MIHIR using 2, 3 and 6 threads

Figure 2. compares the scaling of NCUM on the MIHIR system across various thread counts. The figure illustrates that the NCUM exhibits optimal performance when utilizing 3 OpenMP threads.

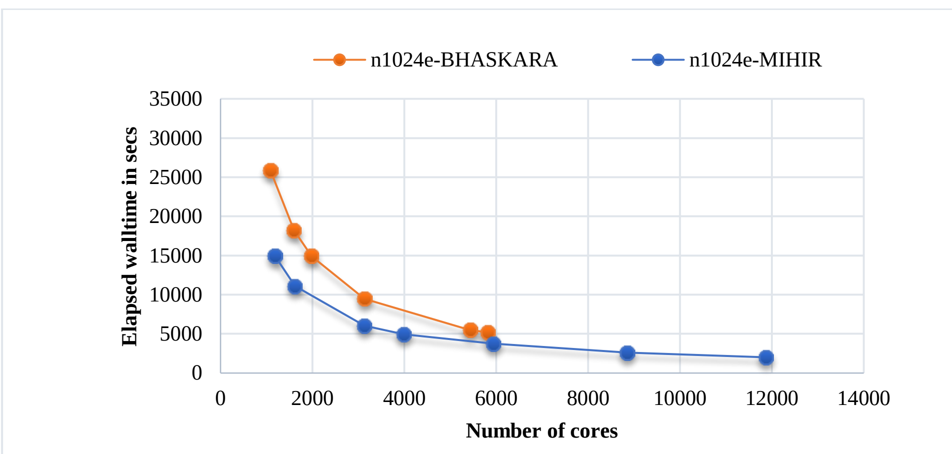

Fig. 3. Performance scaling of NCUM job on MIHIR and BHASKARA Comparison of elapsed walltime with the number of cores

Figure 3 illustrates a comparison between the elapsed walltime to the number of cores for 12 km (n1024e) model run on BHASKARA and MIHIR systems. It is evident from the figure that executing a 12 km (n1024e) job on the BHASKARA system takes more time on fewer nodes, thus doesn't demonstrate efficient performance scaling.

TABLE VI. COMPARISON OF 12 KM (N1024E) AND 10 KM (N1280E) FOR SPEEDUP ACHIEVED ON MIHIR

12 km (n1024e)			10 km (n1280e)		
n_{core}	T_{model}	*speedup = 2868.131/T_{model} (n1024e)*	n_{core}	T_{model}	*speedup = 2868.131/T_{model} (n1280e)*
1188	14921.61	0.1922	1188	24305.73	0.1181
1620	11087.38	0.2586	1404	20751.26	0.1382
3132	6021.23	0.4763	1620	18156.89	0.1579
3996	4915.73	0.5834	3132	9634.09	0.2977
5940	3729.41	0.7691	3996	7586.86	0.3781
8856	2593.92	1.105	5940	5436.49	0.5276
11880	1999.99	1.434	9612	5214.183	0.5501

Table VI presents a comparison of the speedup achieved by executing jobs of resolutions 12 km (n1024e) and 10 km (n1280e) on the MIHIR System. The speedup values are calculated by comparing the elapsed walltime of running NCUM on a single node using the 130 km (n96e) resolution with the run timings achieved on running multiple processors, with 12 km (n1024e) and 10 km (n1280e) resolutions.

T_{model} ($_{\text{n96e single node}}$) = 2868.131 secs

Speedup = $T_{\text{model (n96e single node)}} / T_{\text{model(n1024e/n1280e)}}$

The speedup achieved for 12 km (n1024e) and 10 km (n1280e) resolutions on MIHIR system is 86.6% and 78.5% respectively.

Summary and Conclusion

We have analyzed performance of NCUM at three resolutions on MIHIR supercomputer and subsequently compared the results with the performance on the BHASKARA system. The performance scaling of the 12 km (n1024e) and 10 km (n1280e) on MIHIR supercomputer has been compared, and it is observed that the cost of simulating a model in a day for the 12 km (n1024e) has increased by 34% with the increase in number of cores on MIHIR and by 10.1% on BHASKARA. Comparing cost of simulating a NCUM model for a day for 10 km and 12 km resolution on MIHIR resulted in increased cost for 10 km resolution compared to 12 km. The general trend in observations suggests that BHASKARA is cost efficient compared to MIHIR, but MIHIR demonstrates better scaling capabilities compared to BHASKARA system. Upon examining execution timings of MIHIR and BHASKARA system, with similar core counts (3132 and 3136 cores respectively, considering the nearest core numbers), the execution timings on MIHIR are approximately 1.57 times higher than that on

BHASKARA. A comparison of the speedup achieved by executing jobs of 12 km (n1024e) and 10 km (n1280e) resolution on the MIHIR system resulted in speedup of 86.6% and 78.5% respectively. The performance results of enabling Hyper-Threading on MIHIR resulted in slowing down of NCUM model. As a result, the Hyper-threading has been disabled for NCUM operational executions and NCUM model is running with 12 MPI tasks and 3 threads per node configuration. The observations obtained from execution of NCUM model on both MIHIR and BHASKARA systems yielded valuable insights, aided in understanding the behavior of application on different architectures. This study will contribute to the performance analysis study of NCUM involving OpenMP and MPI as well as potential future enhancements in the source code.

Acknowledgement

The research conducted in this study was carried out utilizing the computational resources of the MIHIR and BHASKARA supercomputers at the National Centre for Medium Range Weather Forecasting. We extend our sincere gratitude to our colleagues and fellow scientists at NCMRWF for their invaluable assistance and guidance throughout the duration of this project. Furthermore, we would like to express our appreciation to the Head of NCMRWF for their continuous encouragement and unwavering support that bolstered us during the course of this endeavor.

References

Mercier, Guillaume & Jeannot, Emmanuel. (2011). Improving MPI Applications Performance on Multicore Clusters with Rank Reordering. 6960. 39-49. 10.1007/978-3-642-24449-0_7.

Sivalingam, Karthee & Lister, G. & Lawrence, Bryan. (2015). Performance analysis and Optimisation of the Met Unified Model on a Cray XC30.

Bermous, Ilia & Steinle, Peter. (2015). Efficient performance of the Met Office Unified Model v8.2 on Intel Xeon partially used nodes. Geoscientific Model Development. 8. 769-779. 10.5194/gmd-8-769-2015.

John P George, S. Indira Rani, A. Jayakumar, Saji Mohandas, Swapan Mallick, A. Lodh, R. Rakhi, M. N. R. Sreevathsa and E. N. Rajagopal (2016). NCUM Data Assimilation System.

M. J. Glover, A. J. Malcolm, M. Guidolin and P. M. Selwood (2016). Stitching Threads into the Unified Model.