Indian Institute of Technology, Kanpur

Proposal for a New Course

1. Course No: **IDC XXX**

2. Course Title: **Parallel Computation of Sparse Matrix Systems**

3. Per Week Lectures: **3** (L), Tutorial: **0** (T), Laboratory: **1** (P), Additional Hours[0-2]: **0** (A),

   Credits (3*L+2*T+P+A): **10** Duration of Course: **Full Semester**

4. Proposing Department/IDP : **Aerospace Engineering and Mechanical Engineering**

   Other Departments/IDPs which may be interested in the proposed course: **ME, CHE, SEE, PHY**

   Other faculty members interested in teaching the proposed course: **Dr. Malay Das (ME)**

5. Proposing Instructor(s): **Dr. Ashoke De (AE) & Dr. Malay Das (ME)**
6. Course Description:

A) Objectives:

Sparse Matrix systems in parallel environments are one of the widely used techniques in the scientific community and industry as well. This course aims to introduce students from interdisciplinary Engineering and science streams to the fundamentals of the sparse matrix systems. The students will learn about the basics of a sparse matrix, parallel programming, and solution techniques of the sparse matrix using a parallel environment. A sparse matrix is the outcome of discretized partial differential equations that represent the conservation laws to simulate fluid flow, heat transfer, and other related physical phenomena. Thereafter, the system of algebraic equations is solved to compute the values of the dependent variable for each of the elements to represent the physical processes. The course is targeted for all engineering and science disciplines to the emerging paradigm of quantum computing.

B) **Contents: considering the duration of each lecture is 50 minutes** (*preferably in the form of 5 to 10 broad titles)*:

S No.	Broad Title	Topics	No. of lectures
1	Basics of Matrix Computations	1) Subspaces, Bases, Orthogonality, Matrices, Projectors, Norms. Floating point arithmetic. 2) Systems of linear equations. Solution of Systems of Linear Equations: matrix LU factorization. Special matrices: symmetric positive definite, banded. 3) Error analysis, condition numbers, operation counts, estimating accuracy. 4) Orthogonality, the Gram-Schmidt process. Classical and modified Gram-Schmidt. Householder QR factorization. Least-squares systems. 5) Eigenvalues, singular values. The Singular Value Decomposition. Applications of the SVD. 6) Eigenvalue problems: Background, Schur decomposition, perturbation analysis, power and inverse power methods, subspace iteration;	6

		the QR algorithm. 7) The Symmetric Eigenvalue Problem: special properties and perturbation theory, Law of inertia, Min-Max theorem, symmetric QR algorithm, Jacobi method. Applications.	
2	Sparse Matrices	1) Sparse matrices and their origin. Graph representation of sparse matrices, sparse graphs, Discretization of Partial Differential Equations. Electrical networks, Information retrieval. 2) Storage schemes for sparse matrices. Regular and irregular structures.	6
3	Direct Solution methods of Sparse matrices	Direct solution methods; Variants of Gaussian Elimination; Permutations and orderings; Band and envelope methods; Cuthill-Mc Kee and reverse Cuthill-Mc Kee orderings; Graph representation; Elimination tree; The frontal and multifrontal approaches; Minimal degree and nested dissection orderings.	3
4	Iterative Solution methods of Sparse matrices	Iterative methods; Projection methods; One-dimensional case: steepest descent, minimal residual methods; Krylov subspace methods; Conjugate gradient (CG) method; basic convergence theory; Connection to Lanczos tridiagonalization and orthogonal polynomials; The idea of preconditioning.	5
5	Eigenvalue problems	Types of problems; Subspace iteration; Krylov methods; Arnoldi's method; The Lanczos algorithm; Nonsymmetric Lanczos.	3
6	Basics of Parallel Programming	1) Introduction; Historical Perspective; Types of parallelism; Parallel algorithms and parallel computing. 2) Parallel computing platforms, Taxonomy, Pipelined-, Vector-, superscalar. Examples of parallel platforms. 3) Memory and cache performance issues, Hierarchical memories, Latency, bandwidth, Caches. 4) Parallel algorithms, design. Parallel performance metrics (Efficiency, load balancing, scalability, ...)	6
7	Parallel Programming Using OpenMP, MPI, and OpenACC	1) Programming shared memory machines - openMP. 2) Programming GPUs, CUDA, openACC. 3) Basic communication operations. Programming with MPI. 4) Programming distributed systems - MPI	6
8	Parallel toolkit	1) Parallel Scientific Libraries: BLAS, LAPACK, SCALAPACK, ARPACK, LIS libraries (only two or three of them will be covered) 2) PETSC	5

C) Recommended pre-requisites, if any (*examples: a- PSO201A,or b- PSO201A or equivalent*):
Undergraduate/Graduate Mathematics, Linear Algebra, and Programming

D) Short summary for including in the Courses of Study Booklet:

This course introduces parallel computing to sparse matrix systems and their applications. It covers parallel architectures, parallel algorithms, and their analysis in the context of sparse linear systems and eigenvalue problems. The course will start with a general discussion of sparse matrices, their origins, and how they are stored and exploited. Then it will briefly cover direct solution methods and iterative methods for solving sparse linear systems of equations and sparse eigenvalue problems. Further, it will discuss other topics related to sparsity, e.g., graph-based algorithms in machine learning, and basic nonlinear techniques. Finally, it will also introduce programming on parallel platforms. Along with the programming medium; OpenMP, MPI, and CUDA for the NVIDIA Graphics Processing Units (GPUs), will also be covered, in conjunction with a quick overview of openACC. The course blends theory with practical issues such as parallel architectures and parallel programming.

7. Recommended text / reference books:

1) ***Introduction to Parallel Computing***, 2nd edition, by V. Kumar, A. Grama, A. Gupta, and G. Karypis (2003).
2) ***Introduction to Parallel Programming*** by Peter S. Patcheco, Elsevier (2011).
3) ***Programming Massively Parallel Processors***, Third Edition: A Hands-on Approach by David B. Kirk and Wen-mei W. Hwu. (2017).
4) ***Using MPI, Portable Parallel Programming with the Message-Passing Interface*** by William Gropp, Ewing Lusk, and Anthony Skjellum, Second Edition, MIT Press, 1999.
5) ***Matrix Computations, 4th edition*** by G. Golub and C. Van Loan. John Hopkins, 2015.
6) ***Numerical linear algebra*** by Lloyd N. Trefethen and David Bau, III. SIAM, 1997.
7) ***Iterative methods for sparse linear systems*** (2nd edition) by Yousef Saad
8) ***Direct methods for sparse linear systems*** by T. A. Davis, SIAM publishing, 2006.

8. Any other remarks: **NA**

Dated: **05/08/24**________ Proposers: **Dr. Ashoke De (AE) and Dr. Malay Das (ME)**______

Dated:________________

DPGC Convener (AE) :__________________________ DPGC Convener (ME) :_____________________

The course is approved / not approved

Chairman, SUGC/SPGC

Dated:________________________________