**PARALLEL IMPLEMENTATION OF THE CONJUGATE GRADIENT METHOD**

**USING MESSAGE-PASSING INTERFACE**

**LLOYD MAWUNYEGA DZOKOTO**

**GLOBAL SOFTWARE DEVELOPMENT , HOCHSCHULE FULDA**

**TABLE OF CONTENTS**

1. **ABSTRACT**
2. **INTRODUCTION**
3. **MESSAGE-PASSING INTERFACE**
4. **CONJUGATE GRADIENT METHOD**
5. **PARALLEL CONJUGATE GRADIENT METHOD**
6. **EXPERIMENTAL RESULTS**
7. **SUMMARY**
8. **REFERENCES**
9. **SOURCE CODES**

**1. Abstract**

The objective of this project is to implement in C programming language both the serial and parallel Conjugate Gradient Method. The parallel variant of CG Method would be implemented using Message Passing Interface (MPI) and an experiment will be conducted to compare the performance of the serial to the parallel Conjugate Gradient (CG) Method.

**2. Introduction**

Cg?

Why paralelize?

Why mpi?

The world’s desire for more computing power cannot be achieved by convectional, single-processor architectures. There is no doubt that many companies are investing into developing new architectures with multiple processors.

**3. Message-Passing Interface (MPI)**

MPI is the most widely used library of subprograms that can be called from C and Fortran programs. It was developed by an open, international forum consisting of representatives from industry, academia, and government laboratories. MPI was carefully designed to permit maximum performance on a variety of systems and based on message passing, a powerful paradigm for programming parallel systems. MPI makes it possible for developers of parallel software to write libraries of parallel program that are both portable and efficient.

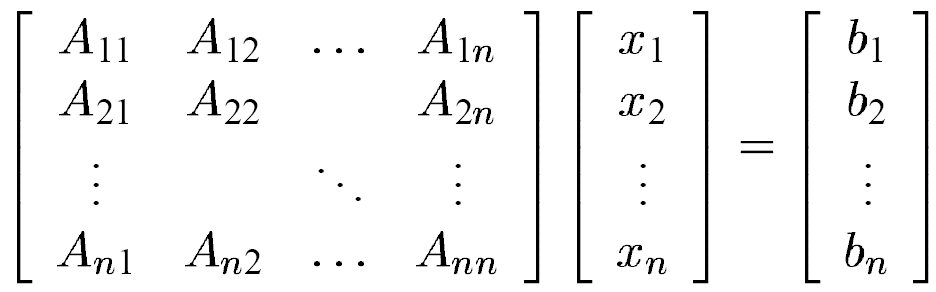
In the early stages of the intr

**4. Conjugate Gradient (CG) Method**

This method is used to solve large linear symmetric positive-definite linear equations of the form:

AX = B (1)

Equation (1) can be represented in matrix terms as:



Where:

A is a known square , symmetric , positive-definite (or positive-indefinite) matrix.

X is an unknown column vector.

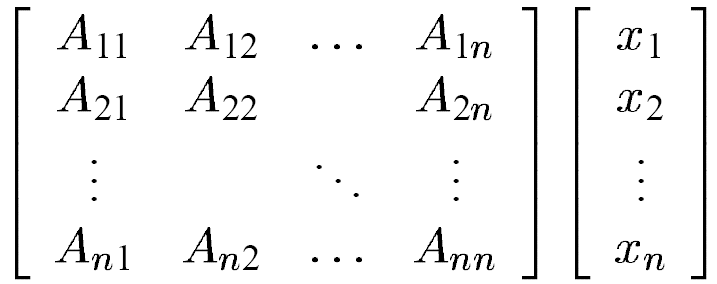
B is a known column vector.

A positive-definite matrix for nonzero vector X is defined as:

XT AX > 0 (2)

Expression (2) can be represented as:

[ *x1 x2 … xn* ]



*>0*

A matrix is defined as symmetric if its equal to its transpose.

A = AT (3)

Most of the iterative algorithms require a stopping criteria to stop the algorithm from executing indefinitely. A large number of the iterative methods use machine epsilon number as the stopping criteria. The CG method also makes use of this machine epsilon number but iterates at most the length of the vector b.

**4.1 Algorithm**

Consider the function:

*Q(x) = 1/2xT . A . x – xT b + c (3)*

Has a minimizer at *x\**, which approximates the exact solution of equation (1)

The CG method generates a sequence of approximate solutions *x0, x1, x2, …, xn* in order to approximate the exact solution *x\** of equation (1). The approximate solutions are computed such that the next approximation approximates the exact solution with decreasing error. Hence the CG method approximates the exact solution with a high degree of accuracy making it one of the most practically used iterative method.

The procedure of the CG method is as follows:

The execution begins with vectors *x0* and *d0* are considered as zero vectors with:

*R0 = b - Ax0 (4)*

The next approximation is then computed as:

*Xk+1 = xk + αk pk (5)*

The value of the new approximation is dependent of the value of the previous approximation *xk, a scalar step αk and a direction vector pk.*

**4.1.1 Algorithm**

* x0 = 0, r0 = b, p0 = r0
* **for** k = 0, 1, 2, 3, . . .n
* αk = (rTkrk) / (pTkApk) step length
* xk+1 = xk + αk pk approximate solution
* rk+1 = rk – αk Apk residual
* βk = (rTk+1 rk+1) / (rTkrk) improvement
* pk +1 = rk+1 + βk pk search direction
* **end**

From the algorithm above, it can be seen that the algorithm performs two operations of matrix-vector multiplications, four operations of inner product and five vector operations. The total number of operations for a symmetric positive-definite matrix can determined from:

*T1 = 2n3 + 13n2 (6)*

With complexity order O(n3).

**5. Parallel Conjugate Gradient (CG) method**

Parallelization involves a computer or computers with multiple processors to perform a common task.

**5.1 Parallel Data Distribution**

Parallelism is achieved for a data-parallel program by dividing the data among processors and each processor processing its own set of data.