CHAPTER : I

GENERAL INTRODUCTION

I. 1. Back Ground and Historical Remarks

Grand challenges, now, in computing technique of interest for examples are:

i) parallel and distributed algorithms

ii) Simulation of novel High Performance Computers (HPC) architectures and networks

iii) High performance visualization and virtual environments

iv) Modeling and simulation of transportation systems

v) Programming environments and tools for parallel and distributed computing

vi) Weather and climate

vii) Chemical and nuclear reaction

viii) Computational science and engineering applications such as:

1) Computational biology and bioinformatics
2) Computational physics
3) Computational chemistry
4) Computational fluid dynamics
5) Computational solid mechanics
6) Predictions of weather, climate , and global changes
7) Computational ocean and earth sciences
8) Combustion system simulation
9) Biomedical applications

ix) Optical interconnections

But in few words, what is parallel computing? and why use parallel processing?, and why optical interconnections?[1].

a) Traditionally, software has been written for serial computation:
    1) To be executed by a single computer having single Central Processing Unit (CPU)
    2) Problems are solved by a series of instructions, executed one after the other by the CPU. Only one instruction may be executed at any moment in time.
b) In the simplest sense, parallel computing is the simultaneous use multiple computing resources to solve a computational problem.
c) The computed resources can include:
    1) A single computer with multiple processors;
    2) An arbitrary number of computers connected by a network
    3) A combination of both.
d) The computational problem usually demonstrates characteristics such as the ability to be:
    1) Broken apart into discrete pieces of work that can be solved simultaneously;
    2) Execute multiple program instructions at any moment in time,
    3) Solved in less time with multiple compute resources than with a single compute resource.
e) Parallel computing is an evolution of serial computing that attempts to emulate what has always been the state of affairs in the natural world many complex, interrelated events happening at the same time yet within a sequence[2]. Some examples are :
    1) Planetary and galactic orbits,
    2) Weather and ocean patterns,
    3) Tectonic plate drift,

4) Rush hour traffic in Local Area (LA),
5) Automobile assembly line,
6) Daily operations within a business,
7) Building a shopping mall, and
8) Ordering a hamburger at the drive through.

f) Traditionally, parallel computing has been considered to be the high end of computing and has been motivated by numerical simulations of complex systems and "Grand Challenge Problems" mentioned before.

g) Today, commercial applications are providing an equal or greater driving force in the development of faster computers. These applications require the processing of large amounts of data in sophisticated ways. Example applications include:

1) Parallel databases, data mining ,
2) Oil exploration,
3) Web search engines, web based business services,
4) Computer aided diagnosis in medicine,
5) Management of national and multi-national corporations,
6) Advanced graphics and virtual reality particularly in the entertainment industry,
7) Networked video and multi-media technologies, and
8) Collaborative work environments

h) Ultimately, parallel computing is an attempt to maximize the infinite but seemingly scarce commodity called time.

i) There are two primary reasons for using parallel computing:

1) Save time-wall clock time, and
2) Solve larger problems

j) Other reasons might include:

1) Taking advantage of non-local resources-using available compute resources on a wide area network, or even the internet when local compute resources are scarce,
2) Cost savings-using multiple "cheap" computing resources instead of paying for time on a supercomputer, and
3) Overcoming memory constraints single computers have very finite memory resources. For large problems, using the memories of multiple computers may overcome this obstacle.

k) Limits to serial computing both physical and practical reasons pose significant constraints to simply building ever faster serial computers:
   1) Transmission speeds-the speed of a serial computer is directly dependent upon how fast data can move through hardware. Absolute limits are the speed of light (30 cm/nanosecond) and the transmission limit of copper wire (9 cm/nanosecond). Increasing speeds necessitates increasing proximity of processing elements,
   2) Limits to miniaturization processor technology is allowing an increasing number of transistors to be placed on a chip, However, even with molecular atomic level components, a limit will be reached on how small components can be, and
   3) Economic limitations-it is increasingly expensive to make a single processor faster. Using a larger number of moderately fast commodity processors to achieve the same (or better) performances less expensive.

l) The future: during the past 10 years, the trends indicated by ever faster networks, distributed systems, and multiprocessor computer architectures (even at the desktop level suggest that parallelism is the future of computing)[2].

By definitions, supercomputers are those hardware and software computing systems that provide close to the best currently achievable sustained performance. The performance of supercomputers, which is normally achieved by introducing parallelism, is typically better than of the vast majority of installed systems. Supercomputers are used to solve complex problems, including the simulation and modeling of physical phenomena such as climate change, explosions, or the behavior of molecules; the analysis of data from sources such as national security intelligence, genome sequencing, or astronomical observation; or the intricate design of engineered products. Their use is important for national security and defense, as well have increased and broadened, supercomputing has become less dominant than it once was. Many interesting applications require only modest amounts of computing, by today's standards. Because of the increase in computer processing power, many problems whose solution once required supercomputers can now be solved on relatively inexpensive desktop systems. That change has caused the computer industry, the research and development community, and some government agencies to reduce their attention to supercomputing. Yet problems remain whose computational demands for scaling and timeliness stress even our current supercomputers. Many of those problems are fundamental to many government's ability to address important national issues. One notable example is the Department of Energy's computational requirements for nuclear stockpile stewardship in USA. Governments have sponsored studies of a verity of supercomputing topics over the years. Recently, questions have been raised about the best ways for governments to ensure that and cost-effectiveness. An example to answer those questions, the National Research Council's Computer Science and Telecommunications Board Convened the Committee on the Future of Supercomputer to conduct a 2-year study to assess the state of supercomputing in the united state and to give recommendations for government policy to meet future needs. This study is sponsored jointly by the Department of Energy's Office of Since and by its Advanced Simulation and Computing (ASC) program. This interim report, presented approximately 6

months after the start of the study, reflects the committee's current understanding of the state of U.S. Supercomputing today, the needs of the future, and the factors that contribute to meeting those needs. After such a short time, the committee is not yet ready to comment in detail on the specifics of existing supercomputing programs or to present specific findings and well-supported recommendations. Although the committee have made considerable progress in understanding the current state of supercomputing and how it go there, it still has much more work to do before it develops recommendations [1].

Many technical, economic, and policy issue need to be addressed in this field. They include :

(1) The computational needs of present and future applications and approaches to satisfying them,

(2) The balancing of commodity components and custom design in supercomputer architectures and the effects of software design improvements and industry directions on that balance,

(3) The interplay of research, development, prototyping, and production in creating innovative advances,

(4) The extent and nature of direct government involvement to ensure that its needs are met, and

(5) The important requirement that the present not to be neglected while the future is being determined.

Although studies touches on each of those topics, the committees have not completed its consideration of them. The particular technical approaches of any program the develops or uses supercomputing represent a complex compromise between conflicting requirements and an assessment of risks and opportunities entailed in various approaches. An evaluation of these approaches requires a detailed understanding of :-

(1) The relevant applications,

(2) The algorithms used to solve those application problems,

(3) The performance likely to be achieved by codes that implement these algorithms on different platforms,

(4) The coding efforts required by various approaches,

(5) The likely evolution of supercomputing technology over multiple years under various scenarios, and

(6) The costs, probabilities, and risks associated with different approaches.

In its final report, the committee will seek to characterize broadly the requirements of different application classes and to examine the architecture, software, algorithm, and cost challenges and trade-offs associated with these application classes keeping in mind the needs of the nuclear stockpile stewardship program, the broad sciences community, and the national security community. The committees believe that it would be unwise to significantly redirect or reorient current supercomputing programs before careful scientific consideration has been given to the issues described above. Such changes might be hard to reverse, might reduces flexibility, and might increase costs in the future. In the period ahead, the committees will continue to learn and to analyze. A workshop focused on applications, 2003, will include a number of applications experts from outside the committee. Its purpose will be to identify both the computational requirements of important applications and the opportunities to adapt and evolve current solutions so as to benefit from advances in algorithms, architectures, and software. In addition, the committee will meet with experts who are developing solutions for applications of particular importance for national defense and security within the Department of Energy (DOE) and the National Security Agency (NSA). The committee will also meet with managers of supercomputing facilities, procurement experts, industrial supercomputers suppliers, experts on computing markets and economics, and others whose expertise will help to inform it [3].

I. 2. Supercomputing Today

High-speed computers are strongly needed not only for solving many scientific and engineering problems, but also for numerous industrial applications. Such applications include, but not limited to, computer aided design, oil exploration, whether predication, space applications and safety of nuclear reactors. These applications need [4] computational modeling and simulation techniques for finding the ultimate answers. The computer modeling and simulation jobs consume extensive processing time involving loops that are executed several times. If these loops can be effectively sub-divided and given to a number of processors, overall computational time can be reduced by a factor depending upon the number of the processors used.

A large number of mathematical models involve computations based on partial differential equations, Monte-Carlo simulations 2-d and 3-d fast furrier transformations, Matrix Multiplication and Matrix Inversion. When these methods are applied to solve complex problems like weather forecasting, computation of air flow around the aircraft, simulation of reactor systems etc., sheer number of operations required for their solution are so large that conventional uniprocessors are unable to solve them in a reasonable amount of time [5].

Over the past ten years, several important events have enabled commodity supercomputing :

i. The dramatic increase in microprocessor performance has continued and accelerated. The Intel Pentium (P5) and Pentium Pro (P6) have improved floating point performance be more than a fact ten over the i486. This has narrowed the performance gap between the high-volume Intel process and the top-of-the-line workstation microprocessors significantly. The difference in raw floating point performance (on typical scientific workloads) between a 200 MHz Pentium Pro and the best DEC, SGI/Caray, HP/Convex, Sun or IBM has to offer is between a factor of three and five,

ii. A free Unix operating system (Linux) [6] has been written and is being used by hundreds of thousand. Originally designed for Intel family machines, releases are now running on Alpha,

Power and Spare architectures. The freely distributable system has promoted development of high-quality devices drivers for nearly every possible peripheral device (notably Small Computer System Interface (SCSI) and Network cards),

iii. Intel designed the Peripheral Component Interconnect (PCI) local bus [7] which offers grater than Mbytes/second communication from the processor subsystem to the outside world, in a processor- independent manner. The PCI bus has become a de-facto standard, and allows peripheral cards (as Fast Ethernet interfaces) to be plugged into only machine, regardless of CPU architecture,

iv. Fast Ethernet has leapfrogged other technologies such as Asynchronous Transfer Mode (ATM) in terms of price/performance. 10 Mbit Ethernet is capable of a bandwidth over 10 Mbytes/sec between two points, with interface which cost less than $100 each. Commodity switches are now available which allow up to 16 Mac to communicate with each other at the full 100 Mbit bandwidth, at a cost of a few hundred dollars, and

v. The Message Passing Interface (MPI) standard has enabled the development of a reasonable amount of portable parallel software Groups working on machines such as the CM-5, Intel Paragon, IBM SP-2, and Cray T3D have sp considerable effort over the past several years to develop such software, and can take immediate advantage of new machines which support the MPI message passing standard and a suitable UNI software environment [5].

Parallel processing, the method of having many small tasks solve one large problem, has emerged as a key enabling technology in modern computing. The past several years have witnessed an ever-increasing acceptance and adoption of parallel processing, both for high-performance scientific computing and for more general-purpose applications. This was a result of the demand for higher performance, lower cost, and sustained productivity. The acceptance has been facilitated by two major developments: Massively Parallel Processors (MPPs) and the widespread use of distributed computing.

MPPs are now the most powerful computers in the world. These machines combine a few hundreds to a few thousands CPUs in a single large cabinet connected to hundreds of gigabytes of memory. MPPs offer enormous computational power and are used to solve computational grand challenge problems such as global climate modeling and drug design. As simulations become more realistic, the computational power required to produce them grows rapidly. Thus, researchers on the cutting edge turn to MPPs and parallel processing in order to get the most computational power possible.

A parallel computer architecture based entirely upon commodity computer components have been implemented. Using 16 Intel Pentium Pro microprocessors and switched fast communication fabric. Sustained performance on scientific application of one Gigaflop have been obtained. During one production astrophysics treecode simulation, They performed floating point operations (1.2 Petaflops) over a three week period, with one phase of that running continuously for two weeks without interruption.

The quest to build increasingly powerful cluster configurations for parallel computing has been enabled by the availability of increasingly capable network interconnects and software. Interconnect technologies such as Myrinet provide moderately high bandwidth and support kernel bypass messaging for low latency. These interconnects have been primarily utilized in clusters whose compute nodes contain only a few processors each. The Berkeley CLUMPS project was one of the first to use larger and more capable Symmetric Multiprocessor (SMP) systems as the basic building blocks for a cluster. SMP systems with larger processor counts make much higher demands of the network interconnect. Indeed, no existing interconnect can supply the bandwidth needed to build a balanced SMP cluster with comparable on-node and off-node bandwidths.

Sun Microsystems has developed the high performance *Sun* Fire™ Link interconnect, which allows multiple network interfaces to be striped to provide multi-gigabyte per second transfer rate. It also supports kernel bypass messaging via Remote Shared Memory (RSM),

whereby memory regions on one machine can be mapped into the address space of another. They [6] have exposed these capabilities to applications via the Sun MPI library, which implements a number of unique features that take maximum advantage of the characteristics of the interconnect. These features include memory-to-memory messaging, deferred connection establishment, interface striping, and multiple transfer protocols.

The characteristics of the interconnect, present the Sun MPI implementation over remote shared memory, has been discussed [6] and closed with a section on performance results a measured on Sun Fire serves clustered with the Sun Fire Link interconnect.

Some novel applications of high performance computing in a discipline now known as "experimental mathematics" was described. A key technique involved in that research was integer relation algorithm (recently named one of ten "algorithms of the century" by Computing in Science and Engineering). This algorithm permits one to recognize a numeric constant in terms of the formula that it satisfied. They presented a variant of PSLQ that is well-studied for parallel computation, and give several examples of new mathematical results that they had found using it. Two of these computations were preformed on highly parallel computers, since they are not feasible on conventional systems[7].

According to the June 2003 list of the 500 most powerful computer systems in the world, the United States leads the world in the manufacture and use of supercomputers, followed by Japan. A number of other countries make or use supercomputers, but to a much lesser degree. Virtually all supercomputers are constructed by connecting large numbers of compute notes, each having on or more processors and a common memory, by an interconnect network (a switch). Supercomputer architectures differ in the design of their nodes, their switches, and the node-switch interface. Higher node performance is achieved by using commercial scalar microprocessors with 64-bit data paths intended primarily for commercial serves or nodes designed specially for supercomputing, rather than the high-volume, 32-bit scalar microprocessors used in workstations

and lower capability cluster systems. The custom nodes tend to use spatial mechanisms such as vectors or multithreading to reduce memory latency rather replaying solely on the more limited latency avoidance afforded buy caches. High-bandwidth, scalable interconnects are typically custom-bulit and more expensive than the lithe **TOP500** list is available at more widespread Ethernet interconnects. Custom switch use is often augenmented by custom node/switch interfaces. The highest-ranked system in the TOP500 list is the Japanese-built Earth Simulator, released in the spring of 2002 and designed specifically vector-based nodes and a custom interconnect. The emergence of that system has been fueling recent concerns about continued U.S. leadership in supercomputing. The system software that is used on most contemporary supercomputers is some variant of Unix, either open sources or proprietary. Programs are written in FORTRAN, C, and C++ and use a few standard applications libraries. All of these supercomputers use implementations of the message passing interface Computer Message Passing Interface (CMPIN) standard to support internodes communications.

Two extremely powerful simulation software packages, in the field of parallel computing, Parallel Simulator Indian Machine (PSIM) and Parallel Virtual Machine (PVM) are extensively used to investigate the farming, function decomposition and domain decomposition in a unit good number of case studies. The case studies in PSM, which cover a good function rage are; sum of a series, numerical integration, discrete Fourier transform, fast Hartley transform, fast robot kinematics modeling, matrix multiplication and dot of product [8].

The results of the above study, have proven that farming has greatly improved the speedup, efficiency and time performance. Also, the volume of data has drastically affected the performance in all cases. Generally, increasing the volume of data improves all the performance measures. That is mainly due to that the communication time is becoming very small compared to task time. Of course, the case studies have to be done to get all the comparison full results. In recapitulation parallelism has proven to be a convenient technique for all the selected applications. The

limitations on PSIM are minor and can easily cured. These limitations are mainly due to simulating or building the parallel environment on a single sequential machine. PVM is an integrated set of software tools and libraries that emulate a general-purpose, flexible, heterogeneous concurrent computing framework on interconnected computers with varied architecture. The overall objective of the PVM system is to enable such a collection of computers to be used cooperatively for concurrent or parallel computation, us computation power increased meaning that efficiency and speedup are increased. To make the comparison bassists unified the network hardware should reach 16 unit, which is expensive. Hopefully, the gained results will help in assessing all parallel performance aspects.

I. 3. Why Optical Interconnect ?

The optical interconnects field has experienced a continuous push into applications with ever-shorter transport path lengths. This drive started only a brief time after the widespread deployment of long-haul optical fiber began about 25 years ago, and has been accelerating ever since. Significant optical interconnect research and development activities now span a length-scale of about seven orders of magnitude, ranging from dense wavelengths division multiplexing-based long-haul transport, routing, and switching—for signal transport in the hundreds of kilometers range, to dense device integration for chip-scale interconnects—where path lengths are on the order of centimeters.

The motivation for this trend stems from the widening mismatch in performance scaling between silicon-based electronic switching speeds and densities, and the metal based interconnection fabrics needed to keep pace with them. The hope is that the steady advances being made in integrated Optical Electronics (OE) array technology can be exploited to overcome the interconnect bottlenecks of metal based technology and sustain the exponential silicon-chip-based performance growth in computing and communications systems. The limitations of metal interconnects at the board, back-plane, and box-to-box levels are already affecting the overall

system performance scalability for many applications. This has resulted in significant inroads for optical interconnect technology in the box-to-box and back-plane domains. Thee achievements are based on advancements in the packaging of parallel optical transceivers with guided wave optical channels that enable higher bandwidth densities and efficiencies as compared to their metal-based competitors. As silicon IC technology continues its march toward deep-sub-micron features sizes, it is widely recognized that the interconnect fabric will be the critical source of performance bottlenecks at the intrachip level. A key question is, therefore, whether and how the optical interconnect trend will continue at the broad-interposer -, and chip level interconnect domains.

A consequence of the wide transport-length range in optical interconnect applications now under consideration is a commensurately wide diversity in research and development. Efforts now underway range from basic material and device research, to integration and packaging research and development, to system- and architecture-level design efforts and studies to determine efficient ways to exploit optical interconnect technology at each level. Perhaps the only common theme in optical interconnect research and development is a focus on efficient packaging—which seems to play a critical roll in the cost and performance of optical interconnect approaches at all levels [9].

Optical interconnections are pervasive at long-haul distances, and, as perchannel data rates rise, conventional electrical interconnections face multiple challenges at increasingly shorter distances. Electrical interconnects face critical tradoffs power consumption, interconnect area, and signal integrity, even at interconnect lengths as short as the board, module, and chip level. Physical material and structural limitations will ultimately force technology changes at the physical layer if interconnect and system performance gains are to continue well into the future. Critical questions facing optical interconnections are how to implement optical interconnects at these shorter distances and, in fact, all distances, in a cost-effective manner, and what are the break points at which it is necessary and at which it is possible to integrate optical interconnection of optical

signal handling and processing into electrical systems will also be justified by some system specifications.

For the next generation of the system, optical technology now stands at a threshold where the integration of optical interconnections and optical functions into board, package, and chip-level electrical systems is projected in industry roadmaps. These projections for the integration of optical interconnections range from 5 to 7 years, but the preponderance of electrical microsystem analysis project that optics will play a role in electrical systems. Critical to the implementation of optics in electronic systems is the method of integration and the cost to integrate the optics. The integration of optical functions such as interconnection into System-On-a-Package (SOP) and System-On-a-Chip (SOC) implementations stands at a similar threshold to that which electronics faced in the 1970s. Electronics in the 1970s stepped into the modern electronic age through the revolution that transformed common circuits from discrete components mounted on boards to integrated circuits in silicon. The unit cost and circuit complexity advantages that directly results from high yield, large area parallel fabrication of circuits in an integrated process such as silicon CMOS have led to the unparalleled increases in computational performance at a diminishing cost that the world has enjoyed for the last 7 years. If optical interconnections can be integrated at the board, package, and chip level with processes that are compatible with, and cost comparable to, electronic manufacturing technologies, then optics will see high volume, board market implementation.

Many quantitative comparisons of interconnection performance have been published discussing electrical and optical interconnections, and the question of how to integrate optical interconnections into an electrical interconnection system is currently a topic of intensive study. Optical interconnect approaches include free-space interconnects with diffractive optical elements, silicon optical bench interconnects, and guided wave interconnections, include substrate guided mode interconnects, fiber-0ptic waveguides, and integrated waveguide. This paper will discuss a

heterogeneous integration approach to optical interconnection interfaces for fiber, free space, and waveguide optical interconnections both singly and in "smart pixel" arrays and will explore a fully integrated planer light wave circuit approach to Microsystems integration with optical interconnection, as well.

One of the most critical questions that must be addressed for optical interconnections and interfaces in electrical systems relates to material compatibility and process compatibility with the wide variety of materials that are used today in electrical Microsystems. The electronic materials used commonly today are not necessarily those that will optimally be used for the integration of optical signals into these Microsystems. The materials choices cover a wide range of organic and inorganic materials, and material development, with a particular emphasis on polymers, is an area of intensive research and progress. Materials are of interest for integrated Microsystems include silicon and Si-CMOS circuits, compound semiconductors and the optoelectronic (OE) and high-speed electronic devices fabricated from the materials, organics such as polymers and epoxies for optical waveguides and electronic substrates, inorganic materials such as silicon dioxide and silicon nitride for passivation, encapsulation, and waveguides, and metals for electrical interconnections. The Microsystems integration of these materials, using processes and structures that do not compromise the performance optimization of individual components in the integrated Microsystems, is an ambitious goal in this field [9].

Optical interconnections offer the potential for gigahertz transfer rates in an environment free from capacitive bus loading and electromagnetic interface. The effectiveness of optical interconnections has been examined from both theoretic and practical perspectives. Over the past decade, much of the research in optical communications networks has focused on applications to Wide Area Networks (WAN's) and Metropolitan Area Networks (MAN's). More recently, specialized High-Speed Local Area Networks (HSLN's) for computer interconnections have been studied and commercial standards have emerged. Other research groups have investigated the

implementation of parallel computers using optical interconnections in multiprocessor applications.

Device technology in electro optics has also matured to a point where small, low power, and low-cost devices exist which are suitable for use in bus-level implementations. Initial efforts have focused on direct technology substitution in board level and chassis-to-chassis links. However, there are obvious limitations to such substitution. For example, any interface between electronics and optics limits the speed of that interface to the speed of electronics.

In switched networks, Time Division Switched (TDS), and Wavelength Division Switched (WDS), implementations have been used to perform message routing in both HSLN and multiprocessor applications. However, since switching device technology has developed more slowly than technology for other components, many recent designs have implemented low-latency "single-hop networks". These networks are composed of groups of processors linked by multiple passive star couplers which efficiently use optical power, and have simple control structures [10].

Optical interconnects have become an integral part of today's networks and are being introduced in high-performance computing systems. Standardization is occurring for data communication within networks, and for server-to-server interconnections and high-performance parallel buses. Optical interconnects provide solutions for almost all the levels in the interconnection hierarchy (from backplanes to rack interconnects to local area networks to long-haul). The main advantages of optical interconnects *are* ***higher density, and higher bandwidth length product.***

Table 1 Optical Signal Processing in Transmission System [11]

Network	Function	Optical Processing experiments and their devices
Signal Processing for Point/Multipoint-to-Multipoint Network	Path-termination Broadband SW Cross-connect Interconnection Distribution	40 Gb/s, 4 bit pattern detection (PLC) 8x8 port (PLC, LN SW) 100 ch (Optical FDM, PLC) 128x128 port (Liquid Crystal) >10000 port (CATV)
Signal Processing for Point-to-Point Transmission	Multi / demultiplexer Modulator / Regeneration Short pulse generation Decision Timing Circuit Equalizer Demodulator/Amplifier	100 Gb/s (Nonlinear SW, LN SW) 20 Gb/s (LN etc.) , 80 Gb/s (Soliton) 350 GHz (CPM) 50 Gb/s (Nonlinear SW) 10 Gb/s (PLL using SLA) 17 Gb/s (DS-fiber) >1 THz (EDFA)
	Transmission Medium	Single-mode fiber with In-Line EDFA > 60 Tb/s-km.

In fact there exists a break-even length above which optical interconnects are more advantageous then electrical ones in terms of power dissipation and bandwidth. For current technologies, optical interconnects are superior for length ~> 7 m for a system that has a maximum bit rate of 10 Gbit/s. Fiber optics is currently the physical medium of choice for single or parallel channels point-to-point links, but free-space is being considered for some applications such as backplanes. For data communication applications, the trend for the next two to five years will be toward increasing the

capacity of the fiber optics by the use of Coarse Wavelength De/Multiplexing (CWDM), and to augment the channel parallelism (one-dimension (1-D) to two-dimension (2-D) arrays of optical devices and fiber) and the channel speed (2.5 Gbit/s/s to 10 Gbit/s/s). similar technologies have also been proposed for the backplane parallel and Wavelength Division Multiplexing (WDM), and for telecom dense WDM.

References [11,12] presented and approached a modeling of the optical interconnect used in networks and advanced high-performance computing systems. The model is generic enough that it can be applied to fiber or free-space technologies, yet precise enough to extract from it salient performance predictions. The developed model is intended as an analysis tool for system builders, and to provide a guide for developing upcoming technologies such as free-space smart pixel-based interconnects [11,12].

Optical interconnects outperform electrical interconnects for long-distance applications. Even for distances as short as 300 meters (m) at bandwidths over 1 Gbit/s, fiber is now the default interconnect choice. Recent research has also suggested that optics has clear advantages even at very short distances, yet optical interconnects are not common in computers.

The reason optical interconnects are not widely implemented in modern computing systems is due to computer architecture design and how interconnects are balanced within those constraints. Despite the fact that optical interconnects have clear advantages in time of flight and bandwidth density, these are not the only design constraints that must be considered. As a consequence, the overall optimization of modern computers has generally precluded the use of highly integrated optical interconnects. The historical scaling of logic, memory, and wires, analyze these access time and density of Dynamic Random Access Memory (DRAM), walk through the steps and timing relations for memory access, and estimate the capacities of electrical interconnects at several levels was discussed. With this background, they should characterized what would be required for optical interconnects to displace wires at the backplane, board, and chip level [13].

I. 4. Evolution in Supercomputing

A major policy issue for supercomputing is the proper balance between investment in alternative approaches that may lead to a paradigm shift (the innovative aspect). Both aspects are important. At this stage in the study, committees all over the world see the following advantages for an evolutionary approach to investment and acquisition. First, much useful work is getting done using the existing systems, and their natural successors can be expected to continue that work. In addition, there are no obvious near-term architectural alternatives: The promising technology breakthroughs, such as processor - in -memory, streaming architectures, and the like, that might revolutionize supercomputing are far of in the future and less than certain. Of course, higher capability would enable better solutions to be obtained faster. But history suggests that even when revolutionary advances come along, they don't immediately supplant existing architectures. Difference problems benefit from different architectures and no one design is universally best. The committees see a need for evolutionary investments in all major approaches to supercomputing that are currently pursued: clusters built entirely of commodity components; scalable systems that reuse commodity microprocessors together with custom technology in the interconnect or the interconnect interface; and systems in which microprocessors, the interconnect and their interface are all customized. although some advantage also accrue form evolutions in software, and organization committees see a need for investment that would accelerated that evolution. The advantages of commodity architectural components might be more easily realized if more applications were redesigned, better custom software was provided, and the cost of bringing software to maturity was better appreciated. The benefit of software investment is that it tends to have continuing values as architectures evolve. At the same time, both the maintenance and the evolution of legacy applications must be anticipated and supported. Finally, the committee observes that uncertainties in policy and inconsistencies over time can be both disruptive and expensive. Unexpected pauses in an acquisition plan or failure to maintain a diversity of suppliers

and products can cause both the suppliers and the skilled workforce to divert their attenuation from supercomputing. It can be difficult and expensive to recover the supply of expertise.[1]

I. 5. Innovation for Supercomputing

Innovation in supercomputing stems form application motivated research hat leads to experimentation and prototyping, to advanced development and testbedes and to deployment and products. All the stages along that path need continuous, sustained investment in order that the needs of the future will be met. If basic research activities are not supported, revolutionary advances are less experimentation and advanced development are not done, the promising approaches never ripen into products.

In supercomputing, innovation is important in architectures, in software, in algorithms, and in application strategies and solution methods. The coupling of these aspects is equally important. Major architecture challenges stem from the uneven performance scaling of different components, in particular, as the gap between processor speeds, memory bandwidth, and memory and network latency increases, new ideas are needed to increase bandwidth and hide (tolerate) latency. Additionally, as new mechanisms are introduced to address those issues, there is need for ways to supply a stable software interface that facilitates exploiting hardware improvements while hiding the changes in mechanism. The need for software innovation is motivated by its role as an intermediary between the application (the problem being addressed) and the architectural platform. Innovation is needed in he ways that system manages the use of hardware sources, such as network communication. New approaches are needed for ways in which the applications programmer can express parallelism at a level high enough to reflect the application solution and without platform – specific details. Novel tools are needed to help application-level software designers reason about their solutions at a more abstract and problem-specific level. Software technology is also needed to lessen future dependence on legacy codes. Enough must be invested in the creation of advances tool and environment support for new language approaches so that users can more readily adopt

new software technology. Importantly, advances in algorithms can sometimes improve performance much more than architectural and software advances do. More realistic simulations and modeling require no only increased supercomputer performance but also new methods to handle finer spatial resolution, larger time scales, and very large amounts of observational or experimental data. Additional applications challenges for which innovation is needed are the coupling of multiple physical system's design by analytic estimation of its properties. Emerging applications in areas such as bioinformatics, biological modeling, and nanoscience and technology are providing both new opportunities and new challenges. [14]

Merrimac uses stream architecture and advanced interconnection networks to give an order of magnitude more performance per unit cost than cluster –based scientific computers built form the same technology. Organizing the computation into the streams and exploiting the resulting locality using a register hierarchy enables a stream architecture to reduce the memory bandwidth required by reprehensive applications by an order of magnitude of more. Hence a processing node with a fixed bandwidth(expensive) can support an order of magnitude more arithmetic units(inexpensive). This in turn allows a given level of performance to be achieved with fewer nodes (a 1-PFLOPS machine, for example, they just 8,192 nodes) resulting in greater reliability, and simpler system management. A sketch of the design of Merrimac, a streaming scientific computer that can be scaled from a $20k 2 TFLOPS workstation to a $20M 2 PFLOPS supercomputer was done. [9,15]

Modern semiconductor technology makes arithmetic inexpensive and bandwidth expensive. To exploit this shift in cost, a high-performance computer system must exploit locality, to raise the *arithmetic intensity* (the ratio of arithmetic to bandwidth) of the application as well as parallelism to keep a large number of arithmetic units busy. Expressing ban application as a stream program fulfills both of these requirements. It express large amounts of parallelism across stream elements and reduces global bandwidth by expressing locality within and between kernels.

A stream processor exploits the parallelism exposed by a stream program, by providing 100s of arithmetic units, and exploits the locality of a stream, by providing a deep register hierarchy. In particular, memory bandwidth is reduces by capturing short-term producer-costumer locality in large local register files, and long-term producer-consumer locality in a stream register file. This locality might not be captured by a reactive cache, more importantly, the stream register file is aligned with individual ALUs and requires and requires global on-chip communication.

Merrimac, a scientific computer system tailored to exploit the parallelism and locality of streams has been designed [9]. The core of Merrimac is a single-chip (90nm CMOS) stream processor that is expected to have 128 GFLOPS peak performance. This processor chip along with 16 high-bandwidth DRAM chips (2G Bytes of memory) from a single Merrimac node. Application experiments suggest that this single-node Merrimac will sustain up to half of peak performance on a range of scientific applications. With an estimated parts cost of less than $1 k per 128 GFLOPS node (including network), a Merrimac machine was expert to provide both capability and capacity –being more cost effective than machines based on commodity microprocessors.

Merrimac employs a high-radix interconnection network to connect 16 nodes (2 TFLOPS) on a single board, 512 nodes (64 TFLOPS) in a cabinet, and 8 nodes (! PFLOPS) in 16 cabinets. The network proved a flat shared address space across the multi-cabinet system with flat bandwidth across a board (16 nodes) and a global bandwidth of 1/8 the local bandwidth any where in the system.

Three representive scientific applications as stream programs have been coded and measured their performance on a simulated Merrimac node. These initial experiments show that typical scientific applications cast as stream programs maintain a high arithmetic to memory bandwidth ratio and achieve a high fraction of peak performance. The applications simulated have

computation-to-memory ratios in the range of 7:1 to50:1, achieving between 18% and 52% of the peak performance of the machine, with less than 1.5% of data references traveling off-chip [15].

Modern Very-Large Scale Integration (VLSI) technology makes arithmetic very cheap (100s of 64-bit FPUs per chip) and bandwidth very expensive (a few words/cycle of off-chip bandwidth). Expressing an application as a stream program exposes parallelism – to take advantage of the large number of arithmetic units and to hide the ever increasing memory latencies – and locality – to reduce the demand on the limited bandwidth. A stream processor exploits this parallelism and locally by providing a deep bandwidth hierarchy that exposes communication so it can be optimized in local register file, a stream processor significantly reduces an application's demand on memory bandwidth.

Merrimac is stream processor tailored for scientific applications. Merrimac is scalable form a 2 TFLOPS single board workstation to a 2PFLOPS supercomputer. A 90nm CMOS stream processor chip with a peak performance of 128 GFLOPS enables Merrimac to sustain on our pilot applications. A high-radix network gives Merrimac a flat global address space with only an 8:1 (local: global) bandwidth ratio. This gives Merrimac a memory efficiency of 250 K-GUPS/$. This relatively flat global memory bandwidth simplifies programming by reducing the importance of portioning and placement.

On the architecture front, alternative stream register file organizations that appear to offer even greater reduction in required memory bandwidth are explored. Investigating how to best use a cache in combination with a stream register file how to give the compiler more control over caching policies and also investigating global communication and synchronization mechanisms that are suitable for use with streams. This includes our scatter-add operation, which reduces the need for synchronization in many applications.

Finally the initial experiments used relatively simple 2D codes running on a single node of a simulated machine. Currently exploring the properties of larger and more complex 3d codes

running across multiple nodes of a simulated machine has been carried out. Initial indication are positive – that these codes exhibit at least as much 'stream' locality as their simpler counterparts.

Since 2001, a new generation of what is called Terascale Computing System (TCS) has been cast. The first is ASCI White at Lawrence Livermore National laboratory and the second is at Pittsburgh Supercomputing Center [16]

A grid of supercomputers was built (summer 2002) where four U.S super computer centers were indeed together into one massive "grid" style computer [17] IBM provided over 1000 server and installed them. Intel powered them with its second generation processed. Such grid networks allow uses to a large processing power over a networks similar to the way information can be shared over the Internet.

As computer networks become cheaper and more powerful, a new computing paradigm is poised to transform the practice of science and engineering.

Driven by increasingly complex problems and propelled by increasingly powerful technology, today's science is as much based on computation, data analysis, and collaboration as on the efforts of individual experimentalists and theorists. But even as computer power, data storage, and communication continue to improve exponentially, computational resources are failing to keep up with what scientists demand of them.

A personal computer in 2001 is as fast as a supercomputer of 1990. But 10 years ago, biologists were happy to compute a single molecular structure. Now, they want to calculate the structures of complex assemblies of macromolecules and screen thousands of drug candidates. Personal computers now ship with up to 100 gigabytes (Gbit/s) of storage – as much as an entire 1990 supercomputer center. But by 2006, several physics projects, CERN's Large Hadion Collider (LHC) among them, will produce multiple petaabytes (10^{15}) byte) of data per year. Some wide area networks now operate at 155 megabits per second (Mb/s), three orders of magnitude faster than the state-of-the-art 56 kilobits per second (Kb/s) that connected US supercomputer centers in 1985.

But to work with colleagues across the world on petabyte data sets, scientists now demand tens of gigabits per second (Gbit/s/s) [18].

What many term the *"Grid"* offers a potential means of surmounting these obstacles to progress. Built on the Internet and the World Wide Web, the Grid is a new class of infrastructure. By providing scalable, secure, high-performance mechanisms for discovering and negotiating access to remote resources, the Grid promises to make it possible for scientific collaborations to share resources on an unprecedented scale, and for geographically distributed groups to work together in ways that were previously impossible.

The concept of sharing distributed resources is not new. In 1965, MIT's Fernando Corbato' and the other designers of the Multics operating system envisioned a computer facility operating *"like a power company or water company"* and in their 1968 article "The Computer as a Communications Device", J. C. R. Licklider and Robert W. Taylor anticipated Grid-like scenarios. Since the late 1960's much work has been devoted to developing distributed systems, but with mixed success.

Now, however, a combination of technology trends and research advances makes it feasible to realize the Grid vision-to put in place a new international scientific infrastructure with tools that, together, can meet the challenging demands of 21 st-century science. Indeed, major science communities now accepts that Grid technology is important for their future. Numerous government-funded R&D projects are variously developing core technologies, deploying production Grids, and applying Grid technologies to challenging applications. (For a list of major Grid projects, see http://www.mcs.anl.gov/~foster/grid-projects.)

Bell and Gray [19] examined past trends in high performance computing and asserted likely future directions based on market forces. While many of the insights drawn from this perspective have merit and suggest elements governing likely future directions for HPC, there are a number of points put forth that they felt require further discussion and, in certain cases, suggest

alternative, more likely views. One area of concern relates to the nature and use of key terms to describe and distinguish among classes of high end computing systems, in particular the authors use of cluster to relate to essentially all parallel computers derived through the integration of replicated components. While arguable and supported by some elements of our community, fails to provide the essential semantic discrimination critical to the effectiveness of descriptive terms as tools in managing the conceptual space of consideration. In [20,21], they presented perspective that retains the descriptive richness while providing a unifying framework. A second area of discourse that calls for additional commentary is the likely future path of system evolution that will lead to effective and affordable petaflops scale computing including the future role of computer centers as facilities for supporting high performance computing environments.

Reif and Yoshida [22] had described and investigated an optical system which they call an optical expander. An optical expander electro optically expands and optical Boolean pattern encoded in d bits into an optical pattern of size N bits. Each expanded pattern is one of the N mutually orthogonal Boolean pattern. They wish the expansion to be exponential so they have d=c[log (N)] for some constant c. An optical expander can be viewed as either an electro optical line decoder which converts d bits of optically encoded binary information to up to N unique optical outputs, or a digital beam deflector which deflects an input laser beam into one of N distinct directions with a control signal of d bits. They showed that an optical expander can not be constructed by using linear optical systems, so a non-linear optical filter must be used. They described two different architectures to implement an optical expander. One uses an optical matrix-vector multiplier and an array of N threshold devices. The other uses log N novel reflection/ transmission switching cells. They then analyzed these architectures in terms of size , energy requirement, and speed.

Their optical expander can help develop various applications in electro optical computing systems. In general, because of I/O constraints and the limited fan-in/fan-out of electrical circuits,

the conventional VLSI technology is not suitable for building a large line decoder. On the other hand, conventional acoustooptic beam deflectors are bulky and limited by capacity-speed product. They found that as an electro optical functional unit, the line decoder finds many applications in optical computing such as optical interconnects and optical memory. Thus, the design and development of optical expanders is vital to optical computing. Our optical expander utilizes high speed and high space bandwidth product connections provided by optical beams in a volume, so it offers fast and accurate operations. Holographic memory system and message routing systems are potential applications of the optical expanders.

I. 6. Optical Computing and Optical Signal Processing

Computing is pervasive in almost all modern scientific endeavors, but nowhere is the need for computing power and efficient computational techniques more apparent than in the optics and imaging fields. In fact, we've known the fundamental laws governing the behavior of electromagnetic waves, Maxwell's equations, since the mid 1800s, but they are difficult to solve under conditions that deviate from the ideal.

Until the introduction of modern computers and computational techniques, solutions to Maxwell's equations under nonideal conditions were intractable. Researchers had to use assumptions and restrictive boundary conditions to make solutions possible. Although these simplifications-scalar diffraction theory being the best example-have elucidated a variety of physical phenomena, they are, nonetheless, based on assumptions that do not always hold true. Modern computational power makes the search for vector solutions to Maxwell's equations under nonideal situations more tractable, hence, researchers can study light's behavior in situations that previously were simply not amenable to computation.

I. 7. Optics and computing

A better understanding of the nature of light is just one example of where the marriage of computing and optics works well. Increases in computational power also have had a large impact in the optical engineering field. Since the late 1970s, for example, researchers have used computed tomography to image the human body. Traditionally, such systems use high-energy photons like x-rays and gamma rays because the physics characterizing this radiation's interactions with matter is well understood. Recent advances in the field of optical tomography, however, have led to tomographic imaging systems that use the near visible spectrum to scan the object being imaged. These advances would not have been possible without modern computer technology.

Recent computer advances have enhanced imaging on both the small scale (such as electron microscopy) and the large scale (such as astronomy). Modern telescopes use a vast amount of computing power just to keep systems running. Some telescopes use deformable mirrors that correct for the negative effects of the Earth's atmosphere. These systems must measure atmospheric distortion and deform the secondary mirror in such a way as to correct for the effect. All of this must be done rapidly because the atmosphere is turbulent and ever changing, a problem that represents an enormous computation burden best solved via efficient, parallel computing techniques.

Finally one of the most visible areas of computing in optics is that of lens design. Numerous techniques and software packages enable the design of multiple-element optical systems that correct for aberrations and other negative effects while retaining acceptable resolution and sensitivity properties. Computers can measure optical properties of lens systems efficiently and examine many different lens designs to best enable a researcher to build better optical systems.

Optical computing and optical signal processing is a mainstream field of research of present days. One should ask, why since it was already promising 30 years ago. The answer lays in three reasons:

1) The semiconductor based microelectronics developed so fast that other technologies were not able to keep pace with it (the speed of processing and the integration density of switching elements is increased threefold in every two year, according to More's Law). But this process is approaching its limits,
2) Thanks to the progress made in the technologies of key devices of optical information processing, optical computers are getting matured enough to help solving electronic computer's bottlenecks, and
3) New computing paradigms were formulated. The essentially parallel opto-electronic computer structures can serve (solve) these paradigms better.

The limits of the processing speed up thanks to the fast progress of microelectronic technologies are summarized (The Moore's law presumably is coming to an end within a decade).

1) Size limit: smallest feature size for lithography has a bottom limit, the number of electrons will be too small for "noise-free switching" in a small volume,
2) Heat dissipation at high density hinders further integration: more dense packing causes heat catastrophe, and
3) Interconnections are limited-planar layer, interconnects between layers-because of the lack of space (surface and volume is occupied by passive, active circuit elements)

That is why new alternative technologies, solutions and principles are sought.

I. 8. Advantages of Optical Computing (OC)

Optical computing has several important and decisive advantages over existing and future electronic computing methods and it is possible to implement the well developed electronic paradigms an principles, among them in general neural computing and especially Cellular Neural/Nonlinear Network (CNN) computing has an eminent position.

The main features of optical computing are:

1) **High degree of parallelism** enabling us to process and program flows (streams): a single instruction or command applies not to a byte or a word but to a whole frame (containing 10^6- 10^7 byte data). Simple optical architectures can perform in a single step 2D Fourier or other integral transformation on a frame. In two steps complex image (or matrix) operations, e.g., correlations (pattern-recognition and classification) can be executed,
2) **High switching rate (frame rate):** presently 1 μs switching rate (1 MHz frame rate), in the near future 1 ns (GHz frame rate), physical limits suggest, that later even pico-second (TeraHz frame rate) will be achievable,
3) **High overall processing speed:** now 10 TerraFlops in the near future 10^{15} bytes/s, later 10^{19} operations per second on bytes will be performed by optical processors (a consequence of 1 and 2),
4) **Freedom and flexibility in interconnectivity:** free space global interconnects (perfect shuffle (crossover), Banyan, crossbar), planar, mixed interconnects can be realized,
5) **Optical storage of huge amount of data is possible:** with high density. Rapid access is possible and divers access schemes (analog/digital, bit-wise, image-wise, serial access of whole frames, random access, associative) have been elaborated.

   In diverse holographic forms storage density of 10^{8} bit/cm^3 can be realized. This huge amount of total storage capacity seems to be reachable, with 10 ns frame access time (what is architecture dependent). A great store of holographic materials is being developed. Its parallel versions fit well to the parallel nature of OC, and
6) **Optical processing is extremely versatile and flexible:** because it can be analog, digital, and hybrid analog/digital, hybrid optical/electronic (photonic), all possessing the advantages mentioned above. Optical processing is directly applicable for matrix.

Optical computing and optical interconnections require high speed opto-electronics devices for data processing [9-13]. These devices (laser sources and laser detectors) must after high bandwidth, small rise time and very low sensitivities of variations.

The optical solution of **interconnect** for high-speed computing systems requires the high quality of Vertical–Cavity Surface Emitting Lasers (VCSEL's).

The Programmable Opto-Electronic Administration Analog CNN *(*POAC*)* computer architecture has been proposed that combines optical preprocessing opto-electronic CNN implementations. Special efforts have to be made for noise reduction for discrimination enhancement. They would introduce optical and detection nonlinearities and apply suitable adaptive thresholds. Other architectures such as the dual axis Joint Transform Correlator (JTC) and VLC with online, spatial domain template input will be used with higher resolution holographic media. Further improvements and application of these techniques can ensure the realization of feedback, which is essential part of the CNN paradigm [23].

The demand for fast identification and tracking of targets e.g. in surveillance systems has been increased dramatically during the last few years. In several other image-processing takes the quick recognition of particular structures is also important.

The very fast, online pre-and post-processing of the flow of image data is inescapable. Optical information processing systems can provide appropriate speed to solve these demands. However, the so far published optical processing system architectures do not seem to be flexible enough to be applicable in different computational tasks.

In recent years, several studies have demonstrated that a CNN type architecture, provides exhaustive programming frame for several complex, image processing tasks. The CNN Universal Machine (CNN-UM) is a massively parallel nonlinear array processor. While several emulated digital and mixed-signal analog implementations are emerging, there have been only a few

attempts to build optical or optoelectronic implementation of the CNN-UM. Optical correlators, however, can implement one of the basic operations of CNN computation: the convolution.

The POAC computer framework consisting of an optically implemented CNN combined with fast and re-programmable optical input VLSI CNN-UM chips present an ideal solution for the above outlined problems. In the optical implementation a large number of templates can be stored and retrieved. In this framework a considerable, in some case the dominant part of the processing can be done at the speed of light, and the rest of the processing on a fast parallel opto-electric device.

A special hybrid opto-electronic CNN computer architecture: was proposed where implementation of *POAC* embedding CNN Universal Chips. They should introduce and analyze this new type of feed forward only optical CNN implementation.

The *JTC* is a powerful optical information processing unit for pattern identification for optical CNN. It shows increased robustness comparing to a matched filter Vander Light Correlator *(VLC).*

I. 9. The Object of the Present Thesis

In this thesis, the design of an optical interconnect is handled, taking into account the following :-

i- The device speed through its 3-dB bandwidth ($f_{3\text{-dB}}$).

ii- Its ultimate bandwidth.

iii- Its available transmitted bit-rate based on soliton propagation.

iv- Its available transmitted bit-rate based on Maximum Division Multiplexing (MTDM) propagation.

I. 10. Organization of the Thesis

The present thesis is organized as follows :

Chapter I : handles a general introduction which includes a back ground, historical remarks supercomputing today, why optical interconnect, evaluation of supercomputing, optical computing and optical signal processing, optics and computing, advantages of optical computing, advantages and disadvantages of JTC realization plus the object of the thesis and its organization.

Chapter II : is a concise literature review on the subject of optical interconnection in high speed optical computing networks.

Chapter III : deals with the basic model, governing equations, and analysis of an optical interconnection, where special emphasis is focused on :

i- Its bandwidth.

ii- Its transmitted capacity.

iii- Its thermal and spectral width sensitivities.

Chapter IV : clarifies the obtained results based on the model of chapter III and a discussion to clarify the impact of the design and these results.

Chapter V : summarizes the major conclusion for the good interconnect performance and its good stability.

Finally the thesis is terminated by :

A- List of (64) references (textbook, magazines, journals, scientific papers, …etc).

B- Designed software for the good performance of the optical interconnect and its stability.

C- Glossary of the important terminology.

D- An Arabic summary.