DEPARTMENT SUMMARY

The Department of Computer Science has active research in several important areas of interest to Department of the Navy. A description of the major areas follows.

Software Engineering

Professor Luqi's research is to enable rapid prototyping of hard real-time systems via a Computer-Aided Prototyping System (CAPS). CAPS is based on a prototyping language with module specifications for modeling real-time systems and combining reusable software. These tools make it possible for prototypes to be designed quickly and executed to validate the requirements. The research focuses on automated methods for retrieving, adapting, and combining reusable components based on normalized module specifications; establishing feasibility of real-time constraints via scheduling algorithms; simulating unavailable components via algebraic specifications; automatically generating translators and real-time schedules for supporting execution; constructing a prototyping project database using derived mathematical models; providing automated design completion and error checking facilities in a designer interface; and establishing a convenient graphical interface for design and debugging.

Rapid prototyping is a means for stabilizing and validating the requirements for complex systems by helping customers visualize system behavior prior to detailed implementation, e.g., for embedded control systems with hard real-time constraints. CAPS supports an iterative prototyping process characterized by exploratory design and extensive prototype evolution. This should enable the first production version of the software to match user needs and reduce the need for expensive modifications after delivery. The current version of CAPS has been used to generate a software prototype of a C31 system with hard real-time constraints. The preliminary result of such an approach has shown great promise.

The major objective of Professor Shing's research is to develop efficient algorithms and tools to support the computer-aided rapid prototyping of real-time embedded systems. The process of design and development of real-time embedded systems is often plagued with uncertainty, ambiguity, and inconsistency. The timing requirements are difficult for the user to provide and for the analysts to determine. It is also very difficult to determine whether a delivered system meets its requirements. Rapid prototyping provides a means to alleviate the risks and difficulties in real-time embedded systems.

Specific topics Professor Shing is investigating include efficient heuristic scheduling algorithms for real-time systems, and incremental attribute-evaluation and software architectures for distributed real-time embedded systems.

Formal methods and associated automated decision aids have a large potential for practical impact that has not been fully realized. To help bring this about, Professor Berzins has developed a formal specification language specifically designed for large-scale applications that include parallel, distributed, and real-time systems. He is currently investigating a variety of methods and tools for partially automating many aspects of software development.

Professor Berzins seeks to develop fundamental theory and practical methods for combining several changes to a software system with mathematically provable guarantees of correctness. The main goal of this research effort is to enable a higher level of computer-aided design in development and maintenance of large software systems. Combining changes to software is a fundamental problem in software engineering. This process is important in all phases of developing large software systems, where multiple changes must be developed concurrently and then combined. This work has potential applications to software maintenance, view integration in specifications, version control in design databases, and multiple inheritance in specification or programming languages.

Professor Berzins has also investigated change merging for specifications and for software prototypes of real-time systems. He integrated a change merging mechanism for specifications with an inheritance mechanism and investigated both applications and formal properties of the resulting structure. He has been working on an analog of the program slicing method for the PSDL language. PSDL presents new problems because it includes explicit real-time constraints and parallel operations. He has also designed an automated design management and job assignment system. The main advance provided by this system is automated scheduling and job assignment for teams of engineers in an environment where plans are uncertain, partially known, and subject to change while the work is in progress.

Professor Shimeall's interests center around a consistent view of the analysis of software. During the development of software there are a variety of properties that must be applied and validated to the software, including correctness, safety, security, and modifiability. While these properties are very diverse, there are a set of approaches that are shared in the validation of these processes.

Professor Shimeall's research has identified three basic approaches to the analysis of software with respect to the above properties: similarity, individuality and source-affinity. Tools have been constructed to apply these approaches to software: Similarity: Reacher - reachability condition analyzer; Falter - activation condition annotator; Spacer -propagation condition analyzer; viewer - user interface to the similarity toolset; Individuality: ITIE - fault/event tree graphical editor; FT2PN - fault tree/Petri net converter; PN2FT - Petri net/fault tree converter; ACT'T - Ada/fault tree converter; Source-affinity: Tparse/An – variable-initialization analyzer; AAPSLC - Ada physical source line counter. Taken together, these tools provide a flexible framework for addressing the analysis of a variety of software properties, and they have been applied to several systems to examine correctness, safety and modifiability concerns. Work in progress includes extending the functionality of these tools, adding additional tools to automate other analysis phases and applying these tools to security and other software properties.

Programming Languages and Foundations

Programming languages play a fundamental role in computing. Well-designed languages are important to a wide variety of areas, including software engineering, modeling and simulation, databases, security, and distributed systems. There are few areas of applied computer science that do not benefit directly from a programming language in some way. Advances in the theory and implementation of programming languages therefore have a very broad impact on applied computer science.

To reason about a system whose operating software is written in some programming language, one needs the ability to reason about its programs. For this purpose, we need a formal semantics for the language and perhaps a programming logic, proven sound with respect to that semantics. Traditional programming logics have gone largely unused in practice because of the effort required to formulate the proper correctness criterion and then to prove it. However, there are "light" logics that are domain specific. Their aim is to make proving specific properties of programs easier. Such properties include reasoning about synchronization behavior among processes on a parallel machine, program termination, memory leaks, and even leaks of classified information. Light logics are more likely to be decidable, meaning a machine can answer, for a given piece of code, whether the code has a desired property.

Part of the Advanced Type Systems in Computing Project at NPS is concerned with developing a light logic for privacy in programs. The logic allows one to reason about a program's ability not to leak sensitive information. The logics are decidable type systems. As such, algorithms exist for performing privacy type inference.

Another major thrust of the project is to identify the rudiments of a programming language. This requires formulating appropriate security and safety properties so that one can prove, with respect to a formal semantics, that certain programs do not violate these properties. New semantic techniques have been developed for this purpose.

The programming languages and foundations effort at NPS is headed by Associate Professor Dennis Volpano.

Computer Graphics and Visualization

The NPSNET Research Group has a decade of experience in developing the software, interaction and networking technology for the large-scale virtual environment (LSVE). The current group (Michael Zyda, Don Brutzman, Rudy Darken, Robert McGhee, John Falby, Eric Bachmann, Kent Watsen and Russell Storms), is focusing on the LSVE network software architecture, web-based interoperability, cross-platform VE toolkits, 3D VE construction, inertial motion tracking, locomotion devices, human modeling in the VE, spatial sound, wayfinding in the VE, and DoD applications of VE technology.

The focus of the NPS Research Group is on the complete breadth of human-computer interaction and software technology for implementing large scale virtual environments (LSVEs). The research is applied to constructing virtual environments useful for the Department of Defense.

The NPSNET Research Group has three main branches, Technology, Interaction, and Applications. The Techno branch focuses on developing the network and software technology for the LSVE. The Interact branch focuses on human-computer interaction technology for the LSVE and on the evaluation of the LSVE for training. The Apps branch focuses on the development of LSVEs useful for the Department of Defense, utilizing the technology developed by the other two branches.

The NPSNET Research Group is a group of faculty (headed by Professors Zyda and Pratt), staff, and students that work in all areas of networked virtual environments. The research group is currently focused on the following virtual environment

(YE) research topics: the large-scale networking of virtual environments (environments greater than 1,000 players), YE network applications protocols, rapidly reconfigurable VE network protocols, Distributed Interactive Simulation (DIS) and High-Level Architecture (HLA) protocols, the real-time walkthrough of large-scale networked VEs, world modeling software for managing large scale networked VEs, the instrumentation of the human body and its representation in the networked YE, hypermedia integration (how we place video, audio, imagery and textual data in the networked YE), and geometric modeling (terrain, building and other c5ject modeling).

The NPSNET Research Group's efforts focus on the development of the above software areas and the integration of proven components of that work into a core software system, NPSNET. NPSNET is currently capable of simulating articulated humans, and ground, air and sea-going vessels in the DIS networked virtual environment. NPSNET can support about 250-300 players using currently available networking and workstation technology. NPSNET is the first 3D virtual environment that is capable of playing across the multicast backbone (MB ONE) of the Internet.

Real-time (3D) computer graphics workstations have progressed to a point where they can be used for Out The Window (OTW) visual simulation systems. The key to the development of these systems is the underlying software. As with most leading edge technologies, the construction of the required software is a black art. As such, the focus of the research has been the development, documentation, and distribution of workstation based OTW visual display systems. A major component of this research is the distribution of the simulation across the network. This allows multiple users to interact with each other in a virtual environment. This is one of the key premises of DIS, inserting humans into the virtual environment where they can "free-play" different scenarios. While this sounds simple, the problems of networks, human/computer interfaces, and data management are significant research topics. While the humans in the virtual environment comprise an integral component of the system, there are not enough manned simulators to sufficiently populate the world.

To provide the additional entities, Professor Pratt is conducting research in the use of traditional constructive combat models and autonomous agents to populate the world. By providing an interface to the traditional models, the work that has been done before in combat modeling can be leveraged. This, combined with the research on autonomous agents, provides a mechanism to provide friendly and opposing forces to complement the manned simulators.

While virtual environments *(YE)* are gaining widespread notoriety as a training tool and general interface to information spaces, little research has been done to show that this technology is in fact useful for training or that it represents an improved interface over conventional techniques.

One of the fundamental obstacles involved with large information spaces and most VE simulators is that of user disorientation. Users cannot navigate a virtual space as well as they can a real space. Professor Darken's interests in this problem come from two different perspectives: (1) navigation aids to improve users' performance on navigation tasks in VEs, and (2) navigation training aids to improve users' performance on real world navigation tasks trained in the YE. Another aspect of this problem has to do with locomotion techniques available to users. Currently, there is no way for a user to walk naturally in a large YE. There are encumbering cables and limited space in which the user must operate. We have recently evaluated a device (the Omni-Directional Treadmill) that attempts to overcome this limitation.

It has recently been shown that we receive more information aurally than we are consciously aware of. In fact, it is believed that sound has a large impact on training effectiveness in VEs. Professor Darken is beginning a research program to investigate this issue within the context of air training. In this, "the information age," it is somewhat surprising that we allow ourselves to be chained to a desk, staring at a monitor and typing on a keyboard. Professor Darken believes usability in computing is not just a matter of how we do things with computers but where we do things with computers. The objective here is to develop a wireless mobile computing environment where information is where users need it when they need it. There is a strong overlap between this and *YE* research as what we- learn about usable interfaces to virtual spaces can be applied to building usable interfaces to real spaces such as ship compartments, buildings, towns, and the battlefield.

One of the key components of a military virtual environment is the geometric description of the terrain database. To address the importance of the terrain database, NPSNET has active research projects in the areas of terrain modifications, culling, and polygon reduction. NPS is quickly becoming one of the leading organizations within DoD for the understanding and conversion of terrain database formats.

Research for the development of highly realistic tactical battlefield simulation systems by Professor Baer is organized around three main areas. These are: (1) development of algorithms for visualizing realistic battlefield effects including 1 meter or better terrain backgrounds, photo realistic targets, and environmental effects and battlefield; obscurants, (2) development of database generation and update systems designed to reduce instrumentation, photographic, and video data to object-descriptor data bases, and (3) development of high speed low cost parallel-processor technologies in order to execute the algorithms and systems resulting from the two previous research areas.

Artificial Intelligence and Robotics

The major objective of Professor Kanayama's research is to investigate fundamental theories in autonomous mobile robotics. Since autonomous self-contained robots have complete freedom in motion, the topic is fundamentally interesting in robotics and M. Furthermore, there are tremendous opportunities for such robots to perform practical tasks in the real world. Professor Kanayama tests and evaluates theories on the Autonomous Indoor Mobile Robot Yamabico-11. Research activities cover abstract mathematical models, intelligent algorithm finding, software development, and hardware construction. Specific topics Professor Kanayama has been investigating include: a smooth vehicle tracking algorithm, sonar interpretation, rigid body motion planning, automated cartography, vehicle control by a steering function, a high-level mobile robot language MML, a real-time hardware/software architecture for mobile robots, motion planning for an autonomous underwater vehicle, and fast gait planning for an underwater walking robot.

The role of robotics in manufacturing is already well established and is an important factor in increasing industrial productivity. In contrast, the introduction of robots into military operations has just begun. This is due in part to justifiable concerns about removing human control from potentially lethal systems, but perhaps to a greater extent is the result of the relatively weak capabilities of mobile robots operating in the unstructured or even hostile environments typical of military situations. Professor McGhee's research is concerned with adapting existing robot technology to suit military applications, and with making theoretical and engineering advances in areas where current knowledge is inadequate for a selected application. He pursues this goal both through the construction of prototype systems for concept demonstration, and by means of real-time graphical dynamic simulation studies in support of such research.

With the recent rapid increase in interest in networked interactive simulations as an alternative to actual field exercises, the accurate modeling of vehicle dynamics needed for mobile robot design studies has taken on a new and larger urgency. At present, Professor McGhee's research is centered around the application of unmanned submersibles to mine countermeasures. Both swimming vehicles and walking vehicles are being considered for this purpose. In the first instance, the NPS Autonomous Underwater Vehicle (AUV) is available for experimental studies. In the second, various vehicles derived from his earlier research on terrestrial walking machines are being considered. Accurate simulation models are being used in both cases to permit not only concept evaluation, but also the development of real-time control software through "hardware in the loop" simulation studies.

The MARIE project of Professor Rowe seeks to build an information-retrieval system for large multimedia databases that exploits the contents of the multimedia. This requires image processing, but especially natural-language processing since descriptive captions are often associated with valuable multimedia data and are much faster to analyze than images. MARIE exploits a large technical lexicon and a trainable statistical parser using statistics on word senses, syntactically-grouped word-sense pairs, parse rules, and rhetorical heuristics. MARIE's image processing uses robust natural-image segmentation methods, together with a neural network for classifying regions; the network also exploits linguistic reference information. MARIE also addresses system-building issues for large multimedia databases by considering the problem as one of efficient information filtering of desired data. Professor Rowe has developed mathematical criteria for optimal such information filtering, including data-par~llel implementations.

The METUTOR project of Professor Rowe helps teachers write and run tutors for tasks involving sequences of actions. With METUTOR, tutors are considerably easier to build than with conventional frame-based tools, while at the same time being smarter in analyzing student behavior. METUTOR tutors use planning methods of artificial intelligence to figure out what a student is trying to do, which gives powerful inference capabilities for finding student errors, categorizing them, and tutoring them. METUTOR permits mapping of domain concepts to graphical elements, which are then combined into a visual display of domain state.

Professor Rowe also conducts research in construction of universal (all-situations) plans for robots moving in natural domains. These are like potential fields but better: They give the provably best thing to do in any situation.

Parallel, Distributed, and Network Computing

Designers of high performance computing systems are increasingly turning to parallel processor systems to achieve high speed at relatively low cost. In such a system, processing elements are duplicated (numbering in the hundreds or thousands) and often the memory is distributed to support parallelism. While the hardware has been rapidly improving, the software for such machines is still archaic. There is no agreement on how to program these machines, nor on the best way to design a

parallel programming language, compiler, operating system, or application program. Thus, the central focus of Professor Lewis' research has been in the design and development of parallel programming environments which support architecture-independent parallel programming.

Recently, in a joint research project between NPS and the Russian Academy of Sciences Systems Programming Institute in Moscow, Russia, Professor Lewis is designing and implementing a parallel programming environment called mpC. This is a language, translator, and scheduling system for automatically mapping a parallel program written in mpC onto an arbitrary parallel computer. The network features of mpC permit rather general expression of the solution to a problem in terms convenient to the programmer. But this may be non-optimal use of the parallel hardware. Therefore, scheduling tools are needed to optimize the performance on a specific machine.

The hardware necessary to create a meta-computer has existed for twenty years. Likewise, heterogeneous programs-consisting of multiple parts, each running in a predefined sequence and on potentially different machines-have been around since the Sixties; these programs have been very latency-tolerant, meaning that the results from one part of a program often had to be written to tape and hand carried to another computer where the next part would execute. The demands of today's environments, particularly that of the military world, can no longer tolerate these latencies. In a crisis situation, where data acquisition, transfer, computation, and display must happen within minutes to be of any use, traditional methods have become outdated and, indeed, dangerous. At a basic level, a meta-computer is a distributed and heterogeneous collection of computers networked together, all coordinated by one or more master schedulers. Such a meta-computer can potentially, and in a matter of minutes, process a request from a hot spot, acquire the data from satellite, compute in a secure facility in the U.S., and present the results to the field commander somewhere in the Middle East. In addition, the meta-computer might be processing hundreds of such requests simultaneously.

Professor Kidd's research generally involves the architectural design of such a meta-computer. To this end, he has created the Heterogeneous Processing Testbed (HPT) in the Heterogeneous Processing Laboratory (HPL). The HPL supports research in meta-computing and heterogeneous processing by providing a fully controlled environment for testing, development and evaluation.

In a distributed and heterogeneous environment, two of the biggest issues are where and when to schedule jobs. Though seemingly settled for "single box" architectures, it is far from solved in a distributed environment, especially one which combines many "boxes" of highly varying architectures. Some of the issues Professor Kidd's research attempts to address are: (1) what optimization criteria should we use, (2) how do you schedule jobs to meet various constraints on when and where they are to execute, such as job A must execute in parallel with job B and before job C, (3) how do you avoid resource contention related to the sharing of resources, such as disks, CPUs and networks, by different jobs, and (4) how do you meet the optimization criteria.

One ingrained assumption underlying Computer Science is determinism. Though never completely true, it has been "good enough" in traditional architectures. As the development of distributed computer evolves, determinism is no longer a valid assumption/simplification. Uncertainties derived from various sources, such as network traffic, shared disk use, and cache use, make job runtimes and other time related events better represented by a probability distribution. To this end, Professor Kidd and Professor Hengsen's research applies probabilistic and statistical techniques in scheduling and learning.

Professor Hensgen is one of the Principal Investigators for the Management System for Heterogeneous Networks (MS HN) Project, sponsored by DARPA under its Quorum Project. This research for MSHN is conducted by several faculty members here at NPS (Hensgen, Kidd, and Irvine), staff and students here at NPS, as well as researchers at NRaD, Purdue, and the University of Southern California. The goal of MSHN is to deliver good end-to-end quality of service to users in environments where both the heterogeneous resources and user set are dynamically changing. MSHN will serve a mixture of applications ranging from compute-intensive to I/O-intensive to interactive and real-time. Professor Hensgen's main area of research within the MSHN project is prioritized management of the multiple shared resources. In particular, it is extremely important to carefully model various classes of shared computing resources which differ greatly from other resources which we typically share.

In addition to the MSHN project, Professor Hensgen and some of her students participate in NRaD's SmartNet project, a scheduling advisor for heterogeneous computing resources. SmartNet has been used within DoD for compute-intensive jobs and the NPS team is currently investigating something similar for communication-intensive jobs.

With her students, Professor Hensgen has also built Graze, a graphical, performance debugger for parallel computing and Concurra, a software system that generates multi-threaded applications whose concurrency is provably correct and which are free from deadlock. One of her current students is investigating the use of Graze to monitor Java programs as well

as its use in determining when certain compiler directives should be used with automatically parallelizing compilers. This will be useful to applications being written for FLEETNUMERIC and the Pacific Disaster Center.

Professor Lundy's research interests are in telecommunications networks and computer networks generally. Most of this work has been in the specification, analysis and testing of communications protocols. This often leads him to suggestions for possible improvements of existing protocols, correcting errors in them, or both. Recent work is in high speed transport protocols, multicasting protocols for reliable communications, and wireless protocols. Professor Lundy is also interested in military communications and in network security. In the past year, Professor Lundy has been studying and evaluating some of the U.S. Army's current networks and their future plans for these networks.

Network guarantees of quality of services required for the transport of multimedia data such as digital audio and video. Specifically, end-to-end transfer delays and loss rate of a multimedia data flow must be bounded below specified values. The objective of Professor Xie's research is to design networks that provide these guarantees. During the past two years, Professor Xie has developed a new network architecture (called Burst Scheduling network), and a set of algorithms for providing end-to-end delay guarantees. The algorithms are highly efficient, suitable for high speed implementation.

Professor Xie's current focus is on the development of an application-level guaranteed statistical service. The service is characterized by: (1) a bound on loss rate of application-level data units (e.g., pictures in a video application), (2) data losses distributed fairly among flows subscribing to the service and uniformly over the duration of each flow, and (3) a deterministic traffic model that promotes statistical multiplexing in the network. More specifically, a flow is modeled as a sequence of bursts, each of which carries the bits of an application-level data unit (ADU). The first and last packet of each burst are marked, and the first packet carries information on the ADU (including the bandwidth requirement). The traffic model enables admission control at the burst level as well as the flow level. To achieve high network utilization, overbooking is allowed in flow level admission control. Burst level admission control, on the other hand, is used to ensure that the capacity of each network channel is not exceeded by bandwidths allocated to flows such that delay guarantees can be provided. The guaranteed statistical service will greatly enhance the ability of existing networks to support distributed multimedia applications such as remote teaching and video on demand.

Computer Security

As the value of the assets stored in computer systems increases, attacks by highly motivated, technically capable opponents using malicious software and subversive techniques become more likely. A scientific foundation exists which may be employed to build secure computer systems and certified software to protect sensitive information. Several areas of research are being pursued.

The widespread use of commercial off-the-shelf (COTS) platforms enforcing security policies with a high level of assurance has been hampered by a lack of compatibility with existing COTS and government-off-the-shelf applications software. Trusted file system research is intended to permit the use of both high assurance security policy-enforcing platforms and the massive body of application software currently available. Professor Irvine's government-industry team plans to build a prototype system providing high assurance controlled sharing of information while allowing users to continue to run their favorite COTS workstation applications. An extension to the initial effort would be the development of a high assurance messaging capability permitting the selection of cryptographic keys and methods based on information security levels.

The use of covert techniques to export sensitive information from trusted systems is also being explored by Professor Irvine. This research will include the analysis of encoding techniques for exploitive purposes as well as empirical studies of the efficacy of these covert techniques.

Databases

Professor Wu's main research interest is in creating a unified database front-end system that provides easy-to-use yet powerful common language to access varying types of relational data management system (RDBMS), and shields the complexity of underlying RDBMS. His system called GLAD II (Graphics Language for Accessing Database) allows users to interact with different relational DBMS by providing a common graphic language called DFQL (Data Flow Query Language) that is based on a data flow diagram. The system automatically translates a user specified DFQL query into the

equivalent query statements recognized by the connected backend DBMS. At present, the prototype connects to Oracle and translates the DFQL queries into the Oracle's SQL statements.

Professor Wu's work is different from other similar-looking work and commercial products in that theirs only support a SQL connectivity, i.e., theirs do not shield users from the complexity of SQL. In contrast, DFQL provides a more logical, higher-level, and consistent query language. Users of DFQL do not have to bother with the poorly designed language features in SQL. In other words, theirs do not eliminate the semantics problem associated with data retrieval. DFQL is a graphic query language based on relational algebra. It has been designed with sufficient expressive power and functionality to allow the user to easily express database queries. DFQL is relationally complete and includes an implementation of aggregate functions. An object-oriented implementation allows programmers to easily create their own DFQL operators from the primitive and other existing user-defined operations. This extensibility of query language is unique, no other query language allows such extensibility. The overall intention is to provide the user with a simple-to-use, yet powerful and extensible tool to implement database queries. A human-factors analysis comparing DFQL and SQL showed DFQL was statistically better than SQL.

139