CMPE 540 PRINCIPLES OF AI

TERM PAPER

Artificial Intelligence Applications In Military : A Survey

by

M. Toygar KARADENIZ

Bogaziçi University

1996

TABLE OF CONTENTS

ACKNOWLEDGEMENTS

During the years, since we, as a group of university students, started our education in computer science, the concepts about *AI*, has grown from a relatively small-scale research activity into a technological and industrial success consisting of machine learning, neural networks, intelligent control. Basic research on this topics, has expanded enormously being a result of development of faster computers, recently. The goal of this particular survey paper was to provide a little insight into the research area of AI in military. The paper consists of both information about what the level of development in field of military AI is and what the most recent applications and research efforts are.

This paper is mainly divided into two parts :

(a) state-of-art
(b) a detailed application

Part one, gives information about the ways the AI is being used in today's military. Here, several applications are mentioned but no very detailed information is given.

Part two is mainly about a specific application in the military field. This part gives detailed information about that information and shows the cases it is useful at. So, this part makes the reader considerably familiar with new concepts of recent developments.

This paper contains, spread throughout it, many references to the AI research literature. These references are important for two reasons. First, they make it possible for the reader to pursue individual concerned studies in greater depth than is possible within the space restrictions of this report. The second reason is that the set of these references are the main support to the content of this report.

This report would not have happened without the help of many people. The content has been greatly improved by the comments of Associate Professor of Computer Science Levent AKIN who is our instructor and our greatest help factor, in the first place.

Special thanks to Sevim OZUTKU and Okan KARAKUS for help and beyond. Sevim kept the survey going under desperate circumstances, and Okan spent innumerable hours increasing the morale and decreasing the stress. we thank them for these things and much, much more.

Bogazici University Library and Internet Connection, provided us the environment in which we could do the actual survey and write this paper. We are thankful to all instructors and personnel for putting up us during the survey process of this paper.

M.Toygar KARADENIZ

ABSTRACT

Project Name **:** Artificial Intelligence Applications In Military : A Survey

Project Team **:** M. Toygar KARADENIZ

Term **:** 1996/1997 1.Semester

Keywords **:** military, artificial intelligence, NAUTILIUS, TINSEL, Eucalyptus, natural language, KOALAS, TERSE, NCARAI, CATA, IFF, CASREPS, InterLACE, MACPlan

Summary **:**

This paper is a survey about current artificial intelligence applications in the field of military. It mainly consists of two parts. In the first part, several mentioned type AI applications developed, are mentioned with a brief description telling what the study is going on about. No further detailed information about these applications, is given and interested reader is invited to the references section. In the second part, a specific application is taken and examined in full detail. This application is a natural language recognition system in fact and interesting enough to be the subject of this survey's second part. More information will be given in the full report.

1. INTRODUCTION

Artificial Intelligence is becoming a more and more common concept in today’s world of science. In nearly every field, an application containing AI can be found to exist. This particular survey aims to investigate the AI applications in military field. Our purpose is to give an clear idea about the state of art in this field in the following sections. Also the potential future improvements on the mentioned applications will be examined when possible.

This paper mainly consists of two parts. In the first part, several AI applications developed in the field of military, are mentioned with a brief description telling what the study is going on about. No further detailed information about the mentioned applications, is given and interested reader is invited to the references section. In the second part, an application is taken and examined in full detail. This application is a natural language recognition system in fact and interesting enough to be the subject of this survey’s second part. More information will be given in the mentioned sections.

As the last word in the introductory part, we, as the survey team, want to thank to you because of your interest in the artificial intelligence subjects and because of your potential patience in reading this paper.

2. STATE-OF-ART AI APPLICATIONS IN MILITARY

As mentioned before, in this survey, we are aiming to give an idea of the current applications of AI in military field and then, examine one of these in detail. In this part, information about many applications is to be given. This will make our idea about the state-of-art in military AI systems, clear.

2.1 Applications On NL

At this point, it may be nice to give some idea about the particular AI application in the military field that we will be examining in detail in the following lines. This project is called ‘Eucalyptus’ and has been proposed and developed by Kenneth Wauchope [1], [2]. Eucalyptus is a natural language (NL) interface to the KOALAS E2 air combat computer simulation belonging to the Naval Air Systems Command. The research goal of this exploratory development project is to demonstrate the added flexibility made available when NL is integrated with an interactive system's graphical user interface. In the integrated interface the user can choose to perform a KOALAS interaction graphically, verbally, or by a combination of the two. The precision and specificity of the graphical interaction mode is complemented by the higher-level generality of NL, reducing both the tiring repetitiveness of the one and the possible ambiguities in the other. The NL discourse facility also maintains a contextual history that the system uses to resolve abbreviated or underspecific user inputs. This system runs on UNIX-based Sun SPARCstations. Eucalyptus consists of a commercial speech recognition front end, a general-purpose NL processor NAUTILUS written in Common Lisp, and a backend translation component. The use of declarative English grammars, lexicons and knowledge bases makes the NL processor fully portable across domains and applications. The

KOALAS application and graphical interface is written in C, and is available in both SunView and X Windows versions. As mentioned, this very interesting application, will be examined in detail in the following part of this survey.

Another research going on NL recognition by computers, is being done by Astrid Schmidt-Nielsen [3]. The objective, here, is to develop more effective interaction methods between users and complex high-performance computer-based systems, to develop a theoretical framework for human-computer dialog by extending current human verbal dialog models into the human-computer domain, by developing and validating natural-language based discourse models for use in visual as well as verbal interactions, by providing a theoretical basis for visualizing abstractions, by developing specification and design methodologies for human-computer dialogs, and by incorporating in the framework methods for processing multimodal human-computer dialog; and to develop dialog metrics by providing a theoretical basis for performance prediction.

As part of a larger research effort in Discourse for Human-Computer Dialog, the Intelligent M4 Systems group which is headed by Kenneth Wauchope, is conducting research in the linguistics (syntax, semantics and discourse pragmatics) of spatial relations as they might occur in verbal descriptions of routes and locative information when a user interacts with a computer-based graphical map display. As a testbed for this research we are using the cartographic database from the LACE combat simulation system developed by the Air Force's Rome Air Development Center. The database covers nearly 280,000 square kilometers of central and southern Germany and includes all major roads, towns, airstrips, waterways, and other objects of military concern such as powerlines and railroads, providing a rich real-world domain for spatial reference in a military map-based environment. To date, a fully pannable, zoomable, mouse-sensitive graphical map display for the LACE cartographic database is developed on Sun workstations, and interfaced our natural language processor NAUTILUS to provide NL database query capability with the option of deictic (combined NL and mouse) reference. A simulated tank unit also responds to verbal route instructions and directional mouse gestures. This base system, ‘InterLACE’ (Interface to LACE), is available to as a testbed for the groups’ research in complex route specification, hypotheticals, presupposition, the semantics and pragmatics of spatial prepositions, and natural language generation of verbal route descriptions.

The mentioned NAUTILUS (Navy AUTomated Intelligent Language Understanding System) is a natural language (NL) processing system written in Common LISP that has been under development at NCARAI since 1986. Its development is being done, to a great extend, by Kenneth Wauchope. The core component of NAUTILUS is the parser from the PROTEUS natural language understanding system developed at New York University's Courant Institute of Mathematical Sciences. The PROTEUS parser accepts an input English string, performs lexical lookup and syntactic analysis (using a context-free phrase structure grammar augmented by procedural restriction rules), and outputs a regularized form (called the translation) represented in a logical operator/operand language. PROTEUS is first used in the TERSE message processing project to analyze and dereference complex shipboard equipment names and as the syntactic analyzer in the InterFIS natural language interface project. At NCARAI the parser's functionality is extended in several areas and an X Windows graphical interface for more effective use in development mode is provided. The remaining modules that comprise NAUTILUS were all developed in-house at NCARAI. TINSEL is a first-pass semantic interpreter that enforces selection restrictions during parsing by matching PROTEUS translations against user-defined case frame patterns and semantic type co-occurrence constraints. It has been extensively tested in a number of domains including the RAINFORM and MUC-3 message sets. FOCAL is a focus-based reference resolution module developed by ONR fellow and MIT graduate student Gina-Anne Levow that resolve anaphoric references and handles quantified noun phrases in a set-theoretic manner. Finally, FUNTRAN uses the outputs of the preceding modules to construct a quantified logic expression suitable for evaluation (as LISP functions and macros) in an application environment. The complete NAUTILUS system was first used as the NL processor in the Eucalyptus project, and in the space of three months in 1994 was successfully transitioned to three new applications: InterLACE (NL interface to a cartographic database with graphical map display), a speech controller for VIEWER (an immersive or "virtual environment" tactical warfare simulation display system), and a prototype interactive electronic repair manual developed by Computer Sciences Corporation in Landover.

As another application in the field, Dennis Perzanowski and John Gurney, currently studying on extracting information in a discourse. This information can be obtained by analyzing several different linguistic properties of the discourse. Various syntactic and semantic triggers provide clues to the complex informational structure of a discourse. For the purposes of this investigation, several discourses were taken from a series of wire service messages dealing with terrorist incidents that occurred in Central America from 1989 to 1991. The corpus is known as the MUC-3 corpus. Because of their inferential properties, a specific syntactic construction, know as 'it-cleft constructions' were isolated from the corpus for investigation. Despite their rarity in the corpus under investigation, this work shows that it-clefts provide additional information, and do not just contrast or emphasize focused linguistic material, such as noun or prepositional phrases. Research to date in this area reveals that, as a subset of the information that can be obtained in a discourse, it-clefts can be used to avoid conflicting or differing interpretations inherent in syntactically related paraphrases, thereby minimizing possible confusion regarding the interpretation of what the author is trying to communicate. In some cases the speaker/writer's point of view or attitude about the subject matter being discussed is also revealed through the use of the it-cleft, whereas a more common syntactic paraphrase would be ambiguous regarding author's attitude.

At the Navy Center for Applied Research in Artificial Intelligence, Kenneth Wauchope and his research group have been working on a project to analyze the free text in CASREPS messages and formatting it into an application-neutral form suitable for inferencing, information retrieval and other AI tasks. CASREPS (Casualty Reports) are Navy messages reporting shipboard equipment failures and attempts at their diagnosis and repair. Although, consisting primarily of information formatted into fixed fields, these reports also include narrative English text describing and amplifying on the formatted portion of the message. A computational linguistic approach had been taken to do the CASREPS analysis. With this approach, each input sentence is subjected to a lexical lookup, parsed against a broad-coverage English grammar, tried through semantic constraints, syntactic regulation and mapped to a frame-like semantic representation. The application task that has been looked for in recent years is the automated highlighting of information in the narrative portion of CASREPS. First prototype system to implement this, was called SUMMARY and only capable of extracting only sentence-local causal and failure information. Hence, the system could recognize the lexical item ‘erratic’ as connoting a failure and could normalize the syntactic structure ‘X be due to Y’ into ‘Y cause X.’ SUMMARY could not make use of intersentential context in doing the job. The next version of the text extraction system is TERSE which includes a model of a piece of equipment that was the topic of that particular CASREP message set. TERSE, also, can analyze the complex prt names and descriptions, resolve multiple references to the same piece of equipment and confirm casual relations. The system was built in Intellicorp’s KEE system development shell and runs on a Symbolics Lisp machine and a Sun workstation.

2.2 Applications On Pattern Recognition

The first study in the field is being done by Behrooz Kamgar-Parsi [4] on the subject of target extraction in images. This images are to be taken as infrared ones and target extraction process is to be done automatically. A fundamental problem in computer vision and image processing is image segmentation or object extraction in intensity or infrared images. Current techniques require either user-supplied parameters or model-based templates. Examples of techniques requiring user-supplied parameters include thresholding and edge detection. They are convenient but they both require threshold values which in general cannot be calculated automatically. Model-based techniques may not require the supervision of a human operator, but they have other deficiencies:

- slowness
- crudeness

Thus, model-based techniques have a low degree of resolution, e.g. they may not distinguish between two types of aircraft. Target extraction is essential for many problems of interest both to the Navy and the commercial sector. A few examples are object recognition, automatic target recognition, scene analysis, etc. Hence, it is of great significance to have a dependable automated technique for target extraction. We have developed a technique for automatic target extraction in infrared images, and have applied it with noticeable success to extraction of aircraft from single images. Note that when the target is far from the camera, even if it is moving its motion is not easily discernible and motion detection algorithms (requiring successive frames) generally fail to produce satisfactory results.

Another project is going on the same point, again, developing a AI application to be used for army. LA Tamburino, a member of Wright Laboratory is studying on machine perception. The object of the machine perception program is to develop advanced processing systems based on adaptive learning and evolutionary computations. Research focuses on analyzing the high-resolution radar classification problems, conducting analytical and experimental studies on real-time computer networks for next generation avionics parallel processors, and studying two-dimensional mesh networks using techniques such as wormhole routing and virtual channels.

2.3 Applications On IFF

'Identification, Friend or Foe' systems are, maybe, the most well-known applications of AI into the field of military. R. Schneider developed an evaluation tool to assess IFF system performance in multiple scenarios ranging from one-on-one to full theater level engagements using statistical techniques. The model was designed to predict the performance of IFF systems in any user-defined scenario. The scenario may include platforms equipped with IFF interrogators/transponders, air traffic control radar beacon system (ATCRBS) interrogators/transponders, and hostile platforms including jammers. The model uses a set of rules to determine if an interrogation/response is successful. The model provides interrogation rates at transponders in a scenario. These rates are used to predict transponder reply efficiency. The interrogation rates and reply efficiencies are used to predict the downlink signal environment at selected interrogator receivers in the scenario. Interrogation rates and reply efficiencies are also used to predict interrogator receiver identification performance. It also includes the effects of jamming on the interrogator receiver, transponder receiver, and the primary sensor. This tool is mainly developed for DoD Electromagnetic Compatibility Analysis Center in support of the Mark XV Identification Friend From Foe System Program Office and NATO Identification System (NIS) Technical Working Group (NTWG). The used hardware and software to make this application run are,

- Computer : VAX/VMS V5.4-1
- Storage : Disk storage requirements vary depending on scenario size
- Peripherals : Printer
- Language : FORTRAN 77
- Document : ECAC-CR-91-045 report describes the engineering theory, functional. But, as the survey team, we can not reach to this report to give a more detailed information about this application, although it said to be unclassified.

As mentioned before, the IFF systems are very popular in the military field. There are many applications developed by using the same idea, but being used by different military members. For example, TRW/Magnavox group developed a combat identification program for Marine Corps. Its primary function is the same : To identify "friend or foe" for armor units in order to reduce wrong decisions. The system design is very like the one mentioned before. The only difference, here, is that Marine Corps is using it, not somebody else.

Dynamic Research Company (DRC) has developed and evaluated data fusion algorithms for target identification. In one application the data was derived from sensors aboard an aircraft, where the computer storage and processing capability is limited, but the need is for near real-time target identification. The standard Bayesian fusion algorithm was not appropriate since it required extensive, precise target information which may not be available, and moreover, it places high requirements on computer storage and processing. On the other end of the spectrum are Voting Schemes which are simple to implement, but they provide no confidence bounds associated with their decision. Since neither Voting Schemes or Bayesian fusion algorithms were ideal, DRC developed a new approach based on "Possibility Theory" which was almost as simple to implement as a Voting Scheme, yet provided confidence bounds for its target identification decisions. DRC, also developed a computer simulation to evaluate the various types of fusion algorithms, and in particular, to compare the effectiveness of the Possibility Theory algorithm versus other types of fusion algorithms. This simulation and other computer programs is used to evaluate the effectiveness of various target identification sensor combinations. The sensor suites included various types of radar derivatives and IFF (Identification-Friend-or-Foe) systems. Several Measures of Effectiveness (MOE) for the computer simulation evaluations including the Exchange Ratio, which is the ratio of hostile killed to friendlies killed, are defined.. This MOE takes both types of identification errors into account: the probability of shooting at a friendly and the probability of failing to shoot at a hostile. In a later application, this simulation has expanded to include other fusion algorithms such as Dempter-Shafer, Rule-Based Inference, and Longitudinal Entropy. A total of six different types of fusion algorithms were evaluated for use in a transportable, automated command and control system.

2.4 Applications On Neural Networks

One of the other projects in the field is to do with clustering. Both because of its theoretical significance and its wide variety of practical applications, clustering has attracted the attention of a large number of researchers over the past several decades. However, virtually all of the proposed clustering techniques attempt to partition the data set into a number of clusters whether or not the partitioning is meaningful. The only published clustering technique which appears to be capable of warning the user when the data set lacks sufficient structure (to be partitioned) is due to Behrooz Kamgar-Parsi [5]. The technique is based on the Hopfield model of neural networks. Work is underway to better understand the potentials of this clustering method.

Next study to consider in the field, is about multilayer neural networks and being studied by Behrooz Kamgar-Parsi. A well-known shortcoming of multilayer neural networks (as well as other machine vision or neural net techniques) is their inability to recognize that they have encountered an unfamiliar object. For example, a multilayer network trained on examples of aircraft and tanks will classify every object (or pattern) in the universe as one of those items, e.g. a tree or a house will also be (mis)classified as either a tank or an aircraft (depending on whether its missimilarity to tanks is less than to aircraft, or vice versa). In early pattern recognition applications of neural networks there was no real need for rejection because every pattern presented to the network would be one of the familiar patterns, e.g. in alphabetical character recognition each pattern would be one of the 26 characters. In an uncontrolled environment, however, many of the patterns which are presented to a network for classification may not belong to any of the classes used during training. For example, a receiver may be interested in the detection of only a few particular signals with certain characteristics, while it is subject to many (previously unseen) incoming signals of different types and attributes. Or, in the project of aircraft recognition when analyzing infrared images of the sky in front of the runway the question cannot simply be the type of the aircraft in the image, because a given frame may not contain any. Using fast image processing techniques, e.g. simple high intensity thresholding, some signal may be detected. The signal, however, may not necessarily be an aircraft as it could be clouds, other objects or just noise. Therefore, the first question is whether the signal that has been picked up is an aircraft. Note, that here we are not dealing with a limited classes of shapes, say (only) different types of aircraft, where neural networks have been traditionally used. This is because the signal can have any shape which amounts to an unlimited number of classes. Therefore, the problem is more difficult requiring rejection capabilities. So, in this project, a technique to have multilayer perceptrons with rejection capabilities, is developed by creating a random deformation technique capable of generating an arbitrarily large number of true and false look-alikes of a given class. The discriminating power of this technique (in visual pattern recognition) is comparable with that of the human eye.

2.5 Applications On Other Subjects

As an interesting application in the field, Alan C. Schultz [6],[7],[8],[9], developed a project named ‘Adaptive Testing of Intelligent Systems’. Traditional validation/verification techniques have difficulty testing the full capabilities of the large software systems that control autonomous vehicles. Current approaches requires a human expert to handicraft test scenarios, observe the systems reaction to those test scenarios, and then refine the scenarios in an effort to understand the weaknesses of the system. So, with this project Schultz, created a tool that can automate the process of testing complex controllers to determine if they might fail in unanticipated situations. The approach attempts to automates the what if role now played by human experts by using NRL advances in the field of machine learning, Given a simulation of the vehicle to be controlled by the complex software, this technique searches for fault scenarios of interest. A fault scenario is a description of faults to be injected into the vehicle given a particular state of the vehicle and its environment. By injecting these faults in the vehicle the testing system can see how the controller will cope in various situations. An automated evaluation function drives the system to find interesting fault scenarios, for example, scenarios where a minimal number of injected faults cause the controller to fail. The greatest advantage of this tool, is that it has no pre-notions of what might fail, and so it often finds situations that might not normally be considered by human test engineers. This is a general-purpose approach to testing intelligent systems, and is expected to contribute to the development of more robust autonomous air, land and underwater systems.

Boeing Defense & Space Group is developing a sophisticated technology that they say, that will change the way wars are fought. The work, under way for more than three years, will allow groups of manned and unmanned strike aircraft to enter combat under the management of an airborne mission controller or an "on-the-scene" pilot/crew member. Boeing, a recognized industry leader in advanced flight and mission management for both tactical and strategic systems, has been developing the technology, called Control Automation and Task Allocation (CATA), under a contract with U.S. Air Force Wright Laboratory at Wright Patterson Air Force Base, Ohio. CATA develops the control strategy for a composite multiship flight management system that will automate mission planning, rerouting, cooperative flight path management, targeting and threat evasion. It integrates flight, mission and battle-management software with computing and communications systems that can be used in a theater of operations. The control technology is a vital design component in the development and employment of unmanned agile strike aircraft such as unmanned fighters. The CATA technology enables a concept for a "strike package" that offers a six ship set of internetted combat vehicles for offensive and defensive operations. It can be tailored to accommodate diverse situations. For example, a combat package could comprise a composite set of manned and unmanned lethal vehicles, a composite set of manned vehicles, or all unmanned aircraft. The system will provide significant advantages to the warfighter through the automation of mission planning, rerouting, cooperative flight path management, targeting and threat evasion. The use of unmanned strike aircraft has advantages over manned strike aircraft. A key benefit will be better evasive maneuverability. CATA is designed so that a mission controller or "on-the-scene" pilot/operator can manage the flight paths of vehicles and allocate strike package assets such as sensors and weapons. From a remote aircraft, such as an AWACS, the mission controller can deploy a strike package that will work best for the mission. When aircraft are flown without a human in the cockpit, extra precautions must be made to assure the aircraft performs as planned. Boeing has met these concerns by integrating flight, mission, and battle management software, computing, and communications systems, vehicle and package integrity management strategies with the required human oversight, whether localized or remote, into one system that can be employed in a theater of operations. Flight testing could begin as soon as 2000, with possible deployment by the U.S. Air Force within five to 15 years.

A.H. Klopf, a member of Wright Laboratory, is currently doing a research on machine intelligence. This research emphasizes adaptive behavior, reinforcement learning, and control theoretic models of nervous system function, as approaches to transitioning the mechanisms of natural intelligence to machine intelligence. The focus is on learning mechanisms and hierarchical network architectures, which will yield robust forms of machine intelligence, in conjunction with arrays of sensors and effectors. Typical investigations involve synthesizing learning system designs (realized in software or hardware) that are tested by using on-line, real-time, closed-loop, goal-seeking interactions between the learning system and its environment.

Relevant applications include dynamic control problems requiring learning, trainable automatic pattern recognition systems, and high-level decision-making systems. Available computational resources consist of MacIntosh IICXs and Quadras, Sun workstations, VAX Station 3s, and a Cray XMP.

The Navy,always, needs autonomous robotic systems that can be deployed in remote and hazardous environments, and that can contribute toward the goal of reducing personnel requirements. Navy applications of autonomous systems include shipboard firefighting, hazardous material handling, surveillance, salvage, and undersea equipment maintenance. So, there is currently a project in Navy by Alan C. Schultz to contribute toward the design of future autonomous systems by extending recent results in the areas of machine perception, behavior-based control architectures, task planning, and machine learning, with a focus on innovative designs for integrating these functional capabilities. Long term goals include answers to fundamental questions such as:

- Can a robotic system learn how much perceptual detail is required for particular behaviors?
- Can it learn to adapt to unexpected changes (e.g., malfunctions) in its sensor/motor capacities?

Most recent research in intelligent mobile robots addresses relatively simple tasks such as following walls or pushing boxes, behaviors that do not require sophisticated interactions among perception, planning, control and learning. In contrast, Alan C. Schultz is particularly interested in more challenging classes of problems involving multiple robots in competitive or cooperative tasks, such as:

- Locating, tracking, evading and targeting another mobile robot in a cluttered environment
- Locating and retrieving hidden items, either in competition or in cooperation with other robots
- Performing material handling tasks while avoiding collisions with other agents performing similar tasks.

The ARPA IFOR/WISSARD project is another interesting project of the Advanced Research Projects Agency. This project is concerned with the development of intelligent computer-generated forces (IFORs) that behave realistically in real time in Advanced Distributed Simulation exercises involving many human and computer-generated agents participating in a simulated battlefield. The goal is the development of IFORs that behave realistically enough that they appear indistinguishable from human agents. The IFORs are being developed by a consortium of university researchers using the Soar system, providing an architecture for general intelligence. The IFORs simulate fixed-wing aircraft engaged in fighter combat against other aircraft and fixed- and rotary-wing aircraft engaged in close air support missions to aid ground forces.

Recently, Kirsten Y. Kissmeyer and Anne M. Tallant, studied on an AI application called MACPlan. MACPlan is a knowledge-based system that helps airlift planners to develop resource-effective airlift plans quickly. The airlift planner's task is rather hard as there is a lot of data involved in the process. This data contains information about cargo and passengers, aircraft, operators of aircraft, airfields and related timing and constraint factors. A planner's main task is to allocate and schedule resources to satisfy potentially thousands of movements. A movement requirement is some quantitiy of passangers or cargo that must be moved from one port to another in a specified time interval. The goal of airlift planning is to get a plan that provides alternative ways to deliver its requirements on time. Flexibility is important because, airfields can be closed and aircraft can fail. An airlift plan consist of some guidelines like the following :

- Use airfields A and B for refueling and reassignment of military aircraft after mission completion.
- Disallow civilian aircraft at airfield C.
- Use civillian aircraft for the bulk of passanger loads.
- From the first to the ninth day of the plan, use three 747-100s from operator A, and increase the amount to four from day 10 to the end of the plan.

MACPlan bases its model of the airlift-planning domain on the Flavors, oject-oriented extension to Lisp. Flavors can be used to represent an extensive amount of declerative knowledge and its natural organization of classes allows many relationships to be derived by inheritance. Its predicates are hand-coded Lisp functions that return two values, either true or false and a justification for the veracity. Constraints and rules contain declerative plan-element information. Predicates are only called by constraints and rules but are themselves independent of the plan-element representation. Forward chaining comprises one of the major mechanisims by which MACPlan reasons. This program, mainly, runs as a stand-alone, multi-process, dual-screen system an a Symbolics 3600 Lisp machine. It is written in Lisp using Symbolics' object-oriented extension Flavors.

3. A DETAILED APPLICATION

3.1 TINSEL

The natural language group at the Navy Center for Applied Research in AI has been building intelligent systems that are basically performing NLU for deriving extracts of Navy message texts and as a NL interface to some particular expert systems. This approach is based on the principles of computational linguistics (CL) and AI. The system is using a syntactic analyser which is actually a parser from the PROTEUS system at New York University. It applies lexical and syntactic knowledge and generates a regularized parse of each sentence input to it. TINSEL (Tandem Interpreter for Selection) is the software product implemented to execute the next stage of the process, lexical and thematic semantics. A clear definition of ‘selection’ may be useful at this point. A selection for a word is the set of words with which that word commonly coexists. The program is written in Common LISP and runs in Sun Common LISP on Sun 3/60, 3/260 and 4/60 workstations under UNIX.

A generic NLU system architecture can be shown as in **Figure 1**.

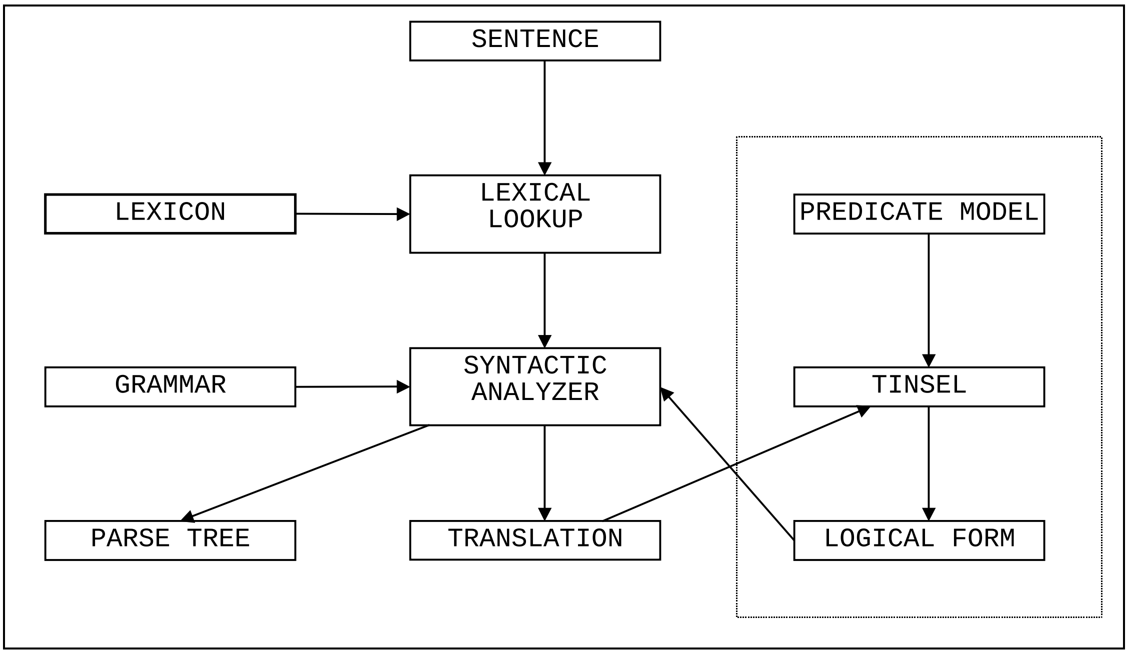

Figure 1

3.1.1 Top-Down Interpretation

TINSEL is basically an interpreter that is an data-driven system program which analyses statements in one particular declerative language (SOOP expressions) by using patterns in another declerative language, prediacte models. TINSEL predicate models are made of two basic parts, frame definitions and mapping definitions. Frame definitions and mapping definitions are all TINSEL needs to do selection and interpretation of most PROTEUS statements. The interpretation is done on two fundemental SOOP types, 'wffs' and 'quants'. 'wffs' are the well-formed formulas and 'quants' are the collection of regularized clauses and NPs.

3.1.1.1 Quants Interpratation

When an quantified expression is input to TINSEL, the interpreter first consults its internal model of SOOP syntax to determine to which SOOP type it belongs to. After identifying it as a 'quant', the interpreter next retrieves all the frame mappings. If there is only one mapping to the frame then interpretation is complete. If not, then there is an ambiguity, there. At this point TINSEL makes several copies of the expression and continues with such many interpretations.

3.1.1.2 Wffs Interpretation

Following is an analysis of how TINSEL studies an wff in top-down mode :

- It begins by recursively invoking itself on each of the several operands of the wff to transform them into interpretations.
- Next, the interpreter generates the frame mappings for that wff.
- TINSEL gets the transfer-data frame given by the user and check whether the frame's selection constraints will permit this mapping to succeed.
- If successful, that interpretation is returmed back, if not, then the second frame mapping is considered as the next.

3.1.2 Bottom-Up Interpretation

The only difference in bottom-up interpretation is that the intepreter does not generally invoke itself recursively on embedded statements. This is because in the bottom-up parsing the left-recursion is already harmless and no further steps are needed to eliminate it, whereas in the top-down parsing, left-recursion is implemented by recursive calls.

3.1.3 Comparison With Like-Systems

At this point, for the purposes of this survey, it will not make much sense to give very detailed knowledge about the operation principles of the TINSEL. But making comparisions to like-systems may give a clearer idea about the power of this particular tool.

3.1.3.1 Linguistic String Parser

The differences between LSP and TINSEL may be summarized as follows :

- LSP does selection by using syntactic patterns of semantic classes, but TINSEL first maps from syntax before checking the selection constraints.
- Selection in LSP is done directly on lexical attributes, but TINSEL interprets syntactic/lexical translations by mapping them to predicate models.
- LSP semantic categories are nonhierarchical word classes.

3.1.3.2 Question Answering System

When used with PROTEUS, an example software called QAS was formed. This was an demonstration NL interface to a student transcript database query system.

The major differences between the QAS and the latest version of TINSEL interpreter can be collected as :

- intermediate representation
- sort hierachicy
- uniform treatment of predicates and types
- attachment of constraints
- thematic roles
- semantic ambiguity
- NP modifiers

3.1.3.3 Allen

Allen uses the logical forms which are based on a case grammar and does the interpretation by the help of lambda calculus. TINSEL works with a case grammar, too, and PROTEUS component uses the lambda calculus. But TINSEL represents the mappings and constraints as declarative patterns. Allen shows these in a rule-based formalism.

3.2 Eucalyptus

Today, the interactive software development has come to a state that the computers can successfully resemble the human-to-human conversation in nearly every extend. One of the most important points among these is the discourse processing, i.e. the ability of tracking the continuity. In the early days of AI research, many people believed that the medium between computer and human would be the natural language. The success of the graphical user interfaces in time, shows that each of these two media has advantages and disadvantages. Most graphical user interfaces, give the use the ability to visualize the environment, point to it, size and select the relevant parts of it. But such interfaces can become tiring and repetitious in use. The NL, conversely has the advantages of logical connectives, quantification and generalized description. But, again, it has the risk of being ambiguous, misunderstood and vague.

In on of his papers, Kenneth Wauchope describes a natural language interface with a limited discourse ability that has been built and integrated into the GUI of a Navy system design and training demonstration, the KOALAS (knowledgeable, observation Analysis-Linked Advisory System) Test Planning Tool. The interface, named Eucalyptus, can handle both the commands and queries which are typed in or spoken through a microphone.

KOALAS is an architecture for designing new generation intelligent systems that integrates the inductive power of human and deductive power of computer. The KOALAS Airborne Early Warning Test Planning Tool is an application built by the Los Alamos National Laboratory for the Naval Air Systems Command to show this philosophy. This system simulates a command and control system for a Navy E2 AEW aircraft. It runs on Sun-SPARC workstations and its GUI has been written by using Sun View. The system has a rule-based tactical advice generator that monitors the state of the simulation and proposes action advises in the form of English sentences.

3.2.1 Working Assumptions :

Eucalyptus has developed over some working assumptions :

- Commercial Speech Recognition : The speech recognition module of Eucalyptus is a commercial one and this research does not have any to do with speech recognition.
- Natural Language Only : Eucalyptus uses only English sentences which include elliptical and fragmentary forms that might normally be used in a

human-to-human conversation in a similar case. So, it is a spoken NL input system, not a voice input one.

- Input Only : Eucalyptus ignores all NL output issues like speech synthesis or generation.
- GUI Unchanged : The KOALAS' GUI is left unchanged. Eucalyptus is just an extension to the original interface.
- No Graphical Meta-language : In Eucalyptus, the GUI and NL interfaces are treated semantically equivalent. Each access the same set of operations but use different ways. For example, a GUI sentence to initialize aircraft trail display might consist of two phases: clicking the Experimenter Control button to see the control panel and then clicking the Aircraft Trails switch. The same NL sentence is simply 'Show Aircraft Trails', not 'Open the Experimenter Control panel and push the Aircraft Trails switch.'

So, the NL interface of Eucalyptus does not manage the GUI, but directly commands to the underlying program. Here, an analogy may be a voice-activated home lightning system that can also be operated by the switches on the wall. 'Lights' command will not be directed to the switch but to the light system.

3.2.2 Where KOALAS And Eucalyptus Interfaces Differ ?

At this point, we may take a generalized look to the Fighter Management menu in order to understand how different the GUI and the NL interfaces interact on the same operations. This will not be a complete set of operations which can be used by the user. Any interested reader is invited to [1],[2].

Six operations can be done by the help of the Fighter Management menu :

- Move: order a deployed aircraft to a new assignment station.
- Deploy : order an aircraft from the carrier to an assignment station.
- Vector : order a fighter out to a hypothetical threat's coordinates.
- Refuel : order an aircraft to a tanker to for refueling.

- Recall : order an aircraft back to the carrier.

Selecting an order from the GUI menu of the system presents the user a dialogue box to fill in the parameters of the selected order. So, the order is sent into the system just as a function call. For example : Vector(fighter, threat). The following sentences will be useful to show how the NL interface resemble the GUI. Suppose that the translation of the button clicks in GUI gives these :

(1) Deploy a fighter to station 5.
Recall fighter 14.
Move fighter 14 to station 5.
Vector fighter 12 to track.
Refuel fighter 14 to tanker 1.

Eucalyptus accepts these, too. It also accepts the simple syntactic variant illustrated in (2). If the sentences are changed a little to address the fighters directly as in (3), the NL interface will accept them, too :

(2) Have a fighter deploy to station 5.
Have fighter 14 return to the carrier.

(3) Fighter 14, return to the carrier.
Vector to track 3, fighter 12.
Refuel at tanker 1.

For each of these verbal inputs, Eucalyptus presents a dialogue box with the sentence understood from the vocal input. So, the user can be sure about the input data accuracy before the order is executed. Final acceptance can be made by clicking the relevant button or giving the vocal response like ‘Okay’ or ‘Never mind’ etc.…

3.2.3 Inside The NL Interface :

The idea in Eucalyptus is not just to produce a multimedia interface, but to give the additional expressive power of the natural language which is over all the graphical interface power. But, the important point, here, is to keep the NL interface still integrated into the GUI. There are three features of NL that distinguish it from direct manipulation. These are its capability for paraphrase, set-theoretic and logical operations and context maintenance.

The set-theoretic and logical operations allows the user to express the object sets by the usage of predicates, quantifiers and logical operations. NL referring expressions in Eucalyptus can be pronouns (it, he, them), headless quantifiers (one, any), locative adverbs (here, there), proper names (fighter 1) or noun phrases : indefinite (an F14, any of the searching UAV, no fighter), definite (that tanker, the fighters moving to station 2), or universal (all the fighters on the carrier). NP pre-modifiers can be nominal (radar trails), adjectival(hypothetical threat), verbal (deployed fighter, moving F14) or genitival (fighter 1’s missiles); post-modifiers are either prepositional phrases (an aircraft on the carrier) or relative clauses (a threat being vectored to).

Eucalyptus also has a querying system to collect information about the internal state of the simulation. This component can handle yes/no questions, Wh-questions (Who is moving ?) and Wh-Noun questions (Which fighters are moving ?). These questions consist of only who, what, where and how many (much) as neither KOALAS nor the NL can handle time (when) or metaknowledge (why, how).

In Eucalyptus, a traditional query system is used in which each query is translated into a quantified logic expression. So when evaluated, it accesses the internal structures of KOALAS data and returns the answer as a sentence. Moreover, Eucalyptus can compose elliptical responses, so that the user meets with a less mechanical and more conversational dialogue.

As mentioned before, discourse capabilities are the main issues that give the true power of NL interfaces. Shortly, discourse is the ability to make decisions to overcome vagueness, ambiguity or abbreviations. Eucalyptus, as its discourse capability, tries to resolve anaphora and ellipsis. Anaphors are abbreviated references to entities already mentioned earlier in a discourse. Ellipsis is the process of leaving a sentence incomplete and context-sensitive on purpose.

3.2.3.1 Anaphora :

The anaphoric referring expressions that can be handled by Eucalyptus are pronouns (it, he, they), headless quantifiers (one, any, each) and demonstratives (that, those, there). It, first, tries to take the anaphoric reading and if it fails, it decides that the reading is non-anaphoric. For example, the treat has a non-anaphoric meaning unless only one treat exists in the expression, but will have an anaphoric meaning once additional threats appear. The anaphoric references become important in the querying process, for example :

Are any fighters vectoring to threats ?
Yes, F14 #1.
Which one is he vectoring to ?
Badger #5.
Are there any fighters still on the carrier ?
Yes, F14 #18 and F14 #19
Deploy one of them and vector him out to that threat.

3.2.3.2 Ellipses :

Eucalyptus keeps up with the elliptical forms by using their pre-logical form as a template to be re-evaluated after replacing some of its arguments with a new argument from the oncoming sentences. Three basic classes of elliptical sentences are well-known and these can all be handled by Eucalyptus. The first consists of repetition like ‘Again’ which just make the sentence to be re-evaluated (1). The second, VP-gapped sentences, are evaluated by making a copy of the prior sentence and just changing the subject with the given one (2). In the third class, any of the arguments may be left unspecified to be replaced with the ones in the prior sentence (3). Moreover, elliptical inputs can be used to correct speech recognition errors (4). The examples are as follows :

(1) Increase the simulation speed.
Again

(2) Which F14s are moving to station 1 ?.
Which UAVs are ?.
Is fighter 1 moving to station 1 ?.
Is fighter 2 ?.
Is fighter 1 moving to station 1 ?.
Recall a UAV that is.

(3) Are any F14s moving to station 1 ?.
Any UAVs ?.
What about station 5 ?.
I meant station 4.

(4) Is fighter 4 moving to tanker ?.
I said fighter 4.
The tanker.

3.2.4 Architecture Of Eucalyptus :

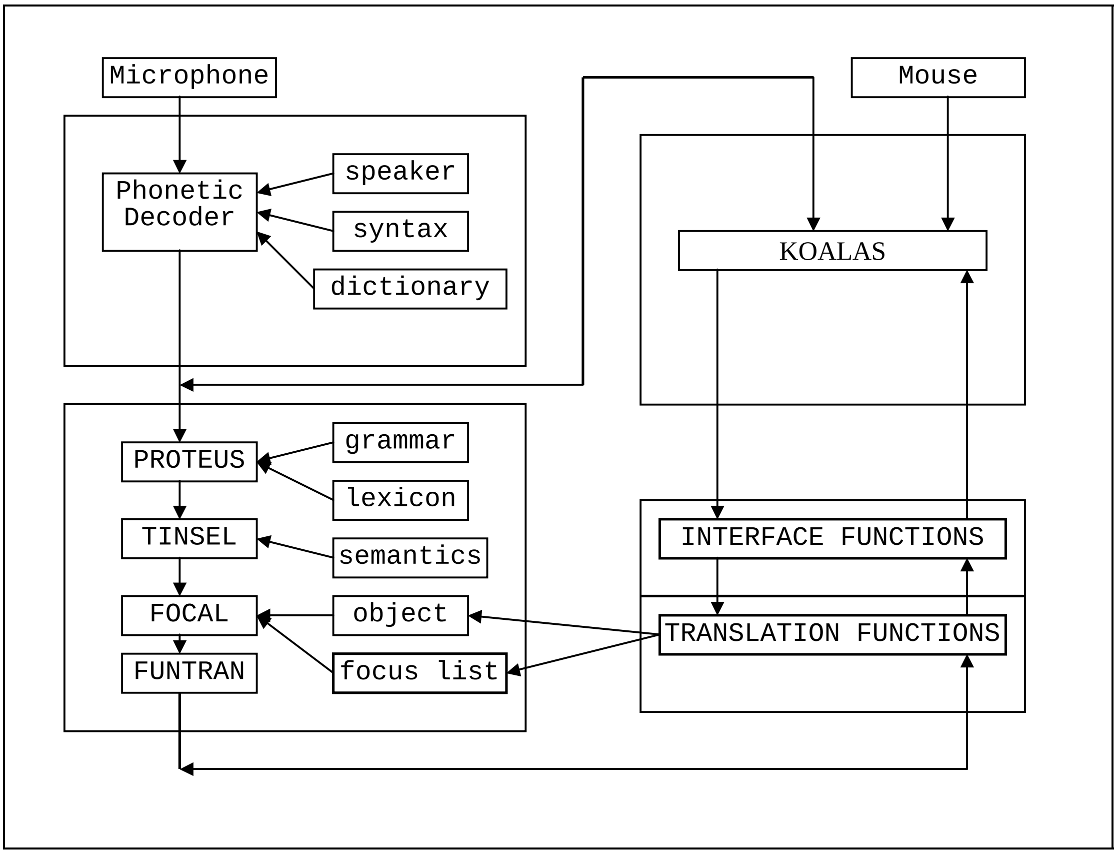

Figure 2

The architecture of Eucalyptus is shown in **Figure 2**. There are three main modules of the system which are the speech recognition component, natural language processor and application specific translator. These subsystems will be made clearer in the following sections.

3.2.4.1 Speech Recognition Component :

Eucalyptus has the commercial Speech Systems Inc. Phonetic Engine 200 continuous speech recognizer. The SSI system needs a finite state grammar of all acceptable utterances which it uses to find the closest match to the input. This grammar for Eucalyptus consists of about 250 non-iterative, non-recursive productions capable of recognizing 115 million different utterances. The phonetic dictionary contains 260 words. Twenty-five are integers, seven are alphabetic characters, thirty-four are apostrophized plurals and the rest are ordinary English words. Moreover, Eucalyptus allows the user to give the numbers as single words (twenty) or as individual digits (two oh).

3.2.4.2 Natural Language Processor :

As soon as a input sentence is obtained, it is passed to the NL understanding component for the conversion to logical form. This component has four modules : the PROTEUS parser developed at New York University's Courant Institute, the TINSEL case-frame interpreter, FOCAL reference resolution module and FUNTRAN quantified- expression builder. The last three are developed in Naval Research Laboratory.

3.2.4.2.1 Syntactic Analysis :

The PROTEUS syntactic grammar is used in Eucalyptus and consists of nearly 180 context-free phase structure rules and 50 procedural syntactic restriction rules. This grammar is based on the grammar developed for the InterFIS project. Its lexicon contains about 425 words, so it is really large.

3.2.4.2.2 Semantic Interpretation :

The TINSEL predicate-argument interpreter runs in accordance with the parser and applies predicate domain constrains during the interpretation. It mainly provides four interpretive functions. Detailed information about these will not be given in this paper. The readers interested, are invited to the detailed report [10]. These functions are :

3.2.4.2.3 Thematic Resolution

- synonymy
- thematic ambiguity
- polysemy

3.2.4.2.4 Quantification :

The FUNTRAN (FUNctional TRANslator) component transforms each TINSEL representation into an first-order predicate calculus expression with restricted quantification. This makes it possible to represent complex noun phases in the logical form as a self-contained expression.

3.2.5 Implementation :

The NL processor source code consists of nearly 6860 lines of Common Lisp. The Eucalyptus translator module is written in both Lisp and C. It contains 2940 of Lisp and 2480 lines of C. KOALAS source is nearly 16150 lines of C code. The data files total about 5750 lines of text. KOALAS execution needs 3.1 MB, Lisp needs 6 MB and Eucalyptus code and data needs about 4.5 MB runtime memory.

3.2.6 Related Work :

There are many other systems developed in recent years that integrate GUI with NL recognizers. For example, XTRA, Shoptalk, CHORIS and CUBRICON user interfaces. CHORIS and CUBICON are map-oriented application interfaces but not NL ones. XTRA allows reference to the contents of fields in a tax computation form interfacing an expert system. It has capabilities just like Eucalyptus. Shoptalk is a factory simulation interface that accepts references to pieces of product and equipment. Phrases can be used to fill in the fields of NL-like dialogue windows, as in Eucalyptus.

3.2.7 Data Processing Examples :

3.2.7.1 SSI Phonetic Decoder Output

```
are all the f fourteen's moving to a station
that fighter 1 is holding
```

3.2.7.2 PROTEUS intermediate syntactic representation

```
(REQUEST PRESENT PROG MOVE
      (ALL N1 F14 PLURAL
      (TO (SOME N2 STATION SINGULAR
            (PRESENT PROG HOLD
            (NULL-DET N3 FIGHTER SINGULAR(IDNUM 1))
            VAR))))
```

3.2.7.3 TINSEL semantic interpretation

```
(REQUEST PRESENT PROG V1 (:CLASS P-MOVE)
      (:PATIENT (ALL N1 (:CLASS P-F14 PLURAL)))
      (:TO-LOC (SOME N2 (:CLASS P-STATION) SINGULAR
            (PRESENT PROG V2 (:CLASS P-HOLD)
            (:PATIENT (NULL-DET N3 (:CLASS P-FIGHTER)
                  SINGULAR (:ID 1)))
            (:AT-LOC N2)))))
```

3.2.7.4 FOCAL alternate focus list

```
((N1  SINGULAR  P-FIGHTER  :  PATIENT(FRIENDLY-1  .  .  .
FRIENDLY-9))
(N2 SINGULAR P-STATION :TO-LOC (STATION-2))
(N3 SINGULAR P-F14 :PATIENT (FRIENDLY-1)))
```

3.2.7.5 FUNTRAN quantified expression

```
(FORALL X1 (SETOF N1 P-F14)
      (EXISTS X2 (SETOF N2 P-STATION
      (EXISTS! X3 (SETOF N3 P-FIGHTER (:ID 1))
      (P-HOLD : PATIENT X3 :AT-LOC N2)))
      (P-MOVE :PATIENT X1 :TO-LOC X2)))
```

3.2.7.6 Quantified expression after performative insertion

```
(TELLIFALL (SETOF X1 (SETOF N1 P-F14)
      (EXISTS X2 (SETOF N2 P-STATION
      (EXISTS! X3 (SETOF N3 P-FIGHTER (:ID 1))
      (P-HOLD : PATIENT X3 :AT-LOC N2)))
      (P-MOVE :PATIENT X1 :TO-LOC X2)))
```

4. CONCLUSION

In this survey, an idea about the state of art AI applications in the military field, today, is tried to be given. Many applications are mentioned with a brief description and one of them is examined in detail.

Although great effort and a lot time has been spent to prepare this survey, we are sorry to report that it is incomplete. It may contain much more topics and information, but because of some problems, it could not. One of the problems that we met when collecting information, is enough interesting to mention here. In our way to prepare this paper, we also search the Internet sources. We have searched nearly a hundred sites on Internet to get information about the AI applications specific to military. But we learned that a large percent of these sites are classified to authorized use and obviously not with willing to share anything. At this point, we have to tell that we believe in the importance of sharing information in Internet. If anybody does not want to share information, then he/she may better not to put it into the net of nets.

As a last word, it is good to see that there are really working and useful -to-some-point AI applications in the field. We hope that the number of these will grow larger and larger in the future to further prove that ‘AI is possible !…’

REFERENCES

1. K. Wauchope, "Eucalyptus: Integrating Natural Language Input with a Graphical User Interface," Navy Center for Applied Research in Artificial Intelligence, Information Technology Division, Feb, 1994.

2. K.Wauchope, "Eucalyptus: An Intergrated Spoken Language/Graphical Interface for Human-Computer Dialogue," Proceedings of the Voice I/O Systems Applications Conference (AVIOS '92), Sept 22-24, 1992, American Voice I/O Society, Minneapolis, MN, pp. 71-76.

3. A. Schmidt, "Human Machine Dialogue," Naval Research Laboratory, Washington DC.

4. B.K. Parsi, "Automatic Target Extraction in Infrared Images," Navy Center for Applied Research in Artificial Intelligence, Naval Research Laboratory, Washington DC.

5. B.K. Parsi, "Clustering with the Hopfield Neural Networks," Navy Center for Applied Research in Artificial Intelligence, Naval Research Laboratory, Washington DC.

6. P.R. Cohen, "Adaptive Interfaces," Technical Report RADC-TR-91-27, Rome Air Development Center, Air Force Command, Griffiss Air Force Base, N.Y, 1991.

7. S.W. Tyler, "Adaptive Interfaces," Technical Report RADC-TR-90-276, Rome Air Development Center, Air Force Command, Griffiss Air Force Base, N.Y, 1990.

8. A.C. Schultz, J.J. Grefenstette and K.A. De Jong, "Adaptive Testing of Controllers for Autonomous Vehicles," Navy Center for Applied Research in Artificial Intelligence, Naval Research Laboratory, Washington.

9. A.C. Schultz, J.J. Grefenstette, K.A. De Jong "Learning to break things: adaptive testing of intelligent controllers," Handbook on Evolutionary Computation, Oxford Press.

10. K. Wauchope, "A Tandem Semantic Interpreter for Incremental Parse Selection," NRL Report 9288, Sept, 1990.

REFERENCES NOT CITED

J. Allen, 1989, Natural Language Understanding, The Handbook of Artificial Intelligence Volume IV.

E. Marsh, J. Froscher, R. Grishman, H. Hamburger and J. Bachenko, “Automatic Processing of Navy Message Narrative,” NRL Report 8893, Aug, 1985.

D. Perzanowski and B. Potter, “InterFIS: A Natural Language Interface to the Fault Isolation Shell.” Naval Research Laboratory.

R. Grishman, “PROTEUS Parser Reference Manual,” PROTEUS Project Memorandum, Department of Computer Science, New York University, July 1986.

J.M. Gawron, “Syntactic Regularization in PROTEUS,” PROTEUS Project Memorandum, Department of Computer Science, New York University, March 1986.

R. Grishman, Computational Linguistics: An Introduction (Cambridge University Press, GB, 1986).

B.J. Grosz, D.E. Appelt, P.A. Martin, and F.C.N. Pereira, “TEAM: An Experiment in the Design of Transportable Natural Language Interfaces,” Artificial Intelligence 32, 173-243 (1987).

C. Barrett and C. Aldrich, “Final Report, KOALAS Test Planning Tool Concept Demonstration: Users Manual,” Sept, 1990, Los Alamos Laboratory.

K.Y. Kissmeyer and A.M. Tallant, “MACPlan: A Mixed-Initiative Approach to Airlift Planning,” 1990.

E. Rich, Artificial Intelligence, New York: McGraw Hill, 1983.

A.C. Schultz, J.J. Grefenstette, and K.A. De Jong, “Test and Evaluation by Genetic Algorithms,” IEEE Expert, v8, n5, 9-14, October 1993.

R. Schneider, “Identification, Friend or Foe Statistical Performance,” DoD Electromagnetic Compatibility Analysis Center.

K. Wauchope, “NAUTILUS (Navy AUTomated Intelligent Language Understanding System),” Navy Center for Applied Research in Artificial Intelligence, Naval Research Laboratory.

A.C. Schultz, J.J. Grefenstette, and W.Adams, “RoboShepherd: Learning a complex behavior,” Robots and Learning Workshop, May 1996.

A.C. Schultz, “Learning Robot Behaviors Using Genetic Algorithms,” Intelligent Automation and Soft Computing: Trends in Research, Development, and Applications, Vol 1, (Eds. Mohammad Jamshidi, Charles Nguyen), Proc. of the First World Automation Congress (WAC '94) and Fifth International Symposium on Robotics and Manufacturing (ISRAM '94), Manufacturing, August 14-17, 1994, p607-612, TSI Press: Albuquerque.

J.J. Grefenstette, “Evolutionary Algorithms in Robotics,” Robotics and Manufacturing: Recent Trends in Research, Education, and Applications, Vol 5, (Eds. Mohammad Jamshidi, Charles Nguyen), Proc. of the First World Automation Congress (WAC '94) and Fifth International Symposium on Robotics and Manufacturing (ISRAM '94), Manufacturing, August 14-17, 1994, p127-132, TSI Press: Albuquerque.

J.J. Grefenstette and A.C. Schultz, “An Evolutionary Approach to Learning in Robots ,” Proc. of the Machine Learning Workshop on Robot Learning, 11th International Conference on Machine Learning, July 10-13, 1994, New Brunswick, N.J., 65-72. (AIC Report No.: AIC-94-014)

A.C. Schultz and J. Grefenstette, “Evolving Robot Behaviors,” AIC Report No. AIC-94-017