WITAS

PM 2004:08

2004-12-30

Wallenberg Laboratory for
Information Technology and Autonomous Systems
at Linköping university, Sweden

Erik Sandewall

The WITAS Dialogue Technology Project

Executive Summary of Progress, December, 2004
and Plans for Continued Research

Related information can be obtained via the following WWW pages:

Linköping university:	http://www.liu.se/
WITAS Project:	http://www.ida.liu.se/ext/witas/
Author's Department:	http://www.ida.liu.se/
The author:	http://www.ida.liu.se/~erisa/

1 Organization, Visions, Goals, and Approach

1.1 Organization

The Wallenberg Laboratory for Information Technology and Information Systems (WITAS) now consists of two projects, namely the Dialogue Technology Project and the Autonomous UAV Technologies Project[1]. The present document describes goals and recent achievements in the former project, as well as remaining work towards the present goals and proposed directions for a new project building on the present Dialogue Technology Project.

In turn, the Dialogue Technology Project has three interacting themes:

- **Application System Theme:** Development of an Operator's Assistant Software System for the WITAS UAV System
- **Knowledge Representation Theme:** Development of methods for representing knowledge about mobile objects and their environment, as well as other spatio-temporal knowledge, with the operator's assistant system as an example and an application area
- **Software Methods Theme:** Development of methods for software systems with the capability of autonomous self-modification, with the operator's assistant system as an example.

The work on the first and the third theme is supported by the Wallenberg Foundation under the WITAS grant. The work on the second theme is supported by CUGS (the National Graduate School in Computer Science), and is outside the scope of this report.

1.2 Vision and Goals for the Application System Theme

Small to medium sized autonomous vehicles (UAV, UGV, etc) will often be used by, and controlled by individual operators which can be soldiers, disaster rescue personnel, etc depending on the type of application. In critical situations it is essential to minimize the cognitive load on the operator for controlling the vehicle, by at least the following measures:

- Possible for the operator to request the execution of an action or a plan (sequence of actions) and have the vehicle perform it by itself without further need for intervention by the operator
- Possible for the operator to set a mission goal and have the system design a plan and carry it out without further need for attention by the operator
- Ability by the system to report on progress towards stated mission goals or failure to achieve them, including a diagnostic of the reasons for the problems and of the possibility for a new try, so that the operator does not have to keep checking the status of ongoing plans

[1] Both names are becoming inaccurate since the Dialogue project deals with operator assistance systems where dialogue is only one aspect, and the UAV project now generalizes to other types of autonomous vehicles. However we keep these names for the time being.

- Possible for the operator to do the above using spoken dialogue in ordinary although restricted language (entire sentences, not just single command words) combined with limited and optional use of maps, diagrams, photographs and video on a portable computer screen
- Possible for the operator to use the concepts and terms of the environment and the application at hand when formulating himself/herself for the above, and ability by the system to use those concepts and terms correctly in its communications to operators.

These goals require considerable development of new technology and constitute an area for qualified research and development. They go much beyond what can be achieved by careful tuning of standard technology for 'user interfaces' using insights about human factors, although of course the technology development must be *combined with* human factors considerations at a certain point. The present project is therefore intrinsically a software technology project and not a 'human factors' project.

1.3 Approach for the Application System Theme

Our system is designed for working in symbiosis with one or more vehicle control systems. We assume a distinction, therefore, between the *autonomous operator's assistant system* (AOA) and the *vehicle control system*, where each operator has his or her AOA. This separation is necessary since each operator may communicate with several vehicles, and vice versa. It corresponds as well to the respective goals of the two WITAS projects at present, where our project is concerned with the AOA.

The AOA is in turn composed of two and eventually three parts: an SGUI (speech and graphics user interface), a DOSAR (language and dialogue system), and an optional reasoning component for situation understanding and learning. The SGUI must be capable of understanding and producing speech in human language (English, Swedish, etc), and of presenting static images and video in such a way that there are well defined connections to input and output speech and to the user's indication when he points into the pictures.

The DOSAR system is the carrier of the autonomy in the AOA, since it contains facilities for receiving mission goals, defining plans and executing those plans. It also contains the processes for understanding language in written form (as produced by the speech input module of the SGUI, for example), for deciding what to say and when to say it to the operator, for asking questions to the operator when something is not clear and is judged to be of essential importance, and so on. - The situation understanding system will be discussed in a later section.

Three additional subsystems are of particular importance for the overall goal besides those mentioned above:

- A *simulator* for the UAV's, other vehicles, and their environment. It is used for testing the AOA software before flights, but it is also intended to be included in the DOSAR system itself in order to provide a kind of 'envisioning' of what happens in the observed world.

- A *standardized interface* between AOA's and other 'intelligent' software systems on one hand, and autonomous vehicles and other autonomous equipment on the other, with high exchangeability on both sides. In our design this interface, called LEAF, will also be responsible for some middle-layer activities such as event recognition and execution of actions during simulation.
- a *video server* that is able to store video coming from the UAV or other video camera carriers, and to play it back to the AOA on request, equipped vith timestamped markup information whereby it is possible to relate objects in the field of vision of the camera to coordinates in the actual physical world.

1.4 Vision and Goals for the Software Methods Theme

Since its inception in year 2002, the WITAS Dialogue Systems Project([2]) has been an attempt to probe into new and alternative methods for software engineeering for a broad range of systems. We started the work with some new ideas in mind, and expected to test, extend and refine them along the way. This aspect of the project has developed considerably during the past year, and we shall now describe how we understand that topic at present.

The direction of work in this theme is to develop methods for the engineering of software systems that 'learn from experience', or more precisely, *software systems that change their structure, their contents, and their behavior as a result of their operation in the application environment during a substantial period of time.* The long-time aspect is significant. Learning and knowledge acquisition is only of interest if it takes place over periods of months and years. The proposed goal must therefore combine methods from artificial intelligence with selected aspects of software engineering, and in particular those aspects of software engineering that are concerned with the entire software life-cycle.

We adopt the following perspective. A *software system* is defined as a collection of programs, procedures, and data that serve an application. We are particularly interested in software systems that are integrated in the sense that they contain their own software development tools. Three levels of self-modification are recognized in such systems:

- **Supervised self-modification**, where a systems developer issues editing or other commands that have the intention and the effect of changing components or structures in the system.
- **Automated self-modification**, where a previously prepared program performs changes in the system. Software product installation using so-called 'wizards' is an example of automated self-modification.
- **Autonomous self-modification**, where the software system performs changes within itself as the result of its operative use in the application environment.

[2]This applies to the Dialogue Systems project in Linköping and not so much to its predecessor activity at Stanford.

Our research interest is in high-level autonomous self-modification (HASM) that affects higher software levels and that is based on an application ontology (explicitly structured model of the application). Other and simpler kinds of self-modification are already in widespread use. This applies in particular to supervised and automated self-modification, but also to atonomous self-modification with low-level autonomy, where for example sets of parameters are updated, or decision trees are reorganized.

High-level autonomous self-modification warrants the use of general methods for autonomous systems, which are already being used or introduced for providing autonomy to the WITAS UAV's and its dialogue system as such. Autonomy in the UAV control system and autonomy in the management of the software system has some features in common. Methods for HASM can not be used in isolation, however, and in particular they must be integrated with existing methods for supervised and automated self-modification in regular software development. This is important because 'learning' and 'knowledge acquisition' in a software system is only meaningful over a considerable timespan, i e months or even years, and during such a period there is going to be many requirements for 'manual' extensions to the software as well. In other words, autonomous self-modification must co-exist with supervised and automated self-modification as integral parts of the software lifecycle.

1.5 Approach for the Software Methods Theme

Our approach for the software methods theme is as follows. We distinguish between a medium-range activity and a long-range activity. In the medium-range activity, which is the major one in size, we develop a software architecture and platform, called the Software Individuals Architecture (SIA) that is used as a basis for the DOSAR part of the dialogue (AOA) system. It is designed with the requirements of all the above-mentioned types of software self-modification in mind. The SIA platform has also been used for one version of the environment simulator and for a document management system, but the DOSAR system is the biggest and most challenging application. In this way, some parts of the AOA project are used as testcases for experimenting with new methods, but other parts (in particular the SGUI part) have not been much affected by this.

The long-range activity concerns the development of new systems software methods for persistent computing, in particular concerning memory management, addressing, and garbage collection. Persistent computing will allow one to reduce or even eliminate the distinction between program execution and software lifecycle, so that in principle a software item 'executes' throughout its 'lifecycle'. This should make it possible to greatly simplify the architecture of systems for autonomous and other self-modification.

2 Activities during 2004

2.1 Group Members and Organization

The following persons have been engaged in our system development activities during 2004 (diacritics omitted):

Erik Sandewall	Project leader
Malin Alzen	Software engineer, January - June
Peter Andersson	Graduate student
Karolina Eliasson	Graduate student
Bjorn Husberg	Software engineer, 20%
Markus Ingevall	Masters thesis (examensarbete)
Hannes Lindblom	Masters thesis (examensarbete)
Susanna Monemar	Software engineer, January - June
Tobias Nurmiranta	Graduate student

In addition, `Joe Steinhauer` is a graduate student who has worked on the knowledge representation theme with CUGS funding. `Anna Maria Uhlin` and `Anki Rune` have been the group's administrators during the spring and the fall semesters, respectively.

The major areas of activity for these persons have been as follows, while omitting many additional but smaller activities:

Erik Sandewall	Project leader, software engineering platform
Malin Alzen	DOSAR, especially dialogue management
Peter Andersson	Simulation system for UAV environment, design for forthcoming LEAF system
Karolina Eliasson	DOSAR, situation understanding using case-based reasoning
Bjorn Husberg	General software support; video server
Markus Ingevall	Knowledge representation
Hannes Lindblom	SGUI, in particular graphics and video facilities
Susanna Monemar	DOSAR, especially language models
Tobias Nurmiranta	Persistent computing system

2.2 Continued Development of the AOA System

One version of our system was demonstrated during the main WITAS demo in October, 2003 and its follow-up with documentation in December of that year. The following are the major activities during 2004 with respect to further development of our experimental system:

- Thorough revision of the DOSAR and SGUI subsystems on the basis of experience gained, laying the ground for considerable additional extensions in 2005 and beyond.
- First step towards network level communication, through a system where one AOA can interact with two (simulated) UAV's at the same time.
- Finishing facilities for planning/scheduling and plan execution.
- Extending the range of input sentences that are understood by the system.
- Collection of test data by arranging for a large number of students to use the actual system.
- Adding Swedish as a second input language.

- Adding the capability for presenting running video, both actual one and flashback, and for pointing into it in order to indicate destinations, line of flight, or areas of flight (or forbidden areas).
- Development of a video server and an adjoint markup server.
- Development of a new simulator with integrated visualization and a number of other services that are important for the AOA.
- Development of facilities for logging dialogues and later on presenting those dialogues as text or as playback.
- Beginning work for a standard interface (LEAF) that will replace the solutions that we have now and that do not have sufficient generality.

The dialogue system was demonstrated live during actual flights at the October, 2003 main demo of the WITAS project. No additional live demos have been made during 2004 due to the rebuilding of the on-board helicopter systems in the other part of WITAS. Successive versions of the integrated system have been tested and demonstrated in the laboratory and against simulated flights. In our best judgement live demos would not have contributed very much technically or scientifically at the present stage, since the testing in 2003 already gave sufficient insight into the problems on that side. It is intended however that additional live tests and demos shall be performed in 2005 as soon as weather and flying conditions permit.

2.3 Graduate Student Progress

The three graduate students in the group have all participated actively in the continued systems development described above. At the same time they have proceeded towards their own licentiate theses, which will be independent works based on the common system platform. All three have written or are finishing their licentiate thesis proposals around the end of year 2004.

Peter Andersson has developed a new software platform that integrates simulation of the UAV and its environment with visualization of that environment. This software platform has considerable generality and is being used for several applications outside the WITAS UAV frameworks as well. He has then proceeded to take an interest in design principles for the interface between Autonomous Operator Assistant systems (AOA) on one hand and unmanned vehicles and sensor stations on the other, with the aim of defining a standardizable middleware that allows various specific systems of each of these kinds to be plugged in.

Karolina Eliasson started by studying the problem of selective remembering and the complementary function of forgetting, which are relevant for the life-cycle perspective of a software system. From this background she now addresses the use of case-based reasoning techniques for the purpose of 'learning' and 'knowledge acquisition' in an AOA system. This will lead to a design for the third major component of our AOA system.

Tobias Nurmiranta works on a novel design for memory management including address translation schemes and garbage collection techniques that provide an efficient basis for persistent computing. This is an important issue in the long range for the use of the techniques that are developed in our project.

2.4 The Software Individuals Architecture

The Software Individuals Architecture has been extensively documented and a number of services have been added to the system kernel, including, in particular, a number of software module management services.

Scientific Publication. The present project started effectively in the second half of 2002 and in early 2003. The natural gestation period before results are ripe for publication is relatively long because of the need for building a system of non-trivial size before meaningful results can be reported. During 2004 we have concentrated on systems building, and the writing of articles has been delayed until 2005.

2.5 Other Contributions

Other contributions, also under the direction of Erik Sandewall, have been made by our software engineer Björn Husberg on 20 pc time and by masters students.

3 Completion of the Present Project

Like many other research projects, ours is open-ended: each goal that is achieved and each result that is obtained usually comes with new insights about additional problems that ought to be addressed. There is therefore no easy answer to the question of when is the project going to be finished. However, the following are the concrete milestones that we would like to use for defining the borderline between the conclusion of the present project and the beginning of proposed new projects.

Completion of licentiate theses by the present three graduate students. This refers to the individual projects of Peter Andersson, Karolina Eliasson, and Tobias Nurmiranta which were described above. It is expected that they will all complete their theses during the second half of year 2005.

Completion of an operator's assistant system that is demonstrated to autonomously define the flight movements of the WITAS UAV during actual flights. All essential capabilities for this exist in the present system, and the remaining problems have to do with the connections with the rebuilt UAV on-board software, including safety aspects during the experiments.

Natural completion of existing facilities. The facility for graphical dialogue by pointing into live video as well as playback video is completed on the SGUI level, but needs to be complemented by a few additions to the language on the dialogue level. This should be done, demonstrated, and reported in order to make it possible to publish the entire work in this direction.

A second item in this category is to complement the existing Swedish-language input facility with corresponding Swedish-language output from the system.

Follow-up of the software methods theme. Analysis of the experience of using our SIA software approach and publication of that analysis are natural parts of the present project.

These activities will require work, during 2005, by Erik Sandewall as project leader, the three graduate students in question, and a limited amount of software engineering support, comparable to the requirement during 2004. Some of the activities are suitable as student projects.

4 Proposed Directions for Forthcoming Projects

The research within the WITAS Dialogue Systems Project and the associated work on spatiotemporal knowledge representation lends itself to be continued in one or more of the following directions, depending on the interest of sponsors and other interested parties. The order of the proposals is from the more applied directions to the more theoretical ones. The given project acronyms are only provisional.

4.1 Knowledge Acquisition for Operator's Assistant Systems: KAOPS

The vocabulary and expressive and conceptual power of our present AOA system is largely the result of our own understanding of the application domain, and our efforts have focused on software architectural issues. This was appropriate for the first years of work, but at this point it is necessary, for moving towards applications, to work actively with representatives of likely user communities in order to revise and extend the conceptual and linguistic bases of the system. The KAOPS project shall therefore concern a systematic study of the vocabulary that is required in interaction with mobile robots. The concrete plans for this project must be formulated in cooperation with one or more user groups.

From a research point of view, this work should result in a so-called *ontology* for cognitive robotics.

4.2 Continued System Development: the KONRAD System

Throughout this project we have been aiming for a network type system containing many autonomous vehicles and many AOA:s (autonomous operator's assistant) systems, but until now the actual system that we have been working with has only included one AOA and one or two simulated or actual UAV:s. We are now ready to proceed to the multi-agent network situation, where the network contains many operators with their respective AOA:s and each AOA can connect to different UAV:s or other autonomous vehicles at different times, and possibly several of them at the same time.

The chosen name for the new network level in our system is KONRAD, for "Knowledge-based Operator assistants in a Network of Robots, capable of Autonomy and Dialogue". This title captures both the network aspect and the fact that the AOA:s provide autonomous assistance to the operators and not merely dialogue.

This systems development project will raise, and rely on work on theoretical issues that are addressed in the HALOS project which is described next.

4.3 High-level Autonomy in the Lifecycle of Software Systems : HALOS

This project will focus on the third level of our AOA system, where attention has so far been given to the first two levels. It addresses both autonomy with respect to the application and with respect to the software system. The former type of autonomy provides additional capabilities for the AOA to perform functions on behalf of the operator in a reliable and transparent way; the latter type of autonomy allows the software system to continuously incorporate past experience into its own structures and also to exchange such 'knowledge modules' with other AOA:s which are then viewed as communicating autonomous agents.

Without being interdisciplinary in any strong sense, this project will anyway require a certain integration of techniques from several backgrounds, in particular:

- Methods for high-level autonomy in software systems (obviously)
- Methods for case-based reasoning for the purpose of problem-solving and language understanding in the AOA
- Methods for high-level learning (closely related to case-based reasoning)
- Methods for software engineering in the life-cycle perspective, such as current work on Architecture Definition Languages for complex software
- Methods for programming languages and systems for persistent objects (i e, objects that prevail throughout the system lifecycle)

In addition, we propose that the existing effort on persistent computation on the computer systems level should be continued: in the long range that approach highly relevant for all work on lifecycle level software techniques.

4.4 Knowledge Representation for Mobile Objects: KREMO

The topic of this activity can be described concisely: it concerns spatiotemporal modelling (= representation of knowledge) with respect to phenomena in the environment of the UAV's and other robotic vehicles, including also the conditions, actions and events that these vehicles are themselves involved in.

4.5 Priorities

The development of an experimental dialogue system has been important in several respects during the current project: it has been an integral part of the entire WITAS effort, and at the same time it has been a testing-ground for proposed new software techniques. Although not reported here, it has also been relevant for the work on knowledge representation during this period. We expect that the same will be true during the coming years, so that the proposed KONRAD project should get priority both for its own sake and because of how it will strengthen the other proposed activities.

The KAOPS project will be the natural next step for transferring the results of the current project towards practical use in the field.

The HALOS project should be the major basic-research effort that continues the present project. It is a very timely topic and our group is in an excellent position to move ahead in this area.

The KREMO project provides theoretical background and depth, in particular to the HALOS project but also to the other two projects. It builds on our contributions to the representation of knowledge about actions and change during the last fifteen years, and gains strength by the interaction with the other and more applied or systems-oriented activities proposed here.

4.6 Funding Opportunities

Obtaining funding for even some parts of the activities proposed above has turned out to be much more difficult than expected. Negotiations were still in (some) progress when the Christmas holiday season set in.