Bio-inspired, Autonomic Model of System Architecture for Protei.

Introduction:

The management of complex, heterogeneous and distributed (network based) system composed of collaborating, seafaring robotic drones (boats) is a challenging task; and unless done effectively can significantly reduce the overall efficiency; degrade capacity to perform various functions as well as limit access to available resources in remote, dynamic and often hostile environment. This is particularly true in heterogeneous, network based environments as the actual structure of the network based system can change depending upon such disparate factors as the application tasks, communication links, hardware, topology and geography and environmental conditions.

The software intensive ***System Infrastructure*** for Protei project will lay the foundations for developing service-oriented and real-time system for monitoring and control of seafaring robotic swarms for such applications as surveillance, containment and collection of oil spills. Apart of the on-board computer(s) to support high functions of the robotic vessel the system is to consist of highly reconfigurable sensor and actuator wireless networks. The ability to adapt to the changing environments requires a step change in the design approaches. The key research challenge is to provide flexible resource management and data access solutions that are effective in a large-scale, heterogeneous system network. The outcomes of the initial design will enhance the position of the AI group to become not only an advisory body but to ensure a sustainable vision for future development of advanced engineering for Protei and co-evolution of and open hardware and software platforms.

The model of hardware and software need to consider high level management (decision making) system wide functions, individual vessel and swarm navigation, control and communication as well as lower (autonomous) functions. The system at some stage would require global information management and application software development facilities in higher level programming languages. We may need to provide/recommend a set of high performance management solutions (Open source), middleware and software component frameworks that are able to facilitate the development of a network based Infrastructure oriented and embedded software for robotic swarms.

AIMS AND OBJECTIVES

This proposal is aimed at addressing the problem of dynamically managing the network of robots and associated resources so that specified communication links, data rates and priorities as defined by the real-time, Protei management system can be achieved. This entails a development of infrastructure oriented software and algorithms for construction and simulation of real-time, embedded and mission critical solutions in resource constrained environments. Initially, this work will be conducted using the tools and research facilities available at the University of Southampton, UTS, UNY, … .

SIGNIFICANCE AND INNOVATION FOR PROGRAM

Significance

The software methodologies and tools are core aspects of all networked oriented system infrastructure software. These infrastructure oriented software are implemented to efficiently combine, manage data and control robotic swarms. At some stage, infrastructure oriented software is to play of enormous significance in such areas as: Protei's infrastructure management, environment monitoring, vessel's seafaring ability, adaptation/articulation of all major parts, sensing and actuation, security and safety, education as well as many other areas that depend critically upon software technology. However, building application that make best use of AI and WSAN technology in terms of practicality and economics (including time to deploy) cannot be fully realised without a consensus by majority of application developers on adequate methodologies and tools. In the context of remote management of infrastructures, such methodologies and tools for robotic network systems can significantly improve development life cycle the value of embedded devices and sensor infrastructure, reduce the cost of information management and offer technically and economically significant as well as viable implementations to many participating institutions.

Innovation:

The proposed set of solutions need to bring the cutting edge technology of infrastructure oriented software, OODB tools, software integration and platforms and AI techniques in robotic networks, their remote access along with the distributed processing to fundamentally transform the way infrastructure management is undertaken. Due to their dependence on tight coupling between the user applications, infrastructure oriented software, the AI solutions, on-board computer(s) and the sensors/actuators at the low level might be difficult to deploy and highly inflexible and hence are not suited to work in such Infrastructure oriented software. In addition, the new architecture that is proposed could benefit from 3TZ Remote Group-work paradigms co-developed at UTS. The management paradigm adds proprietary actuating and sensing robotic systems to the list of shared resources and brings in a new set of challenges and complexities to the Protei's open hardware/software landscape. The solutions based on the Protei paradigm are innovative in sense that they facilitate sharing the same computer hardware/software, sensing/actuating resources and services with other organizations and authorities.

Advancing the knowledge base of the discipline:

The scope for cross disciplinary knowledge advancements when discussing Protei's hardware and software of architecture is significant. First and foremost is the advancement associated with the application of autonomics and information AI to improve the engineering methods of analysis, simulation and prediction. Second is the advancement associated with the development of cost effective and robust robotic network architecture for on-board vessel and remote operations. Third is the advancement associated with the application of biomimetic and AI paradigm that will enable the sharing of resources for multiple infrastructure oriented software concurrently. Owing to the richness of the field and the number of open problems in the domain, there is significant scope for several serendipitous advancements in the knowledgebase of several disciplines.

The following **new methodologies and technologies** will be developed in the course of this project

1. New design paradigms for infrastructure networks with distributed processing and AI for autonomic robotic network infrastructure management.

2. New and open Service Oriented Architectures that support access to the fused/processed remote sensor/actuator information by possibly multiple applications.
3. Advanced infrastructure oriented software tools for development and integration of real time, context sensitive and proactive software Infrastructure for mission critical robotic networks/infrastructures.

APPROACH AND METHODOLOGY

Our research approach consists of the following stages:

1. Produce a working definition of Service Oriented Architecture-like infrastructure software for WSAN based and embedded robotic systems. The output of this stage might be a document describing the architecture and how it applies to the needs of Protei. The document will include such high level design components as: Service Configuration, Service Activation, Fault Data Collection, Performance Data Collection and Usage Data Collection modules.

2. Identify the mechanisms, policies and possible parameters for enforcing control and management of individual robotic drones and their swarms. Outcome of this stage is a document identifying the policies, algorithms and equipment parameters that can be used for controlling and managing Protei.

3. Develop algorithms for management execution of real-time functions of Protei system architecture in the test-bed environment. During this stage performance management software for the equipment currently available in the collaborators research facility will be developed.

4. Develop suitable algorithms for task allocation within individual embedded devices/motes and across groups of these devices. A small scale simulation environment can be used to test different resource allocation techniques. In this phase, simple search techniques can be applied. More sophisticated, scalable resource allocation techniques will be the focus for future.

5. Participate with collaborating parties to develop a WSAN demonstrator that will execute in the water environment to demonstrate dynamic/adaptive modifications of resource/device settings to achieve a given control priority setting. This will include the management and decision-making software for the current Protei environment. This demonstrator can be used to show how the management and decision making systems work as well as test the system performance.

Objective 1:
Develop new architectures for robotic swarms, telerobotics and sensornets

The key end user requirement for this project could be summarised as follows: develop an open source, cost effective and reliable architectures which would allow collaborators to play a significant role in rapid development of various user Infrastructure hardware and software and at the same time make the best use of emerging computer and network device platforms to facilitate information processing at the individual WSAN motes, local level controllers (intelligent on-board computers) and the central computer(s). One of the key advantages of such approach is that by processing information locally, the system can reduce the amount of data that needs to go on the networks. More importantly, there would be no need to send all that irrelevant or redundant data through the network and thus burdening the transaction requirements of the systems. In line with this view, the proposed architecture aims to allow the processing to move from wherever it is to a point closer to the sensors and actuators. This has the potential to fundamentally transform the efficiencies associated the computing and network support infrastructure. The proposed system architecture (hardware and software) will be developed for Protei's multi-sensor/actuator, multi-application environment as depicted in the two figures below:

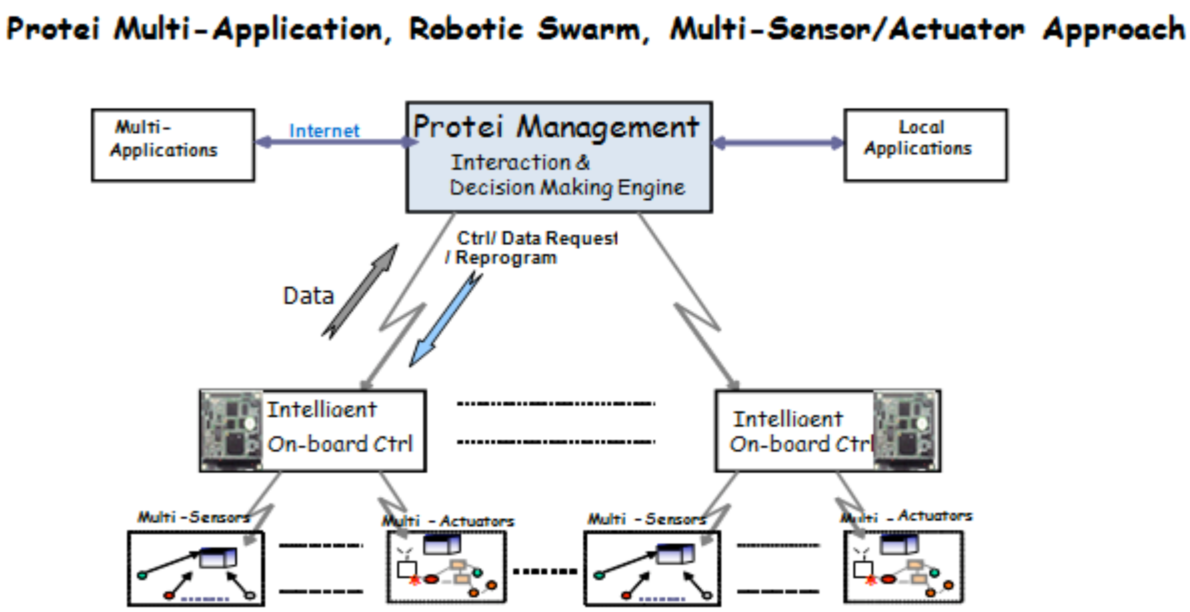

Figure 1: Conceptual Architecture for Robotic and WSAN software infrastructure

Although most of the processing may be done closer to source, still one needs an efficient Internet and Web Services like and possibly Cloud Computing (grid middleware) access. The proposed remote information management approach is illustrated in Figure 2. Web user logs into the web system to request services that include the main functionalities of the web system such as view, analysis and control of required information. The web browser transfers the user request to the web server. The web server sends the service request to the information server for information AI. The information is retrieved from the data storage server that could in principle be embedded in the SNAP. The data server processes the information using opportunistic information techniques, returns the AI results to the data storage server, and returns the results to the web server to display on the web pages. The web user will interact with the web pages for next service request.

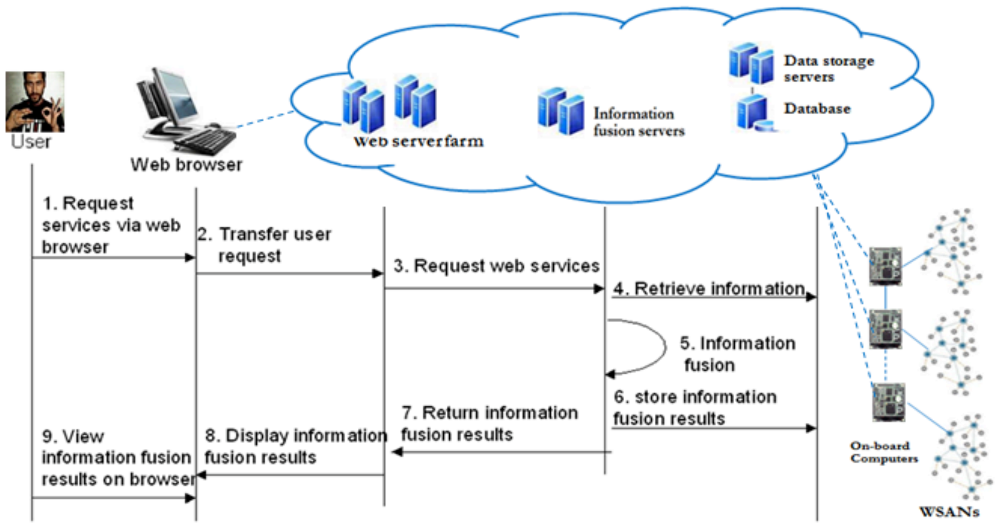

Figure 2: Web information management sequence

Objective 2: Develop new software engineering design paradigms and AI algorithms to support mission critical, autonomous, multi-application, heterogeneous, distributed and high volume traffic systems executing in hostile environments.

Low cost sensors and actuators with embedded computing and communication capabilities are enabling a new paradigm where sensors and actuators become a shared resource, just like the way computers have formed a grid to share the computing power. The Protei architecture at presentation and business logic layers would still require significant computational power. Hence, the vessel's higher level functions would be provided by a dedicated on-board CPU(s) (Linux box). While, the Protei system management can rely on Web services and Cloud computing infrastructure (grid middleware). The vision is to push the envelope of modern solution by enabling sensors and actuators to form a grid and deliver, autonomous services to different Infrastructure software.

As Infrastructure software will expand to incorporate multi-sensor/actuator feeds such systems are subject to severe bandwidth loading and potentially may require large amounts of computing power and storage. As we move to broader multi-sensor/actuator installations these limitations might be further exacerbated. Biology inspired, alternative approach would be able to dramatically reduce the bandwidth, computing and storage requirements while allowing multi-sensor/actuator information and control to be shared between many diverse Infrastructure software solutions. The human body is an excellent example of a true multi-sensor/actuator system being used for a multitude of diverse infrastructure software. As can be seen in the Figure 3 much of the bodies data is being constantly processed by the brain at the Unconscious and the Semiconscious levels in the background without the direct involvement of the reasoning or cognitive part of the brain. In this way, the brain is capable to obtain information from the Semiconscious parts, when required and then integrate, via data/ctrl AI, with the Conscious world. If the human brain was not compartmentalised in this specific manner, the Conscious part of the brain would be completely overloaded with just trying to process the data or events required for the control of Unconscious body functions such as respiration, blood circulation, heart rate, temperature control not to mention immune, endocrine or cutaneous system functions. The only time that this Unconscious world communicates with the Conscious part of the brain is when serious anomalies are detected, i.e. a high temperature or low sugar levels is signalled to the Conscious world where reasoning then takes place followed by Conscious actions - take a pill, eat, call the doctor etc.

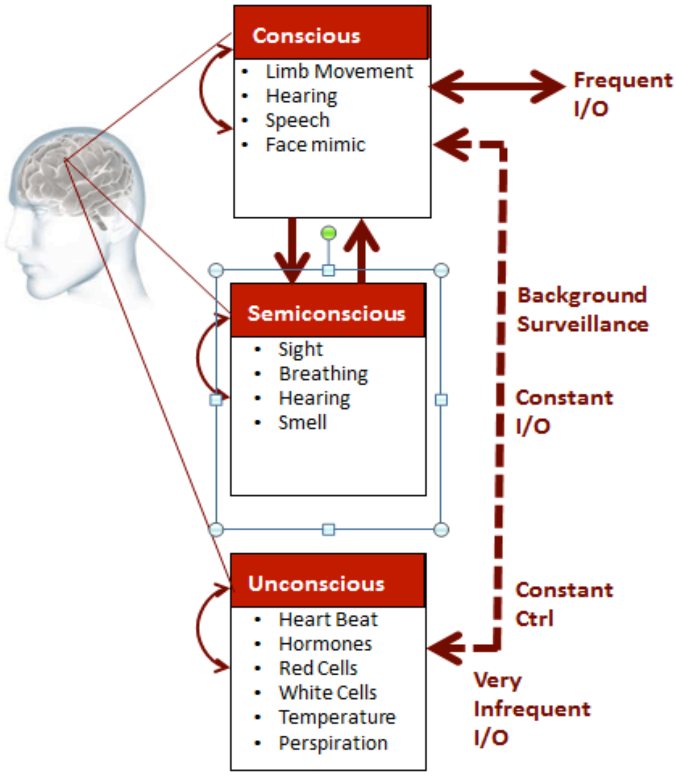

Figure 4: A Sensor Network with Feedback & Data/Ctrl AI

However, the breakdown of computing and communication for optimal dat/ctrl AI is a non-trivial problem. The fundamental question is: what processing or control should be executed at the sensor/actuator level, if it has some computing power and what needs to be done at the intelligent controller (Linux box) level; And ultimately what computations need to be carried out at the Protei system management level (application layer). One of the suggested approaches might be to first break down the information AI problem in terms of its distribution of communication and computing load in a highly flexible manner and then mapping the information AI on to the proposed architecture. It is proposed to use the emerging IEEE 1451 TEDS standards for effective discovery and maintenance methodologies of transducer (sensor and actuator) infrastructure.

<u>Objective 3: Provide software developers with new tools for analysis, design, integration and management of embedded and WSAN oriented infrastructure software</u>

Software tools and technologies play the critical role of in WSAN and information AI Infrastructure software. Advanced software development tools are not only to enable rapid design implementation and integration of infrastructure oriented software but also to facilitate transaction processing, manage demands for computing resources, support flexible software composition, provide data from all sensors for the comprehensive assessment of processing conditions and allow software constructs which can support combination of real-time decisions or perceptions from multiple sources. With the advent of wireless sensor and actuators, MEM devices, application developers need to be in a better position to overcome the traditionally over-conservative, less-transparent, labour intensive and costly approach in terms of design development and maintenance of software.

Notes take by Peter Keen.

Meeting between: Alex Phillips (University of Southampton), Zenon Chaczko (UTS), Gabriella Levine (UNY), Peter Keen (KML), Etienne Gernez (DNV)

16 September 2011, 1100 BST

agenda: (loosely speaking)
1) Clear statements of student research topics to be conducted
2) The separation and complimentary nature of these topics
3) Standardisation of code and platforms
4) Future capacity for integration of single vehicles (Soton focus) with multiple vehicle networks (UTS focus)
5) Any additional items (possible funding sources/model? items of a more esoteric nature - preferred AI models for instance? Any identified candidates yet? etc)

Minutes:

The meeting began after an hour and a half of informal chat between Zenon and Peter where they exchanged various files of interest concerning Texas Instrument control systems that Zenon had previously used and had great confidence in. A selection of these files accompany this email for participant information.
File list:
- szza058a.pdf
- MSP430 series.pdf
- UAV-paper-v3-2.pdf

Alex, Gabby and Etienne join the chat at approximately 1100 and attempts to hold the meeting using voice and text are made which prove unsatisfactory and eventually the meeting is continued using text only. Etienne left the meeting at this point to attend to previous commitments.

Alex indicates he is currently looking to acquire a small model yacht for a student to use and that he has some positive leads. He presents the following draft of a thesis topic:

"The Protei project is seeking to develop open hardware sailing drones which could be used to assist in the cleaning after oil spills. As part of this project we are looking to develop autonomous control strategies for autonomous sailing boats. Using knowledge gained from our existing autonomous under water vehicle the Student will be expected to retrofit a control system and sensor suite to a model scale radio control yacht and develop autonomous sailing strategies which enable the model to complete an upwind-downwind course."

Gabby indicates her interest in the TI MSB430 as a platform for development

Discussion is held over the split between single and multiple vehicle systems to be developed separately by Southampton and Sydney respectively with sufficient liaison to maintain future compatibility, particularly in the areas of protocols and local intelligence.

Peter asks how much intelligence needs to be local and how much can be group intelligence. Zenon indicates that group intelligence is expected to bring certain benefits though all agree that each individual should have sufficient competence to complete a limited number of tasks on it's own taking into account the special, and non-trivial, situation of autonomous sailing. It is considered that the interplay of these complimentary goals will be beneficial to the development process.

Peter poses the question: So do we approach this from the bottom up or the top down? Start with individual competence or define the wider network function and partition at the simplest autonomous unit?
Zenon responds by proposing three rules: (1) Separation: do not move too close to nearby birds,
(2) Alignment: move in the general direction of the flock, and
(3) Cohesion: steer towards the general center of the flock.
as a way of illustrating that individual competencies are probably the most fruitful place to start, though this 'bottom up' process may be determined by hardware limitations. Gabby questions Zenon on what the specific hardware limitations might be. Zenon responds by listing power consumption; cost; redundancies; and suggests opting for micro controller units (MCU) or PSOC style chips for use on individual vehicles.
Alex suggests that in the case of sailing units a brain; compass; a measure of roll; GPS; wind direction and speed are required. He adds that for multiple yachts you also need some communication (a much easier problem for yachts than submarines). The choice of hardware for the brain and its resulting implications for language selection are an important point as well. He expresses a preference to use a low-power linux box running ROS since this allows him to directly copy and paste, code between the AUV and an autonomous yacht. He notes that there are power implications of this strategy however.
Zenon agrees that a brain model would work but also indicates a wish to pursue the use of MCU for specific functions.
Peter suggests that perhaps within individuals certain basic tasks could be dealt with by pre-programmed responses (MCU level) leaving the Linux based CPU to consider higher level functions at greater leisure.
Alex mentioned a degree of success using parallel arduino MCUs with a linux PC (Mini-itx) running higher level functions in the context of the Southampton AUV.
Zenon offers his own experience with zigbee and CC2530 communications between a PC and robotic drones.
Generally it is accepted that this model, of a linux box overseeing multiple MCU's is worth pursuing at the individual vehicle level.

Peter summarises by speculating that within the swarm the communication traffic between the Linux box and MCU's would be less relevant, to which Zenon agrees adding that currently the fusion of data coming from the swarm is beyond the capability of a single MCU, reinforcing the need for individuals to be equipped with a more powerful computational platform for higher level functioning.

Peter questions whether the swarm data and onboard agent could be used to 'educate' an MCU for more sophisticated behaviour.
Zenon confirms that this is currently the only practical way to create and distribute a new learned behaviour.
Peter comments that the linux box on board needs to be able to sit in the middle of the process and not only form it's own world model, but also educate it's minions (in the form of the MCU) but take advice from the swarm and adapt accordingly. So the Linux brain on an individual is key to this division of labour.
Zenon suggests that it would be nice to explore automated survival schema that run on MCU's for the situation where the CPU has failed and a limited set of MCU's remain to recover the vehicle. Alex agrees this would be a really nice feature to have.
Peter suggests the use of genetic algorithms for generating and prioritising scenarios that are subsequently installed on MCU's to provide an adaptive way of fine tuning these schema.
Zenon agrees it would be possible and that it may even be possible to place adaptive seeds on an MCU allowing them to be somewhat adaptive once the CPU has failed and a vehicle wishes to return to a safe haven.
Gabby asks if the goal today was to define a couple of projects based on a defined platform, but the consensus was to continue the discussion and exchange of ideas for a little longer via email and reconvene the meeting within a week to focus on these more detailed issues.

Meeting concludes 1303 16/09/2011