On Conceptual Design of Intelligent Mechatronic Systems

George Rzevski

Magenta Corporation Ltd, London EC3V 3ND, UK
Department of Information Systems and Computing, Brunel University, Uxbridge UB8 3PH, UK

Published in:
Mechatronics 13 (2003) pp 1029 – 1044

Abstract – We have technology now to design networks of small intelligent units capable of competing and/or co-operating with each other on specified tasks and making decisions under conditions of uncertainty through a process of negotiation. In highly dynamic environments, such distributed systems are capable of achieving considerably better results in terms of performance/cost ratio and reliability than conventional centralised large systems and structures. The major elements of these systems are intelligent agents, which are software objects capable of communicating with each other, as well as reasoning about received messages. The paper discusses conceptual design of mechatronic systems based on multi-agent technology.

Keywords – Conceptual design**;** Intelligent machines; Intelligent mechatronics; Intelligent networks; Multi-agent systems.

1. Introduction

Current global market conditions are volatile and unpredictable. In the post-industrial society [1] supported by the information economy [2] we shall have to learn to live with complexity, dynamics and uncertainty of demand and supply conditions. Traditional automated systems are rigid and are not capable of responding rapidly to changes in demand and supply. The automation, in its present form, does not deliver agility. My recent experience as a consultant to a major automobile manufacturer confirmed how large are disruptions and consequent costs caused by frequent changes in order specifications. These losses are rarely publicised but they are real and are likely to increase with time.

It follows that it is necessary to develop a new design philosophy for both organisational and technological systems, a philosophy that will ensure that systems are able rapidly to respond to unpredictable changes in their environments with a view to maintaining or improving their performance [3 - 5].

Digital information and communication technologies have reached the level of development that enables designers to achieve this objective.

The massive use of digital technology is assured by its continuous improvement in performance/price ratio. According to the well-known Moore's Law, every eighteen to twenty four months chip density and hence computing power doubles while cost remains constant. We have good evidence that since 1960s the development of computer technology has strictly followed this law. The implication is that in the near future the cost of electronic tags will be less than that of barcodes. We can expect therefore physical objects, including living systems, to be tagged and thus endowed with the ability to communicate with each other, opening extraordinary opportunities for advanced mechatronics.

Artificial Intelligence (AI) has matured and is now capable of providing innovative solutions to many practical problems where there is a need to replace automation with intelligence. This is particularly true for Distributed AI as exemplified by multi-agent systems. Some commentators have named the new interest in AI as "the second coming of artificial intelligence".

2. Fundamental Concepts

Let us review key concepts underlying conceptual design of intelligent mechatronics systems, as used in this paper.

2.1. Mechatronic Systems

It is quite common now to refer to multi-technology systems that include mechanical, electrical, electronic and software components, as Mechatronic Systems [6], [7].

I propose here to classify mechatronic systems according to their behavioural characteristics into

- Automated Mechatronic Systems
- Intelligent Mechatronic Systems
- Intelligent Mechatronic Networks

An *Automated* mechatronic system is capable of handling materials and energy, communicating with its environment and is characterised by self-regulation, which enables it to respond to predictable changes in its environment in a pre-programmed fashion. An overwhelming majority of current mechatronic systems belong to this category. These systems are not equipped to cope with the complexity, dynamics and uncertainty inherent in new markets and will not be considered in this paper.

An *Intelligent* mechatronic system is capable of achieving given goals under conditions of uncertainty. In contrast to automated systems, which are, by definition, pre-programmed to deliver given behaviour and are therefore predictable, intelligent systems may arrive at specified goals in an unpredictable manner. They are endowed with flexibility, which means they are capable of responding to frequent changes in their environments without being re-programmed. This qualitative difference in their behaviour is a result of the separation of the domain knowledge from the mechanism for problem solving.

Intelligence can be designed into a system using traditional AI methods such as expert systems, fuzzy logic or neural networks, but the most cost-effective and powerful implementation is through the use of distributed artificial intelligence, where a community of intelligent agents decides on the optimal or near-optimal action through a process of negotiation.

Examples of such systems include intelligent machine tools, intelligent robots, intelligent geometry compressors, autonomous road vehicles, self-parking cars, pilot-less aircraft and goal-seeking missiles. Autonomous mechatronic systems will be referred to in this paper also as Autonomous Mechatronics Systems or simply Intelligent Machines.

A most interesting variety of intelligent systems is a network of mutually interconnected intelligent systems, or an *Intelligent Mechatronic Network*.

Intelligent mechatronic networks are capable of deciding on their own behaviour by means of negotiation between constituent autonomous units (the network nodes). Each of constituent units is itself an intelligent mechatronic system. Even more impressive is their ability to improve their own performance by *self-organisation* (changing relations between constituent components with a view to improving the overall network performance).

The most advanced intelligent networks pursue a continuous *evolution* (disconnecting and thus eliminating less useful constituent units and connecting new units perceived by the network to be beneficial for achieving current or future goals).

Fleets of spacecraft, colonies of intelligent agricultural machinery, intelligent manufacturing systems and swarms of intelligent parcels are examples of such networks. Self-organising and evolving networks will almost certainly dominate the next decade as the most sought after engineering systems.

2.2. Intelligence

There is no agreed definition of Intelligence. It is considered to be too complex a concept for a neat and precise definition. My view is that if we call a class of systems "intelligent", we should define in what way these systems differ from the rest.

I suggest that the following definition of intelligence is quite adequate for our purpose:

Intelligence is the capability of a system to achieve its goals under conditions of uncertainty.

Where, the uncertainty is caused by the occurrence of unpredictable internal events, such as component failures, and/or external events, eg, unforeseeable changes in the system environments.

To exhibit intelligent behaviour a system must have access to the knowledge on the domain in which it operates, and to act upon this knowledge in response to, or in anticipation of, external inputs (rather than to passively react to input data in a pre-programmed manner). In most cases, to "act upon knowledge" means selecting a pattern of behaviour, which takes advantage, or neutralises undesirable consequences, of unpredictable events. It is important to note that when an intelligent system meets a new problem it must find a solution by the trial-and-error method, just like human beings [8].

2.3. Distributed Intelligence

The term Distributed Intelligence implies that the system has many interconnected decision-making units that share the responsibility for system behaviour. Each unit may access the centrally stored knowledge and/or its own local knowledge, the latter arrangement usually improving the overall system performance. A distributed intelligent system is typically a network with decision-making unites as nodes and communication channels as links. The key feature of a distributed intelligent system is its *Emergent Intelligence*, that is, intelligence created through the interaction of stakeholder units. Relatively simple units when connected into a complex network are capable of generating a rather superior intelligent behaviour [9]. Such systems are often compared to colonies of ants or to swarms of bees.

2.4. Multi-Agent Technology

Distributed intelligence is very often implemented by means of the multi-agent technology [10]. The key elements of this technology are Intelligent Agents [11].

An *Intelligent Agent* (also called *Smart Agent* or *Software Agent*) is a software object capable of communicating with other intelligent agents, as well as with humans, with a view to achieving a given task.

Intelligent in this context implies being capable of

- Comprehending tasks that need to be performed
- Choosing the most effective strategy and tactics for the task in hand
- Selecting relevant correspondents (other agents or humans)
- Composing meaningful messages and sending them to selected correspondents
- Interpreting received messages
- Making decisions on how to respond to the content of received messages making sure that the decision contributes to the achievement of system goals
- Acting upon these decisions

A *Multi-Agent System* (*Swarm of Agents*, *Team of Agents*, *Society of Agents*) is a system consisting of intelligent agents competing or co-operating with each other, with a view to achieving system objectives. The process of negotiation between agents creates the emergent intelligence of the system. There is evidence to claim that the larger the number of agents within a multi-agent system, the greater its Emergent Intelligence.

The architecture of multi-agent systems as developed by Magenta Corporation based on my ideas is given below.

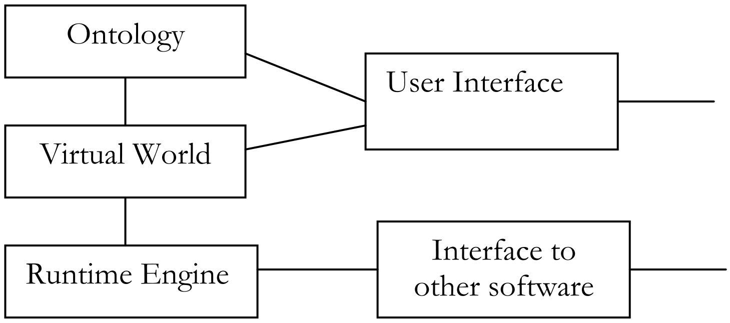

Fig 1. An Architecture of Multi-Agent Systems

Ontology contains extensive specification of knowledge on the domain in which the system operates. The knowledge is structured in terms of classes of *Objects*, *Properties*, *Attributes*, *Scripts and Relationships* and thus resembles an augmented semantic network. The performance of agents critically depends upon the quality of the domain knowledge stored in ontology and associated databases. Ontology can be modified by users of the system and in some cases will evolve through internal processes of eliminating its own useless components and experimenting with new ones.

Virtual World is the place where software agents are created when needed, where they interact with each other and are destroyed when their useful life comes to an end. Since each software agent represents a person, organisational unit or a physical object from the domain under consideration, the Virtual World is a dynamic model of the Real World. The challenge is to design a Virtual World that will reflect all relevant situations observable in the Real World, and all changes of these situations.

Runtime Engine together with *Extensions* contains all the algorithms and protocols required for proper functioning of agents. The Engine is as complex as a multi-tasking operating system. It supports parallel running of a very large number of agents and enables their interaction at great speed. The current version of the engine supports typically 500,000 agents working in parallel and exchanging 50,000 messages per second.

Interface links the multi-agent system with users and other software. The interface with other software is based on international standards, including XML and COBRA. Multi-Agent Systems can be ported to all standard platforms.

For the projects described in this paper, individual agents have been designed to be relatively simple. They act within rules, guidelines and constraints stored in ontology scripts. The intelligence of the system *emerges* from the interaction of a very large number of simple agents.

3. Principles of Conceptual Design

Conceptual design is an early stage of design in which designers select concepts that will be employed in solving a given design problem and decide how to interconnect these concepts into an appropriate system architecture.

One of the most important secrets of the successful design is to keep design options open as long as possible. At the beginning of every design process there is a large variety of candidate solutions to a given design problem and a considerable uncertainty about which of these solutions will be best suited to the given specification. This is particularly true when the designer has to meet highly dynamic or badly articulated user requirements. The fundamental rule is – keep reducing the uncertainty in a controlled manner, step by step. Delay committing yourself to a particular design solution until absolutely necessary.

One way of keeping as many as possible design options open is to postpone the selection of physical components, which will be included in the final system, and initially limit the design considerations to choices of concepts. In other words, first select concepts, which will be employed in solving a design problem, then decide how to interconnect these concepts into an appropriate system of concepts, and proceed to Physical Design, that is, to selecting a physical implementation for each constituent concept [12] only after a thorough validation of the Conceptual Design.

The equally important design trick is to abandon the conventional end-to-end design process and to consider all aspects of a design problem concurrently – Concurrent Engineering. This is particularly important for multi-technology systems, where materials handling, energy conversion and information processing interact and a high-quality design cannot be achieved without simultaneously considering all three flows [13], [14].

Conceptual design normally results in a diagrammatic description of links between conceptual blocks, known as *System Architecture*. The term Architecture has however a far wider meaning. In all domains in which it is used it describes how a system fits into its environment and how system components interface with each other. For example, the term Machine Architecture signifies the interface between hardware and software in a computer. Building Architecture describes how a building interfaces with its environment and with its users, and how building components are connect to each other, eg, how windows and doors fit into elevations and how walls and roofs are interconnected. Standard architectural solutions are usually given distinctive names. Thus, in computer engineering we talk about advantages and limitations of Von Neumann Architecture. In building industry we distinguish features of Gothic, Norman, Tudor or Georgian Architecture. Similarly, in mechatronic systems we argue about differences between Hierarchical and Networked Architecture.

During the conceptual design of mechatronics systems the main architectural choice is between a hierarchy and a network.

The conventional wisdom teaches designers to take full advantage of the economy of scale. It proclaims that building large systems will be always more economical than constructing a large number of small ones. Since large systems require an architecture that is reasonably transparent to enable an effective management, only one type of architecture satisfies this requirement – the hierarchy. Big organisations are partitioned into divisions; divisions into departments, departments into sections, etc. Big mechatronic systems are partitioned into subsystems; subsystems into modules, modules into sub-modules, etc.

Centralised command and control hierarchies have dominated our lives in social, political and business arenas. The doctrine was in force for centuries and is so ingrained in our subconscious that it is rarely questioned.

It is now time to realise that the economy of scale is valid only under one condition, namely, that the environment into which the system is built is stable. When conditions in that environment are subject to frequent and unpredictable changes, large systems exhibit a weakness. They are over-organised, ie, their constituent units, designed for efficient passing of instructions and reporting, are not capable of autonomous decision-making, creativity and innovation.

The organisational structure that is more suited to dynamic environments is a network of autonomous units, that is, units empowered to make decisions following flexible, agreed general rules, without waiting for instructions. In organisational systems networked structures are implemented through the teamwork and extended enterprise concepts whilst in mechatronics systems by designing the units to have communication and decision-making capabilities, in other words, endowing them with artificial intelligence.

There is a real possibility that a part of the conceptual design of mechatronic systems will be, in the near future accomplished by negotiation between intelligent agents [15], [16].

The above discussion could be summarised as a number of conceptual design principles.

(1) *The two key characteristics of any system aimed at operating in a dynamic environment should be Agility, ie, the ability to rapidly change the system behaviour in response to, or in anticipation of changes in its environment and*

Autonomy, ie, the ability to decide when and how to change the system behaviour without waiting for external instructions.

It is important to note that to achieve the required responsiveness systems must have some built-in redundancy. The lean system by definition cannot be agile. To achieve autonomy systems must have highly developed perceptive and cognitive components, which implies complex information processing and knowledge management modules. Autonomous systems are by definition to a certain degree unpredictable.

(2) *Systems should be designed as Networks of Autonomous Units, rather than Hierarchies.*

The Metcalf's Law, states that *the value of a network is equal to a square of the number of its nodes.* The implication is that the increase in utility of a network, as it grows, is polynomial whilst the increase in expenditure for building extra nodes is linear. Therefore the economy of networking is much more powerful than economy of scale. The Internet is of course the prime example of Metcalf's law at work. As the number of computers connected to the Internet increases, the value for users of being connected goes up in a non-linear fashion. In other words, by connecting smaller units into networks we generate certain *emergent* behaviour not detectable when the same units are independent.

The scope of this law encompasses human networks. We have all experienced the *emergent* performance of a team that is more than a sum of performances of individuals.

There is an additional reason for favouring networks over hierarchies. When the decision-making is distributed to network nodes, which are close to sensors and actuators, the system is capable of reacting far more swiftly to unexpected events than a centralised system with long reporting/instruction paths between information sources and executive mechanisms. The same applies to human networks.

(3) *All decisions on system behaviour should be made through negotiation among affected constituent units. Negotiations must lead to the increase of the specified overall System Value.*

Since most decisions affect more than one node in a network, it is necessary to involve all affected units in the decision-making process. The negotiation is the mechanism to support this involvement. Also, when the power of decision-making is devolved to constituent units, there is a need for a mechanism that would ensure that decisions are made with a view to improving the overall system performance. The rule must be that every outcome of a decision should increase an overall performance measure. This step-wise increase in performance through negotiation is analogous to the distributed hill-climbing optimisation technique or to numerical relaxation method.

(4) *In pursuing the goal of increasing the system value, units may, at their discretion and with the approval of affected units,*

- *Compete and/or co-operate with each other (Autonomy)*
- *Construct, deconstruct and reconstruct links with each other (Self-organisation)*
- *Disconnect units considered ineffective and connect new promising units, temporarily or long-term (Evolution).*

This principle effectively gives freedom to network nodes to pursue any strategy that will increase the overall system performance. The observations of networked systems in operation confirm that in complex applications the ability to switch occasionally from one strategy to another offers a substantial overall performance improvement.

(5) *During the conceptual design stage, considerations of concepts related to the three fundamental elements of a mechatronics system, namely, processing of information (communication and control), conversion of energy and movement of materials should be done concurrently and interactively.*

Current mechatronics design practice of considering mechanical structures, energy systems and control and communication systems independently, in an end-to-end fashion, and then attempting to integrate them, is wholly inappropriate for problems characterised by complexity, dynamics and uncertainty, as illustrated by the case study below.

4. A Case Study

Let us look at a real but suitably disguised case study that I have used recently when delivering Masterclasses in conceptual design of mechatronic systems to senior technical personnel of a company designing and supplying complex systems for warships. The case study goes as follows.

A supplier received an order for two warships that are to be controlled by sophisticated digital systems and thus could be considered as a new type of a highly advanced mechatronics system. The ships were designed and built but could not be made to work in time for delivery. The problem that caused endless delays and overspending could be described in very simple terms: the performance of the three key constituent systems of the ship, namely, the power system, communication system and weapon system, could not be synchronised for the warship to be able to fulfil its main function, that is, to execute the precision launching of missiles without interference from power and communication systems.

The delivery of the warships to a client had to be delayed for a year, and it took the supplier further two years after the delivery to remedy the situation and achieve the required level of synchronisation.

This was clearly a conceptual design failure. A number of wrong decisions were made in the early stages of design. Firstly, it was a mistake to attempt to design separately the three constituent mechatronics systems (the plant) and their control system. The concurrent design would have ensured a better co-ordination. Secondly, designers should have recognised that the operating conditions of a sea-going vessel involved in military engagements would be extremely volatile and unpredictable, and that the level of synchronisation demanded by the new military and communication technologies incorporated into the weapon, power and communication systems would be so high that the situation called for a distributed intelligent control rather than a conventional centralised control system. A network in which the three critical systems are nodes empowered to autonomously negotiate with each other when and how to achieve the required synchronisation would be relatively straightforward to design, simulate and implement, but designers had no required skills to do that.

5. Examples of Conceptual Designs

The following examples are mostly based on my personal experience, unless otherwise stated.

5.1. An Intelligent Machine Tool

My first attempt at designing an intelligent mechatronics system followed the ideas expressed in the seminal work on the philosophy of distributed control in humans, *The Society of Mind,* by Marvin Minsky [17]. The prototype was developed as an early experiment in testing usefulness of multi-agent systems in manufacturing. The problem was formulated as the design of a distributed information processing system for a simple metal cutting machine tool. The material and energy flows in the machine were not considered. The validation was done by simulating rather than building the proposed design.

A requirements analysis showed that major information processing functions were: (1) Controlling processing speed, (2) Scheduling, (3) Condition monitoring, (4) Ensuring safety and security and (5) Record keeping and reporting. Clearly these functions combined and cut across conventional machine control systems, job shop scheduling systems, maintenance systems, accounting systems, etc. The novelty was that by considering a machine-tool as a node in a workshop mechatronics network, the design problem was greatly simplified and, at the same time, the information system was made much more flexible – nodes could be connected and disconnected at will whenever the host machine-tool would change its operational state, without the rest of the system being adversely affected.

The decision was made that an intelligent agent would control each information processing function, as follows:

- The *Performance Agent* was given the task of deciding and maintaining the optimal cutting speed.
- The *Maintenance Agent* was given the task of monitoring the condition of the tool. In the case of tool damage, the Maintenance Agent was programmed to initiate a negotiation with the Performance Agent whether to terminate the process, slow down, or continue, and replace the tool at the next opportune time, depending on the seriousness of the damage.
- The *Scheduling Agent* negotiated the loading of the machine tool with other Scheduling Agents by a kind of auction in which capacities were matched with orders.
- The *Safety Agent* monitored the immediate environment of the machine tool making sure that operators, or mobile robots, would not enter the danger zone.
- The *Bookkeeping Agent* kept records and sent reports on the machine operation.

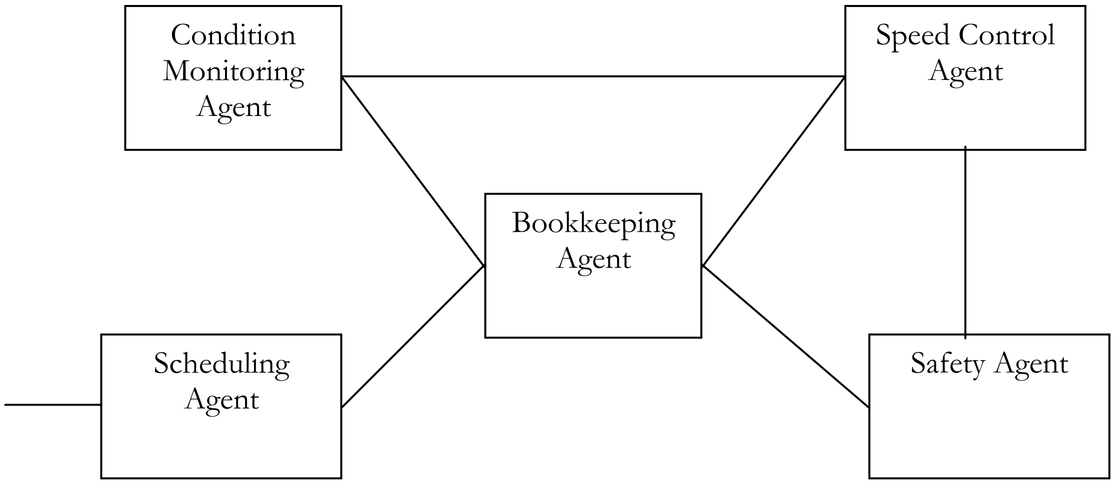

Fig. 2 A Multi-Agent System for a Machine Tool

Simple communication protocols in the form of production rules were used to guide the negotiation process and avoid stalemate. Each original prototype agent was a miniature expert system with a knowledge base consisting of up to 10 rules and a reasonable fund of facts. For example, the knowledge base of the Performance Agent contained data on characteristics of a selection of metals and rules for choosing optimal processing speeds for these metals. The knowledge base of the Maintenance Agent contained data on typical damaged tools and probabilities of each particular type of damage causing a tool breakdown.

Keeping agents highly specialised enabled them to operate each in a narrowly defined knowledge domain, which demanded only a small knowledge base for each agent. Having a collection of small knowledge bases instead of one large base considerably simplified the development and maintenance processes. The multi-agent system proved to be simple to develop and highly flexible. What was however even more important, the group of five agents showed clearly an *emergent* behaviour, which was far more sophisticated than the behaviour of each individual agent.

5.2. Intelligent Geometry Compressor

Axial turbo compressors are used in many areas of industry where large quantities of air or gas have to be moved or compressed. Typical examples are in jet engines, the larger gas turbines, gas-line pumping and in many process plants. All turbo compressors are limited in their performance by the aerodynamic phenomena of stall and surge, where the flow of the gas becomes unstable and can reverse in direction. Stall and surge, if allowed to develop, can cause significant mechanical damage to the compressor.

Present designs of axial compressors use fixed geometry rotor blades and fixed geometry stator vanes, with a limited capability to vary vane angles against pre-set limits, using simple control algorithms. The operating point for the compressor is designed to give an adequate safety margin from the surge line, therefore avoiding the possibility of stall or surge in operation. This *surge margin* puts a limit on both the work that the compressor can do and it's efficiency. For example, large industrial turbo compressors can absorb powers of the order of 50 MW, with annual operating costs of between £5M and £20M, depending on the duty cycle and power charges applied. It is understandable therefore that the cost for even a small drop in efficiency due to a provision for pressure and/or flow with adequate surge margin, is very significant. The avoidance of surge has important safety implications in aerospace. Surge under extreme manoeuvres has resulted in the loss of aircraft, the most public of which was the loss of the Russian Tu 144 Supersonic Transport at the Paris Air show in 1973, with the loss of 14 lives.

Supported by a group of industrial organisations and experienced turbo-machinery designers, the author’s team decided to tackle the problem of aerodynamic instability in compressors in a novel way. The decision was made to reconsider the fundamentals of compressor design by removing the usual assumption of fixed or partially variable geometry and to apply concepts from the intelligent network paradigm.

As a result, a design of an axial compressor with variable geometry emerged, where an intelligent agent individually controls each movable element. Agents are then connected into a network and empowered to negotiate among themselves relative positions of movable elements with a view to achieving a performance as close as feasible to the optimum under continuously changing aerodynamic conditions. The overall behaviour of the compressor emerges from the interaction of agents.

The proposed intelligent geometry compressor will operate by using sensors to monitor the aerodynamic conditions at each movable element. Sensor information will be used by local agents, which, through the process of negotiation, will make control decisions and instruct actuators to incrementally adopt the flow path geometry that ensures optimum performance for current aerodynamic conditions.

The intelligent geometry compressor is a genuine mechatronic system comprising mechanical components characterised by variable geometry (eg, vanes), sensors, actuators, digital hardware, software and embedded artificial intelligence.

The implications for reliability are staggering. Utilising embedded processing power it becomes feasible to design into the compressor:

- Self-diagnosing (monitoring compressor conditions and identifying faults when they occur),
- Self-repair by reconfiguration (isolating faulty parts and thus making them harmless)
- Graceful degradation of performance (repositioning remaining healthy parts to achieve a reduced but acceptable level of performance; also, in case of a serious failure of critical elements, such as actuators, agents can revert to the fixed geometry mode of operation)

Early simulation results indicate that an axial compressor designed as a network of movable elements could be operated over a significantly enlarged envelope without risk of stall or surge [18].

6. Intelligent Mechatronic Networks

It is now economically feasible to attach electronic tags to physical objects and to supply people with smart cards containing quite sophisticated silicon chips, which communicate with each other at a distance. This technology enables people and physical objects to be linked into intelligent networks.

Schemes of this kind are already in operation and show a considerable advantage over conventional systems. For example, several Swiss skiing resorts have operated smart card based systems for controlling access to ski lifts for several years now, making handsome profit and serving satisfied customers. Skiers walk with their smart cards tucked in their pockets passing the tag readers located at critical access points throughout the resort, without ever having to show their passes or actively operate electronic locks.

6.1. Intelligent Logistics Network

The project is based on the assumption that in the future, as a result of rapid changes in demand and supply, the transportation capacity will be considered as a commodity and that the transportation customers and service providers will trade in transportation options [19]. After acquiring transportation options customers would release their goods packaged as Intelligent Parcels into an Intelligent Logistics Network expecting parcels to autonomously find their way to their destinations. During the transportation, customers will be given opportunities to communicate with their intelligent parcels via the Internet in case of a need to change their destinations. Parcels will be able to autonomously re-negotiate the required transportation facilities to their new destinations. The whole transportation process will be monitored and displayed on a website.

The main physical components of an intelligent logistics network are:

1. *Intelligent Parcels*, equipped with electronic tags capable of storing information on weight, dimensions and storage conditions of constituent goods, their destination and the credit limit up to which the parcels are allowed to pay for re-routing without asking the owners for approval. The tag may be passive or, exceptionally, contain an agent capable of negotiating the parcel's routing.

2. *Intelligent Parcel Processors*, located in parcel reception and loading areas, on parcel handling robots, in warehouses and on transportation facilities. The Processors are capable of communicating with electronic tags at a distance, reading and rewriting their content and making decisions concerning the manipulation of parcels (picking them up, storing them, loading them, etc.). They are also connected via the Internet or intranets/extranets to websites accessible to customers and transportation suppliers.

The proposed Intelligent Logistics Network is a genuinely distributed, self-organising logistics system offering a reduction in transportation times and costs. In addition, by its capability to re-route parcels and thus avoid unnecessary transportation, it will offer an ecologically sound solution to the current environmental damage caused by the excessive movement of goods across the globe.

It is quite obvious how a logistics network described above could be interconnected with a retail network in which intelligent parcel processors are placed on retail shelves, in customer trolleys and checkouts enabling customers to select their purchases, place them in their trolleys and push them to their cars without queuing, whilst the system calculates the amounts their spent, checks their credit rating, asks for permission to charge their charge cards, conducts the transactions and sends requests for stock replenishment.

6.2. Intelligent Network of Agricultural Machinery

Current agricultural tractors are far too heavy and expensive. Because of their weight they tend to damage the soil structure, which requires expensive remedies, and because of their costs only large agricultural complexes can use them. According to some estimates by agricultural experts almost a half of the tractor weight and cost is due to the requirement to have a cabin and associated equipment, whose sole role is to protect operators and provide them with reasonably comfortable working conditions. In spite of this operators are not satisfied with their working conditions primarily because they are bored.

It is perfectly feasible with current technology to replace heavy tractors with networks of small and agile agricultural machinery equipped with sensors and actuators and controlled by networks of agents. A network of such machinery would behave not unlike a colony of ants. Because of their scalability such networks would be suitable for work on small and large fields. They could help us go back to small farming.

6.3. An Intelligent Family of Robots

Although the UK space effort has been reduced almost to zero, there are some small funds available for innovative work in space exploration. A project has been recently completed concerned with the development of technologies for the autonomy and robustness in space, involving three universities and two spacecraft manufacturers from this country. Consortium members have developed conceptual prototype robots for the exploration of Mars. Following the design principles described in this paper, the originally envisaged single robot has been replaced by a family of five much smaller intelligent robots. Each member of the family has a limited intelligence and is potentially able to undertake simple tasks such as placing scientific instruments onto a correct location and to provide a variety of services to other members of the family, e.g., cleaning their solar cells if they get covered by the space dust and helping them to get out of small crevasses. The cost and weight per unit performance for this family is likely to be below the cost and weight of an equivalent single robot. Their size would offer an important advantage in packaging for launch and delivery.

6.4. Other Intelligent Networks

Perhaps the most interesting and advanced work on the development of intelligent networks is taking place in the USA space sector. Fleets of small low orbit communication satellites are being built instead of large and expensive geo-stationary units. Armadas of very small space ships, capable of communicating with each other and providing a limited mutual support, will be sent into the deep space to relay scientific information at a cost of an order of magnitude less than an equivalent single spacecraft. More information is available on www.nasa.gov

7. Conclusions

Contrary to a belief expressed in many publications that Engineering is Applied Science, I base my research and development work on the premise that Engineering is a synthesis of Art and Science. The Science component provides the universal laws of behaviour enabling engineering designers to predict physical behaviour of engineering objects and systems. The Art component makes Engineering a discipline firmly rooted in the socio-economic context and therefore dependent on time and location. At the beginning of the 21st Century in well developed Western countries we are in transition from manufacturing-based (industrial) to information-based (service-oriented) socio-economic system. As a consequence it is counter productive to continue designing engineering systems as well-integrated large hierarchies. Under new market conditions characterised by high-level dynamics and frequent occurrence of unpredictable events it is necessary to change the conceptual engineering design paradigm to incorporate ideas of agility, intelligence, co-operation and networking.

I argue in this paper that the configuration best suited to new socio-economic conditions is a network of autonomous, intelligent decision making units able to arrive at decisions through the process of negotiation. The proposed paradigm is well illustrated by a number of case studies and examples.

Acknowledgements

I am most grateful to Professor Paul Wiese, formerly of Rolls Royse and Ricardo Aerospace, for suggesting and contributing to the Intelligent Compressor Project; and to Gwyn Morgan for developing an excellent multi-agent proof-of-concept simulation of autonomous compressor vanes. I acknowledge the creativity and skills of Dr Peter Skobelev and the group of Russian software developers from Magenta Development Ltd, Samara, who have developed a generic multi-agent engine, ontology and virtual world capable of supporting the design of intelligent mechatronics systems and networks. Thanks are due to DTI for supporting the Space Robots project, as well as to all my colleagues contributing to the conceptual design of the robot family.

References

1. Drucker, P; Post-capitalist Society, Butterworth-Heinemann, 1993.
2. Shapiro, C; Varian, H R; Information Rules: A Strategic Guide to the Network Economy, Harvard Business School Press, 1999.
3. G Rzevski; Strategic Importance of Engineering Design. Journal of Design and Manufacturing, Vol 2, No 1, 1992.
4. Rzevski, G; A Framework for Designing Intelligent Manufacturing Systems. Computers in Industry, 34 (1997) 211-219.
5. Rzevski, G, Prasad, K; The Synergy of Learning Organisations and Flexible Information Technology. AI and Society v12, pp87-96.1998.
6. Rzevski, G; Mechatronics: Designing Intelligent Machines. Butterworth-Heinemann in association with the Open University, 1995.
7. Burr, J; Mechatronics. PhD thesis.Technical University of Denmark, Copenhagen, 1990
8. Popper, K; Conjectures and Refutations: the Growth of Scientific Knowledge. Routledge & Kegan Paul, 1972.
9. Holland, JH; Emergence. Oxford University Press, 1998.
10. Rzevski, G; On Behaviour and Architectures of Autonomous Intelligent Agents: An Engineering Perspective. Proceedings First International Round-Table on Abstract Intelligent Agents, ENEA, Rome, 1993.
11. Muller, JP, Wooldridge, P M, Jennings, N R (eds.); Intelligent Agents III, Agent Theories, Architectures and Languages, Springer-Verlag. 1996.
12. French, MJ; Conceptual Design for Engineers. 2nd ed. Design Council, 1985.
13. Rzevski, G; FACADE: Concurrent Engineering Applied to Multi-Technology Products. Proceedings of the International Workshop on Concurrent/Simultaneous Engineering Frameworks and Applications, Lisbon, April 1995.
14. Stacey, M K, Sharp, H C, Petre, M, Rzevski, G, and Buckland, R A; A Representation Scheme to Support Conceptual Design of Mechatronic Systems. In Gero, J S, Sudweeks, F, (eds) Artificial Intelligence in Design 96. Kluwer Academic Publishers, 1996.
15. Stacey, M K, Rzevski, G, Sharp, H C, Petre, M, and Buckland, R A; Intelligent Support for Conceptual Design: A Flow Modelling Approach. Proceedings of ICED 97, August 1997.
16. Cambel, M; Cagan, J; Kotovsky, K; A-Design: An Agent-Based Approach to Conceptual Design in a Dynamic Environment. Research in Engineering Design, Vol 11, pp 172-192, 1999.
17. Minsky, M; The Society of Mind. Heinemann, 1985.
18. Morgan, G; Intelligent Geometry Compressor: The Application Of Multi-Agent System Technology to the Design Of Intelligent Machines. PhD Thesis, Brunel University, 2002.
19. Rzevski, G; Engineering Design for the Next Millennium: The Challenge of Artificial Intelligence. The 86th Thomas Hawksley Memorial Lecture, IMechE, 9 December 1998.