



Rapid response manufacturing through a rapidly reconfigurable robotic workcell

I-Ming Chen*

School of Mechanical and Production Engineering, Nanyang Technological University, Nanyang Ave, Singapore 639798, Singapore

Received 1 May 2000; received in revised form 5 September 2000; accepted 10 September 2000

Abstract

This article describes the development of a component-based technology robot workcell that can be rapidly configured to perform a specific manufacturing task. The workcell is conceived with standard and inter-operable components including actuator modules, rigid link connectors and tools that can be assembled into robots with arbitrary geometry and degrees of freedom. The reconfigurable “plug-and-play” robot kinematic and dynamic modeling algorithms are developed. These algorithms are the basis for the control and simulation of reconfigurable robots. The concept of robot configuration optimization is introduced for the effective use of the rapidly reconfigurable robots. Control and communications of the workcell components are facilitated by a workcell-wide TCP/IP network and device-level CAN-bus networks. An object-oriented simulation and visualization software for the reconfigurable robot is developed based on Windows NT. Prototypes of the robot workcells configured to perform the light-machining task and the positioning task are constructed and demonstrated. © 2001 Elsevier Science Ltd. All rights reserved.

Keywords: Modular robot; Component technology; Agile manufacturing; Reconfigurable automation

1. Introduction

An automated manufacturing system usually consists of a collection of material-processing and handling devices such as CNC machines, robots, part feeders, conveyors, and sensors. These devices are traditionally designed and commissioned with the intent that they will be operated with few significant changes so long as the specific product is being manufactured. Certain flexibility has been given to those devices through programmable controllers. However, it is time consuming and not cost effective to reconfigure them for other products. In the rapidly changing and highly competitive global economy environment, such drawbacks reduce the attractiveness of automation systems for manufactures that are involved in high-mix, low-volume production. For manufacturers adopting the fixed automation systems, not being able to respond quickly to market changes will place them at a disadvantage. The Nanyang Technological University and Gintic Institute of Manufacturing Technology thus initiate a project aiming at improving

agility and flexibility in automated manufacturing systems.

From the aspect of product design, component standardization through a modular architecture has clear advantage in the areas of cost, product performance and product development. Modular product architecture directly links to the effectiveness of a manufacturing system [1]. In the study of the computer industry, from hardware, software to VLSI design, it is found that modularization plays a very important role in the advancement of the entire industry. Modularity even reshapes the firms and markets that “play host” to the evolution of a set of modular designs [2]. Instead of focusing on the outcome from a manufacturing system, the project here looks into changing the architecture of the manufacturing tools so that they can be rapidly configured and deployed based on the functional needs. The key to the concept of rapid reconfiguration and deployment lies in the “plug-and-play” component-based technology. In this project we aim to develop a rapidly reconfigurable robotic workcell. As the robots are complex elements in a manufacturing cell, kinematics, control and coordination of robotic systems are sophisticated compared to other elements/equipment in a cell. A reconfigurable robotic workcell can fairly set the benchmark

* Corresponding author. Tel.: + 65-790-6203; fax: + 65-791-1859.
E-mail address: michen@ntu.edu.sg (I-Ming Chen).

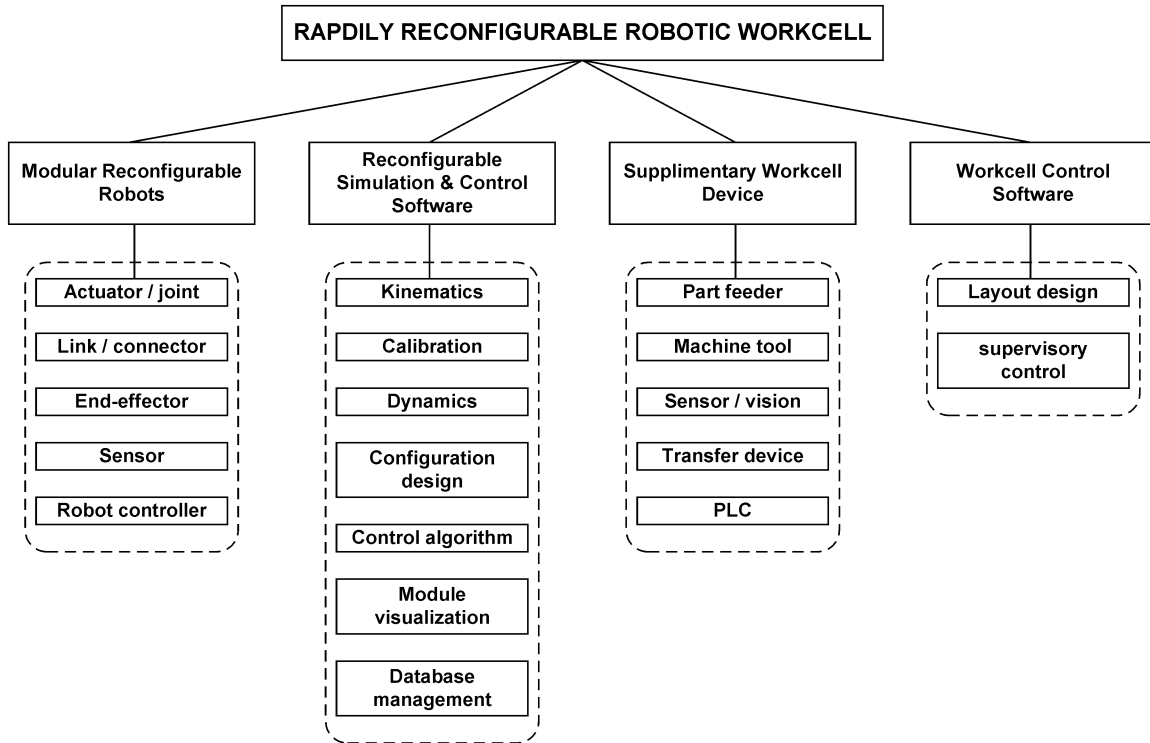


Fig. 1. Rapidly reconfigurable robotic workcell system.

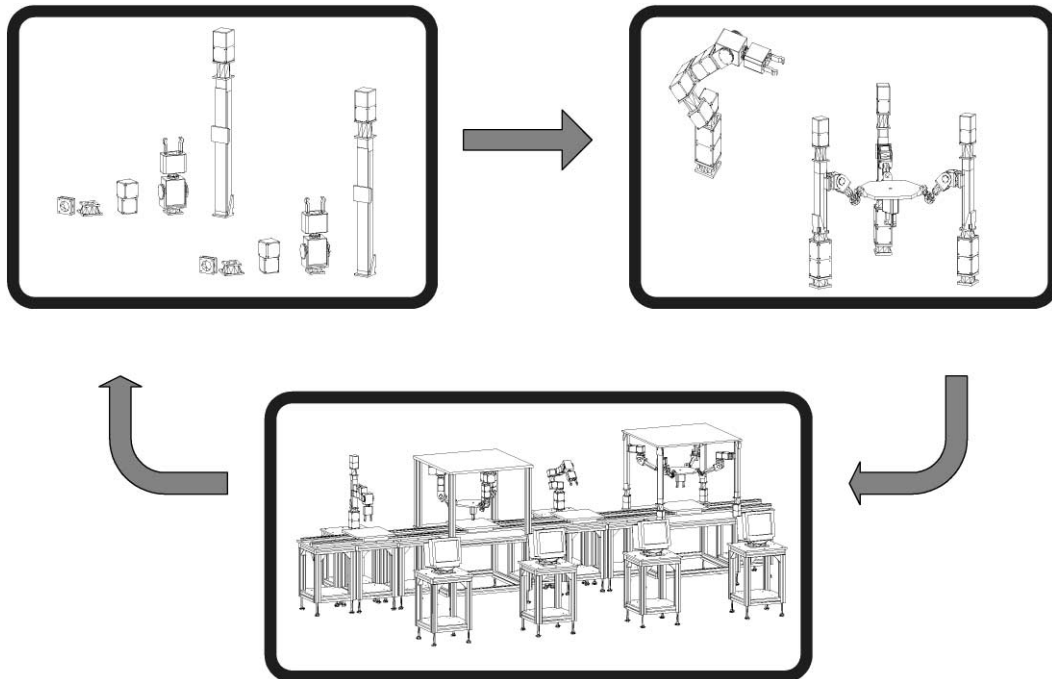


Fig. 2. Deployment of a rapidly reconfigurable robotic workcell.

on the future trend of reconfigurable manufacturing systems. Fig. 1 illustrates the system layout of a reconfigurable robotic workcell. In this system, workcells are made of standard interchangeable modular components, such

as actuators, rigid links, end-of-arm tooling, fixtures, and sensors. These components can be rapidly assembled and configured to form robots with various structures and degrees of freedom (DOF). The robots, together with

other peripheral devices, will form a complete robotic workcell to execute a specific manufacturing task or process (Fig. 2). The corresponding intelligent control and simulation software components are then reconfigured according to the change of the workcell configuration. The maintenance and upgrade of the system are simplified by replacing the malfunctioned or outdated components. Converting a manufacturing line from one product to another can be very fast in order to keep up with the rapidly changing marketplace. Through the Internet and the E-commerce drive, it is even possible to set up mass customization production systems [3] based on the reconfigurable system.

This article covers the following five major aspects of the development of a reconfigurable robotic workcell prototype, namely:

- (1) *Robot/workcell hardware component design* — to cope with complex functions required by the robots or other devices through the use of mixed types of modules while maintaining reconfigurability and inter-operability (Section 3).
- (2) *Reconfigurable and “plug-and-play” kinematics and dynamics modeling* — to ensure rapid deployment and unified control operation of the robots with different structures and DOFs (Section 4).
- (3) *Robot configuration optimization* — to establish optimal and effective use of the rapidly configurable system based on an appropriate selection of module components according to the task requirements (Section 5).
- (4) *Control of robot/workcell components* — to provide reconfiguration capability at the workcell level and coordinate tasks among various workcell devices and robots (Section 6).
- (5) *Simulation software for robot/workcell* — to produce CAD-featured environment for the reconfigurable system for visualization and simulation, and provide real-time interface to the control of reconfigurable robots (Section 7).

A prototype of the robotic workcell configured to perform light-machining tasks has been constructed and demonstrated in the International Industrial Automation Exhibition (IA'99) in Singapore in October 1999. This workcell consists of a 7-DOF redundant serial-typed robot to pick and place the workpiece, a 6-DOF articulate parallel modular robot to machine the workpiece, and a 1-DOF linear motion stage to move the workpiece in between the two robots. A workcell with a 3-DOF planar parallel robot and a workcell with a 6-DOF sliding-typed parallel robot are prototyped for positioning and machining purposes, respectively. The physical implementations of these workcells are covered in Section 8. Performance evaluation of the reconfigurable robotic workcell and trade-off using reconfigurable systems in current manufacturing facilities are discussed

in Section 9. Summary of the current work and future research aspects of the reconfigurable robotic workcells are concluded in Section 10.

2. Related research effort

The concept of the reconfigurable robotic workcell was originated from the research of modular robots. The modular robots can be further categorized into the industrial modular robot system, which is the focus of this project, and the self-reconfigurable modular robot consisting of identical units. In the modularization of industrial robots, the granularity of the components is usually based on their basic functions, i.e. motion actuation and tooling. Thus, the design of modules is highly differentiated into actuator modules, passive joint modules, and tooling modules, etc. Several prototypes of industrial modular robotic systems have been developed and demonstrated including the “Reconfigurable Modular Manipulator System” (RMMS) [4], the “cellular robot system” (CEBOT) [5,30], and other modular systems [6–8]. Basically, these systems have serial-typed (or open-chain) geometry with large working envelopes. These serial-typed modular robots are suitable for assembly, trajectory tracking, welding, and hazardous material handling. Parallel modular robots are also developed for light-machining tasks [9]. As indicated in [9], modular design can reduce the development cycle of the parallel robot significantly. Furthermore, it allows a trial-and-error approach to construct a parallel robot that is impossible with the integrated design approach.

For self-reconfigurable robots, the major emphasis is on the autonomous reconfiguration capability, which requires high uniformity in the design of the modules. In the research of [10–12], the modules are all designed as fully identical self-contained units (can be termed as “atoms”) with actuation, communication, and intelligence capabilities. Because self-reconfiguration of a large number of modules is difficult to achieve under gravity influence, applications of such systems could be limited to outer space or under the sea.

In applying modular and reconfigurable concept in the design of a manufacturing system, however, the work of “Agile Assembly Architecture” (AAA) applied a similar uniform-module design approach for manufacturing assembly tasks [13]. In AAA, an entire assembly line can be set up by using a set of identical assembly stations. Each assembly station is capable of 4-DOF motions that are equivalent to the motion of a 4-DOF SCARA-type assembly robot. The tools are modularly designed so that they can be interchanged among different assembly stations for different assembly tasks. Because of the identical design, the kinematic characteristics of the assembly station are limited to assembly type of tasks.

3. Robot/workcell hardware components

The hardware components of the reconfigurable workcell are developed around multi-DOF robots and low-DOF motion stages. To cope with rapid change of task requirements in the production line while maintaining inter-operability of components, using mixed-module design is necessary. Robots with serial, parallel, or hybrid geometries can be constructed for tasks requiring different accuracy, stiffness, and dexterity, etc. Key components, such as actuators and grippers/fixtures, can be interchangeable among different robots and devices. In principle, all of the robots and motion stages in the workcell are constructed using standard *fixed-dimension modules* and easy-to-fabricate *variable-dimension modules*. The fixed-dimension modules include active and passive joint modules and gripper modules that can produce rigid motions. The variable-dimension modules consist of rigid links, connectors, and platforms that can be designed and fabricated in a very short time. The mechanical, communication, and power interfaces between the modules are all standardized to ensure interoperability. Through the use of mixed types of modules, task-optimized workcell configuration design and rapid deployment can be achieved at the same time.

3.1. Fixed-dimension modules

The standard fixed-dimension modules include actuator modules, passive-joint and spherical-joint modules, and end-effector modules. The actuator modules are required to initiate either rotary or translational 1-DOF motion. For the sake of modularity, the actuator modules used are compact self-contained mechatronic drive units. Each of the drive units contains a built-in motor, a controller, an amplifier, and the communication interface. The inter-module communication and power transmission are through inter-connection cables and interfaces. The end-effector module employs a similar self-contained design.

In this project, we utilize a series of self-contained mechatronic drives and end-effector units that are commercially available for rapid development. Both revolute (Fig. 3a) and prismatic (Fig. 3b) actuator modules are employed. The revolute actuator has a cubic or double-cube design with multiple connecting sockets so that two actuator modules can be connected in many different orientations. The prismatic module employs a lead screw mechanism with a variable stroke length. The mechanical connection between modules is through manually fastened bolts and nuts. The inter-module communication is through the CAN-bus protocol (Control Area Network) and the RS-485 serial interface.

Three types of passive-joint modules (without actuators) are in-house designed and fabricated for parallel robot configurations: the pivot joint (Fig. 4a), the rotary



Fig. 3. Standard actuator modules.

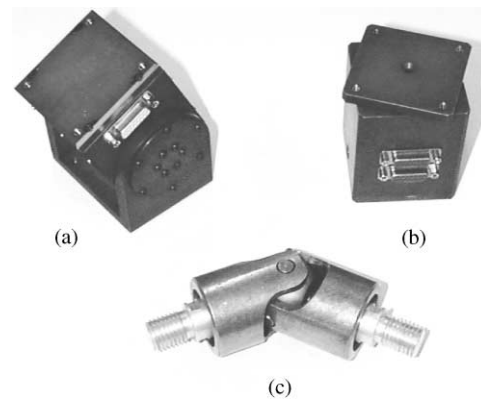


Fig. 4. Passive joint modules.

joint (Fig. 4b), and the spherical joint (Fig. 4c). Angular displacement sensors are built into the passive rotary- and pivot-joint modules for forward displacement sensing of the parallel robots. The sensor readings are transmitted to the robot controller through the CAN-bus. The spherical joint has 3-DOF motion capability serving as the connection between the moving platform and the legs of the parallel robot. There is no sensor built into this module.

In designing the compact active and passive modules, it is found that two crucial factors will affect the overall performance of the module: compact layout design of 3-D sub-components and thermal management of the modules. From our experience with the current system, the thermal management plays a crucial role in the reliability of the actuator module. And yet, this usually contradicts with the minimum space requirement for compact design. A new design paradigm for compact component layout with thermal management is necessary for further development of the reconfigurable workcell.

3.2. Variable-dimension modules

The robot modules like the rigid links, module connectors, and the mobile platform that can be custom-made

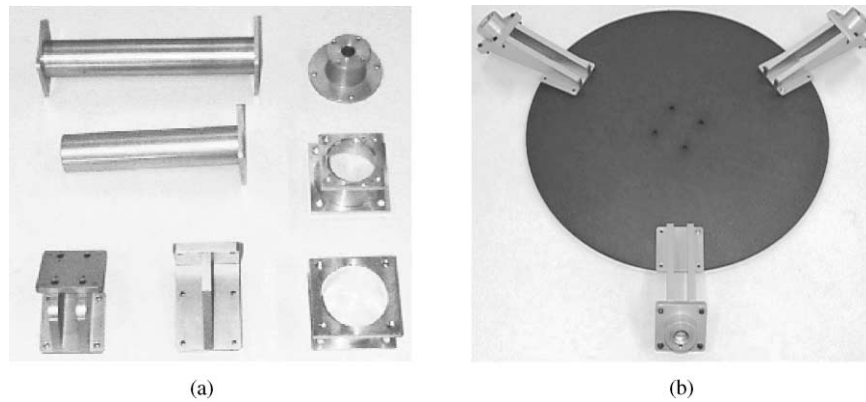


Fig. 5. Variable-dimension modules.

easily may have arbitrary dimensions. These modules usually have simple geometry and can be rapidly designed and fabricated based on the functional requirements. To allow for dimensional change in the module design provides end-users the ability to rapidly fine-tune the kinematic and dynamic performance of the completed robot, especially the size and geometry of the workspace and dexterity of the end-effector. Furthermore, the closed-loop structure of robots with parallel geometry imposes kinematic constraints on the dimensions of the robot sub-assembly. Thus, it makes the construction of a useful parallel robot configuration exclusively from standard modules difficult [9]. A set of links with various geometrical shapes and dimensions and a hexagonal mobile platform have been designed and fabricated as shown in Fig. 5a and b, respectively.

3.3. Interface design

To ensure inter-operability of the modules, interface design of all the modules must follow a standard. The design of communication and power interfaces needs to follow the industrial standard with specific form factors. However, the design of the mechanical interface may vary from systems to systems. The major considerations in designing a mechanical interface are rigidity, ease of connection, and accuracy. The rigidity of a connection is crucial to the performance of a reconfigurable system and is the first priority in the module design. The rigidity of the connection depends on the mechanical design, the material used, and the fabrication technique. Ease of connection is usually achieved through the design of the so-called “quick-coupling” mechanisms. Many modular robot systems use quick-coupling mechanisms to achieve the rapid configuration capability. However, extensive effort will be put in the design of such kinds of mechanisms. Furthermore, quick-coupling mechanisms are usually proprietarily designed and owned by the developers. To achieve inter-operability among systems developed by different vendors is very difficult. In our

module hardware design, a very simple manual connection method is used. It is based on the flange-typed rigid connectors/links (Fig. 5a). The flanges connect active and passive joint modules through bolts and nuts tightly. Stepping links and angled-plates are also designed with flanges. Although it lacks quick-connection capability, it allows different types of modules to be connected together easily. The accuracy of the connecting interface is maintained by specially designed locating Dove-pins in between the flange and the module. To maintain connection accuracy is important as robots are usually used for precision works. However, kinematic calibration techniques are developed here to specifically cope with the inherent inaccuracy in the module connections.

4. Reconfigurable and “Plug-and-play” robot modelings

For a robot system built from modular components without fixed DOFs and geometry, the derivation of the kinematic and dynamic models becomes difficult, as there are almost endless robot configurations to be considered. Conventional robot controllers require the kinematic models of the robot to be manually derived and hand-coded into the system. In principle, this is not suitable for reconfigurable robot systems. A new framework is proposed to automate the robot model generation process [14] (Fig. 6). This framework consists of a component database, a representation of modular robot assembly, and geometry-independent modeling techniques. The database stores the *static* CAD data of the components. A kinematic graphical representation of the assembly configuration of a modular robot, termed an *Assembly Incidence Matrix* [15], keeps track of the *dynamic* robot configuration data. Geometric modeling techniques based on the product-of-exponentials formula (POE), the modern approach for rigid body screw motion, utilize both static and dynamic data to generate the required models. When the robot is reconfigured, the corresponding kinematic and dynamic robot models are

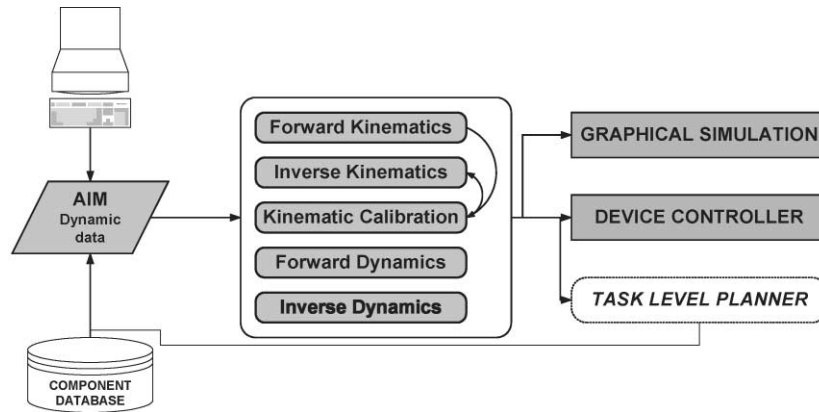


Fig. 6. Reconfigurable robot model generation.

reconfigured accordingly and are ready for control and simulation. This approach can be applied to all serial-typed robot configurations and a class of parallel-typed robot configurations.

4.1. Kinematic models

Unlike conventional robots using Denavit–Hartenburg (D–H) parameters for kinematic models, the kinematics of modular reconfigurable robots is formulated based on the local product-of-exponentials presentation [16]. The POE modeling method can uniformly describe the robot joint axes using generic line coordinates regardless of the type of the joints. Also due to the fact that POE is a complete description of the rigid body motion, not a minimal representation like the D–H parameters, the formulated robot kinematics has robust performance on the singularity problems that arise in kinematic calibration and numerical inverse kinematics.

The Local POE modeling method is defined on two neighboring robot links as follows. Let link $i - 1$ and link i be two adjacent links connected by joint i as shown in Fig. 7. Link i and joint i are termed as link assembly i . Denote the body coordinate frame on link assembly i by frame i . The relative position and orientation of frame i with respect to frame $i - 1$, under joint displacement q_i , can be described by a 4×4 homogeneous matrix as

$$T_{i-1,i}(q_i) = T_{i-1,i}(0)e^{\hat{s}_i q_i}, \quad (1)$$

where $\hat{s}_i \in se(3)$ is the twist of joint i expressed in frame i and $T_{i-1,i}(0) \in SE(3)$ is the initial pose of frame i relative to frame $i - 1$. $SE(3)$ represents the Euclidean group of spatial rigid body motions and $se(3)$ is the Lie algebra of $SE(3)$.

For serial-typed robot configurations, the construction of the forward kinematic model is an incremental and recursive process using Eq. (1) similar to the physical assembly of the modules. The incremental model construction can deal with serial and branching modular robot configurations with arbitrary DOFs. After the

forward kinematic model is completed, the differential of the forward kinematic function can be obtained. A general numerical inverse kinematics for modular robots with arbitrary DOFs is then formulated based on this differential. The solution of the inverse kinematics can be obtained through the iterative Newton–Raphson method [17]. Because the differential is expressed in $se(3)$, not the matrix differential, simulation results have shown that this numerical inverse kinematics converges in fewer steps than the numerical inverse kinematics using D–H parameters method, and is robust for trajectory tracking [17]. A closed-form inverse kinematics for reconfigurable robots using the *subproblem* approach [18] is also developed [19]. The subproblems are the solutions to the POE equations with two or three twist axes of different geometries. The complete inverse kinematics POE equation of a multi-axis robot is then solved by reconfiguring several subproblems representing the same kinematic sub-structures of the robot. Hence, the subproblems are also re-usable and reconfigurable.

For parallel-typed configurations, the derivation of forward and inverse kinematics becomes very complicated because both forward and inverse kinematics solutions may not be unique for certain configurations. Typically, the forward and inverse kinematics of parallel robots are derived based on a specific configuration and geometry. Here we proposed a unified Local POE approach to

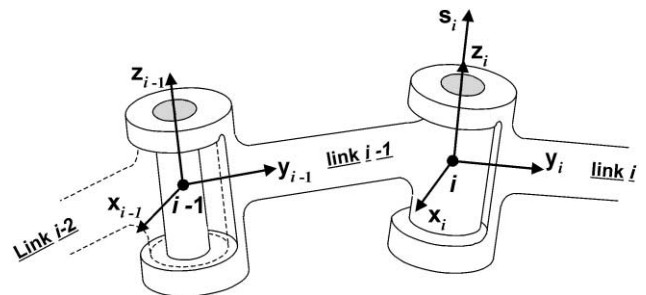


Fig. 7. Local POE model.

derive the forward and inverse kinematics for a class of non-redundant parallel robots having 3–6 legs [9]. For practical implementation reason, the forward kinematics is obtained using either the sensor-based or the numerical method. The inverse kinematics is determined by using the subproblem approach for the individual leg [18].

4.2. Dynamics models

The dynamics model of the robot is required for high-speed operation. The modular robot dynamics is based on Local POE representation as well. The formulation starts with the recursive Newton–Euler method. The generalized velocity, acceleration, and forces are expressed in terms of linear operations on $se(3)$, the Lie algebra of the Euclidean group $SE(3)$. Based on the relationship between the recursive formulation and the closed-form Lagrangian formulation for serial robot dynamics, the closed-form equation of motion can be derived [20]. The derivation of the parallel robot dynamics follows a similar approach.

4.3. Kinematic calibration

The machining tolerance, compliance, and wear of the connecting mechanism due to frequent module reconfiguration may introduce errors in positioning the end-effector. Hence, kinematic calibration is a must for modular robots with serial or parallel configurations. The robot calibration follows the Local POE representation of the robot kinematics. The Local POE model varies smoothly with the change of joint axes, which makes the model singularity free. In our proposed calibration model, the robot errors are assumed to be in the initial positions of the consecutive modules because the Local POE model is a zero reference method. Based on linear superposition and differential transformation, a 6-parameter error model is established for serial-typed robots [21]. This model can be obtained through the automatic generation process depicted in Fig. 6. An iterative least-square algorithm is employed to find the error parameters to be corrected. The corrected kinematic model is then updated in the robot controller for operation. Our simulation and experiment have shown that the proposed method can improve the position accuracy up to two orders of magnitude, or to the nominal repeatability of the robot after calibration with measurement noise. A typical 6-DOF articulate-type modular robot can reach a position accuracy of 0.1 mm compared to an accuracy of 1 mm before the calibration.

For parallel-typed robot configurations, the calibration consists of two stages: self-calibration of the parallel structure and position calibration of the end-effector. The self-calibration is to correct the kinematic parameters of the parallel structure of the robot mechanism

due to assembly and other factors; the end-effector calibration considers the error of the entire robot from the base to the end tooling. Based on the Local POE representation, two self-calibration models of a class of 3-legged parallel robots are formulated utilizing the error in the leg-end distance and measurement residues in the passive joint sensors, respectively [22,23]. Because the errors are expressed in a linearized calibration model, an iterative least-square algorithm is used to determine the error parameters. Computer simulation and experiment results indicate that the proposed parallel robot self-calibration method can improve the position accuracy to about an order of magnitude. After the self-calibration is completed, calibration of the end-effector becomes straightforward. Similar to the calibration of serial-typed robots, a linear error model can be formulated and an iterative least-square algorithm is used to determine the error parameters.

5. Robot configuration optimization

Because of the modular design, the reconfigurable workcell system can achieve optimal design at the component level but may not obtain optimal performance at the system level. Task-driven robot configuration optimization becomes necessary to establish sub-optimal performance for the overall robotic workcell.

Typically, the problem of robot configuration optimization can be stated as “finding an assembly of robot modules that can achieve a certain task requirement based on an inventory of modules”. The assembly of a reconfigurable robot can be treated as a compound entity with finite number of constituents. Finding the most suitable task-oriented robot configuration then becomes a discrete design optimization problem. A task performance related objective function is formulated. Discrete optimization techniques, such as genetic algorithms (GA), the simulated annealing method (SA), and other artificial intelligence techniques are employed to find solutions [15,24,29]. However, due to the complexity of the problem involved, the computation effort required for GA and SA is tremendous [25]. To alleviate the computation problem, we are using the “software agent” concept to determine the optimal configuration [26]. Software agents provide an alternative approach to deal with problems or systems with high complexity by using a number of small, intelligent and autonomous programs, termed agents, to tackle a problem collectively. Our proposed master–slave agent architecture for solving the problem of robot configuration optimization is shown in Fig. 8. In this architecture, the slave agents, also the mobile agents representing arbitrary robot configurations will be dispatched to a cluster of networked computers. Kinematic agents, which are resided on the networked computers, evaluate the performance of and

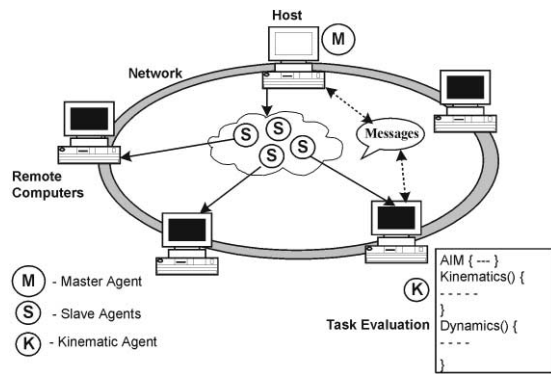


Fig. 8. Agent-based robot configuration optimization.

improve a dispatched robot configuration individually. The master agent controls the dispatch of a pool of slave agents, provides two-way communication among the kinematic agents, and determines the task-optimal configuration based on the messages sent back by the kinematic agents. Because the computation of optimal robot configurations is executed in a distributed and asynchronous manner, the proposed agent architecture can be scaled up to determine the optimal configurations of multiple robots under different task requirements simultaneously, and thus, achieves concurrent design of multiple robots with multiple objectives. For a reconfigurable robot workcell with multiple robots, this is a viable tool for the workcell layout design.

The proposed software agents are developed based on IBM's Aglet software development kit (ASDK) using Java language. Because of the cross-platform compatibility of Java, this agent-based robot configuration optimization software is platform-independent. At this moment, it is implemented on a cluster of 10 networked Pentium II and III PCs with speeds ranging from 200 to 400 MHz. The simulation results show that the proposed software agent approach is about 4–6 times faster than using GA to solve the same problem with equivalent computing power.

Note that the proposed approach is a generic implementation. The criteria used in selecting the optimal configuration depend largely on the task requirements and the performance of the robots. We have proposed a Reduced DOF philosophy to minimize the total number of actuator modules employed in a serial-typed modular robot for a given task [24]. With less number of modules, the robot can carry more payloads instead of the distal modules. Furthermore, the robot can be operated at higher speed with better dynamic response.

6. Control and communication

The control of a reconfigurable workcell and its devices requires: (1) the capability to cope with reconfigura-

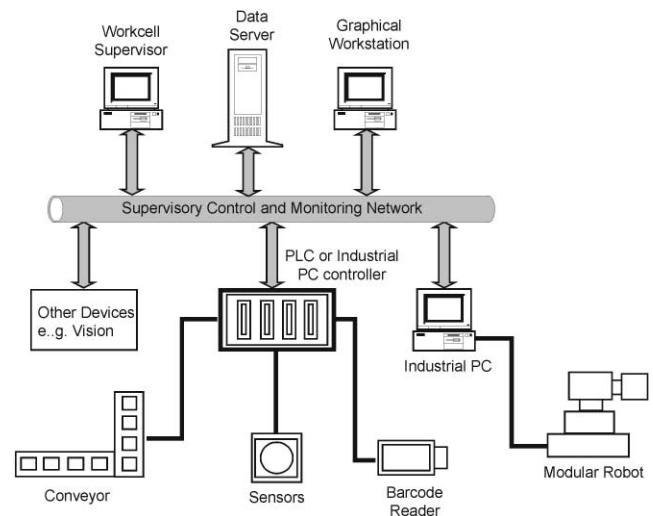


Fig. 9. Workcell-wide communication network.

tion of devices and (2) task coordination among devices. Therefore, two levels of control are used in the workcell: supervisory control of the workcell activities and real-time control of individual workcell devices. The workcell supervisor performs supervisory control of the devices and coordinates the workcell activities by collecting reports from the device controller regarding the status of the workcell, and then, based on these reports, dispatching device task instructions to the device controllers. The device controllers control the individual devices. Each device controller is equipped with its own real-time control system to ensure that the assigned tasks are completed properly. The control and communication architecture of the workcell is illustrated in Fig. 9.

- *Device controller*: A device controller is essentially an interface between the device and the workcell supervisor. A device could be a PLC, a sensor, a vision system, or a conveyor system. Since devices of different types from different vendors have their particular characteristics of operation, a device controller's function is to hide such particularity from the workcell supervisor to lessen the analytical and computational burden of the supervisor, thus improving the "reconfigurability" of the workcell. Operationally, a device controller performs the following three types of tasks: (i) reporting device status, (ii) processing instructions from the workcell supervisor, and (iii) controlling the device to execute the instruction.
- *Robot controller*: Because the active modules are self-contained mechatronic units, the control loop of the modular robot is closed at the joint level. Each actuator module has its own individual motion controller. A serial CAN-bus network moderates the inter-module communication. The reconfigurable "plug-and-play" robot model generation algorithms

described in Section 4 are implemented on the host PC. Based on these algorithms, low-level robot trajectory generation and control are developed. Once a robot is fully constructed and initialized, the host PC can identify the configuration of the robot, generate the necessary models, and coordinate the motion control of the robot. Basically, this host PC robot controller is also a device controller.

- *Workcell-wide supervisory control network*: All the devices are connected to a workcell-wide digital communication network to facilitate control and communication. The choice of network implementation will ultimately depend on the bandwidth requirement, the connectivity of the network with the device controllers, and the reliability of the networking protocol. To facilitate remote internet-enabled operation, a fast TCP/IP Ethernet LAN is selected as the workcell-wide network.
- *Workcell supervisor*: During the workcell operation, the supervisor receives reports from the device controllers regarding the status of the devices, and then, based on the status reports, issues instructions to the device controllers. The model of workcell supervisory control is based on the discrete event control approach [27,28].

7. Simulation software for robot/workcell

To visualize and simulate the performance of an assembled robot, such as reachability and workspace, a robot simulation software application is necessary. The Simulation Environment for MODular Robot System (a.k.a. SEMORS) is a Windows NT-based object-oriented software application developed for this project. Based on the proposed Local POE models and AIM

data structures, SEMORS can offer uniform model construction effort (kinematics, dynamics and calibration) across computer simulation and real-time control of arbitrary robot configurations. The basic graphical user interface of SEMORS is illustrated in Fig. 10. SEMORS is intended to be a uniform interface for all modular robots and is portable to modular robot systems from different vendors. It will be used both for simulation and for on-line execution of a task, regardless of whether the robot is executing (or is simulated to be executing) the task as a stand-alone application, or as part of a workcell process. Thus, it allows the user to quickly integrate the hardware components into modular robots, and to manage their operations in the reconfigurable workcell. Key features of SEMORS include:

- module and robot builder;
- 3D graphical task simulation;
- “Universal” inverse kinematics;
- full dynamics models;
- trajectory and task planning;
- transparent workcell network connectivity.

In addition to the simulation of modular robots, extended features like robot configuration planning/optimization and module database management to be incorporated into SEMORS are currently under development. The task-based robot configuration optimization mentioned in Section 5 is essentially a platform-independent system which can be integrated into SEMORS easily. With the capability of task-based robot configuration optimization, designing the modular robot configuration using SEMORS becomes no longer an ad hoc approach. The software system will provide end-user an optimized robot configuration according to the input task requirements. The user does not need to start the design work from scratch. Rather, based on the

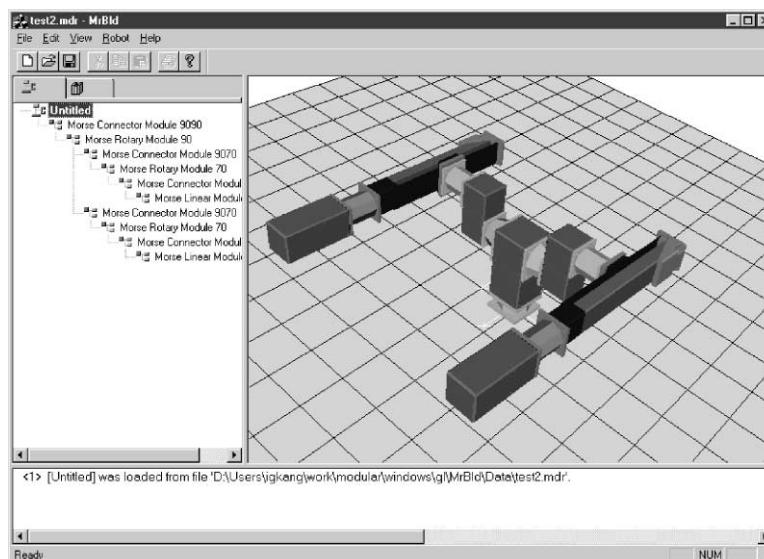


Fig. 10. User interface of SEMORS.

result of optimization, he can fine-tune the suggested robot design or layout. The development effort and time for the workcell can be greatly reduced.

8. Case studies of reconfigurable workcells

8.1. Light-machining workcell (multiple robots)

To effectively demonstrate the concept of a complete reconfigurable robotic workcell, we have constructed a prototyped workcell for light-machining tasks for an industrial exhibition in 1999 (Fig. 11). This workcell is to be built with multiple reconfigurable robots along with other supporting devices under a unified modular approach.

8.1.1. Preliminary design stage

To make use of the advantages of both parallel- and serial-typed robots, we intend to make the workcell to perform a complete milling operation of a workpiece, starting from picking up the object, transferring the object to a milling robot, starting the milling process, and returning the workpiece back to a storage rack. Based on this preliminary concept, we decide to use two reconfigurable robots in this workcell: one is a serial-typed robot for the pick-and-place operation, and the other is a parallel-typed robot for the milling operation because of its structural rigidity. The task is to perform milling operation on a dome-shaped top of a cylindrical workpiece with 15 cm in diameter. A workpiece transfer system should be used in between the two robots.

8.1.2. Robot configuration selection and construction

Based on the preliminary task description, the workcell is configured with a 7-DOF redundant serial-type

robot, a 6-DOF articulate RRRS parallel robot, and a 1-DOF linear motion stage. From the robot configuration optimization, a 4-DOF SCARA-type robot is sufficient to perform the task. Deploying a redundant robot here is to demonstrate that the proposed model generation algorithms used in SEMORS and in robot control are universally applicable for any configuration.

The configuration design of the parallel robot follows a systematic approach [9]. In principle, a three-branch parallel structure is used because of the structure stiffness and dexterity. Each branch consists of three rotary joints (two are active and one is passive) and a passive spherical joint. Once the geometry is determined, the workspace analysis is performed. From the result of this analysis, the lengths of the rigid links and connectors are determined. Because of the modular design, the actuator modules can be freely located at the nine revolute joints. The workspace of the robot changes according to the locations of the actuator modules. A disk-shaped moving platform is attached to the three branches. An end-mill tool actuated by an intelligent motor is mounted at the center of the platform. This motor uses the same control interface as the standard actuator modules. Because of the lack of the force sensor, the task is only carried out in simulated manner, i.e., the end-mill tool only goes through the milling path without touching the surface of the workpiece.

The 1-DOF linear motion stage uses two standard modules: one rotary module to drive the linear slide and one gripper module to hold the workpiece, to ensure uniformity in the workcell control. The specifications of the robots and the motion stage are listed in Table 1. The complete module components used in this system is listed in Table 4.



Fig. 11. Light-machining workcell.

Table 1
Specifications of the light-machining workcell

7-DOF redundant serial robot

Work envelope	Approx. sphere, SR = 1200 mm
Max speed	750 mm/s
Repeatability	± 0.10 mm
Max payload	5 kg (excluding end-effector)
Weight	16 kg (excluding base)

6-DOF RRRS articulate parallel robot

Work envelope	Approx. hemisphere, SR = 500 mm
Max speed	500 mm/s
Repeatability	± 0.05 mm
Max payload	25 kg (excluding end-effector)
Weight	30 kg (excluding base)

1-DOF linear motion stage

Effective stroke	$L = 1500$ mm
Max speed	500 mm/s
Repeatability	± 0.025 mm
Max payload	45 kg (excluding fixture)
Weight	35 kg

8.1.3. Workcell construction and fine-tuning

After the robots and motion stage are constructed, the robot controllers are connected to the robots. Two Pentium II based industrial PC robot controllers are used to perform high-level trajectory control of the serial robot and the parallel robot, respectively. The kinematic models of the two robots are generated automatically in SEMORS and stored in the robot controllers. Kinematic calibration of both robots is performed before the operation. The kinematic calibration is performed by using an articulate-typed coordinate measuring equipment, called “Spin Arm”, and the calibration models described in Section 4.3. The obtained calibration data are transferred to the robot controller and then SEMORS computes and updates the corrected kinematic models of the robots automatically. Because of its simplicity, the control of the motion stage is done by one of the robot controllers for this implementation.

8.1.4. Finalize task sequence and control of the workcell actions

With updated kinematic models, the detailed task sequence of all robots (Table 2) is laid out. The tasks are then programmed into the respective robot controllers. The two robot controllers are connected to a closed-loop workcell LAN running at 10 MB/s. A separate notebook computer is also connected to the workcell network performing supervisory control of the workcell through SEMORS running on the individual robot controllers. The task sequence of the workcell is monitored and supervised by the notebook supervisor.

Based on the actual construction, to assembly the described 7-DOF serial-typed robot takes two users about 30 min altogether. The time to construct the parallel robot requires two persons about 2 h because of the complexity of the structure. Adding the time to install the motion stage, calibrate the robots and fine-tune the workcell hardware, it will take about 4 h in total to complete the entire workcell setup excluding the time spent on the preliminary design stage.

Table 2
Task sequence^a

1	Robot A picks up workpiece from fixture
2	Robot A places workpiece on the motion stage
3	Motion stage moves workpiece under Robot B
4	Robot B performs milling task
5	Robot A shifts locations of un-processed workpieces
6	Robot B finishes milling task
7	Motion stage moves processed workpiece back
8	Robot A picks up processed workpiece from motion stage
9	Robot A places processed workpiece to the fixture

^aRobot A: 7-DOF serial robot, Robot B: 6-DOF parallel robot.

8.2. Other parallel robot workcells (single robot)

Besides the light-machining workcell with multiple robots, we also use the robot modules to construct a variety of parallel robot workcells with only a single robot for demonstrating the rapid prototyping and rapid deployment capabilities.

8.2.1. 3-DOF planar parallel robot

A 3-DOF planar parallel robot workcell is shown in Fig. 12. It is constructed using three revolute actuator modules, six passive joint modules, six rigid links, and a mobile platform. The planar parallel robot uses three identical branches. There are three revolute joints located on each of the three branches. Based on the planar kinematics, the robot has three DOFs. Therefore, only three revolute actuators are needed in this system. The other six joints are constructed with passive joint modules. Because of the kinematic structure, this system can produce high positioning accuracy along the motion plane and has high rigidity perpendicular to the motion plane. The specifications of this robot are listed in Table 3. The workspace of the mobile platform can be modified through module reconfiguration by either changing the lengths of the rigid links or by relocating the actuator modules. By changing the locations of the actuator modules along the branches, for example, from the base joint to the middle joint, the workspace of the robot changes accordingly as the Jacobian of the system is different. Because of the modularity, this kind of reconfiguration can be easily accomplished. The module components used in this workcell are listed in Table 4.

8.2.2. 6-DOF RPRS sliding parallel robot

A 6-DOF parallel robot workcell with large workspace is constructed using three linear actuator modules as shown in Fig. 13. This robot also employs the three-leg design. There are two revolute joint modules (one active

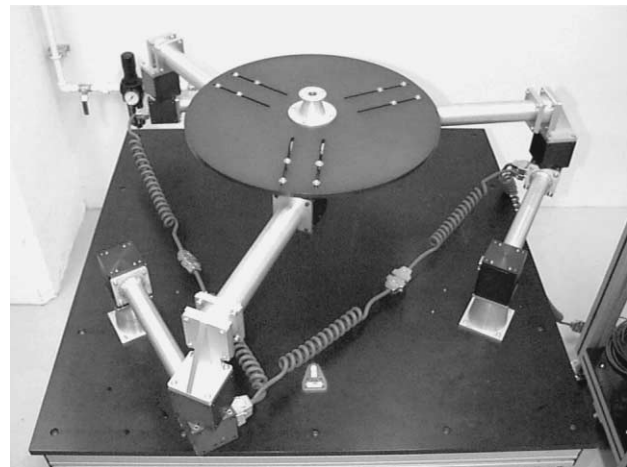


Fig. 12. 3-DOF planar modular parallel robot.

Table 3
Specifications of the parallel robot workcells

<i>3-DOF planar parallel robot</i>	
Max workspace	Circle, $\phi = 400$ mm
Max linear velocity	200 mm/s
Max angular velocity	180°/s
Repeatability	± 0.10 mm
Max payload	> 100 kg
Weight	17.5 kg (excluding base)
Assembly time	30 min
<i>6-DOF RPRS sliding parallel robot</i>	
Max workspace	Cylinder, $\phi 400$ mm \times 450 mm
Max linear velocity	400 mm/s
Max angular velocity	180°/s
Repeatability	± 0.1 mm
Max payload	50 kg
Weight	85 kg (excluding base)
Assembly time	60 min

and one passive) and a linear actuator module for each branch. Spherical joints are attached in between the mobile platform and the branches. The kinematic geometry of the robot is effectively a planar 3-DOF parallel robot with three additional linear actuators as the base joints. Therefore, the workspace of the robot (both reachable and dexterous workspaces) is large in comparison with the articulate 6-DOF parallel robot assembled for the light-machining workcell. Similar to the planar parallel robot, the workspace of the robot can be changed by reconfiguring the location of the passive and active revolute joint modules. The specifications of the robot are listed in Table 3. The types of the modules and components used are listed in Table 4.

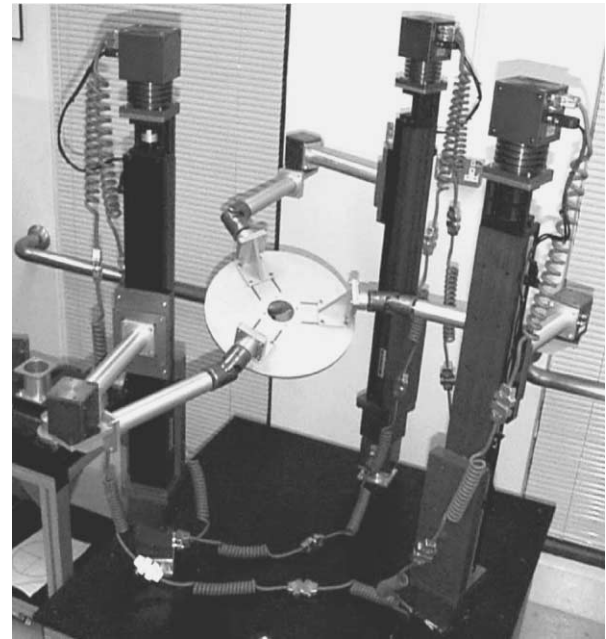


Fig. 13. 6-DOF RPRS sliding parallel robot.

9. Evaluation of workcell performance

To directly compare the performance of the prototype reconfigurable robotic workcell we have constructed with other automated manufacturing systems is not feasible because of the availability of the manufacturing system hardware and the working conditions of the prototype. Instead, we will define a number of qualitative and quantitative criteria to highlight the important

Table 4
List of module components of the reconfigurable robotic workcells^a

Robot configuration	DOF	Active joint module						Passive joint module	
		Revolute		Prismatic		Wrist		Rotary	Spherical
		90	70	90	70	90	70		
Serial redundant	7	2	3				1		
Planar parallel	3		3					6	
3-RRRS articulate parallel	6	3		3				3	3
3-RPRS sliding parallel	6	3	3					3	3
Linear motion stage	1	1							
Link/connector									
Robot configurations	DOF	Cylindrical		Threaded (connecting spherical joints)	Taped 90–70	Angular		Platform	
		90	70			90	70		

^aNote: (1) The number in the parentheses refers to the length of links in millimeters. (2) The sizes of the modules/links have two series: 70 and 90, meaning that the basic dimensions of the modules are 70 and 90 mm, respectively.

performance evaluation issue of the reconfigurable workcell based on our experience.

9.1. Evaluation criteria

Based on our experiment with the reconfigurable workcell, the performance of a reconfigurable robotic workcell can be evaluated at three different levels: the component level, the robot level, and the workcell level (Table 5). At the component level, the interface design is crucial. The rigidity and alignment accuracy and compatibility with the industry standard are the major consideration. The use of a quick-connection mechanism will affect the speed and convenience of the assembly/reconfiguration set up but is not the first priority as indicated in Section 3.3.

The criteria defined for the active and passive modules are mainly based on their design specifications. A set of module specifications, such as speed, accuracy, repeatability, can be defined based on the types of the modules. They are also comparable to the specifications of industrial actuators. Compactness, reliability and maintainability of the modules pose as critical economical factors during the decision-making process in the deployment of such a system. Hence, these criteria should be considered thoroughly in the early design stage.

The criteria defined for the assembled robots are primarily generic for different robot configurations. As the stiffness and workspace of a robot depend on its intrinsic kinematic structure, they should not be considered as measures for the reconfigurable robot. Instead, the reconfiguration time, reconfigurability of the control software, and management of robot CAD-database become important. The time for robot calibration is needed as inaccuracy may occur during module connection as described in Section 4.3. The evaluation criteria for the workcell are defined in a similar manner. The automatic layout generation, time for system setup and calibration, compatibility with part feeders, and the network connectivity are the crucial factors.

9.2. Observation during the implementations

From our actual workcell implementation, we found that using component-based technology to design the robotic workcell indeed provides the convenience of rapid prototyping and rapid deployment. Because the robot usually has complex kinematic structure, the ability to construct a physical prototype rather than a computer-simulated prototype is very helpful for the developers and the end-users. It also eases the effort of system integration because of the uniform module

Table 5
Workcell evaluation criteria

<i>Component level</i>	
Interface	Rigidity (tolerance) Alignment accuracy Quick connection mechanisms? Integrated mechatronic interface? (mechanical, communication, power, service)
Active/passive modules	Industrial standard (inter-operate with other vendors) Module performance (max speed, max force/torque, accuracy, repeatability) Compactness (volume/weight) Reliability (life span) Maintainability (convenience of upgrading internal hardware and software) Module selection and diversity Allow customization by the end-user?
<i>Robot level</i>	
Hardware	Assembly/reconfiguration time Positioning accuracy (end-of-arm tooling) Repeatability Tracking accuracy Time for calibration
Software	Automated configuration design? Configuration design time Automatic simulation/control model generation? Module database management?
<i>Workcell level</i>	
Hardware	Assembly/reconfiguration time Compatibility with other automation equipment (e.g. part feeders) Time for calibration
Software	Automated layout design? Layout design time Unified control and supervision architecture? Workcell network architecture

standard and the interface standard. As indicated in Table 4, the modules and links used for the three prototyped workcells are highly inter-operable. When a robot can be configured with arbitrary structure freely, using multiple reconfigurable robots to build a robotic manufacturing system becomes possible. The manufacturers can benefit from the system reconfiguration and modularity. In addition, the control architecture and the programming method of the modular reconfigurable robot are compatible with conventional robotic systems. Therefore, manufacturers can adopt a gradual change-over policy from a fixed automation system to a reconfigurable manufacturing system, which will be economically viable.

However, based on current state-of-the-art technology, the reconfigurable robotic system also has some drawbacks. Current technology on the actuators still does not allow us to design very compact and high output-to-weight ratio drives. As a modular robot must carry all of its actuator modules during the movement, the payload and dynamic properties of the robot are inferior to the integrated industrial robots. Strategies must be developed to limit the number of modules used in a reconfigurable robot system. The lack of industrial standard on the module connection interface poses a threat in the proliferation of the reconfigurable system. Unlike the computer industry, modular design is very successful because of the existence of large dominant players who can set the industry standard. With a common industrial standard, modular design becomes a common practice and enables horizontal integration of the manufacturers. In the robotics and automation industry, the absence of a dominant player will keep the cost of the module developers high because of the lack of the economic scale. Acute business strategy in promoting the reconfiguration technology will be very important. Lastly, the reliability of the modules is crucial for the success of such a reconfigurable system. From our user experience, current actuator modules are not as reliable as the motors used in the conventional robots because of the lack of experience in designing compact integrated drives (Section 3.1). Without reliable modules, the user must maintain a number of spare modules to keep the system down time to the minimum. The extra modules add on the cost for the overall system that actually defeats one of the advantages using a reconfigurable system. How to achieve reliable module design is a very important topic in the reconfigurable system design.

10. Summary

We have presented the development of a fully reconfigurable robotic workcell in five aspects, namely, the hardware component design, the approach for reconfigurable robot models, robot configuration optimization,

control of the robot and workcell, and reconfigurable robot simulation software — SEMORS. We have also demonstrated several prototyped workcells configured to perform various tasks. From the prototype construction, we can confirm the advantage of using mixed modules in constructing the complex parallel robot configurations. The development of plug-and-play kinematics, dynamics, and calibration robot models are also verified through the actual implementation in the robot controller and the simulation software. The performance of Ethernet implementation of the workcell supervisory control is satisfactory based on the actual deployment in IA'99. Overall speaking, the hardware and software setup of the entire workcell can be completed within hours with proper configuration and layout design. Though the current reconfigurable robotic system has pros and cons in large-scale deployment, the design of those key components, like the actuator modules, can be improved significantly in the near future. Currently, we are also testing out a workcell configuration with five robots to perform maintenance tasks for aircraft honeycomb structures. The next phase of the project is to make the system possess the capability of rapid development from concept to installation. More effort will be spent on the full automation of the back-end robot configuration optimization and workcell layout design. Algorithms for configuration optimization of parallel robots and mission-driven workcell layout design need to be developed and incorporated into the software system so as to achieve this purpose.

Acknowledgements

Team Members of this project: Assoc. Prof. Song Huat Yeo, Assoc. Prof. Guang Chen, Dr. Guilin Yang, Dr. Peter Chen, Dr. Weihai Chen, Dr. Wei Lin, Mr. In-Gyu Kang, Mr. Wee Kiat Lim, Mr. Edwin Ho, Mr. S. Ramachandran, Ms. Yan Gao, and Mr. Chee Tat Tan. This project is supported by Gintic Institute of Manufacturing Technology, Singapore, under Upstream Project U97-A006, and Ministry of Education, Singapore, under Academic Research Fund Project RG64/96 and JT ARC 7/97.

References

- [1] Ulrich K. The role of product architecture in the manufacturing firm. *Res Policy* 1995;24:419–40.
- [2] Baldwin CY, Clark KS. *Design rules vol. 1: the power of modularity*. USA: MIT Press, 2000.
- [3] Pine BJ. *Mass customization*. USA: Harvard Business School Press, 1993.
- [4] Paredis CJJ. et al, A rapidly deployable manipulator system. *International Workshop on Some Critical Issues in Robotics*, Singapore, 1995.

- [5] Fukuda T, Nakagawa S. Dynamically reconfigurable robot system. Proceedings of the IEEE International Conference on Robotics and Automation, 1988. p. 1581–6.
- [6] Cohen R, Lipton M, Dai M, Benhabib B. Conceptual design of a modular robot. *ASME J Mech Des* 1992;114:117–25.
- [7] Matsumaru T. Design and control of the modular robot system: TOMMS. Proceedings of the IEEE International Conference on Robotics and Automation, Nagoya, Japan, 1995. p. 2125–31.
- [8] Tesar D, Butler MS. A generalized modular architecture for robot structures. *ASME J Manuf Rev* 1989;2(2):91–117.
- [9] Yang G, Chen I-M, Lim WK, Yeo SH. Design and kinematic analysis of modular reconfigurable parallel robots. Proceedings of the IEEE International Conference on Robotics and Automation, Detroit, USA, 1999. p. 2501–6.
- [10] Yim M, Duff DG, Roufas KD. PolyBot: a modular reconfigurable robot. Proceedings of the IEEE International Conference on Robotics and Automation, San Francisco, USA, 2000. p. 514–20.
- [11] Rus D, Vona M. A physical implementation of the self-reconfiguring crystalline robot. Proceedings of the IEEE International Conference on Robotics and Automation, San Francisco, USA, 2000. p. 1726–33.
- [12] Murata S, Kurokawa H, Kokaji S. Self-assembling machine. Proceedings of the IEEE Conference on Robotics and Automation, San Diego, CA, USA, 1994. p. 441–8.
- [13] Hollis R, Quaid A. An architecture for agile assembly. Proceedings of the American Society of Precision Engineering, 10th Annual Meeting, Austin, TX, 1995.
- [14] Chen I-M, Yang G, Yeo SH, Chen G. Kernel for modular robot applications — automatic modeling techniques. *International J Robotics Res* 1999;18(2):225–42.
- [15] Chen I-M, Burdick J. Determining task optimal modular robot assembly configurations. Proceedings of the IEEE Conference on Robotics and Automation, Nagoya, Japan, p. 132–7, 1995.
- [16] Chen I-M, Yang G. Configuration independent kinematics for modular robots. Proceedings of the IEEE Conference on Robotics and Automation, Minneapolis, MN, USA, 1996. p. 1845–9.
- [17] Chen I-M, Yang G. Numerical inverse kinematics for modular reconfigurable robots. *J Robotics Systems* 1999;16(4):213–25.
- [18] Murray RM, Li Z, Sastry S. A mathematical introduction to robotic manipulation. FL, USA: CRC Press, 1994.
- [19] Gao Y. Decomposable closed-form inverse kinematics for reconfigurable robots using product-of-exponentials formula. Master of engineering thesis, School of Mechanical and Production Engineering, Nanyang Technological University, Singapore, 2000.
- [20] Chen I-M, Yang G. Automatic model generation for modular reconfigurable robot dynamics. *ASME J Dyn Systems Meas Control* 1998;120(3):346–52.
- [21] Chen I-M, Yang G. Kinematic calibration of modular reconfigurable robots using product-of-exponentials formula. *J Robotic Systems* 1997;14(11):807–21.
- [22] Yang G, Chen I-M, Lim WK, Yeo SH. Design and self-calibration of modular parallel robots. SPIE International Symposium on Intelligent Systems and Advanced Manufacturing, Boston, MA, USA, 1999. p. 224–35, 1999.
- [23] Yang G, Lim WK. End-effector calibration for a self-calibrated modular parallel robot. Technical Report, School of Mechanical & Production Engineering, Nanyang Technological University, 2000.
- [24] Yang G, Chen I-M. Task-based optimization of modular robot configurations — MDOF Approach. *Mech Mach Theory* 2000; 35(4):517–40.
- [25] Paredis CJJ, Khosla PK. Agent based design of fault tolerant manipulators for satellite docking. Proceedings of the IEEE Conference on Robotics and Automation, Albuquerque, NM, USA, 1997. p. 3473–80.
- [26] Ramachandran S, Chen I-M. Distributed agent based design of modular reconfigurable robots. Proceedings of the 5th International Conference on Computer Integrated Manufacturing, Singapore, 2000. p. 447–58.
- [27] Chen PCY, Chen I-M. An enumerative approach to scheduling of reconfigurable manufacturing workcells. Proceedings of the 5th International Conference on Control, Automation, Robotics, Vision, Singapore, 1998. p. 1664–8.
- [28] Ho HL, Chen PCY, Chen I-M. Model and control synthesis of a manufacturing workcell. Proceedings of the 5th International Conference on Computer Integrated Manufacturing, Singapore, 2000. p. 780–97.
- [29] Han J, Chung WK, Youm Y, Kim SH. Task based design of modular robot manipulator using efficient genetic algorithm. Proceedings of the IEEE International Conference on Robotics and Automation, Albuquerque, NM, USA, 1997. p. 507–12.
- [30] Fukuda T, Ueyama T. Cellular robotics, micro robotic systems. Singapore: World Scientific Publishing Co, 1994.