Multimedia Contents

22. Modular Robots

I-Ming Chen, Mark Yim

This chapter presents a discussion of modular robots from both an industrial and a research point of view. The chapter is divided into four sections, one focusing on existing reconfigurable modular manipulators typically in an industry setting (Sect. 22.2) and another focusing on self-reconfigurable modular robots typically in a research setting (Sect. 22.4). Both sections are sandwiched between the introduction and conclusion sections.

This chapter is focused on design issues. Rather than a survey of existing systems, it presents some of the existing systems in the context of a discussion of the issues and elements in industrial modular robotics and modular robotics research. The reader is encouraged to look at the references for further discussion on any of the presented topics.

Modularity in design engineering refers to a compartmentalization of elements. Most often modularity in complex systems occurs as a result of taking a complex system and dividing it into pieces in order to better understand the simpler elements and parallelize the design efforts. Modularity also facilitates the replacing of elements either for repair or upgrading new functionality. The alternative to a modular approach is an integrated approach where systems are designed as a whole. While integrated approaches tend not to be as easy to repair, upgrade or reconfigure, they do have fewer constraints on element design and therefore can be made more optimal. Integrated approaches can focus on lowering cost or having higher performance. In mechanical devices, the choice between modular or integrated architectures can have a large impact on the range of application as well as cost or performance [22.1].

22.1 Concepts and Definitions

For example, a hand drill with modular attachments can expand itsrange of functionality from drilling holes to screwing bolts or buffing surfaces. For robotics, the same impact applies. However, robots have an inherent complexity that lends itself to modularity,such as actuator modules, sensor modules, and sometimes computation modules. In the following sections, definitions will be given for different kinds of modularity frequently seen in robotics.

22.1.1 Concept of Modularity

The product design literature can be looked at as encompassing robotics – which can be considered as an industrial or research product. The architecture of product modularity can be categorized into three subtypes: slot, bus, and sectional modularity [22.1]:

- *Slot architecture*: Each of the interfaces between components is of a different type from the others, so that the various components in the product cannot be interchanged.
- *Bus architecture*: There is a common bus to which the other physical components connect via the same type of interface.
- *Sectional architecture*: All interfaces are of the same type and there is no single element to which all the other components attach, i. e., there is no base component. The assembly is built up by connecting the components to each other via identical interfaces (Fig. 22.1).

Such classifications provide a good definition of modular robots as follows:

- If a complex robotic system adopts a slot- and bus-modularity design approach for its internal structure and architecture, not the external configuration, it can be called a *modularly designed robotic system* benefiting from design parallelization. Such a robot may have a unified and integrated configuration that cannot be changed from outside.
- If a robot adopts a bus- and sectional-modularity design approach for both internal structure and external configuration, it can be called a *modular robot*. The users can reconfigure the compartmentalization and interchange functional modules with some level of effort.

22.1.2 Definition and Classification of Modular Robots

Any system can be reconfigured by destructing and reconstructing it, for example in the worst case, using a blowtorch and milling machine. The key element we must define is the level of effort required to reconfigure. We propose here three levels from the lowest to the highest level of effort of the user:

1. The system reconfigures itself. It is *self-reconfigurable*.
2. The system is reconfigured by a lay user with or without special tools typically in matter of seconds to minutes.
3. The system is reconfigured only by an expert with specialized tools.

This chapter will focus on modular robots that give rise to *plug-and-play* reconfiguration of the system for task and function changes, levels 1 and 2. Level 1 systems are self-reconfigurable systems with sectional modularity. Level 2 systems in this chapter focuses on reconfigurable modular manipulators with a finite set of modules of different functions.

The modular manipulator type of robotic systems are natural evolution of industrial robot manipulators that consist of a number of specific functional modules, such as actuator modules, link modules, and end-effector modules. Subsequently, robots with the serial and branching topology, such as humanoid robots, legged robots, mobile manipulators adopt a similar approach for modularity as these functional modules form the basis of a robotic system.

The self-reconfigurable modular robots grew out of the concept of self-evolution and self-configuration of biological cells with identical units. Such a robot normally consists of a large number of a small set of types of mechatronic units that possess actuation, connection, communication, and computing capability that can be assembled together in arbitrary forms and also reconfigure itself.

Although the two types of modular robots originated from different fundamental concepts, the goals to provide a large number of possible robot configurations for different tasks with the same set of basic robot modules are the same.

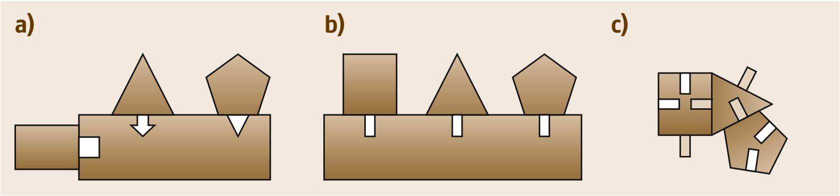

Fig.22.1a–c Modular architecture types: **(a)** slot architecture **(b)** bus architecture **(c)** sectional architecture

22.2 Reconfigurable Modular Manipulators

Reconfigurable modular manipulators are robot arms that have elements that can be rearranged.

22.2.1 Background of Modular Manipulator Systems

The simplest form of modular manipulator comes in the form of automatic tool changers also called quick-change end-effectors. These are optional tools that can be attached to the end of a robotic arm. Automatic tool changers are standard equipment for many CNC (computer numerically controlled) milling machines and lathes since the 1960s. They enable the machines to drill holes of different sizes, or cut different shapes. Although CNC machines are not often considered industrial robots, they share the same elements (actuators, sensors, and computation) and their function is more limited. Most industrial robot arms can be equipped with automatic changing end-effectors by adding a wrist that has the compatible interface for a variety of grippers and end-effectors. These devices are available commercially from Schunk (Germany), ATI industrial automation (USA), Destaco, Amatrol, RE2 (USA), RAD, and others.

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 prototype modular robotic systems have been built and demonstrated, including the reconfigurable modular manipulator system (RMMS) [22.2], several generations of the cellular robotic system (CEBOT) [22.3], and modular manipulator systems developed by University of Toronto [22.4], University of Stuttgart [22.5], University of Texas at Austin [22.6], and Toshiba Corp. [22.7].

Basically, these systems have serial-type (or open-chain) geometry with large working envelopes. These serial-type modular robots are well suited for assembly, trajectory tracking, welding, and hazardous material handling. Parallel modular robots have also been developed for light-machining tasks [22.8]. As indicated in [22.8], modular design can reduce the development cycle of the parallel robots significantly. Furthermore, it allows a trial-and-error approach for the construction of parallel robots that is impossible with the integrated design approach.

With globalization of world manufacturing, the concept of modular manipulators has quickly gained industrial attention. A full-scale reconfigurable robotic system workcell consisting of three modular robots with a total 15 axes was successfully showcased in 1999 [22.9] (Fig. 22.2). The modular robot workcell has a serial type 6-degrees-of-freedom (DOF) robot for the pick-and-place action of the work piece, a 2-DOF robot for work piece transfer, and a 6-DOF parallel robot for milling operations on the work piece. All the robots are built from the same set of modular components, including actuator modules, link modules, and tool modules.

The German company Amtec, later acquired by Schunk, developed the first commercially available modular manipulator system called PowerCube. Schunk has subsequently developed its industrial manipulators and automation systems based on PowerCube with some success [22.10]. Today there are many robotic systems with a wide spectrum of applications that are built around modular robot components.

The modular robot concept also proliferated in the hobby and educational robot sectors around the year 2000 by the introduction of well-packaged self-contained servo motor modules into inexpensive robotic devices, such as Robotis (Korea) and Kondo (Japan), as well as Lego (Denmark) and other toy companies making educational robots.

Mobile robots with legs, wheels, and tracks also belong to this class of modularity where they are configured for different task requirements such as those needed for disaster relief, rescue, and surveillance purposes. Two tracked modular mobile robots designed with multiple track segments [22.11] and reconfigurable tracks allowing serial and parallel connections [22.12] have been demonstrated. The work in [22.13] contains an in-depth review of the development in modular mobile robots.

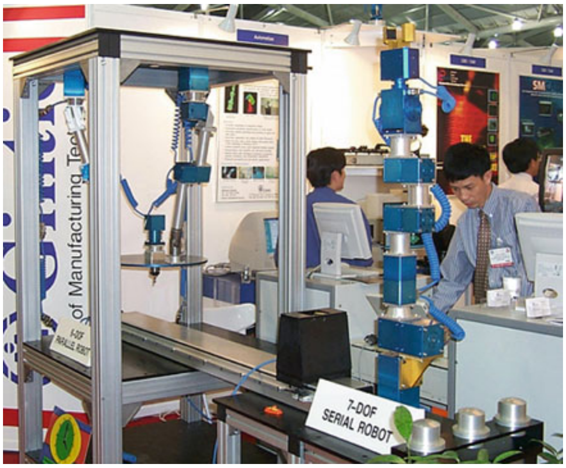

Fig. 22.2 15-axis Reconfigurable Robotic Workcell (after [22.9])

22.2.2 Module Design Issues

A modular robot consists of the two main features found in a modular product: 1) a one-to-one mapping from functional elements to the physical components of the product and 2) decoupled interfaces between the components of different modules [22.1]. For modular manipulators, the essential components are the base, positioning, and orienting mechanisms composed by actuator modules and link modules of different dimensions and geometry, and the end-effector module. For legged and wheeled mobile robotic systems, the motion generation mechanism modules are essential.

The actuator modules normally use DC or AC motors as a 1-DOF *rotate* or *pivot* joint module typically with compact high reduction ratio transmission mechanisms [22.2, 4, 6, 7, 9]. Some modular systems also adopt 1-DOF linear modules for larger workspace envelope [22.4, 9] and 2-DOF joint modules for compact dexterous motions [22.10]. The actuator module shown in Fig. 22.3 has two independent linear and rotary motion capabilities suitable for compact assembly tasks [22.9]. The actuator modules are typically designed with similar geometry but with different dimensions and power ratings for different application requirements.

The link modules connecting units in between the actuators function as reachable workspace extenders. Some systems adopt a standard fixed-dimension connection module [22.3, 4, 6] and some use variable dimension modules that can be customized to satisfy arbitrary design constraints [22.9]. In some systems [22.2], the link module becomes part of the actuator module so that the module acts as an actuator as well as the connecting structure.

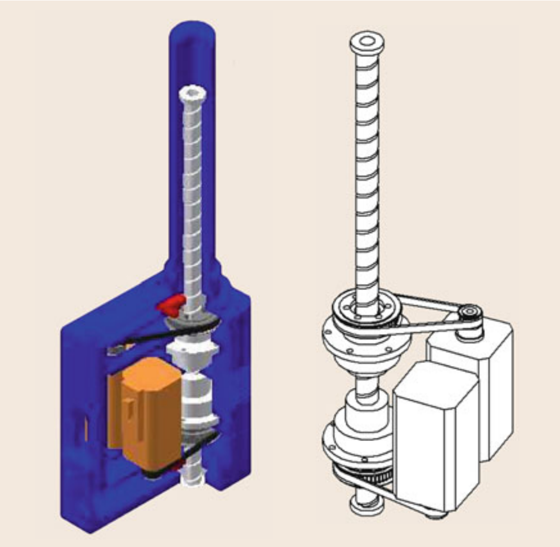

Fig. 22.3 A 2-DOF translate-turn module with ball-screw and ball-spline mechanism

22.2.3 Interface Design Issues

The mechanical connecting interface between modules in a modular manipulator needs to satisfy three basic requirements:

1. Stiffness
2. Fast reconfiguration
3. Interchangeability.

Thus, the design of mechanical connections or docking mechanisms is a critical issue. In a fully or semisupervised robotic system, like some modular manipulator systems [22.2–9], the connecting mechanism is designed to be manually operated for reliability and safety reasons. In a fully autonomous system, the connecting mechanism needs to be designed typically with an extra actuator and locking mechanism for carrying out the connection automatically. This is the case for most of the self-reconfigurable modular robots.

In order to meet the requirement of interchangeability, the electronic and communication interface for the modular system normally adopts common communication network architecture with plug-and-play capability similar to local area network (LAN). There are a number of existing industrial standard network protocols for real-time robot control suitable for such applications, like CAN-bus, RS485, and IEEE 1394. The progressive development of industrial automation protocols will facilitate the implementation of modular robot communications. While many of the early systems use wired multidrop bus architectures for communications, multirobot systems have used fast local message forwarding [22.14] and wireless networking.

22.2.4 Modeling of Modular Manipulators

Challenges to model the modular manipulator systems come from the lack of uniform formalisms of the unfixed robot configuration and geometry and the errors accumulated from assembly and dis-assembly of the modules. Hence, the first effort in modular robot modeling was the introduction of a graph-based technique with additional module assembly information for the representation of a modular robot configuration [22.15]. This work introduced a modular robot representation scheme, termed assembly incidence matrix (AIM) for distinct modular robot configurations. There are several subsequent extensions and variations of AIM for broader categories of modular robots including modular mobile manipulators [22.16–18].

Once the modular robot configuration can be distinctly defined with the type of modules, the connection sequence, module orientation, kinematic, and dynamic models of the robot can be obtained through an automatic generation algorithm [22.19]. Kinematic model generation can be achieved through the conventional Denavit–Hartenberg (DH) parameterization [22.2–4, 17] or the coordinate-free local product-of-exponential (POE) approach [22.8, 9, 19]. However, the DH method does not provide a clear distinction between the arranging sequence of modules in a robot chain, and it is an initial position-dependent representation. The local POE formulation of the kinematics and dynamics based on the theory of Lie groups and Lie algebras for rigid motion in $se(3)$ and $SO(3)$ can avoid this problem. Furthermore, the POE representation can avoid the singularity conditions that frequently occur in the kinematic calibration formulated by the DH method [22.20]. Thus, POE representations provide a uniform and well-behaved method for handling the inverse kinematics of both calibrated and un-calibrated robot systems. In local POE modeling, the joint axes are described in the local module (body) coordinate systems, it is progressive in constructing the kinematic models, so it conveniently resembles the assembling action of the physical modular robot components.

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. In the POE 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 can be established for serial-type robots [22.19]. This model can be obtained through the automatic generation process. An iterative least-square algorithm employed to find the error parameters to be corrected. The corrected kinematic model is then updated in the robot controller for operation. The 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 [22.20]

A formulation of the dynamic model of modular manipulators starts from a recursive Newton–Euler algorithm [22.21, 22]. The generalized velocity, acceleration, and forces can be expressed in terms of linear operations on $se(3)$ [22.23]. Based on the relationship between the recursive formulation and the closed-form Lagrangian formulation for serial-robot dynamics discussed in [22.24, 25], the AIM can assist in the construction of the closed-form equation of motion of a modular robot in any generic topology with redundant and non-redundant configurations [22.19].

22.2.5 Configuration Optimization

Due to the modular design, the modular manipulator can be optimal at the component level, but may not obtain optimal performance at the system level. Task-driven robot configuration optimization becomes necessary to establish locally optimal performance for the overall robotic system. 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 configuration of a modular 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 using a task performance related objective function. Discrete optimization techniques, such as genetic algorithms (GAs), the simulated annealing (SA) method, and other artificial intelligence techniques have been employed to find solutions [22.26–28].

The criteria used in selecting the optimal configuration depend largely on the task requirements, which mostly describe the necessary robot trajectories or key postures. *Yang* and *Chen* proposed a reduced DOF approach to minimize the total number of actuator modules employed in a serial-type modular robot for a given task [22.29]. With fewer modules, the robot can carry more payloads instead of carrying distal modules. Furthermore, the robot can be operated at higher speed with better dynamic response.

22.3 Self-Reconfigurable Modular Robots

Self-configurable systems can rearrange their own topology. An example is shown in VIDEO 2. There are dozens of research groups who have constructed many versions of self-reconfigurable robots [22.3, 30–46], with many approaches for programming them [22.36, 47–61]. As of 2012, over 800 papers and a book [22.31] have been written.

These systems are characterized by many identical modules that can be rearranged into a variety of shapes and configurations and by being highly scalable with simulated systems having hundreds, thousands, or millions of modules. These systems have three promises:

1. Low cost from batch fabrication of repeated modules
2. High robustness from redundancy and the ability to self-repair
3. High versatility from the ability to reconfigure and adapt to changing situations [22.62].

Practically speaking, none of the promises have been proven, though the promise of versatility is getting close. These systems have exhibited a wide variety of locomotion and manipulation including: legged walking with between 2 and 14 legs; riding a tricycle [22.63]; rolling like a tread [22.14]; snake-like locomotion (lateral and rectilinear undulation, concertina, sidewinding) [22.64]; manipulation of large objects with multiple arms/fingers [22.65]; manipulation of small objects; climbing stairs, fences, poles, in pipes; self-reconfiguration between dozens of shapes and many others [22.31]. Figure 22.4 and VIDEO 1 show a self-reconfigurable modular robot SMORES [22.66]. Other self-reconfigurable robots like ATRON [22.40] is shown in VIDEO 5 and M-blocks is shown in VIDEO 3.

22.3.1 Types of Self-Reconfigurable Modular Robots

Self-reconfiguring systems can be classified into three types based on the style of reconfiguration: chain, lattice, and mobile [22.62]. The chain systems reconfigure by using chains of modules that form and break loops [22.36, 62]. They tend to be well suited for work on the environment, as they can form articulated limbs. The lattice systems have modules which have nominal positions sitting on a regular lattice and tend to be better at self-reconfiguration as moving to neighboring lattice positions makes collision checking easy [22.33, 34, 39, 43]. The mobile systems have modules that individually maneuver on terrain and reconfigure by moving on the environment to relocate themselves in a conglomerate [22.38, 67].

Of the systems that have been implemented to date, some that have been shown to be most capable (judging by number of demonstrations) are the hybrid chain-lattice systems: Superbot [22.14], MTRAN III [22.68] and CKbot [22.69]. Recent additions to this group are Johns Hopkins University [22.70], iMobot [22.71], and SMORES [22.66] which are hybrid in all three areas, chain-lattice-mobile systems.

22.3.2 System and Module Design Issues

There is an interesting phenomenon called second system syndrome [22.72], where designers include many features in the second version of a system they design. They often include many more features than the system may need. This is especially problematic in modular reconfigurable systems with repeated modules; any feature added to one module has the possibility of its effect multiplied by n, the number of modules. For example, an increase of d in computational processing power (e.g., microprocessor without interlocked pipeline stages, MIPS) in one module will result in a system increase of dn.

Another similar phenomenon for designers is called *feature creep*, where more and more features are added as the system is being designed. Often this has a cumulative linear effect on cost, but worse, there is an exponential effect on reliability.

The simplest form of robustness analysis assumes independent probabilities that a module may fail during a specified function. If one module succeeds during that function with probability p then the system with $n$identical module has a probability of success of p^n, under the assumption that all modules must be functioning to succeed. Clearly this is problematic for large n.

As systems scale up in number, a key to make things work is to be sure that the system does not depend on every module working perfectly. Indeed, if only X numbers are needed, one would wonder why one would use more than X, especially if the only impact is reduced reliability. Here, one strategy is to ensure that the solution is devised in a way where the performance improves with the number of modules. If modules fail, then the system can gracefully degrade. In systems with tightly coupled actions between modules, this can be difficult

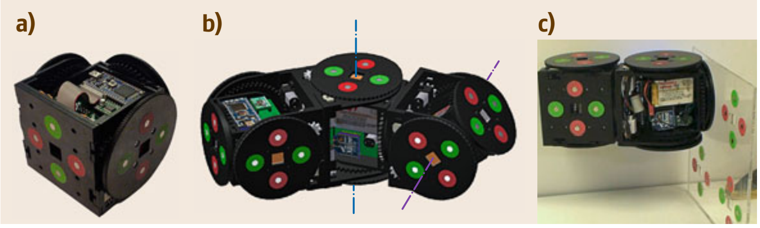

Fig. 22.4 **(a)** One SMORES module with four main actuators and four docking faces. **(b)** Three SMORES modules attached together. **(c)** Two modules moving on a lattice

to implement. In tasks where only binary metrics exist, (success or failure) using the optimal X modules, setting aside the extras may be the best solution.

One of the aspects that researchers find most interesting about self-reconfigurable robots is examining what happens when modules scale up in number (and also typically, scaled down in size). As the numbers increase, the number of shapes/configurations increases and concomitantly, the types of activities that are possible increase as well.

Control and planning get very complicated quickly though. Simply enumerating the number of isomorphic configurations has resulted in a PhD thesis without completely solving the problem [22.15].

While researchers have simulated hundreds, thousands, and even millions of modules, no physical system has been demonstrated with more than a few hundred to date. The largest single system so far is Kilobot [22.73], which has very simple mobile modules that swarm together and actually do not connect rigidly. The modules communicate wirelessly in a one-to-many (broadcast) fashion. For rigidly connected systems the largest number of modules demonstrated in one connected system was a 14-legged 48-module Polybot system [22.74].

Main Actuation

Every module has some form of actuation that enables the modules to move from one position to another or to do some work on the environment. We call it the *main actuator*. By far the most common main actuator is a DC electric motor as it is the lowest cost and easiest to implement.

In chain systems, the main actuator typically drives a revolute joint so that a chain of modules forms an articulated arm. In lattice systems, the main actuator typically moves the module (or a neighboring module) along a constrained 1-DOF path that can be translation or rotation. In mobile reconfiguration, the main actuator enables locomotion, usually through wheels or treads.

The main actuator is typically the largest component in a module and so has been the focus when trying to make modules smaller. To date, the smallest module with onboard actuators, that both attaches and detaches uses shape memory actuated module that was 2 cm [22.75]. An even smaller 1 mm module was created at Carnegie Mellon University [22.76] using electrostatics as an actuator, but it did not attach to other modules.

On the other extreme, the largest module is GHC [22.77], an 8 m^3 helium-filled cube with shape memory actuators on the edges to rotate these floating balloons about edges, attaching electrostatically. DARPA is also sponsoring a project to look at reconfigurable maritime craft constructed in a 20′ ISO container form factor. University of Pennsylvania has demonstrated scale models at 1/12 scale, with main actuators being thrusters to maneuver the parallelepipeds in water [22.78].

22.3.3 Interface Design Issues

The main component of modular systems differentiating it from an integrated system is the interface between modules. When talking about modular systems, the amount of *modularity* can be measured by the number of these interfaces. For self-reconfigurable modular robotic systems in this chapter the number of interfaces in one connected component can be as small as six or as large as millions.

In the most general case, every interface must do two things: 1) attach and 2) detach. When they attach they must do two things: 1) form a physical coupling and 2) allow a flow through the interface for supply power and information (often done electrically).

The interfaces can be *gendered* [22.36], that is two mating interfaces are not identical, one has male features sticking out and the other female features to receive the male features. They can be *ungendered* with no protruding or receding mating features. Or they can be *hermaphroditic* [22.39] where interfaces contain both male and female features. Ungendered and hermaphroditic interfaces can have identical interfaces on both sides. This increases the number of possible arrangements over gendered modules; however, ungendered and hermaphroditic components are typically more complex than gendered interfaces.

Each module can have multiple interfaces. If we consider the number of configurations possible with m interfaces per module assuming identical modules and ignoring any physical constraints such as self-collision, we obtain m^n possible configurations. The number of non-isomorphic configurations is much less, but is very difficult to enumerate in general. However, one special case to consider is two interfaces per module. In this case, topologically, there are only two different configurations, a single chain, and a loop. For this reason, most systems have three or more interfaces.

In addition, each interface can have multiple symmetries that allow different kinematic relationships for the same adjacency relationship. For example many systems have either a 2-way connector (modules may be optionally be rotated 180° to each other) or a 4-way connector (90° rotations.) This p-way symmetry leads to $(pm)^n$ total possible configurations.

Mechanical Interfaces

Mechanical interfaces rigidly attach two modules together with the ability to detach as well. Strategies to enable this capability include using nuts and bolts, magnetic bonding [22.39], electrostatic bonding [22.77], probe and drogue [22.79], and a variety of physical hook-type mechanisms. Each mechanical interface has three performance metrics: bond strength, acceptance range, and interface precision/stiffness.

Bond strength can be characterized by the amount of force required to separate two modules in its weakest direction [22.80]. The acceptance range is the tolerance to position errors as two docks approach to dock [22.81]. Interface precision/stiffness indicates the amount of position error (deflection) that can occur contributed by the interface under load. This can come from elastic deflection, *slop* at a joint in a mate, or misalignments in non-self-guiding docks.

For self-reconfiguring systems, attaching and detaching is automatic and the area of acceptance is required to be much larger than typical manually assembled modular systems. This is in large part because the cost of precision is high and bond precision/stiffness is typically small. The ideal mechanical interface has high bond strength, high acceptance range and high bond stiffness, all of which comes with increased weight, size, and cost. However, different styles of reconfiguration emphasize different aspects.

Lattice reconfiguration systems do not need high acceptance range as modules move in highly constrained manner when reconfiguring, usually with one degree of freedom from one lattice position to neighboring one. However, high bond stiffness is emphasized, as these modules do not often have actuated degrees of freedom to compensate for errors in position. For example, a sequence of modules in a lattice that are forming a loop might not close properly because one section of modules may sag under gravity. This sag was noticeable in the 3-D lattice module [22.82].

Chain reconfiguration modules can deal with lower bond stiffness since they usually have degrees of freedom that can compensate for errors in deflection. However, docking even with this compensation is not trivial and most systems require as large an acceptance range as possible.

Bond strength for systems that reconfigure tend to be much lower than the one would find in a manually assembled system. For example, magnetic bonding methods are easy to implement, typically have good precision and wide range of acceptance, but are weak. Manual systems that are bolted together have high bond strength, but are very complicated if they are made automatic with a very small acceptance range. The variability in bond strength versus the stiffness in joints can be used to vary the overall stiffness of a conglomerate system [22.80]. This variable stiffness can be used to make a compliant material that can conform to surfaces or a stiff material that will not bend under load.

Depending on the task, bond strength may not be required to be high. For example, in mobile reconfiguration systems that primarily move on the ground, e.g., forming trains such as *millibot* [22.67] or *swarms* [22.38], the worst case situation is when the conglomerate tries to cross a gap with modules cantilevered over the gap. Otherwise, the conglomerate is nominally supported at all times and the bond only sees dynamic friction and inertial forces. But in general, higher bond strength does not hurt and keeps a system from falling apart from static or dynamic loads.

Power and Communications Interface

Power and communications interfaces are typically electrical, though in the case of the related work in quick-change end effectors, pneumatics is often supplied too. In any case, the primary parameters of concern are the number of lines that must be transmitted between the interface, and the type of lines (e.g., pneumatic, high voltage, high current, fiber optic, etc.). The type of line will determine the size (which usually limits the number of lines) as well as the precision alignment required at the interface.

In self-reconfigurable systems, the typical interface passes electric power and a separate electronics communications bus on which all modules talk, though they are sometimes combined [22.77]. A key consideration for electronic lines is not only that the correct lines have good contact when mated, but also that wrong contacts do not touch during the docking process due to position error. For most self-reconfiguring systems maximizing the tolerance to position errors is the most important [22.56].

Communication between modules is essential for any self-reconfigurable system, whether the control is distributed or centralized. There are primarily two forms of communication: a global communications bus (any module talks to any other module) or a local communications medium (modules only talk to their neighbors through the modular interface.)

These two forms have different implications. Global buses are typically much faster than local methods of equivalent cost. There are no issues of varied latency between modules as with local methods. One problem with global buses is that there is no mechanism for modules communicating with each other to know the relative physical position of each module. Local methods get this for free. Local methods, can also emulate global architectures by message passing and routing. Local methods are also more robust to physical errors (such as shorting a communications line to power) containing failures to locally. However, they are just as

Table 22.1 Characteristics of used communication protocols

Method	Speed (kbs)	Range (m)	Power (mW)	Address space	Rel. cost	Notes
Ethernet/CAT	1k–1m	100's	500	≈4 billion	$$$$	Complex stack (IPV4)
CANbus	10−1000	100	≈ 200	≈ 2000	$$$	Robust (base frame)
Bluetooth	1k–3k	10	≈ 100	≈1 million	$$$	3 s wake-up
BLE (wibree)	1k	10	10	≈1 million	$$	Low power Bluetooth
Zigbee	20−250	10−75	≈ 30	64 bit	$$	Wakeup 15 ms
RS485	100−10 k	1200	5	256	$	Robust elec. protoc
SPI	1−12 k	≈ 10	5	Out of band	$	Simple sync 5 wire

vulnerable to software errors such as flooding a network with garbage messages.

There are many choices for global communication protocols falling into two categories: wired and wireless. Wired is the most common and has many possible protocols, mostly requiring that the bus is multidrop. Popular wired protocols include Ethernet, EtherCAT, RS-485, SPI, and CANbus. Popular wireless protocols include Zigbee, Bluetooth, and 802.11. Important characteristics of communication protocols include the speed, real-time aspects, address space, and cost. A comparison is shown in Table 22.1 with typical values of common implementation.

The importance of the speed depends (or often dictates) the architecture of distributed control. Communications to individual modules can range from 100 times a second (for direct remote control) to several times per minute for higher level behavior control (e.g., hormone control [22.50]). Many protocols have automatic recovery from bad packets, which has obvious importance. Ethernet uses random backoff retransmission after a collision, which makes the protocol non-deterministic. This makes it difficult to have guarantees of real-time performance. The protocol is very fast so it is possible to ignore this as messages will get through with small latency with high probability. EtherCAT is Ethernet for Control Automation Technology that is better suited for real-time control. CANbus (Control Area Network bus) is a well-established bus used in the automotive technology that can be real-time, and robust, though it is typically slower than Ethernet or EtherCAT.

22.4 Conclusion and Further Reading

After more than a quarter century of research and development in modular robotics, a number of modular robotic systems have successfully entered the industry automation, education and entertainment markets. These successful modular robotic systems indicate that modular design offers advantages in the areas of product variety, application variety and creativity. However, the cost of such systems, no matter at the module level or at the system level, has room to be reduced to help mass adoption.

The cost structure of a modular robot is closely linked with the design of the individual modules and the systemic architecture to be conceived as well as the market demand and expectation. From the development history of LEGO bricks to the Schunk Powercube modules, it is clear that module design functions are becoming simpler (reducing cost) and increasing variability (increasing the user base). Hence, for reconfigurable modular manipulators, mobile systems, and self-reconfigurable modular robots, focusing on system designs to meet the expectation of the end user is the current trend.

Besides system design, standardization of mechanical and communication interfaces is critical. Like other electronic and industrial products, electronic and communication interfaces for modular robots could adopt existing industry standards depending on the form factors, connecting performance, reliability, etc. The mechanical interface design normally does not have industry standard to follow due to variety in mechanical specifications on the form factor, loading capacity, rigidity of the joints. Hence, many novel mechanical connection designs can be explored for modular robots.

Future research continues to explore increasing the number of modules. As they approach hundreds and thousands, there are interesting questions that arise as modules become more tightly coupled than current efforts [22.73]. Issues include how to deal with the increased likelihood that some modular elements are not functioning completely correctly. Biologically inspired mechanisms such as intentional cell-death and cell-replacement may become a required part of very large systems.

As the modular approach has increasingly large numbers of modules, there are many more configurations and resulting capabilities from those configurations. Future research will need to address the problem of determining appropriate or optimal configurations for arbitrary tasks. This may lead to a better understanding of robotic tasks in general.

The most recent review article on modular mobile robots can be found in [22.13]. In [22.30], the technical challenges and future of self-reconfigurable modular robots are reviewed. Interested readers may find further information on self-reconfigurable robots in [22.31].

Video-References

VIDEO 1 SMORES
available from http://handbookofrobotics.org/view-chapter/22/videodetails/1

VIDEO 2 4x4ht4a
available from http://handbookofrobotics.org/view-chapter/22/videodetails/2

VIDEO 3 M-Blocks: Momentum-driven, magnetic modular robots self-reconfiguring
available from http://handbookofrobotics.org/view-chapter/22/videodetails/3

VIDEO 5 ATRON robot showing robust and reversible execution of self-reconfiguration sequences
available from http://handbookofrobotics.org/view-chapter/22/videodetails/5

References

22.1 K. Ulrich: The role of product architecture on the manufacturing firm, Res. Policy **24**, 419–440 (1995)

22.2 C. Paredis, H.B. Brown, P. Khosla: A rapidly deployable manipulator system, Proc. IEEE ICRA (1996) pp. 1434–1439

22.3 T. Fukuda, S. Nakagawa: Dynamically reconfigurable robot system, Proc. IEEE ICRA (1988) pp. 1581–1586

22.4 R. Cohen, M. Lipton, M. Dai, B. Benhabib: Conceptual design of a modular robot, ASME J Mech. Des. **25**, 114–117 (1992)

22.5 K.H. Wurst: The conception and construction of a modular robot system, Proc. 16th Int. Symp. Ind. Robots (ISIR) (1986) pp. 37–44

22.6 D. Tesar, M.S. Butler: A generalized modular architecture for robot structures, ASME J. Manuf. Rev. **2**(2), 91–117 (1989)

22.7 T. Matsumaru: Design and control of the modular robot system: TOMMS, Proc. IEEE ICRA (1995) pp. 2125–2131

22.8 G. Yang, I.-M. Chen, W.K. Lim, S.H. Yeo: Kinematic design of modular reconfigurable in-parallel robots, Auton. Robot. **10**(1), 83–89 (2001)

22.9 I.-M. Chen: Rapid response manufacturing through a rapidly reconfigurable robotic workcell, Robotics Comput. Integr. Manuf. **17**(3), 199–213 (2001)

22.10 Schunk GmbH, Lauffen, Germany, http://www.schunk.com/

22.11 W. Wang, W. Yu, H. Zhang: JL-2: A mobile multirobot system with docking and manipulating capabilities, Int. J. Adv. Robotic Syst. **7**(1), 9–18 (2010)

22.12 B. Li, S. Ma, J. Liu, M. Wang, T. Liu, Y. Wang: AMOEBA-I: A shape-shifting modular robot for urban search and rescue, Adv. Robotics **23**(9), 1057–1083 (2009)

22.13 P. Moubarak, P. Ben-Tzvi: Modular and reconfigurable mobile robotics, Robot. Auton. Syst. **60**, 1648–1663 (2012)

22.14 H.C.H. Chiu, M. Rubenstein, W.M. Shen: Multifunctional superbot with rolling track configuration, Proc. IEEE/RSJ IROS (2007)

22.15 I.-M. Chen, J.W. Burdick: Enumerating non-isomorphic assembly configurations of a modular robotic system, Int. J. Robotics Res. **17**(7), 702–719 (1998)

22.16 J. Liu, Y. Wang, S. Ma, Y. Li: Enumeration of the non-isomorphic configurations for a reconfigurable modular robot with square-cubic-cell modules, Int. J. Adv. Robotic Syst. **7**(4), 58–68 (2010)

22.17 H. Liu, H. Wang, S. Li, L. He: Research on the topology description and modeling method for reconfigurable modular robots, Proc. ASME/IFToMM Int. Conf. Reconfig. Mech. Robot. (ReMAR) (2009) pp. 129–135

22.18 E. Meister, A. Gutenkunst: Self-adaptive framework for modular and self-reconfigurable robotic systems, Proc. 4th Int. Conf. Adapt. Self-Adapt. Syst. Appl. (ADAPTIVE) (2012) pp. 30–37

22.19 I.-M. Chen, G. Yang, S.H. Yeo, G. Chen: Kernel for modular robot applications – Automatic modeling techniques, Int. J. Robotics Res. **18**(2), 225–242 (1999)

22.20 I.-M. Chen, C.T. Tan, G. Yang, S.H. Yeo: A local POE model for robot kinematic calibration, Mech. Mach. Theory **36**(11), 1215–1239 (2001)

22.21 J.M. Hollerbach: A recursive lagrangian formulation of manipulator dynamics and a comparative study of dynamics formulation complexity, IEEE Trans. Syst. Man Cybern. **10**, 730–736 (1980)

22.22 G. Rodriguze, A. Jain, K. Kreutz-Delgado: A spatial operator algebra for manipulator modeling and control, Int. J. Robotics Res. **10**(4), 371–381 (1991)

22.23 R. Murray, Z. Li, S. Sastry: *A Mathematical Introduction to Robotic Manipulation* (CRC, Boca Raton 1994)

22.24 R. Featherstone: *Robot Dynamics Algorithms* (Kluwer, Dordrecht 1987)
22.25 F.C. Park, J.E. Bobrow, S.R. Ploen: A lie group formulation of robot dynamics, Int. J. Robotics Res. **14**(6), 609–618 (1995)
22.26 I.-M. Chen, J.W. Burdick: Determining task optimal modular robot assembly configurations, Proc. IEEE ICRA (1995) pp. 132–137
22.27 J. Han, W.K. Chung, Y. Youm, S.H. Kim: Task based design of modular robot manipulator using efficient genetic algorithm, Proc. IEEE ICRA (1997) pp. 507–512
22.28 C. Paredis, P.K. Khosla: Agent based design of fault tolerant manipulators for satellite docking, Proc. IEEE ICRA (1997) pp. 3473–3480
22.29 G. Yang, I.-M. Chen: Task-based optimization of modular robot configurations: Minimized degree-of-freedom approach, Mech. Mach. Theory **35**(4), 517–540 (2000)
22.30 M. Yim, W.M. Shen, B. Salemi, D. Rus, M. Moll, H. Lipson, E. Klavins, G. Chirikjian: Modular self-reconfigurable robot systems – Challenges and opportunities for the future, IEEE Robotics Autom. Mag. **14**(1), 43–52 (2007)
22.31 K. Stoy, D. Brandt, D.J. Christensen: *Self-Reconfigurable Robots – An Introduction* (MIT, Cambridge 2010)
22.32 D. Rus, M. Vona: Self-reconfiguration planning with compressible unit modules, Proc. IEEE ICRA (1999)
22.33 S. Murata, E. Yoshida, K. Tomita, H. Kurokawa, A. Kamimura, S. Kokaji: Hardware design of modular robotic system, Proc. IEEE/RSJ IROS (2000)
22.34 K. Kotay, D. Rus: Self-reconfigurable robots for navigation and manipulation, Proc. IEEE/RSJ IROS (1997)
22.35 C. Ünsal, H. Kiliççöte, P.K. Khosla: I(CES)-cubes: A modular self-reconfigurable bipartite robotic system, Proc. SPIE, Sens. Fusion Decent. Control Robotic Syst., Bellingham (1999) pp. 258–269
22.36 A. Castano, W.M. Shen, P. Will: CONRO: Towards deployable robots with inter-robots metamorphic capabilities, Auton. Robots **8**(3), 309–324 (2000)
22.37 W.H. Lee, A.C. Sanderson: Dynamic analysis and distributed control of tetrobot modular reconfigurable robotic system, Auton. Robots **10**(1), 67–82 (2001)
22.38 E. Tuci, R. Gross, V. Trianni, F. Mondada, M. Bonani, M. Dorigo: Cooperation through self-assembly in multi-robot systems, ACM Trans. Auton. Adapt. Syst. (TAAS) **1**(2), 115–150 (2006)
22.39 J.W. Suh, S.B. Homans, M. Yim: Telecubes: Mechanical design of a module for self-reconfigurable robotics, Proc. IEEE ICRA (2002)
22.40 M.W. Jorgensen, E.H. Ostergaard, H.H. Lund: Modular Atron: Modules for a self-reconfigurable robot, Proc. IEEE/RSJ IROS (2004) pp. 2068–2073
22.41 M. Hafez, M.D. Lichter, S. Dubowsky: Optimized binary modular reconfigurable robotic devices, IEEE/ASME Trans. Mechatron. **8**(1), 18–25 (2003)
22.42 P. White, V. Zykov, J. Bongard, H. Lipson: Three dimensional stochastic reconfiguration of modular robots, Proc. Robotics Sci. Syst. **1**, 161–168 (2005)
22.43 B. Kirby, J. Campbell, B. Aksak, P. Pillai, J. Hoburg, T. Mowry, S.C. Goldstein: Catoms: Moving robots without moving parts, Proc. Natl. Conf. Artif. Intell. **20**, 1730–1731 (2005)
22.44 S.C. Goldstein, J.D. Campbell, T.C. Mowry: Programmable matter, Computer **38**(6), 99–101 (2005)
22.45 A.A. Goldenberg, N.M. Kircanski, P. Kuzan, J.A. Wiercienski, R. Hui, C. Zhou: Modular, expandable and reconfigurable robot, US Patent 552 3662 (1996)
22.46 H. Lipson, J.B. Pollack: Towards continuously reconfigurable self-designing robotics, Proc. IEEE ICRA (2000)
22.47 J.E. Walter, E.M. Tsai, N. Amato: Choosing good paths for fast distributed reconfiguration of hexagonal metamorphic robots, Proc. IEEE ICRA (2002)
22.48 A. Castano, P. Will: Representing and discovering the configuration of CONRO robots, Proc. IEEE ICRA (2001) pp. 3503–3509
22.49 K. Støy, W.M. Shen, P. Will: Global locomotion from local interaction in self-reconfigurable robots, Proc. 7th Int. Conf. Intell. Auton. Syst. (IAS-7), Marina del Rey (2002)
22.50 P. Will, B. Salemi, W.M. Shen: Hormone-controlled metamorphic robots, Proc. IEEE ICRA (2001) pp. 4194–4199
22.51 A. Kamimura, S. Murata, E. Yoshida, H. Kurokawa, K. Tomita, S. Kokaji: Self-reconfigurable modular robot – Experiments on reconfiguration and locomotion, Proc. IEEE/RSJ IROS (2001) pp. 606–612
22.52 R. Fitch, D. Rus, M. Vona: Basis for self-repair robots using self-reconfiguring crystal modules, Proc. IEEE Int. Auton. Syst. (2000) pp. 903–910
22.53 G.S. Chirikjian, A. Pamecha, I. Ebert-Uphoff: Evaluating efficiency of self-reconfiguration in a class of modular robots, J. Robotic Syst. **13**(5), 317–338 (1996)
22.54 F.H. Bennett, E.G. Rieffel: Design of decentralized controllers for self-reconfigurable modular robots using genetic programming, Proc. 2nd NASA/DoD Workshop Evol. Hardw. (2000)
22.55 Z. Butler, R. Fitch, D. Rus: Distributed control for unit-compressible robots: Goal-recognition, locomotion, and splitting IEEE/ASME Trans. Mechatron. **7**(4), 418–430 (2002)
22.56 M. Yim, Y. Zhang, K. Roufas, D. Duff, C. Eldershaw: Connecting and disconnecting for chain self-reconfiguration with PolyBot, IEEE/ASME Trans. Mechatron. **7**(4), 442–451 (2002)
22.57 Y. Zhang, M.P.J. Fromherz, L.S. Crawford, Y. Shang: A general constraint-based control framework with examples in modular self-reconfigurable robots, Proc. IEEE/RSJ IROS (2002) pp. 2163–2168
22.58 H. Bojinov, A. Casal, T. Hogg: Emergent structures in modular self-reconfigurable robots, robotics and automation, Proc. IEEE ICRA (2000)
22.59 P. Pirjanian, C. Leger, E. Mumm, B. Kennedy, M. Garrett, H. Aghazarian, S. Farritor, P. Schenker: Distributed control for a modular, reconfigurable cliff robot, Proc. IEEE ICRA (2002)
22.60 K.C. Prevas, C. Unsal, M.O. Efe, P.K. Khosla: A hierarchical motion planning strategy for a uniform

self-reconfigurable modular robotic system, Proc. IEEE ICRA (2002)
22.61 W.W. Melek, A.A. Goldenberg: Neurofuzzy control of modular and reconfigurable robots, IEEE/ASME Trans. Mechatron. **8**(3), 381–389 (2003)
22.62 M. Yim, D.G. Duff, K.D. Roufas: Polybot: A modular reconfigurable robot, Proc. IEEE ICRA (2000) pp. 514–520
22.63 M. Yim, Y. Zhang, D. Duff: Modular Reconfigurable Robots, Machines that shift their shape to suit the task at hand, IEEE Spectrum **39**(2), 30–34 (2002)
22.64 M. Yim, C. Eldershaw, Y. Zhang, D. Duff: Limbless conforming gaits with modular robots. In: *Springer Tract. Adv. Robot.*, Vol. 21, 2006) pp. 459–468
22.65 M. Yim, J. Reich, A.A. Berlin: Two approaches to distributed manipulation. In: *Distributed Manipulation*, ed. by K. Bohringer, H.H. Choset (Kluwer, Dordrecht 2000) pp. 237–261
22.66 J. Davey, N. Kwok, M. Yim: Emulating self-reconfigurable Robots – Design of the SMORES system, Proc. IEEE/RSJ IROS (2012)
22.67 H.B. Brown, J.M. Vande Weghe, C.A. Bererton, P.K. Khosla: Millibot trains for enhanced mobility, IEEE/ASME Trans. Mechatron. **7**(4), 452–461 (2002)
22.68 S. Murata, K. Kakomura, H. Kurokawa: Toward a scalable modular robotic system, IEEE Robotics Autom. Mag. **14**(4), 56–63 (2007)
22.69 M. Yim, B. Shirmohammadi, J. Sastra, M. Park, M. Dugan, C.J. Taylor: Towards robotic self-reassembly after explosion, Proc. IEEE/RSJ IROS (2007) pp. 2767–2772
22.70 M. Kutzer, M. Moses, C. Brown, M. Armand, D. Scheidt, G. Chirikjian: Design of a new independently-mobile reconfigurable modular robot, Proc. IEEE ICRA (2010)
22.71 H. Cheng, G. Ryland: Design of iMobot, an intelligent reconfigurable mobile robot with novel locomotion, Proc. IEEE ICRA (2010)
22.72 F.P. Brooks: *The Mythical Man-Month* (Addison-Wesley, Reading 1975)
22.73 M. Rubenstein, R. Nagpal: Kilobot: A robotic module for demonstrating behaviors in a large scale (2^{10} units) collective, Proc. IEEE ICRA (2010) pp. 47–51
22.74 Y. Zhang, M. Yim, C. Eldershaw, D. Duff, K. Roufas: Scalable and reconfigurable configurations and locomotion gaits for chain-type modular reconfigurable robots, Proc. IEEE Symp. Comput. Intell. Robotics Autom. (CIRA) (2003)
22.75 E. Yoshida, S. Murata, S. Kokaji, K. Tomita, H. Kurokawa: Micro self-reconfigurable modular robot using shape memory alloy, J. Robotics Mechatron. **13**(2), 212–218 (2001)
22.76 M.E. Karagozler, S.C. Goldstein, J.R. Reid: Stress-driven MEMS assembly + electrostatic forces = 1mm diameter robot, Proc. IEEE/RSJ IROS (2009)
22.77 M.E. Karagozler, J.D. Campbell, G.K. Feddore, S.C. Goldstein, H.P. Weller, B.W. Yoon: Electrostatic latching for inter-module adhesion, power transfer, and communication in modular robots, Proc. IEEE/RSJ IROS (2007)
22.78 I. O'Hara, J. Paulos, J. Davey, N. Eckenstein, N. Doshi, T. Tosun, J. Greco, Jungwon. Seo, M. Turpin, V. Kumar, M. Yim: Self-assembly of a swarm of autonomous boats into floating structures, Proc. IEEE ICRA (2014) pp. 1234–1240
22.79 L. Vasilescu, P. Varshavshaya, K. Kotay, D. Rus: Autonomous modular optical underwater robot (amour) design, prototype and feasibility study, Proc. IEEE ICRA (2005)
22.80 P.J. White, S. Revzen, C.E. Thorne, M. Yim: A general mechanical strength model for programmable matter and modular robotic structures, Robotica **29**(1), 103–121 (2011)
22.81 N. Eckenstein, M. Yim: The X-Face: An improved planar passive mechanical connector for modular self-reconfigurable robots, Proc. IEEE/RSJ IROS (2012) pp. 3073–3078
22.82 S. Murata, H. Kurokawa, E. Yoshida, K. Tomita, S. Kokaji: A 3-D self-reconfigurable structure, Proc. IEEE ICRA (1998) pp. 432–439