Afreen Siddiqi
Post-Doctoral Research Associate
Department of Aeronautics & Astronautics,
Massachusetts Institute of Technology,
Cambridge, MA 02139
e-mail: siddiqi@mit.edu

Olivier L. de Weck
Associate Professor of Aeronautics &
Astronautics and Engineering Systems
Engineering Systems Division,
Department of Aeronautics & Astronautics,
Massachusetts Institute of Technology,
Cambridge, MA 02139
e-mail: deweck@mit.edu

Modeling Methods and Conceptual Design Principles for Reconfigurable Systems

Reconfigurable systems can attain different configurations at different times thereby altering their functional abilities. Such systems are particularly suitable for specific classes of applications in which their ability to undergo changes easily can be exploited to fulfill new demands, allow for evolution, and improve survivability. This paper identifies the main factors that drive the need for reconfigurability and proposes methods for modeling reconfigurable systems. A survey of 33 different reconfigurable systems is also presented to provide broader insights and general design guidelines for reconfigurable systems. [DOI: 10.1115/1.2965598]

1 Introduction

A common theme in the requirements for many future systems is that they should be able to respond to changing needs. Reconfigurable systems, i.e., those that can change their configurations, can potentially satisfy changing system requirements. The need for reconfigurability is generally driven by three main factors (see Fig. 1).

(1) Multiability: the system performs multiple distinctly different functions at different times
(2) Evolvability: the system changes easily over time by removing, substituting, and adding new elements and functions
(3) Survivability: the system remains functional, possibly in a degraded state, despite a few failures

These three situations either uniquely or in a combination envelope almost all cases for which reconfigurability may be desired.

The requirement for multiability can be due to the resource efficiency considerations or for performance enhancement. In both cases, the essential requirement on the system is that it needs to fulfill multiple objectives/goals, and thus should be capable of executing them collectively, but not necessarily simultaneously. Reconfigurability in systems is also often an enabler of change over time. The future configurations may be known a priori at the time of design or they may be unpredictable in which case the evolvability helps to manage uncertainty. Reconfigurability can also be a means for enhancing a system's survivability. In the event of partial failure, reconfigurable systems can potentially configure to a state in which some level of functionality is maintained. Furthermore, they can enhance the safety margins (through reconfigurations), and thus effectively reduce the probability of experiencing failure.

The life cycle of a reconfigurable system can be partitioned into phases, as shown in Fig. 2. The reconfiguration phase in which the system undergoes changes in its configuration (after it has been fielded) sets these systems apart from the traditional nonreconfigurable ones. Therefore, new frameworks that account for the time related changeability aspects of reconfigurable systems can be useful for analysis and design for reconfigurability.

1.1 Literature Review. From a review of current literature, it becomes clear that there has been an increasing effort toward the study and design of systems that can change or reconfigure. Figure 3 shows how the number of journal articles on this topic has increased over the past few years. The most extensive work related to reconfigurable systems has been done in the computing domain. Field programmable gate arrays (FPGAs), devices that allow circuits to be defined in software and provide for easy reconfigurations, have advanced greatly in recent years [2]. Avionics systems that can evolve with changing environmental conditions of a spacecraft have been developed to enhance survivability in long duration space missions [3]. Various concepts of software-radio based transponders for communication satellites have been proposed that allow for on-orbit reconfigurations [4,5].

Manufacturing systems have also received significant attention for reconfigurability. The National Research Council has identified reconfigurable manufacturing systems (RMS) as the number one priority technology for future manufacturing in 2000 [6]. Several researchers have especially focused on developing design methods for reconfigurable machine tools (RMT). The general scheme is that from a set of commercially available modules (that constitute a module library) the optimal RMT design is formulated so that it has the desired reconfigurability in its degrees of freedom, work piece geometry variability, spindle orientation, etc. [7].

Reconfigurability has also been increasingly incorporated in the design of new air and space systems. Unmanned aerial vehicles (UAVs) that can morph their wings in-flight in order to efficiently carry out different roles have been the subject of much recent research and development [8,9]. These UAVs can undergo large changes in wingspan, area, and sweep angles so that the same vehicle is able to effectively perform loiter/reconnaissance and attack roles in a single mission. For reconfigurable spacecrafts, two kinds of approaches have been mainly explored. The first involves a "plug and play" architectural approach in which spacecrafts are built from a standard set of modules that supposedly allow for faster manufacturing and potentially on-orbit reconfigurations/upgrades, etc. [10]. The second approach involves modular architectures based on common elements of form such as truncated octahedrons [11] and tetrahedrons [12], etc.

Several research efforts have also focused on developing frameworks and methodologies for designing changeable (sometimes also referred to in literature as flexible or adaptable) systems. Some examples include a framework that has been developed for multiattribute decision making in flexible system design [13], transformer design theory [14] that deals with form related aspects of systems capable of configuring into different states, and selection integrated optimization methodology [15] that can be used to study adaptability in design variables.

It is important to recognize that analysis methods specifically

Contributed by Design Theory and Methodology Committee of ASME for publication in the Journal of Mechanical Design. Manuscript received April 26, 2007; final manuscript received March 4, 2008; published online September 10, 2008. Review conducted by Yan Jin.

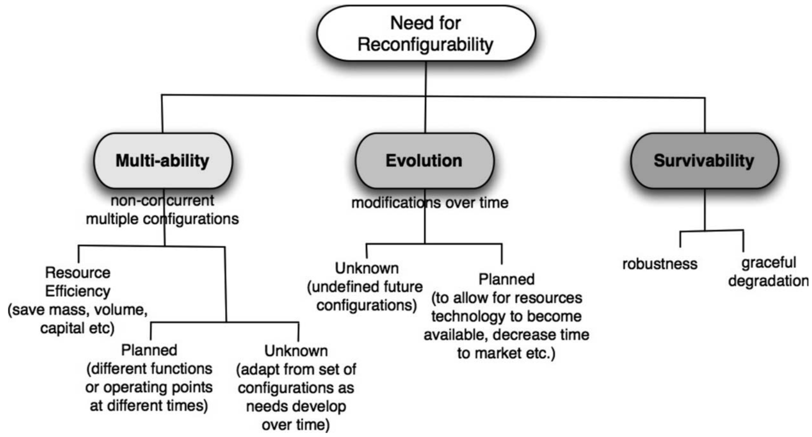

Fig. 1 Properties enabled by reconfigurability

catering to the key characteristics of reconfigurable systems (i.e., their time varying nature) can be especially effective for producing good designs. In this regard, a few modeling methods for analyzing reconfigurable systems are presented. These methods can be used for a wide variety of systems, and their applicability is illustrated through examples of a planetary rover and a morphing aircraft. The modeling techniques are then followed by a review of the physical design of several different kinds of reconfigurable systems. Based on this review, some guidelines are formulated that can help in the development of good design concepts for reconfigurable systems.

1.2 Reconfigurable Systems: Definition. Reconfigurable systems can be defined as those systems that can reversibly achieve distinct configurations (or states), through alteration of system form or function, in order to achieve a desired outcome within acceptable reconfiguration time and cost. Reconfigurable systems in this discussion are thus limited to those that do not go through only a one-time change. Rather they are capable of changing states repeatedly in the operational stage of their life cycle.

The set of configurations S of a system is taken to be the (usually finite) collection of specific discrete or continuous states of the system, in which there is a different form and/or functional behavior.

Figure 4(*a*) shows an object-process diagram (OPD) [16] of a reconfigurable system, as defined above. It is shown that in a reconfigurable system the attributes of the system form, the externally delivered function, and/or the attributes of the function are affected by the process of reconfiguration. For example, consider a recently developed morphing UAV, the MXF-1, which can reversibly reconfigure its wings during flight. It can achieve an in-flight area change of 40%, a span change of 30%, and a wing

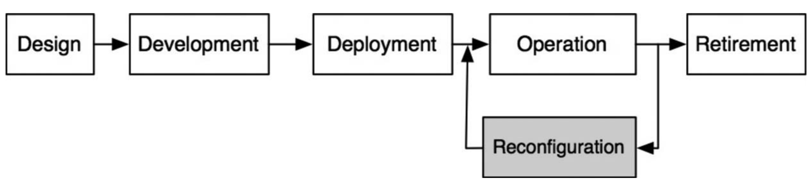

Fig. 2 Stages in life cycle of a reconfigurable system

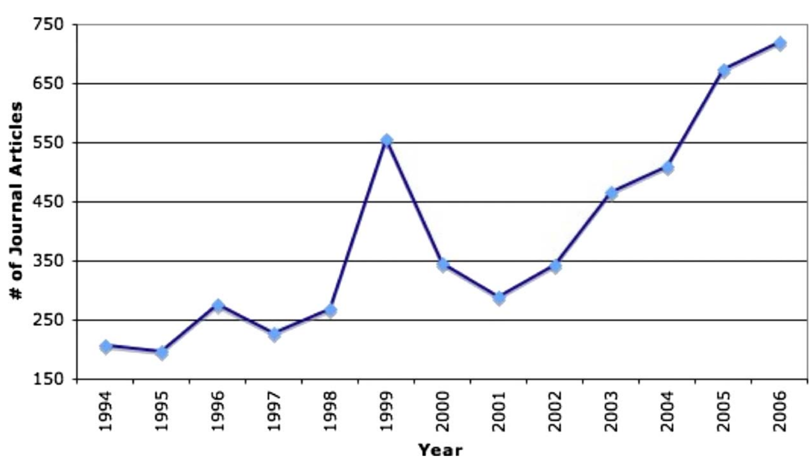

Fig. 3 Number of journal articles published with the keyword "reconfig" in the title or abstract (Ref. [1])

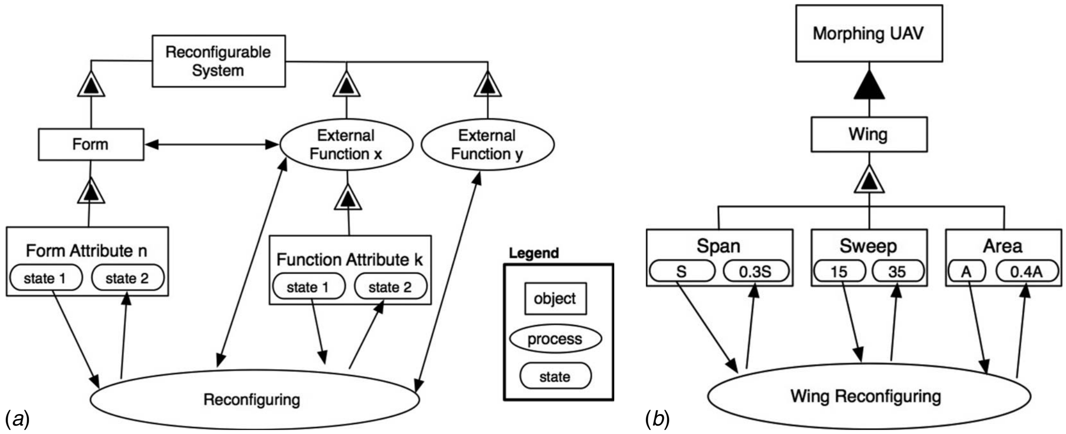

Fig. 4 Object-process diagrams of reconfigurable systems: (*a*) OPD of a reconfigurable system; (*b*) OPD of a morphing UAV

sweep variation of 15–35 deg [9]. An OPD of such a vehicle is shown in Fig. 4(*b*). The MXF-1 wing has an articulating lattice structure. Its skin is of a material that can undergo strains in excess of 100% and conforms to the wing's changing skeleton. The geometry change is achieved through a series of internal linear electromechanical actuators. The reconfigurable wings optimize the aircraft for dramatically different flight conditions—for instance, from an efficient, high-altitude loiter shape to an efficient high-speed attack mode [9]. Figure 5 shows a subscale prototype of this UAV.

2 Modeling Reconfigurable Systems

Since reconfigurable systems can attain different configurations over time, the modeling methods used for their analysis should account for their time varying nature. It is proposed that Markov models and control-theoretic approaches can be used to model such systems effectively. These approaches are developed in this section, and their applicability for analyzing dynamical aspects of reconfigurable systems is demonstrated through a few examples.

2.1 Markov Models. A Markov process is a probabilistic model of usually a complex system, which employs the concepts of states and state transitions. Reconfigurable systems can therefore be studied with Markov models in a very natural way. The Markov theory has been fairly well developed since its first introduction by the Russian mathematician A. Markov in 1907 [18].

2.1.1 Discrete-Time Markov Chains. A discrete-time Markov chain is a process in which the system's state changes at certain discrete-time instants. Suppose a system can exist in N finite and discrete states that belong to a set $S=\{1,2,\ldots,N\}$. A time ordered set, $T=[t_1,\ldots,t_n,\ldots,t_f]$ can be defined, and the system state at a time instant t_n can be denoted as X_n. Then, for a Markov process the following assumption holds [19]:

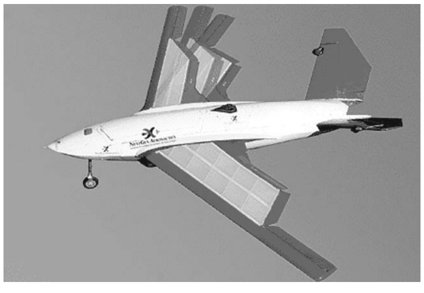

Fig. 5 NextGen MXF-1 morphing UAV [17]

$$p_{ij}(n) = Pr\{X_{n+1}=j|X_n=i\}, \quad i,j \in S \tag{1}$$

where $p_{ij}(n)$ is the probability of transitioning from States i to j at time t_n. The probability law of the next state X_{n+1} depends on the past only through the present value of the State X_n. In other words, the Markovian property refers to a condition where the memory of previously visited states between $t_1,\ldots,t_{n-1}$ is irrelevant. The $p_{ij}(n)$ are also called the single-step transition probabilities since they define the transition probabilities for one-time step only. The transition probabilities from a State i sum to 1.

Using the transition probabilities between each state pair, the complete system can be described by defining a single-step transition probability matrix $\mathbf{P}(n)$:

$$\mathbf{P}(n) = \begin{bmatrix} p_{11} & \cdots & p_{1N} \\ \vdots & \vdots & \vdots \\ p_{N1} & \cdots & p_{NN} \end{bmatrix} \tag{2}$$

This matrix provides full information regarding the behavior of the system at a particular time instant. In order to determine the individual state probabilities, a vector $\pi(n)$ is defined where $\pi(n)=[\pi_i(n)]_{1\times N}$, and $\pi_i(n)$ is the probability of being in State i at time t_n. The state probabilities at time t_{n+1} are then simply

$$\pi(n+1) = \pi(n)P(n) \tag{3}$$

If the initial state of the system is known (i.e., $\pi(0)$ is given), then the above relationship allows the state probabilities to be calculated at any t_n.

At any time instant, the state probabilities need to sum to 1, i.e.,

$$\sum_{i=1}^{N} \pi_i(n) = 1 \tag{4}$$

The case in which the single-step state transition probabilities do not depend on time is known as the *time-homogeneous Markov chain*. For systems whose state probabilities reach a limiting value as n gets larger (which is when the system is not periodic and is homogeneous), the asymptotic or steady-state behavior of π is [18]

$$\pi(\infty) = \lim_{n\to\infty} \pi(0)P^n \tag{5}$$

Since at steady state, the relation $\pi=\pi P$ has to hold true: $\pi(\infty)$ can be computed by solving the simultaneous equations given by

$$\pi_j(\infty) = \sum_{i=1}^{N} \pi_i(\infty)p_{ij}, \quad j=1,2,\ldots,N \tag{6}$$

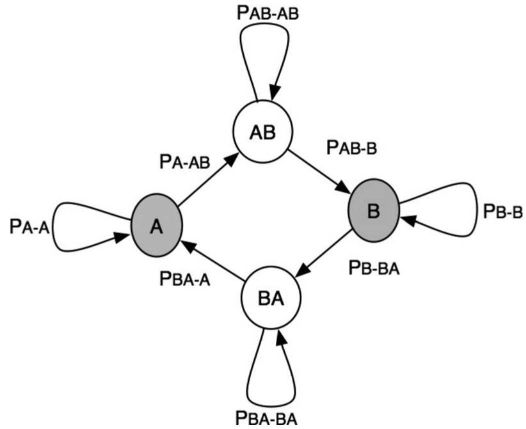

Fig. 6 Markov model of a generic reconfigurable system

Since these N equations are linearly dependent, in order to get a nontrivial solution, only $N-1$ equations are used, and the Nth equation is the property that all state probabilities should sum to 1.

$$\sum_{i=1}^{N} \pi_i(\infty) = 1 \tag{7}$$

It can be seen that the homogeneous Markov chains are quite amenable for analysis since the state probabilities can be predicted for any future time given an initial state. This allows calculation of useful statistics such as average time spent in a particular state, mean time before a certain state may be reached for the first time, and so on [19].

In the case of reconfigurable systems, the homogeneous model is applicable if the single-step transition probabilities are the same over the system's operational time.

A general model of a reconfigurable system can be constructed, as shown in Fig. 6. Two types of states are shown: *operational states* and *reconfiguration states*. Operational states are those in which the system operates and carries out its useful function. In the figure, they are States A and B (that are shaded). The reconfiguration states are those in which the reconfiguration processes are carried out, and the system may be (but not necessarily) fully or partially disabled.

For the most general case, it can be assumed that each operational state can configure into every other operational state after passing through a specific reconfiguration state. Thus, State A can transition to State AB (which is a reconfiguration state in which the system reconfigures from A to B), and then State AB goes on to State B (which is the state in which the system is in operational configuration B). A similar path is shown for B to BA to A. The self-transition probabilities $p_{AB\text{-}AB}$ and $p_{BA\text{-}BA}$ in the reconfiguration states can be used to account for factors such as reconfiguration delay or failure in any particular reconfiguration attempt. In essence, they capture the expected delay in moving from one operational state to the next. In most real systems, it is usually not possible to enter each operational state from every other operational state. However, the most general case is considered here for completeness.

For the type of system shown in Fig. 6, for k operational states, there are $k(k-1)$ reconfiguration states. In the general case, Fig. 7 shows a generic operational state and a reconfiguration state. In order to clearly differentiate between the types of states, each operational state is denoted by a single variable i or j, etc. The reconfiguration states are denoted by two variables, so, for instance, ij is a reconfiguration state that converts State i to State j. A hyphen in the subscript of the transition probabilities serves to indicate the initial and final states between which the transition takes place. Thus, $p_{i\text{-}i}$ is the self-transition probability of State i,

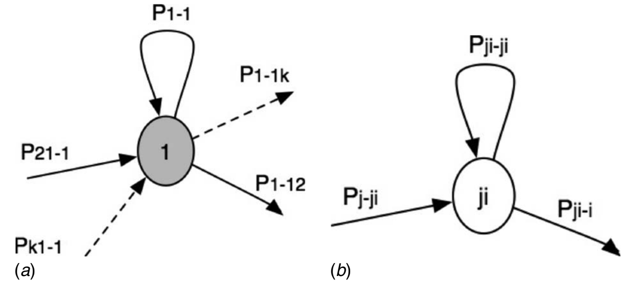

Fig. 7 Markov model of operational and reconfiguration states

while $p_{i\text{-}ij}$ is the transition probability from i to State ij. With this notation (and omitting the argument (∞) for the πs for simplicity), for the type of systems shown in Figs. 6 and 7, the results given in Eq. (6) lead to the following equations for steady-state behavior:

$$\pi_i = p_{i-i}\pi_i + \sum_{j=1}^{k} \pi_{ji} p_{ji-i}, \quad j \neq i \tag{8}$$

$$\pi_i = \frac{\sum_{j=1}^{k} \pi_{ji} p_{ji-i}}{1 - p_{i-i}}, \quad j \neq i \tag{9}$$

where

$$\pi_{ji} = \frac{p_{j-ji}}{1 - p_{ji-ji}} \pi_j \tag{10}$$

Combining the above two relations gives

$$\pi_i = \frac{1}{1 - p_{i-i}} \sum_{j=1}^{k} \frac{p_{j-ji} p_{ji-i}}{1 - p_{ji-ji}} \pi_j, \quad j \neq i \tag{11}$$

Since the transitions to reconfiguration states have the specific form, as shown in Fig. 7, it is also true that

$$p_{ji-ji} + p_{ji-i} = 1 \tag{12}$$

Using this result, Eq. (11) becomes

$$\pi_i = \frac{1}{1 - p_{i-i}} \sum_{j=1}^{k} p_{j-ji} \pi_j, \quad j \neq i \tag{13}$$

To obtain a nontrivial solution, Eq. (7) is also used. In order to have a relation only in terms of the operational states, Eq. (7) can be written by explicitly separating out the operational (π_j) and the reconfiguration state (π_{ij}) expressions:

$$\sum_{j=1}^{k} \pi_j + \sum_{i=1}^{k} \sum_{j=1}^{k} \pi_{ij} = 1, \quad i \neq j \tag{14}$$

Using Eq. (10), it becomes

$$\sum_{j=1}^{k} \pi_j + \sum_{i=1}^{k} \sum_{j=1}^{k} \frac{p_{j-ji}}{1 - p_{ji-ji}} \pi_j = 1, \quad i \neq j \tag{15}$$

Thus, the asymptotic k operational state probabilities can be determined directly using the $k-1$ linear equations given by Eq. (13) and the kth equation given by Eq. (15). Note that it is assumed that the self-transition probabilities p_{i-i} and p_{ji-ji} are not 1.

A simple example of a planetary rover can now be used for illustrative purposes. It is considered that the rover has three distinct configurations during its operation. Suppose the three configurations, or states, correspond to its operation on flat terrain (which mainly involves driving activities and is denoted as State D), its ascent or descent of a crater slope (which is denoted as C to indicate climbing up or down), and its exploration of a crater

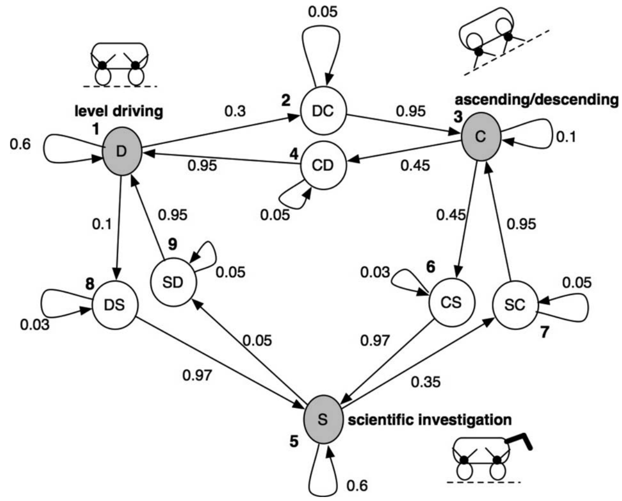

Fig. 8 Markov model of reconfigurable rover

floor (mainly carrying out scientific investigations and is denoted as S), respectively. The reconfigurable rover can employ wheels or legs for its locomotion. Therefore, it is conceivable that the wheels are used in the flat terrain driving configuration, while the legged configuration is used when climbing/descending slopes, etc. Suppose the terrain in which the rover will operate is fairly well known. Its characteristics (based on crater density, etc.) are such that a Markov model, shown in Fig. 8, can be constructed. This is only a notional case, and the assumption is that a typical mission can perhaps be described in this way given relevant parameters such as terrain profile, robot speed, mission duration, reconfiguration processes, and chances of irrecoverable failure in each state, etc.

The evolution of the operational state probabilities, π_i for $i=1,3,5$, for a number of time steps is shown in Fig. 9. Each iteration can represent a certain period of time (days, weeks, etc.) Using Eqs. (13) and (15) the steady state analytic results are found to be $\pi_1=22.32\%$, $\pi_3=17.08\%$, and $\pi_5=24.8\%$. This is in agreement with the results shown in the plots of Fig. 9, which were computed numerically.

In many cases, it is often desirable or even required to model the failure aspects of the system as well. In such a situation, a model can be employed in which a *failed* state F, from which the system cannot recover, is also factored in. This type of model would essentially describe a system in which the failed state is an *absorbing state* from which once the system enters it can never leave. There are analytical means of determining the expected number of iterations after which a system will enter an absorbing state, and those can be employed to determine failure time, etc. [19].

For the rover example, Fig. 10 shows a Markov model in which a failed state has been included. Various insights can be gained about the system's behavior through such a model. For instance, Fig. 11 shows the chances of being in any operational state (i.e., $p_o=\pi_D+\pi_C+\pi_S$) and the chances of being in any reconfiguration state (which is p_r, and is the sum of all the probabilities of reconfiguration states). Since the failure probability is very low initially, the p_r is almost equal to $1-p_o$. It can thus be determined how likely it is for the system to be performing useful work, (i.e., it exists in an operational state). This type of analysis can help in evaluating the effectiveness of the design.

With these models one can also determine the effects of changing probabilities of self-transition of reconfiguration states. These essentially capture characteristics of the reconfiguration process and can be used in making specific design choices. Figure 12 shows how the total probability of being in any operational state D, C, or S (i.e., $\pi_D+\pi_C+\pi_S$) is affected when the self-transition probabilities p_{DC-DC}, p_{CD-CD}, p_{SC-SC}, p_{CS-CS}, p_{DS-DS}, and p_{SD-SD} are reduced by 10% from the values shown in Fig. 10. Thus, by varying these self-transition probabilities, which correspond to design issues of the reconfiguration processes, the effect on the system performance can be determined.

The example of a reconfigurable UAV (described in the previous section) can also be used to further illustrate the application of this method. Figure 13 shows a Markov model that describes the states of the UAV along with the notional transition probabilities (which in reality would be based on military strategic scenarios, likelihood of finding attack targets in an area, etc.) It is assumed that the vehicle can exist in two operational states—a swept low aspect ratio wing configuration for dash (D) and an unswept high

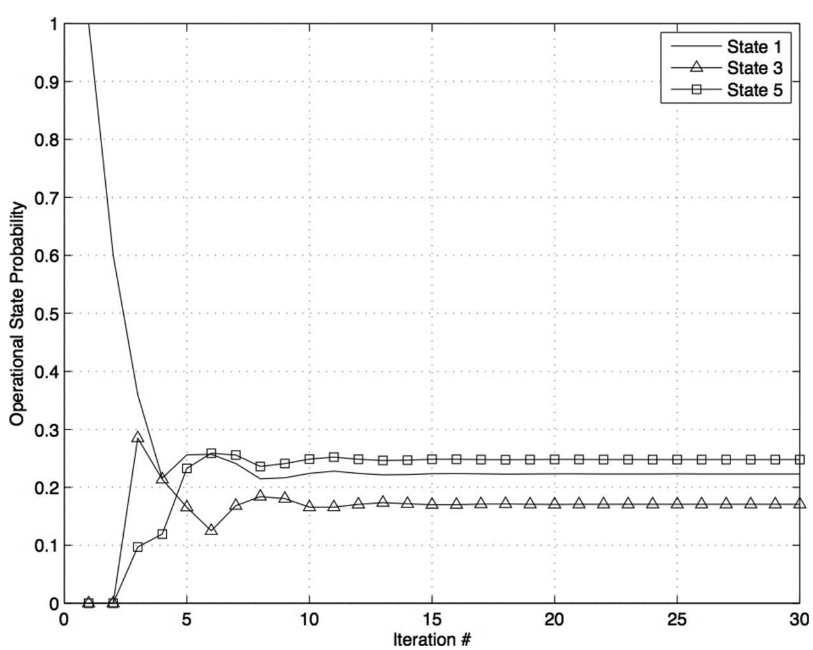

Fig. 9 Rover operational state probabilities

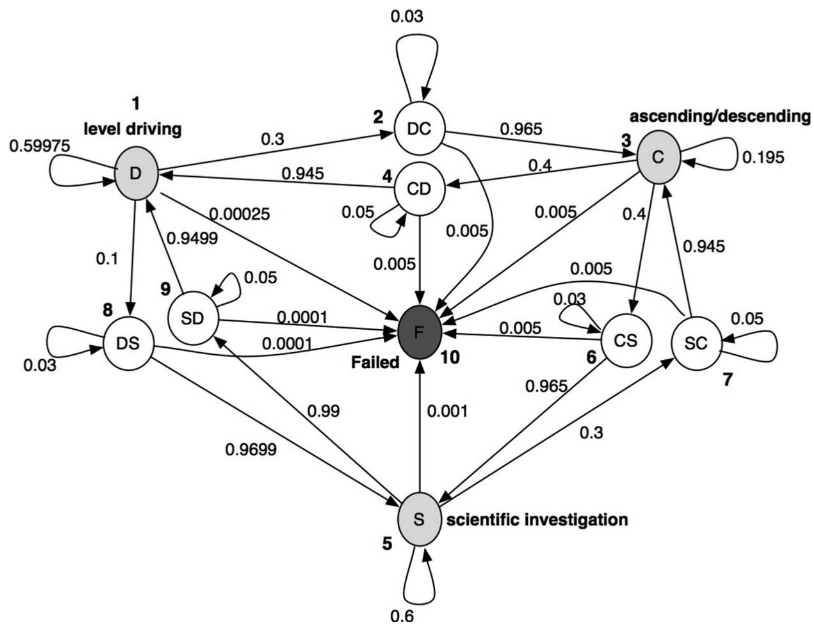

Fig. 10 Markov model of reconfigurable rover with a failure state

aspect ratio wing configuration for loiter (L). The aircraft also has reconfiguration states that correspond to the two operational states along with a nonrecoverable failure state (F).

The evolution of state probabilities in this case was computed for 400 steps, and the first 80 iterations are shown in Fig. 14.

Figure 15 shows the probability of the system being in the failed state. Three plots were generated by varying the transition probabilities leading to State F (such as p_{D-F}, p_{L-F}, etc.). In the first line (from the top), the failure probabilities were increased by 20% from the values shown in Fig. 13. In the second plot, the failure probabilities are those shown in Fig. 13, while in the third plot they have been reduced by 20%. With decreasing values of those transition probabilities (denoted collectively as P_f in Fig. 15), the chance of being in the failed state decreases. Depending on how much that decrease is, trades can be carried out between reliability and cost (since it can be expected that there will be increased cost in bringing down the failure probabilities). In this simple model, only one irrevocable failed state is considered (since the primary aspect of interest is the multiability of the UAV). For a more detailed analysis of survivability considerations, such as of graceful degradation, more states will need to be factored, capturing the degraded behavior/performance.

These examples illustrate the use of homogenous Markov models as potential tools for analysis of reconfigurable systems. In many actual applications, determining the appropriate values for transition probabilities can be difficult. Furthermore, the applicability is limited to a class of systems in which no active decision rule is used to determine the state to which the system should reconfigure rather the probabilities are assumed to be known. There are many types of systems in which the reconfiguration to a state will depend on some exogenous variable that will make some states desirable and some undesirable as a function of time. The different kinds of Markov models can be used in such cases and are discussed in the following section.

2.1.2 Time Varying Markov-Based Models. A more general case is the one in which the single-step transition probability matrix $\mathbf{P}$ is not the same for every time step. This is known as the

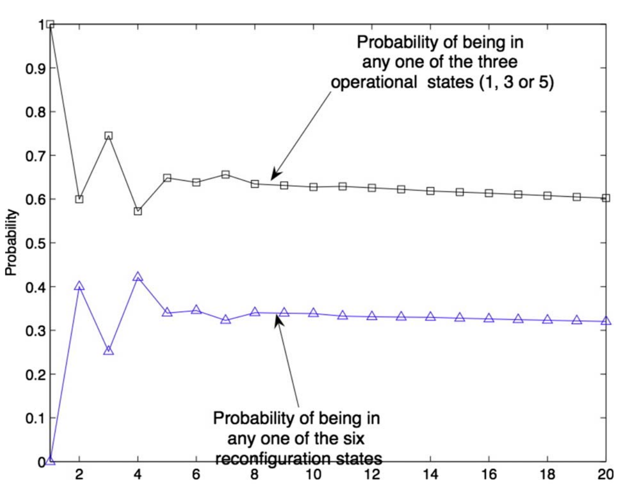

Fig. 11 Operational and reconfiguration state probabilities

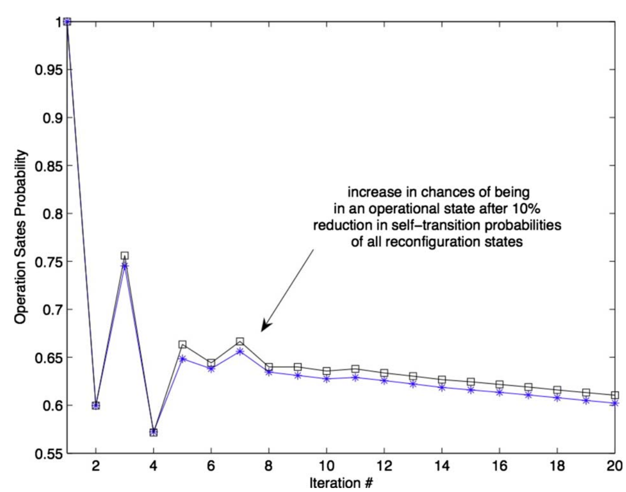

Fig. 12 Effect of 10% change in self-transition probabilities of reconfiguration states

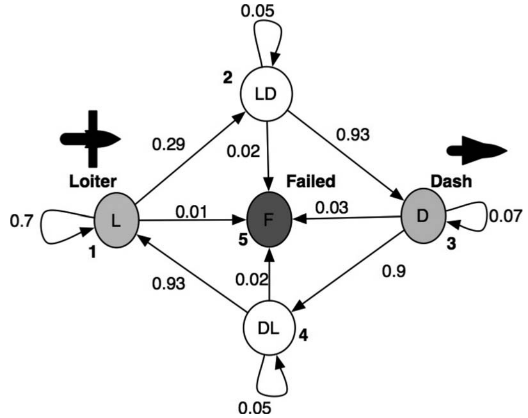

Fig. 13 Markov model of a reconfigurable unmanned aerial vehicle

nonhomogeneous Markov chain [20]. In this case, Eq. (5) does not hold, and analytical solutions to the asymptotic behavior of the system are not possible (except for the periodic cases). Indeed the system may not have a steady-state as is possible for the case of homogeneous systems.

The multistep transition probabilities are computed by taking a complete product of each $\mathbf{P}(n)$ [18]:

$$\Phi(m,n) = \mathbf{P}(m)\mathbf{P}(m+1), \ldots, \mathbf{P}(n-1), \quad m \leq n-1 \qquad (16)$$

Most reconfigurable systems can be better described through nonhomogeneous models, with a further assumption that the state transition probabilities at each time instant are conditioned on some external time-varying process $u(n)$. Thus for any State-Pair i and j (and using the notation with the hyphen)

$$p_{i-j}(n) = f(u(n),i,j) \qquad (17)$$

This $u(n)$ can be mapped to a corresponding behavior of the system such that some objective J is achieved. From a given set of finite states that the system can attain in one step, there will be a state i^* that is most desirable for the system to have for a particular $u(n)$ and a particular formulation of J. In fact, each state will have an associated J_i for the given $u(n)$, and the state transition probabilities will be according to this J_i. The optimal state i^* will have the highest probability for the system to transition into, followed by the next best state, and so on. The states that are worse than the existing configuration can be modeled to have extremely low transition probability for that time period. For a given input $u(n)$, the optimal operational state is

$$i^* = \arg\max J(u,S) \qquad (18)$$

where J is some function that needs to be maximized, and S is the set of system states. Also as described above,

$$p_{m-mj} > p_{m-mk}, \quad J_j > J_k > J_m \qquad (19)$$

where p_{m-mj} is the probability that operational state m will transition to the reconfiguration state mj and so on. Note, that $u(n)$ could be a vector, making $J(u,S)$ more complex. Also, one of the reasons that p_{ij} is probabilistic is due to the estimation error of $u(n)$ among other factors. As an example, consider a reconfigurable satellite constellation that is capable of changing its coverage to different locations on the globe in response to disaster relief efforts. For simplicity, it is assumed that it can attain a set of discrete states. The input to this system is the stochastic process of disaster occurrence (e.g., floods, earthquake, and hurricane) in various regions of the globe. Given a certain demand at a particular time $u(n)$, there is an optimal state (among the discrete and finite set) that the satellite constellation should adopt so that it maximizes its communication service and coverage to the required region. The Function J can thus be the revenue (or perhaps some metric of service), and the state that maximizes J should get the highest transition probability for that time step. Thus, the transition probabilities will be such that p_{j-ji^*} are high (i.e., the probability of moving from a suboptimal state to its corresponding reconfiguration state that leads to i^* will be high). Also, if the system is already in the optimal state, it will stay there, i.e., $p_{i^*-i^*}=1$ and $p_{i^*-i^*j}=0$. In other words, it will be an absorbing state but only during the period of time when its optimal. Since in the nonhomogeneous case, the absorbing states change with time (with the exception of irrevocable failure), the long-term behavior

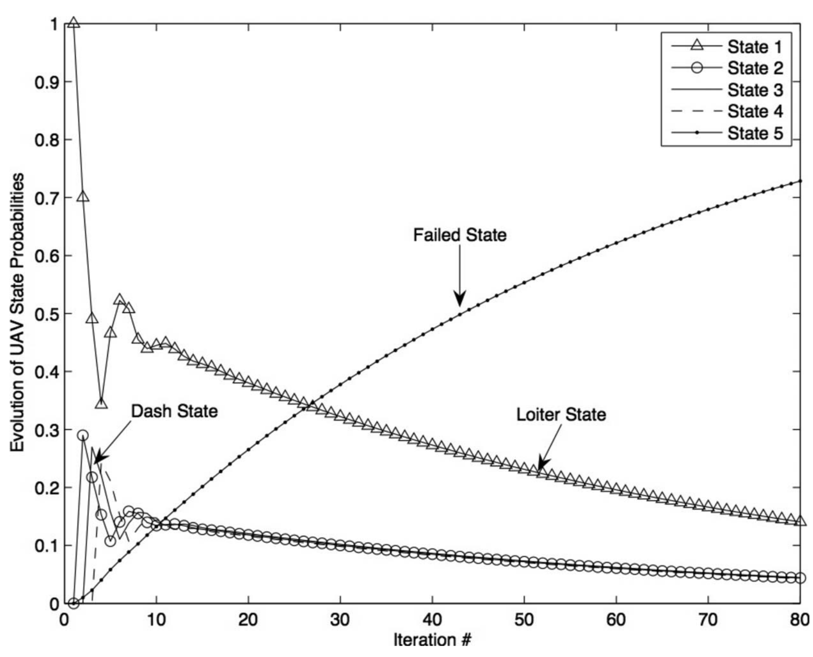

Fig. 14 Evolution of UAVs state probabilities

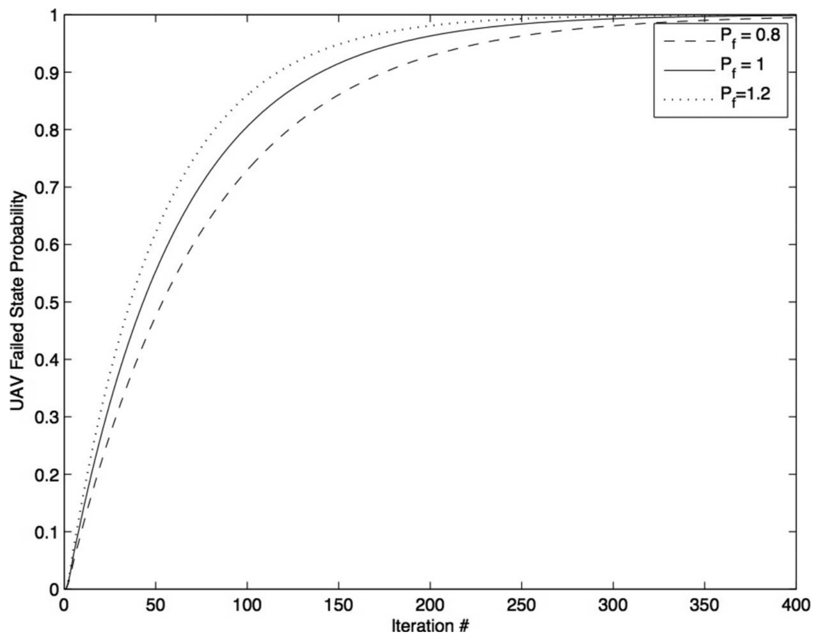

Fig. 15 Effect of 20% increase and decrease in failure probability over time

cannot be predicted analytically (unless $u(n)$ is periodic).

One way to assign p_{ij}s is to establish a rule that a state transition will occur if the "net benefit" F_{ij} is positive, where

$$F_{ij} = \Delta J_{ij} - C_{ij} \tag{20}$$

$$\Delta J_{ij} = J_j - J_i \tag{21}$$

The ΔJ_{ij} essentially is the difference in performance of the two States i and j and should be positive for a transition to occur (i.e., the system moves toward a better state). Additionally, C_{ij} represents the cost of transitioning from i to j. Note that by this definition, remaining in the same state incurs no additional benefit or cost:

$$F_{ij} = 0, \quad i = j \tag{22}$$

If $F_{ij} > 0$ then it means that there is a net benefit to be gained by transitioning from i to j even while accounting for the costs involved. One possible formulation for p_{ij} can be

$$p_{ij} = \frac{F_{ij}}{\sum_j F_{ij}}, \quad j \in S' \tag{23}$$

where $S' \subseteq S$ and consists of states for which $F_{ij} > 0$. A detailed discussion and application of this method to reconfigurable planetary rovers is provided in Ref. [21].

2.2 Metacontrol Framework. Another scheme of modeling and representing reconfigurable systems can be based on concepts and tools of classical control theory. Control-theoretic approaches have been used in a metasense for a variety of applications ranging from modeling organizational dynamics to human-machine interaction [22]. These are systems that cannot be described through physical dynamical equations; nonetheless the tools of control theory lend themselves to studying certain basic time-related characteristics of such systems.

It is proposed that a control-theoretic approach can also be applied to a class of reconfigurable systems that can undergo two kinds of reconfigurations: *online* reconfiguration and *off-line* reconfigurations. In the former case, the system continues its operation as it reconfigures, while in the latter case it is nonoperational during the reconfiguration process. For instance, a rover may undergo online reconfigurations as it explores the surface of a planet, or it may be reconfigured off-line (by a crew in a manned exploration mission).

Figure 16 shows how online and off-line reconfigurabilities can be modeled in a generic manner. Typically, there is some change in the system's environment (such as planetary surface terrain for a rover or strategic conditions in a military zone). Those changes are sensed and mapped to some corresponding desired attribute of the system (such as the desired characteristics of a rover's wheel in response to terrain conditions, wing geometry of a UAV in

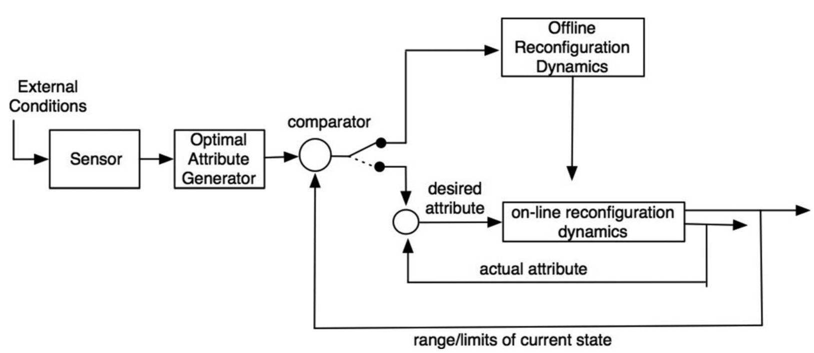

Fig. 16 Generic reconfigurable system

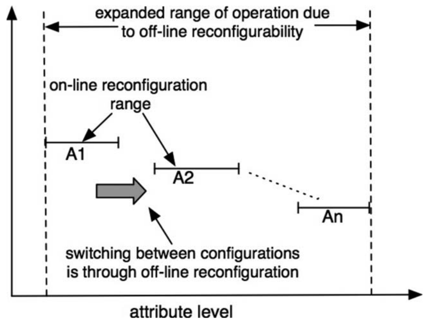

Fig. 17 Online and off-line reconfigurations

response to changing mission needs.) The reconfigurable system tries to achieve the desired configuration through its online reconfiguration loop (which is essentially the traditional type of active control in many modern systems). If the online reconfiguration bandwidth, i.e., the limits to which the system can configure online, are not sufficient for fulfilling the desired objective, then the system goes through its off-line reconfiguration process. Through off-line reconfiguration, it undergoes changes such that it can achieve the desired states once it is operational again.

The set of states or configurations that the system can adopt through off-line reconfiguration can be represented as

$$\mathbf{A} = \{A_1, A_2, \ldots, A_n\} \tag{24}$$

Each configuration A_i has its specific online reconfiguration range δA_i defined as

$$\delta A_i = a_i^{\max} - a_i^{\min} \tag{25}$$

This defines the extent to which the system can adapt itself while in operation in a particular state A_i. Through the combination of online and off-line reconfigurabilities, the total range of an attribute that the system can change through reconfiguration is enlarged (as illustrated through Fig. 17). It is possible, however, that this total range/reconfiguration bandwidth may not always be continuous. If the example of a reconfigurable UAV is considered again, one can conceive of a design that can go through both online and off-line reconfigurations. The online reconfiguration would involve in-flight wing geometry changes. The off-line reconfigurations (performed between sorties of the vehicle) can potentially involve switching out components (perhaps even wing modules), payload, etc., such that the most suitable ones are used for different usage scenarios.

Note, that, traditionally, what is being referred to as online and off-line reconfigurabilities has been studied separately [23,24]. This approach however, by combining the two types of reconfigurations, can allow for exploring a larger design space for the system. The metacontrol framework expands the spectrum of possibilities that can be assessed for the system.

The three methods discussed above provide ways to model and assess the time related aspects of reconfigurations. The Markov approach allows for analyzing operational, reconfiguration, and failure states. Figures 12 and 15 show examples of how the effects of changes in self-transition or failure probabilities can be evaluated and then possibly traded in the design process. One can quantify the effect of change in failure probabilities through such analysis and then translate the results into design (or performance) requirements for subsystems. Such requirements in turn guide decisions on selecting components. The analysis of reconfiguration time aspects (as allowed by the metacontrol framework) can lead to a definition of requirements on the physical design such as choice of actuators and mechanisms. Similarly, through the analysis of A (number of off-line configurations) and δA_i (extent of online reconfigurability), one can produce requirements for system form. Such as in the case of a reconfigurable UAV, the number of wing types (that can be fitted on to the aircraft) and the extent of each wing's in-flight reconfigurability would be determined so that the overall system performance across all the operational scenarios of interest is satisfied to some desired level/goal. A more detailed discussion on the application of these methods can be found in Ref. [21], where it is shown how the states of a reconfigurable wheel may be determined for application in planetary surface exploration vehicles.

It should be noted that these modeling methods have to factor in the available computational resources. The computational requirement may become significant for a real world complex system. In such a case, the issue may be addressed by balancing the time fidelity of the simulations versus the number of modeled states so that the problem is reduced to a computationally manageable size.

3 Survey of Reconfigurable Systems

The preceding sections discussed various theoretical notions—definitions and modeling techniques—for studying reconfigurability. In order to gain broader insights for implementation of reconfigurability, a survey was conducted on a set of different types of reconfigurable systems. This section presents some of the common trends that were observed and guidelines that were elicited from analyzing their designs.

3.1 Systems Description. A set of 33 different systems ranging from a simple potentiometer to a large complex radio telescope system, the very large array (VLA), was selected for the study. Some of their basic information is given in Table 1 and more details are provided in Ref. [25].

The *Functional Type* in Table 1 has been assigned based on the definitions proposed in Ref. [43]. The letters E, M, and I represent energy, matter, and information, respectively. The number 1 is associated with functions that transform or process the operand, 2 is for transport or distribute, 3 is for store or house, 4 is for exchange or trade, and 5 is for control or regulate [43]. It can be seen that the selected systems mostly are described by M1, M2, M3, and I1 types, i.e., most of these systems are matter or information processing (M1, I1), or mass transportation (M2), or storage/housing (M3). Note that these types are associated with the *primary* function of the system, for there are certainly several other subprocesses that are of different types and enable the primary function to be carried out.

It should be noted that these systems range from conceptual stage (e.g., variable diameter compound helicopter) all the way to full production and deployment stage (e.g., SMART car). Some kind of data such as cost, etc., however, could not be obtained for all the 33 systems, and thus there are some empty cells in Table 1.

3.2 System Requirements. The three system properties that were identified for driving the need for reconfigurability in Sec. 1 were matched against the stated requirements for each systems. In order to get any meaningful insights, the systems were classified in three application categories of commercial/consumer items, air/space systems, and manufacturing/test systems. Systems 1–15 (i.e., from potentiometers to race car in Table 1) were the commercial/consumer items, systems 16–18 (NI-RIO to RMS) were manufacturing/test, and systems 19–33 (Polybot to VLA) were designated as air/space systems.

In this analysis, the multiability property was refined into two subcategories of "resource efficiency" and "multiple configurations" (see Fig. 1). These categories along with the other two (evolvability and survivability) were then assigned to each system based on their stated objectives and requirements as found in their description and other relevant literature sources. A 33×4 matrix was created in which a value of 1 was assigned to the element in the ith row and jth column if the ith system fulfilled the jth need subcategory. The fraction of systems assigned to each of these four categories (for each of the three application domains) was determined and is shown in a plot in Fig. 18.

It can be seen that the different classes show varying trends in

Table 1 Set of systems used for reconfigurability analysis. Legend: E, energy; M, mass; I, information; 1, transform; 2, transport or distribute; 3, store or house; 4, exchange or trade; 5, control or regulate

No.	Name	Func-type	Mass (kg)	Volume (m^3)	Cost ($)
1	Potentiometer	E1	0.01	1×10^{-6}	2.5
2	Airpot (adjustable shock absorber) [26]	E1	0.07	7.6×10^{-5}	20
3	LEGO	M1	1	3×10^{-3}	14
4	Vaccum	M4	7.5	6.88×10^{-2}	650
5	Food processor	M1	6	1.5×10^{-2}	150
6	Sewing machine	M1	9.18	4.7×10^{-2}	130
7	Convertible stroller [25]	M2	4.09	2.9×10^{-1}	70
8	Digital photoframe	I3	0.82	8.6×10^{-4}	200
9	USM Haller table [27]	M3			5000
10	3-in-1 crib	M3	36.4	1.44	250
11	Sofa bed	M3	41.8	2.74	550
12	Adjustable bed	M3	61.4	1.02	2700
13	Convertible car	M2	1590	12	35,000
14	SMART car [28]	M2	730	5.63	14,000
15	Flexible race car [23]	M2	500	7.99	99,500
16	Reconfigurable input/output (NI-RIO) device [29]	I1	2	9.35×10^{-4}	2000
17	Reconfigurable discrete die [30]	M1	3980	2	500,000
18	RMS [6]	M1	32,700	166	1×10^{6}
19	Polybot (reconfigurable modular robot) [31]	M2	3.6	2.3×10^{-3}	10,000
20	LARA (reconfigurable rover) [32]	M2	50		
21	SRR (sample return rover) [33]	M2	10		
22	Solar maximum mission (SMM) [34]	I1	2320	6.36	1.2×10^{8}
23	SWARM (reconfigurable spacecraft) [35]	I2	25	3.16×10^{-2}	
24	Xilinx Virtex II Pro FPGA	I1	0.001	7.29×10^{-6}	250
25	Evolvable hardware (EHW) [3]	I1		6.25×10^{-6}	
26	Long life spacecraft avionics [36]	I1		4.05×10^{-3}	180
27	Reconfigurable communications equipment (RCE) [37]	I1	16	1.82×10^{-4}	500
28	Reconfigurable patch antenna [38]	E4		1.25×10^{-6}	80
29	TTC transponder [5]	I1	2.9	6.97×10^{-3}	
30	MXF-1 (Morphing UAV) [9,39]	M2	45.45		
31	F-14 Tomcat [40]	M2	33,800	1820	3.8×10^{7}
32	Variable diameter compound helicopter (VDCH) [41]	M2	9550	90.6	1×10^{8}
33	Very large array (VLA) [42]	I4	6.2×10^{6}	376,000	7.9×10^{7}

the reasons for their reconfigurability. The consumer items are dominated by the two subcategories of multiability. They either are reconfigurable due to resource efficiency requirements or have multiple-configurations for some planned or unplanned usage needs. Space systems, on the other hand, are motivated by requirements of survivability and evolution for unknown configurations in addition to multiability. In manufacturing, the system resource efficiency, the multiple-configurations, and the evolution aspects all play a key role.

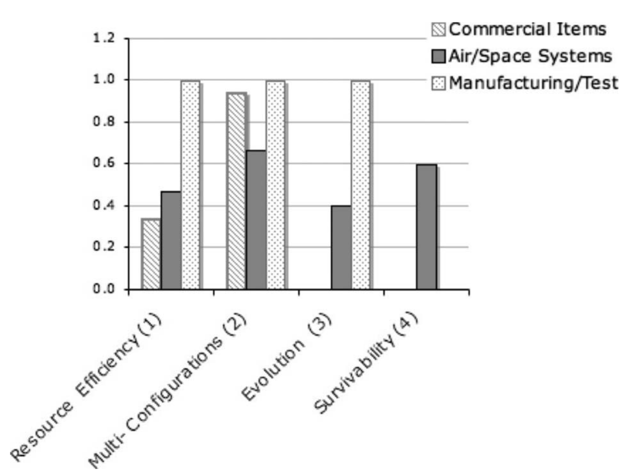

Fig. 18 Reconfigurability drivers for systems in three application domains

3.3 Reconfiguration Time. The time related aspects were studied for the systems for which relevant data could be obtained. The reconfiguration time, denoted as T_r, is the time a system takes to reconfigure from one state or configuration to its next state/configuration (see Fig. 6). Depending on the context it may include the total time spent in determining the new configuration (e.g., in case of evolution) in addition to the actual time spent in carrying out the reconfiguration processes on/by the system.

The surveyed systems for which reconfiguration time information could be obtained are listed with relevant data in Table 2.

For consumer items such as vacuum cleaners and food processors, the time was estimated from personal experience as a user of these systems, whereas for other systems such as the F-14, and VLA, the reconfiguration time was obtained from actual data. In the case of the solar maximum mission (SMM), the reconfiguration time was taken to be the total time it took for servicing the craft (which was 2 days, since the first attempt to capture the satellite failed on the first day of the mission) [34]. It does not include the time of getting the mission approved, ready, and launched to carry out the reconfiguration. In the case of SWARM also, the time is the docking time for the various modules based on the assumption that the docking-undocking procedures are what primarily constitute the reconfiguration process in the SWARM system [25]. For NI-RIO the data are again based on

Table 2 Reconfiguration times

No.	Name	Reconfiguration time	Average state occupancy time	Life (yr)
4	Vaccum	30 s	15 min	2+
5	Food processor	60 s	20 min	2+
6	Sewing machine	10 s	3 min	2+
7	Convertible stroller	3 min	30 min	1+
8	Digital photoframe	5 min	1 day	3+
10	3-in-1 crib	30 min	12 months	2+
11	Sofa bed	5 min	8 h	5+
12	Adjustable bed	60 s	2 h	2+
13	Convertible car	3 min	30 min	
14	SMART car	2 days	3 months	
16	NI-RIO	15 min	8 h	5–7+
17	Reconfigurable discrete die	30 min	20 h	5+
19	Polybot	30 s	5 min	
22	SMM	2 days	5 yr	5+
23	SWARM	120 s	30 min	
30	Morphing UAV	10 s	5 min	
31	F-14 Tomcat	6 s	15 min	30+ (7200 op h)
33	VLA	14 days	3 months	30+

personal experiences as a user, and for the reconfigurable discrete die, the data are based on a first order estimate and are meant to only capture the order of magnitude of the time.

The average time the system spends in a particular state or configuration, denoted as T_s, was considered to be the average state occupancy time. For instance, the VLA stays in each of its four configurations A, B, C, and D for three months on average. So the T_s for the VLA is 3 months. For consumer items, and other items such as NI-RIO or the reconfigurable discrete die, the T_s was again estimated from general experience and basic assumptions.

The log of the ratio of T_s and T_r was computed and plotted, as shown in Fig. 19. An interesting observation that can be made is that there is a lower bound of approximately 1 for this ratio, i.e., the reconfiguration time is never more than *at least* $1/10$ of the time the system spends in a particular configuration. One might state that reconfiguration times that exceed 10% of the useful time in any given state render reconfiguration undesirable. The truly acceptable maximum T_r depends on the exogenous dynamics (also discussed earlier in Sec. 2). Note that this fact distinguishes reconfigurable systems from pure redesign or unplanned remodeling. This 10% rule can perhaps be used as a useful rule of thumb when considering the design of a system that needs to reconfigure between different states. It should be noted that where the data have been estimated from experience or on a first order level, the lower limit for the state occupancy time and an upper limit for the reconfiguration time have been used. Therefore, in reality, on average the state occupancy time for most of these systems will be longer, and the reconfiguration time will be shorter. The ratio will therefore be even higher, and the proposed ratio of at least one-tenth should still hold.

3.4 Architecture. The architecture of the 33 systems was also evaluated. From the review of their physical design, it emerged that all the systems could be categorized into three main types based on their modularity (see Table 3). The first type can be termed as *self-similar modular* since all the modules in that type are exactly or almost identical (such as in the case of Polybots, SWARM, and VLA) The second type can be defined as *reconfigurand modular* since these are systems in which only the reconfigurand (the thing that is reconfigured) is implemented as an independent module in the system. It typically has well defined interfaces for easy reconfiguration, while the rest of the parts/subassemblies are well integrated. Examples of such systems are vaccums, food processors, and SMART cars. The third type did not exhibit any appreciable modularity and was integral in its architecture from a reconfigurability perspective (such as the F-14 and morphing UAV).

There is an increasing degree of integration, or decreasing level of modularity in going from the first to the third type. In the self-similar modular systems, large constitutive chunks of the system can (and do) undergo reconfiguration, while in the reconfig-

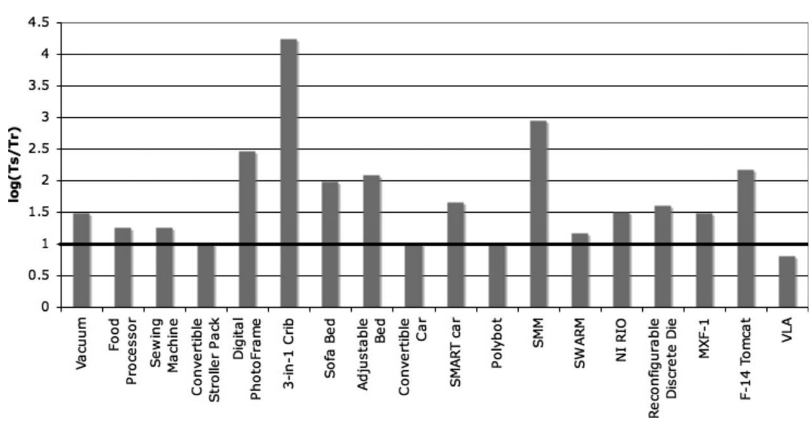

Fig. 19 Reconfiguration time ratios

Table 3 Architecture types

No.	Self-similar	R-modular	Integral
1	Polybot	Vaccum	Potentiometer
2	LARA	Food processor	Airpot
3	Xilinx FPGA	3-in-1 crib	Sewing machine
4	EHW	SMART car	Convertible stroller
5	SC avionics	SMM	Digital photoframe
6	RCE		Sofa bed
7	Reconfigurable die		Adjustable bed
8	VLA		Convertible car
9	LEGO		Flexible race car
10	USM haller table		Sample return rover
11	SWARM		Reconfigurable patch antenna
12	NI-RIO		TTC transponder
13	RMS		MXF-1
14			F-14 Tomcat
15			VDCH

urand modular systems only a smaller chunk relative to the rest of the system is reconfigured. In the integral case, the chunk is also small and its degree of reconfiguration is also limited (consists mostly of transposition) and cannot be removed from the system easily.

4 Principles for Reconfigurable System Designs

Based on the review of the selected systems along with additional consideration of many other reconfigurable systems, a few general principles were synthesized.

4.1 Principle of Reconfigurability

For every configuration of a reconfigurable system, there exists a corresponding dedicated system that is AT LEAST equal in performance.
A good reconfigurable design is one in which the performance of each configuration approaches that of the corresponding dedicated system.

This principle can be easily proven by comparison of an application specific integrated circuit (ASIC) and a FPGA (for which a similar law has been proposed [44]). An ASIC solution can be implemented on a FPGA, and then if the extra (unused) elements, i.e., interconnects, and logic blocks, are removed, the resulting system will be one that uses less power, is more dense, and is even cheaper when produced in volume. This notion can be extended to any system in general. For instance, consider the case of a morphing UAV; for every configuration that the UAV can assume, a corresponding fixed aircraft can be built that would at least be equal and probably be even better in performance.

Keeping this principle in view, comparison metrics can be developed for specific systems to test the goodness of the reconfigurable designs. A few specific examples of such metrics and their application in reconfigurable system design can be found in Ref. [25].

4.2 Principle of Self-Similarity

Systems with self-similar modules, have highest degree of reconfigurability.
Common modules should be maximized across configurations.

Systems composed of identical or very similar modules are the easiest to reconfigure radically (hence are greatly *reconfigurable*). This has been qualitatively proposed earlier [45]; however, the survey of systems illustrates this notion empirically. LEGOs, LARA, Polybot, SWARM, avionics based on identical generic modules [36], etc., are all self-similar systems that exhibit a high degree of reconfigurability. Their form can be radically altered, and their functions (not just functional attribute) can be completely different. Table 3 shows that of the 33 systems that were studied, 13 or approximately 40% of them exhibit self-similar architecture. This high proportion is indicative of the effectiveness of common modular architecture in achieving system reconfigurability. Figure 20 shows a schematic of a self-similar system along with a prototype of the Polybot robot [31] that is based on such an architecture. A detailed discussion and analysis of self-similarity and reconfigurability can be found in Ref. [46].

4.3 Principle of Information Reconfiguration

Maximize the informational nature of the element under frequent reconfiguration.
Reconfiguration costs of informational elements and interfaces is usually low.

Maximizing the informational nature is desirable since it is easier to change information than physical matter/material. Reconfiguring a system informationally can thus be easier than reconfiguring it physically.

A few examples of systems that demonstrate this principle are reconfigurable displays such as digital photo frames or touch screen control panels. Virtual instruments (software based measurement and control instruments [47]) powerfully show the benefits of this approach. In virtual instrumentation, the traditional hardware implementation of a specific measurement system is replaced through an equivalent personal computer (PC)-based reconfigurable hardware and software solution [29]. For instance, in a traditional oscilloscope the number and type of input channels, signal range, and many other functions are predefined and fixed. In a reconfigurable PC-based measurement solution, the user can easily change the configurations and create new and different measurement systems, as required over time (see Fig. 21).

Software-radio based satellite transponders also serve as good illustrative examples. Figure 22 shows how in a proposed reconfigurable satellite transponder, the traditional hardware filters and

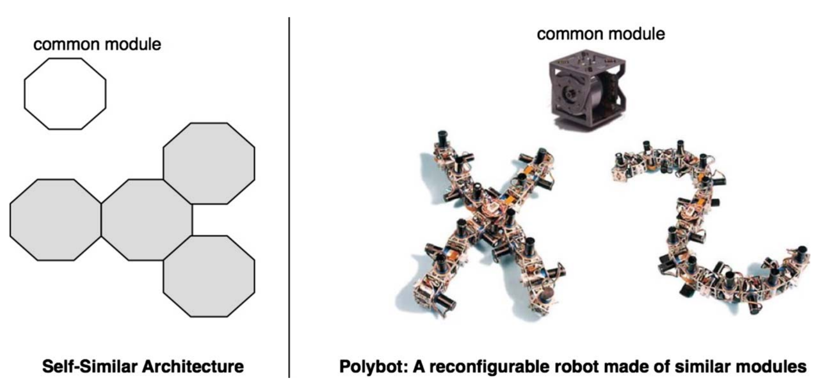

Fig. 20 Self-similar architecture in reconfigurable systems [31]

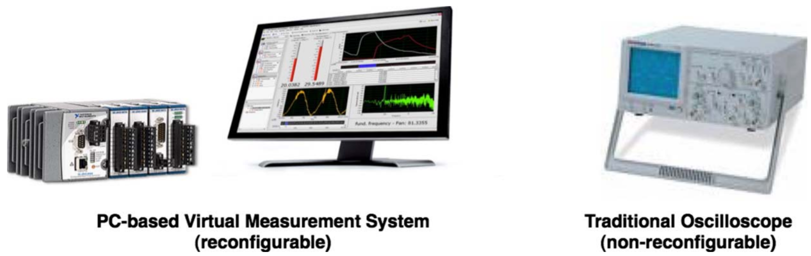

Fig. 21 Reconfigurable and traditional measurement systems

other circuit elements will be replaced with software [4]. The physical nature is thus reduced (through reduction of hardware circuitry) and the informational nature is increased (with the implementation of digital signal processing solutions).

Another example of the application of this principle is in changing traditional physical connections between subsystems to wireless links. The physical manifestation (in the form of structural connection, electrical wires, etc.) is changed into a more information based implementation (through infrared waves that transmit necessary information between sensors, actuators, etc.) Physical interfaces have spatial constraints, while wireless interfaces do not and thus lend themselves naturally to easy reconfigurations. Wireless interfaces have therefore been identified as the most adaptable interface [48]. There has been a noticeable trend toward designs (of reconfigurable systems) in which the subsystems are only linked through a wireless interface. Some specific examples include reconfigurable modular spacecraft that fly in formation and communicate wirelessly [49] to effectively function as one satellite but are able to undergo radical reconfigurations. In drive-by-wire technologies in cars, the physical connections between the passenger compartment and chassis are eliminated through information based (wireless) links. The passenger inputs to the brakes and steering are communicated wirelessly to the wheels and steering system. This allows for removal of all physical connections between the passenger compartment and the chassis, and thus enables radical reconfigurations of the car such as being able to attach different types of passenger compartments/interiors [50]).

It is important to note that increasing the informational nature of a system can be expensive. Implementation of digital solutions (versus analog) can come with higher cost. The cost of reconfigurability (nonrecurring cost incurred in making the system reconfigurable in the first place) maybe higher, but the cost of reconfiguration (recurring cost incurred while carrying out reconfigurations) is usually much lower. It is therefore beneficial to increase the informational nature of those system that have to reconfigure frequently.

5 Conclusions

Reconfigurability in systems can be the technical means of responding to change. Reconfigurability is thus important when the underlying aim is to design not just for the short term cost and performance goals but also for long-term life-cycle issues.

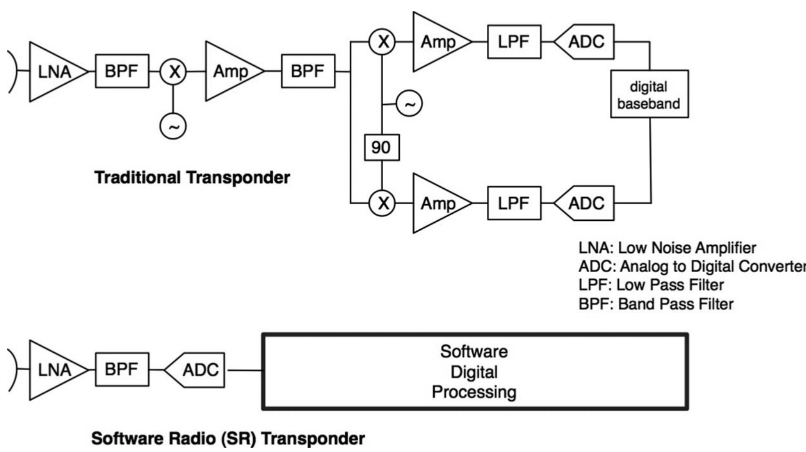

Fig. 22 Reconfigurable satellite transponder

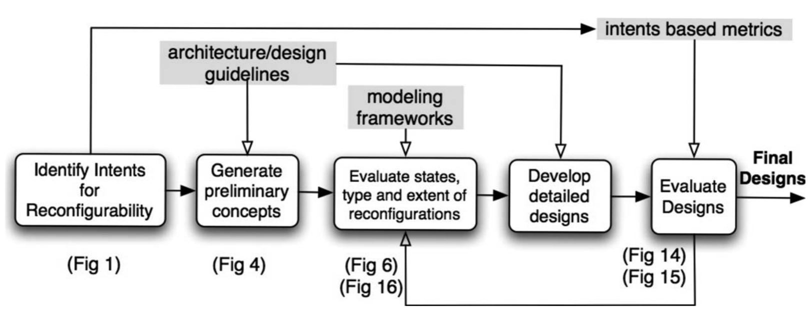

Fig. 23 Reconfigurable system design process

The modeling methods and high-level principles proposed in this paper can aid in the design process of reconfigurable systems. Figure 23 illustrates how they can fit in a general process of designing for reconfigurability. In the first step it is necessary to establish the driving factors for system reconfigurability (see Fig. 1). The identification of the intents allows among other things a value based assessment of the concepts that are generated at a later stage. In the second step, preliminary concepts are generated (Fig. 4(*a*)) and are then used as a basis for modeling and evaluating various system states/configurations. The Markov modeling and metacontrol frameworks can be employed toward this end (Figs. 6 and 16). They can also aid in performing trades between few configurations with large reconfiguration bandwidth, (i.e., the number of elements, n in set A is small and the δA_i are large) and many configurations with small bandwidths (n is large while δA_i are small). The results from the modeling and analysis process, are then used for developing detailed designs of the system. This process can draw from the principles and guidelines that were summarized in the previous section. Once a set of designs is obtained, the evaluation metrics based on the driving requirements (such as resource efficiency and survivability) can then be used for a meaningful comparison (Figs. 14 and 15). This process will be an iterative one in which the designs are refined until the desired goals are satisfied after which the final designs are selected.

For large, complex, reconfigurable systems that may exist in several states/configurations, additional research in analytic techniques is required that would be applicable in the detailed design stage of the process. The ultimate objective in the design of any reconfigurable system is driven by the principle of reconfigurability, i.e., the ideal design of a reconfigurable system would be one in which each state/configuration matches closely with an corresponding optimally designed fixed system. Collaborative optimization or Bilevel integrated system synthesis (BLISS) [51] can potentially be employed toward this end. These methods have been used to optimize various subsystems of a complex system. In their application to reconfigurable system design, instead of subsystems the various states or configurations of the system would be improved. The resulting overall design will thus be one in which each state of the system has been optimized within the given constraints.

Acknowledgment

This research was supported by the Richard D. DuPont Fellowship, and the National Aeronautics and Space Administration's Contract No. NNK05OA50C.

Nomenclature

p_{ij} = transition probability from State i to j
$P(n)$ = single-step transition probability matrix
C_{ij} = cost of reconfiguring from State i to j
F_{ij} = net benefit in transitioning from State i to j
J_i = system performance in State i
δA_i = online configuration range of State A_i
$\pi_i(n)$ = probability of being in State i at time step n
$\Phi(m,n)$ = multistep transition probability from m to n

References

[1] Compendex, http://www.engineeringvillage2.org.
[2] Underwood, K., 2004, "FPGA vs. CPUS: Trends in Peak Floating-Point Performance," *Proceedings of the ACM International Symposium on Field Programmable Gate Arrays*, Feb. 24, Monterrey, CA.
[3] Keymeulen, D., Zebulum, R., Jin, Y., and Stoica, A., 2000, "Fault-Tolerant Evolvable Hardware Using Field-Programmable Transistor Arrays," IEEE Trans. Reliab., **49**(3), pp. 305–316.
[4] Mondin, M., Presti, L., and Scova, A., 2002, "A software Radio-Based Reconfigurable Transponder for Space Applications," Wireless Communications and Mobile Computing, **2**, pp. 839–845.
[5] Cabo, J., Bravo, R., Lopez, E., and Casas, O., "Galileo TTC Transponder: A Multimode In-Orbit Reconfigurable Transponder."
[6] Mehrabi, M., Ulsoy, A., and Koren, Y., 2000, "Reconfigurable Manufacturing Systems: Key to Future Manufacturing," J. Intell. Manuf., **11**, pp. 403–419.
[7] Chen, L., Xi, F., and Macwan, A., 2005, "Optimal Module Selection for Preliminary Design of Reconfigurable Machine Tools," J. Manuf. Sci. Eng., **127**, pp. 104–115.
[8] Joshi, S. P., Tidwell, Z., Crossley, W. A., and Ramakrishnan, S., 2004, "Comparison of Morphing Wing Strategies Based Upon Aircraft Performance Impacts," *45th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference*, April 2004, Palm Springs, CA, paper No. AIAA-2004-1722.
[9] http://www.nextgenaero.com/success_mfx1.html.
[10] Miller, J., Guerrero, J., Goldstein, D., and Robinson, T., 2002, "Spaceframe: Modular Spacecraft Building Blocks for Plug and Play Spacecraft," *16th Annual/USU Conference on Small Satellites*, 2002, Logan, UT.
[11] de Weck, O. L., Nadir, W., Wong, J., Bounova, G., and Coffee, T., 2005, "Modular Structures for Manned Space Exploration: The Truncated Octahedron as a Building Block," *First Space Exploration Conference: Continuing the Voyage of Discovery*, 30 Jan.–1 Feb. 2005, Orlando, FL, Paper No. AIAA-2005-2764.
[12] Clark, P., and Rilee, M., 2004, "Bees for Ants: Space Mission Applications for the Autonomous Nano Technology Swarm," *AIAA First Intelligent Systems Technical Conference*, Sept. 2004, Chicago, IL, Paper No. AIAA-2004-6303.
[13] Olewnik, A., Brauen, T., Ferguson, S., and Kemper, L., 2004, "A Framework for Flexible Systems and Its Implementation in Multiattribute Decision Making," ASME J. Mech. Des., **126**, pp. 412–419.
[14] Singh, V., Skiles, S., Krager, J., Wood, K., Jensen, D., and Szmerekovsky, A., 2006, "Innovations in Design Through Transformation: A Fundamental Study of Transformation Principles," *ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Sept. 10–13, 2006, Philadelphia, PA, Paper No. DETC2006-99575.
[15] Khire, R., and Messac, A., 2006, "Selection-Integrated Optimization (SIO) Methodology for Optimal Design of Adaptive Systems," *ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Sept. 10–13, 2006, Philadelphia, PA, Paper No. DETC2006-99322.
[16] Dori, D., 2002, *Object-Process Methodology*, Springer, New York.
[17] http://www.spaceagecontrol.com/s050ai.htm.
[18] Howard, R., 1971, *Dynamic Probabilistic Systems Vol. 1: Markov Models*, Wiley & Sons, New York.
[19] Bertsekas, D., and Tsitsiklis, J., 2002, *Introduction to Probability*, Athena Scientific, Belmont, MA.
[20] Feller, W., 1968, *An Introduction to Probability Theory and Its Applications*, Vol. 1, 3rd ed. Wiley & Sons, New York.
[21] Siddiqi, A., Iagnemma, K., and de Weck, O., 2006, "Reconfigurability in Planetary Surface Vehicles: Modeling Approaches and Case Study," J. Br. Interplanet. Soc., **59**.
[22] Shinners, S., 1967, *Techniques of System Engineering*, McGraw-Hill, New York.
[23] Ferguson, S., and Lewis, K., 2006, "Effective Development of Reconfigurable Systems Using Linear State Feedback Control," AIAA J., **44**(4), pp. 868–878.
[24] Asl, F., Ulsoy, A., and Koren, Y., 2000, "Dynamic Modeling and Stability of the Reconfiguration of Manufacturing Systems," *Japan-USA Symposium on Flexible Automation*, July 23–26, 2000, Ann Arbor, MI.
[25] Siddiqi, A., 2006, "Reconfigurability in Space Systems: Architecting Framework and Case Studies," Ph.D. thesis, Massachusetts Institute of Technology, Cambridge.
[26] 2006, www.enidine.com.
[27] Codrington, A., 2002, Allmodpro.
[28] McGray, D., 2004, "Hot Wheels," *Wired Magazine*, October.
[29] http://www.ni.com/compactrio/about.htm.
[30] Walczyk, D., Lakshmikanthan, J., and Kirk, D., 1998, "Development of a Reconfigurable Tool for Forming Aircraft Body Panels," J. Manuf. Syst., **17**(4), pp. 287–296.
[31] Yim, M., Roufas, K., Duff, D., Zhang, Y., Eldershaw, C., and Homans, S., 2003, "Modular Reconfigurable Robots in Space Applications," Auton. Rob., **14**, pp. 225–237.
[32] Clark, P., and Rilee, M., 2004, "Lara: Near Term Reconfiguration Concepts and Components for Lunar Exploration and Exploitation," *55th International Astronautical Congress*, October 4–8, 2004, Vancouver, Canada, Paper No. IAC-04-IAA.3.8.1.08.
[33] Schenker, P., 2000, "Reconfigurable Robots for All Terrain Exploration," Proc. SPIE, **4196**, pp. 454–468.
[34] Quinn, J., Repairman in Space.
[35] Rodgers, L., 2005, "Concepts and Technology Development for the Autonomous Assembly and Reconfiguration of Modular Space Systems," M.S. thesis, Massachusetts Institute of Technology, Cambridge.
[36] Chau, S., Sengupta, A., Tran, T., and Bakhshi, A., 2002, "Ultra Long-Life Spacecraft for Long Duration Space Exploration Missions," *53rd International Astronautical Congress*, October 10–19, 2002, Houston, TX, Paper No. IAC-02-U.2.05.
[37] Nishinaga, N., Takeuchi, M., and Suzuki, R., 2004, "Reconfigurable Communication Equipment on SmartSat-I," IEIC Technical Reoprt, Vol. 104, No. 212, pp. 133–134.
[38] Simons, R., Chun, D., and Katehi, L., 2001, "Microelectromechanical Systems (MEMS) Actuators for Antenna Reconfigurability," National Aeronautics and Space Administration, Glenn Research Center, Tech. Rep. No. NASA/CR-2001-210612.
[39] 2006, "Nextgen to Build Morphing UAV," Flight International, June 6, 2006,

URL: http://www.flightglobal.com/articles/2006/06/06/207057/nextgen-to-build-morphing-uav.html
[40] http://www.aerospaceweb.org/aircraft/fighter/f14/.
[41] Wells, V., Rutherford, J., and Corgiat, A., 1999, "Mission and Concept Evaluation for a Multirole, Mission-Adaptable Air Vehicle," Aircraft Design, **2**, pp. 65–80.
[42] http://www.nrao.edu/intro.nm_astro.html.
[43] Magee, C., and de Weck, O. L., 2004, "Complex System Classification," *14th Annual International Symposium of the International Council on Systems Engineering (INCOSE)*, June 21–24, 2004, Toulouse, France.
[44] Hauck, S., 1998, "The Future of Reconfigurable Systems," *5th Canadian Conference on Field Programmable Devices*, June 1998, Montreal, Canada.
[45] Fricke, E., and Schulz, A., 2003, "Design for Changeability (DFC): Principles to Enable Changes in Systems Throughout Their Entire Lifecycle," J. Syst. Eng., **8**(4), pp. 342–359.
[46] Siddiqi, A., and de Weck, O., 2006, "Self-Similar Modular Architectures for Reconfigurable Space Systems," *57th International Astronautical Congress*, October 2–6, 2006, Valencia, Spain, Paper No. IAC-06-D1-1.4.03.
[47] Johnson, G. W., ed., 1998, *LabVIEW Power Programming*, McGraw-Hill, New York.
[48] Gu, P., Hashemian, M., and Nee, A., 2004, "Adaptable Design," CIRP Ann., **53**(2).
[49] Nishinaga, N., Ogawa, Y., Takayama, Y., Takahashi, T., Kubooka, T., and Umehara, H., 2003, "Softsat: Reconfigurable Communication Satellite System," 21st International Communications Satellite Systems Conference and Exhibit, April 15–19, 2003, Yokohama, Japan, Paper No. AIAA-2003-2420.
[50] 2006, http://www.fueleconomy.gov/feg.fcv_benefits.shtml.
[51] Sobieszczanski-Sobieskr, J., Altus, T., Phillips, M., and Sandusky, R., 2003, "Bilevel Integrated System Synthesis for Concurrent and Distributed Processing," AIAA J., **41**(10), pp. 1996–2003.