

Exploring the effectiveness of parallel systems in distributed design processes subjected to stochastic disruptions

SOUROBH GHOSH,¹ ERICH DEVENDORF,² AND KEMPER LEWIS³

¹Department of Mechanical Engineering, Massachusetts Institute of Technology, Cambridge, Massachusetts, USA

²Information Directorate, Air Force Research Laboratory, Rome, New York, USA

³Department of Mechanical and Aerospace Engineering, University at Buffalo, SUNY, Buffalo, New York, USA

(RECEIVED February 18, 2014; ACCEPTED May 5, 2014)

Abstract

During the design of complex systems, a design process may be subjected to stochastic disruptions, interruptions, and changes, which can be described broadly as “design impulses.” These impulses can have a significant impact on the transient response and converged equilibrium for the design system. We distinguish this research by focusing on the interactions between local and architectural impulses in the form of designer mistakes and dissolution, division, and combination impulses, respectively, for a distributed design case study. We provide statistical support for the “parallel character hypothesis,” which asserts that parallel arrangements generally best mitigate dissolution and division impulses. We find that local impulses tend to slow convergence, but systems also subjected to dissolution or division impulses still favor parallel arrangements. We statistically uphold the conclusion that the strategy to mitigate combination impulses is unaffected by the presence of local impulses.

Keywords: Collaborative Engineering; Complex Systems; Design Architecture; Distributed Design; Impulses; Process Modeling

1. INTRODUCTION

In the design of complex systems, it is often necessary to decompose problems into their constituent parts. Decomposition of complex systems simplifies their representation and breaks them into manageable pieces that can be individually analyzed (Bloebaum et al., 1992; Browning, 2001; Wiecek et al., 2005). To acquire an understanding of the aggregate system, design managers generally follow a three-step process: deconstruct the system into its component parts, analyze the parts to understand them individually, and reassemble the individually understood parts to obtain system-level knowledge.

Because decomposition forms the basis for analyzing engineering problems, it is natural for it to play an important role in the structure of design processes. Design problems that have been decomposed into a set of individually solved subproblems separated by spatial, temporal, or organizational boundaries are called decentralized design problems. In this paper, we examine a specific type of decentralized problem for complex system design called distributed design. In distributed design each subsystem minimizes its own individual objective

function subject to a set of constraints by modifying the value of those design variables under their unilateral control.

Distributed design problems have been widely studied by the design community (Yang & Jin, 2008; Wernz & Deshmukh, 2010). An increasing number of problems fall under the category of distributed design systems. For example, there is a growing tendency for large corporations to shift a portion of their design work to suppliers (Ward et al., 1995; Engardio & Einhorn, 2005). In most studies of distributed design, the subsystems exhibit deterministic behavior. However, at times a design process will experience unexpected disruptions, interruptions, and changes, or “design impulses.” These impulses can have a significant impact on the convergence time of the overall system (Gurnani & Lewis, 2008).

In this paper we examine distributed design in the context of two types of design impulses: local and architectural. Although these impulses have been individually studied (Devendorf & Lewis, 2010; Ghosh et al., 2012), the interactions between distinct categories of impulses occurring in a design system have not been rigorously investigated. We provide statistical support for the architectural impulse mitigation strategy suggested in Ghosh et al. (2012). We propose and empirically demonstrate the “parallel character hypothesis,” which states that architectures that feature increasingly parallel

Reprint requests to: Kemper Lewis, Department of Mechanical and Aerospace Engineering, 207 Bell Hall, University at Buffalo, SUNY, Buffalo, NY 14260, USA. E-mail: kelewis@buffalo.edu

arrangements generally yield the fastest convergence times after the application of dissolution or division impulses. We then examine the interaction between local and architectural impulses, because the strategies recommended for mitigating them directly conflict with one another.

Using a 5 designer, 16 design variable distributed design system as a benchmark case study, we investigate the relationship between the strategies recommended to mitigate local and architectural impulses. We study the role of impulses in the transient response of the case study using a Monte Carlo simulation to evaluate each configuration. We examine the convergence time of transient system responses in this work and consider the changes in convergence times that occur due to architectural changes and impulses. We do not focus on the convergence properties of the individual subsystem optimization problems. The results of the simulation provide the fitness function for identification of the architecture with the fastest mean convergence time and with the least variation across convergence time using a genetic algorithm.

To establish the underlying basis for the Monte Carlo simulation, we discuss the definition and simulation method for each impulse type in Section 3. In Section 4 we discuss the experimental design and case study parameters, and analyze the results. We provide background into distributed design in Section 2.

2. DISTRIBUTED DESIGN SYSTEMS

The properties used to evaluate distributed design systems include stability and transient response. We discuss these properties in Section 2.2 and 2.3 in conjunction with the concept of solution–process architecture. The game theoretic concept of noncooperation underlies both properties, and we discuss these assumptions in Section 2.1.

2.1. Problem assumptions

Distributed design principles provide a model to analyze the behavior of decomposed design problems with noncooperating or noncoordinated subsystems. In decomposition approaches, either the overall design system is naturally decomposed or a design manager delegates design responsibilities to different subsystems.

A distributed design process is iterative, and it possesses a nonhierarchical design structure with design subsystems that selfishly minimize their own objective function. Design subsystems have responsibility to determine the value of their assigned design variables. The major advantages of distributed design, as compared to other multidisciplinary optimization approaches, are that it is straightforward to implement for a design system and requires limited oversight in its execution. Such approaches include game theoretic formulations (Vincent, 1983) and other multidisciplinary design optimization approaches of independent subspaces (Martins & Lambe, 2013).

The disadvantages of the distributed design approach include the potential for the design process to diverge and

that there is no guarantee the converged solution is optimal. In spite of this, distributed design models provide insight into many design problems. The noncooperative subsystem properties that characterize distributed design include the following: designers have knowledge of only their own objective function; designers act unilaterally to minimize their own objective function; designers have complete control over specific local design variables; and designers communicate by sharing the current value of their local design variables.

Distributed design problems exist as stand-alone problems and as subproblems in a larger design task. As long as these subproblems fulfill the four assumptions of noncooperation, distributed design techniques can effectively model the system behavior. For example, when the number of iterative loops in a design structure matrix cannot be reduced to zero, the subproblem can often be modeled as a distributed design problem (Denker et al., 2001).

Distributed design problems have two characteristics of primary concern for system analyzers. The first is the equilibrium stability, which determines if the system equilibrium is convergent, divergent, or a saddle point. The second is the transient response, which determines the time required to reach the equilibrium solution for convergent systems. We discuss these characteristics in the next sections.

2.2. Equilibrium stability

Vincent (1983) established the foundation for stability in distributed design using a game theoretic approach. Chanron and Lewis (2004, 2006) extended Vincent's game theoretic model by representing the subsystem objective functions using a state space to apply control theoretic principles to determine system stability. Another approach analyzed equilibrium and convergence rates of coupled systems using only subsystem input/output information found in a design structure matrix representation (Smith & Eppinger, 1997). Chanron et al. (2005) demonstrated that some design systems have multiple equilibria, each of which may be stable, unstable, or a saddle point. The approach in Chanron et al. (2005) included nonquadratic objective functions using nonlinear system theory and Lyapunov functions to identify convergent regions of the design space. While this paper considers systems with linear rational reaction sets, Chanron's characterization of nonlinear systems demonstrates the broad applicability of these techniques and the ability to expand our research to more general nonlinear cases.

Initial research in distributed design assumed that the structure of the design problem, called the solution process architecture, played no role in determining system stability (Smith & Eppinger, 1997; Chanron & Lewis, 2004). However, Devendorf and Lewis (2011) demonstrated that process architecture changes can influence the stability, and they presented an approach to determine the stability for a set of parallel/sequential architectures. The influence of solution process architecture on transient response was first investigated by Devendorf and Lewis (2013), and they showed that the rate

and characteristic shape (e.g., exponential decay or decreasing amplitude sinusoidal) of convergence could be changed with architecture changes. Before examining the broader role of solution process architecture in determining the properties of distributed design systems, we first discuss transient response and then process architecture in Section 2.3.

2.3. Transient response

The transient response of a system describes the convergence rate and the shape of the convergence curve for the entire system. We refer to convergence rate in this work as the settling time as measured by the number of iterations. An iteration represents a single step through the entire system optimization process with each subsystem solving its individual optimization problem once. We do not consider the algorithmic convergence related to the solution of each subsystem's individual optimization problem. In this paper, we examine the settling time aspect of transient response. Our definition of settling time, consistent with control theory, is the number of iterations required for a system to converge to within 2% of its final value as measured from its initial value (Ogata, 2005). We use settling time as a metric for our experiments to provide a repeatable definition for convergence and to remain consistent with previous work examining transient response. Previous research (Devendorf & Lewis, 2013) demonstrated the dependence of settling time on process architecture. From a heuristic perspective, social network theory has shown that the settling time is dependent on the relationship between the process architecture and subsystem coupling (Devendorf et al., 2010).

Taken collectively, previous work examining equilibrium stability and transient response has demonstrated that process architecture is fundamental to the analysis of distributed design systems. In the context of distributed design, the term *architecture* refers to the ordering of the solution process. It does not refer to product architecture, which is an independent and significant

area of design research (Ulrich & Eppinger, 2012). Instead, it refers to the ordering in which the design subsystems are solved. This ordering can include both parallel/sequential and simultaneous elements. We show an example of a purely parallel or sequential and a purely simultaneous architecture in Figure 1 along with a hybrid architecture that incorporates aspects of both.

A change to the process architecture modifies design variable sharing in the system. When subsystems iterate in parallel with one another, they cannot share their design variable determination until the next iteration. For example, in the case of the hybrid architecture in Figure 1, Subsystem 3 will perform its optimization based on the new value of y before Subsystem 1 has the opportunity to incorporate y into the determination of its own design variable values.

From a structural perspective, specifying the process architecture is the only way to modify the behavior of a distributed design system without changing its constituent subsystems or violating the underlying noncooperative assumptions. Understanding the underlying deterministic behavior of distributed design systems remains important, but it neglects those cases when the system experiences stochastic inputs, interruptions, and changes. We discuss and define these design impulses in Section 3.

3. IMPULSES IN THE DESIGN PROCESS

Understanding the manner in which stochastic elements or disruptions propagate through decentralized decision networks is of critical importance. In this paper, we remain consistent with the definitions presented in Ghosh et al. (2012) and refer to the stochastic elements present in distributed design process as *design impulses* and define a design impulse as

unintended inputs occurring during a design process that cause a change to the aggregate process architecture or in the designers' variable values, variable ownership, objective functions, or constraints.

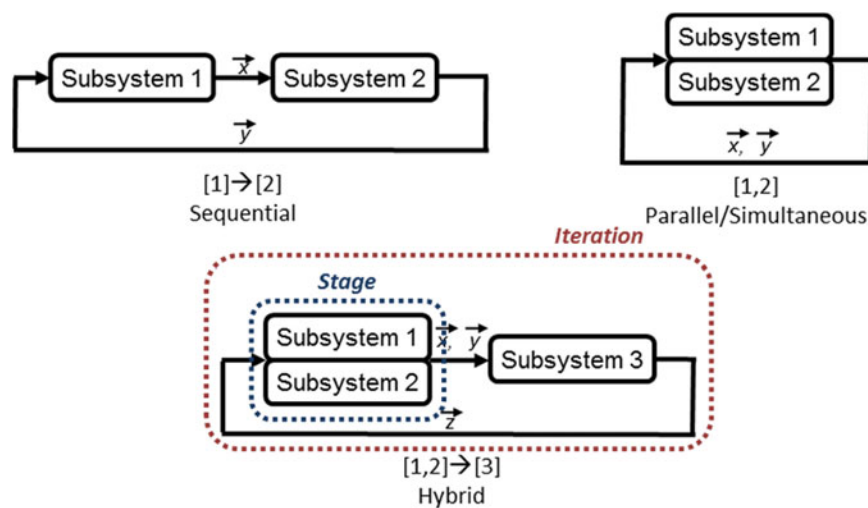


Fig. 1. Solution process architectures.

We deliberately use the term *impulse* because it is consistent with the terminology used to describe stochastic elements in control theoretic representations, which have been extensively applied to model decentralized decision networks. Because our focus is on the process architecture of distributed design systems, an impulse could potentially alter the equilibrium and transient response of the system. These changes may not always result in inferior system performance or a decrease in convergence time. However, they introduce uncertainty into iterative decision processes. Our goal is to identify mechanisms to remain resilient to impulses, potentially at the cost of beneficial impulses.

The definition of a design impulse is intentionally broad due to the variety of different impulses. We break design impulses into three categories, based on how they interact with the design system: *local impulses*, *external impulses*, and *architectural impulses*. An event triggers impulses that influence a design system, and a single event can result in multiple impulses. Decomposing events into their constituent impulses and identifying a verifiably exhaustive set of impulses is an area of future work. The impulses we identify in this work describe a large number of commonly observed events. This paper examines the interaction between local and architectural impulses, and we define these impulses in Sections 3.1 and 3.3, respectively, and discuss the techniques used to model each impulse type. In Section 3.2, we define external impulses by way of presenting our definition for architectural impulses in Section 3.3, although formally modeling external impulses is a topic of future work.

3.1. Local impulses

The first type of impulse, a local impulse, describes impulses that originate within a decision network and are defined as

impulses caused by an event whose origin can be attributed to design resources controlled by, and whose initial impact occurs in, a single design subsystem.

Local impulses propagate across decision networks as subsystems share and exchange decision information across the network. The propagation path depends on the coupling that exists between subsystems in decentralized decision networks.

In this paper, we consider the local impulses that occur due to bounded rational designers. In bounded rationality, decision makers have deterministic preferences but make stochastic choices (Thurston, 2001). Gurnani and Lewis (2008) used bounded rationality to model mistakes in distributed design and to analyze the equilibrium properties of bounded rational decision makers. The model we use in this paper is consistent with Gurnani and Lewis's work and is identical to the model used by Devendorf and Lewis (2010) to investigate the role of local impulses in transient response. Figure 2 provides a general formulation for a bounded rational decision maker.

In Figure 2, X^* is the expressed preference of the decision maker for decision variable X , $V(p,q)$ is a value function that

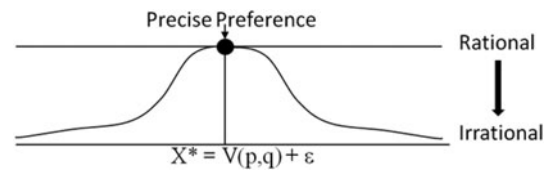


Fig. 2. Bounded rational decision maker.

represents the precise preference of the decision maker, while ϵ represents the stochastic portion of the decision. The value function $V(p,q)$ is dependent on p , the local variables controlled by the designer, and q , the nonlocal variables in the system not controlled by the designer. Together, p and q yield the designer's rational reaction set, which embodies the value function V and is a direct representation of the fully rational and precise preference of the decision maker. We represent ϵ as a normal distribution centered at the origin, and the standard deviation of ϵ captures the level of irrationality the decision maker exhibits. As the distance from the precise preference in Figure 2 increases, the decision becomes more irrational.

3.2. External impulses

In contrast to local impulses, we define external impulses accordingly:

an impulse caused by an event outside the decision network whose origin cannot be traced to any design subsystem in the system.

In the same way coupling between subsystems can cause changes from a local impulse to propagate across a decision network, changes from an external impulse can have a similar impact. For example, a common example of an external impulse for an automotive manufacturer would be a change in global petroleum prices, affecting demand for certain types of vehicles. As noted earlier, formally modeling these impulses is not the focus of this paper.

3.3. Architectural impulses

Local and external impulses both describe changes to individual subsystems in a distributed design system. Architectural impulses address broader scenarios where the structure of the decision network itself changes. We define an architectural impulse as

an impulse caused by events that originate either locally or externally and cause a restructuring of the process architecture.

In contrast to local or external impulses, which we define as affecting the specific parameters internal to the subsystem design problem, architectural impulses change the coupling links between subsystems and are not restricted in their origin.

We distinguish architectural impulses from local and external impulses based on their influence on the actual structure of the process.

Examples of potential architectural impulses might be a breakdown or enhancement in communication between subsystems, the removal or addition of a subsystem from or to a decision network, or the development or degradation of ties between subsystems. We further subdivide architectural impulses into three classifications: dissolution, combination, and division, and discuss them in Sections 3.3.1, 3.3.2, and 3.3.3.

3.3.1. Dissolution impulse

Dissolution impulses eliminate one or more design subsystems from the aggregate design system. Strategies to handle the design variables associated with the dissolved subsystem include distributing them to the remaining design teams, assigning them a new subsystem, or freezing them at their previous value. Freezing design variables is an attractive option to respond to a dissolution impulse because reassigning them can destabilize the system due to design variable coupling effects. Investigating the difference between the strategies used to mitigate this impulse is a topic of future work. An event that may result in a dissolution impulse is the bankruptcy of the organization controlling the design of one of the subsystems. Figure 3 provides a graphical representation for a five-subsystem distributed design system undergoing a dissolution impulse.

The representation in Figure 3 shows five individual subsystems, D1 to D5. The arrows connecting the subsystems represent their dependence on design variable information from another subsystem. For example, in Figure 3 the arrow from D4 to D5 indicates that a design variable from D4 appears in the objective function of D5. The dissolution impulse shown by Figure 3 illustrates the removal of D1 from the design process. This removal leaves two subsystems, D2 and D3, previously linked to D1 without a source to specify those design variables. D4 does not directly require input from D1, but it now has less influence on the decision process because it can no longer indirectly control the other subsystems through inputs passed through D1. To continue the design process, a design manager or the subsystems must agree on

a strategy to deal with the loss of D1, including reassigning or freezing design variable values.

3.3.2. Division impulse

While dissolution impulses remove subsystems from a design system, a division impulse increases the number of subsystems. Division impulses occur when an event divides a design subsystem into two or more separate and distinct subsystems. Each subsystem has its own design variables, objective function, and constraints, which must encompass at least the original shared design variables and may create new shared variables. An event that may cause a division impulse is the sale of a division of an organization working on a project still in process by the parent company. We graphically represent division impulses in Figure 4.

We represent a division impulse in Figure 4 as D4 splitting into two subsystems. One challenge associated with division impulses is partitioning design responsibilities to the emerging subsystems. Even when dividing design responsibilities is easy, dividing the design variables themselves presents a challenge. Furthermore, new design variable coupling from the now independent subsystems may also emerge due to the division.

3.3.3. Combination impulse

In contrast to division impulses, combination impulses occur when two or more design teams merge into a single subsystem. Combination impulses reduce the number of subsystems in a design process like a dissolution impulse, but the affected subsystem retains a presence in the design system. The combined subsystem retains control of design variables associated with each subsystem and may have a new aggregate objective function. An example of an event causing a combination impulse is a buyout or a merger. Figure 5 shows a combination impulse for a five-subsystem design problem.

We represent a combination impulse in Figure 5 by combining D3 with D5. This combination requires the merging of objective functions for D3 and D5. The resultant subsystem can influence the stability and transient response of the overall system. We examine all three types of architectural impulses in conjunction with local impulses in Section 4 using a five-subsystem case study.

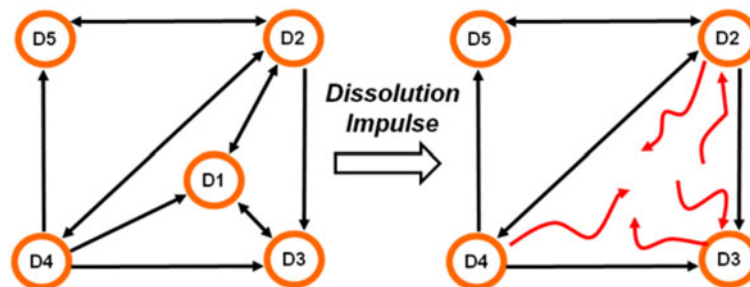


Fig. 3. Dissolution impulse.

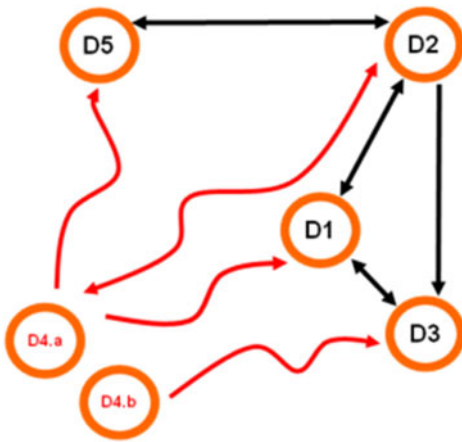


Fig. 4. Division impulse.

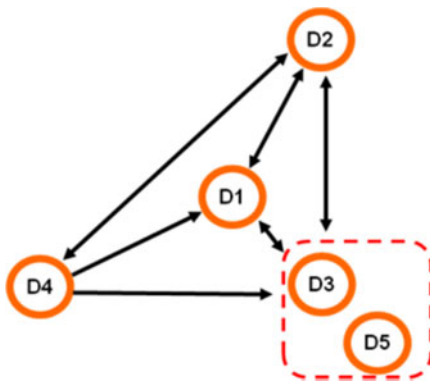


Fig. 5. Combination impulse.

4. AGGREGATE IMPULSE EXPERIMENT

In this section we examine process architecture in the context of a system subjected to local and architectural impulses. We differentiate this work from Ghosh et al. (2013) by statistically validating our hypothesis on which architectures perform best in the presence of architectural impulses and by examining the statistical significance for the aggregate impulse results. We describe the experimental design in Section 4.2 and discuss the results in Section 4.3. First we introduce the five-subsystem problem used as our case study in Section 4.1.

4.1. Case study problem structure

Our experiment examines the same case study used in the initial study of large distributed design systems as well as previous individual studies of local impulses and architectural impulses. We use this case study because it is large enough to have a variety of potential architectures and the behavior of the system is largely well understood because previous research has used this case study to demonstrate more general insights into distributed design systems (Chanron et al., 2005; Chanron & Lewis, 2006). The five-subsystem example shown conceptually in Section 3.3 to illustrate the types of

architectural impulses has couplings consistent with this case study. We explicitly give the objective function associated with each subsystem in Table 1 and provide the design variables controlled by each subsystem. The objective functions shown in Table 1 are all quadratic and to be minimized, and each subsystem controls at least two design variables. These design variables couple every subsystem to at least one other subsystem as shown in the functional dependence table in Table 2.

The breadth of different process architectures for a five designer problem means a wide range of potential settling times exist for the system. Figure 6 demonstrates the wide range of potential settling times across the design system.

In Figure 6 the architectures are grouped into bins based on the number of iterations required to converge. The height of the bars indicates the number of process architectures with a specified range of convergence times shown on the x axis. The mean convergence time for the simulated architectures is 25 iterations. The two fastest converging architectures are shown in Figure 7a. The slowest converging architecture is shown in Figure 7b and took 39 iterations to reach an equilibrium.

One interesting result of the architectures shown in Figure 7 is the relative strength in the link between the different subsystems, roughly measured by the number of connections to other subsystems in Table 2. Conventional wisdom has held that subsystems with strong links to one another should be arranged sequentially while subsystems with weak links are arranged in parallel (Rogers, 1996). For example, the other four subsystems are all strongly linked to SS2. However, in the fastest converging architectures, SS2 is not arranged sequentially as previous work has suggested is appropriate.

For the study of local impulses in (Devendorf et al., 2010), it was found that one of the fastest converging architectures,

Table 1. Five subsystem case study

<p>Subsystem 1: x_1, x_2</p> $f_1(x) = 9.41x_1^2 + 1.80x_2^2 + 6.06x_1 + 1.62x_2 + 8.16x_1x_3 + 2.82x_2x_4 + 6.20x_2x_5 + 9.82x_1x_6 + 4.26x_2x_7 + 2.37x_1x_8 + 1.25x_1x_9 + 2.23x_2x_{10} + 3.47x_1x_{11} + 0.23x_2x_{12}$
<p>Subsystem 2: x_3, x_4, x_5</p> $f_2(x) = 6.55x_3^2 + 7.57x_4^2 + 5.68x_5^2 + 4.29x_3 + 8.84x_4 + 5.25x_1x_5 + 2.13x_5x_8 + 5.34x_5x_9 + 3.58x_5x_{15} + 2.34x_5x_{14} + 4.57x_4x_{15} + 4.12x_5x_{16}$
<p>Subsystem 3: x_6, x_7</p> $f_3(x) = 7.81x_6^2 + 5.49x_7^2 + 5.43x_6 + 7.51x_7 + 7.87x_1x_6 + 4.57x_2x_7 + 4.52x_3x_6 + 1.23x_4x_6 + 2.12x_3x_7 + 3.26x_6x_{10}$
<p>Subsystem 4: $x_8, x_9, x_{10}, x_{11}, x_{12}$</p> $f_4(x) = 9.88x_8^2 + 9.86x_9^2 + 6.49x_{10}^2 + 9.48x_{11}^2 + 6.4x_{12}^2 + 5.43x_8 + 1.23x_9 + 7.84x_{10} + 0.32x_{11} + 5.43x_{12} + 1.30x_3x_9 + 1.94x_3x_{11} + 5.75x_3x_{12} + 4.32x_4x_8 + 0.12x_4x_{10} + 4.56x_4x_{11} + 3.26x_4x_{13} + 4.89x_5x_8 + 2.3x_5x_9 + 1.51x_5x_{10} + 3.20x_5x_{11}$
<p>Subsystem 5: $x_{13}, x_{14}, x_{15}, x_{16}$</p> $f_5(x) = 9.52x_{13}^2 + 7.75x_{14}^2 + 9.93x_{15}^2 + 8.12x_{16}^2 + 2.79x_{13} + 0.5x_{14} + 9.37x_{15} + 6.53x_{16} + 5.43x_3x_{13} + 6.41x_3x_{14} + 0.12x_3x_{16} + 6.27x_4x_{13} + 5.43x_4x_{15} + 1.232x_5x_{14} + 6.77x_5x_{16} + 1.38x_8x_{13} + 4.31x_{10}x_{14} + 2.64x_{10}x_{15} + 3.41x_{10}x_{16}$

Table 2. Functional dependence table

	f_1		f_2			f_3		f_4				f_5				
	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	x_{11}	x_{12}	x_{13}	x_{14}	x_{15}	x_{16}
f_1	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█
f_2	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█
f_3	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█
f_4	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█
f_5	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█

[1,2,4] → [3,5], was also the most robust to impulses. This was because the local impulse propagated most quickly through that architecture, and that in turn enabled each subsystem to adjust to its presence. This is important, because it means that the same propagation property that enabled the system to remain robust to impulses also enables it to converge quickly to its equilibrium.

4.2. Statistical testing of the parallel character hypothesis for dissolution and division impulses

Strategies for mitigating architectural impulses as investigated in Ghosh et al. (2012) yielded a preference for systems that featured the fewest number of serial connections when applying dissolution or division impulses, which in this work we term as the *parallel character hypothesis*. That is, architectures that featured increasingly parallel arrangements generally yielded the fastest convergence times after the application of dissolution or division impulses. Combination impulses did not uphold this hypothesis, because of the inher-

ent instability that these impulses cause, which is discussed in further detail by Ghosh et al. (2012).

Before investigating whether the parallel character hypothesis also applies to our evaluation of aggregated design impulses, we evaluate its validity via statistical testing. For these tests, we generate architectures and measure their transient response using the same procedure as described in Ghosh et al. (2012), except that each architecture is now run for a total of 500 trials as opposed to 10. This sample size allows us to assume that the mean iteration count produced for each architecture is representative of that architecture’s performance.

Using the mean convergence time data, we categorize each architecture into subpopulations according to the number of sequential connections it exhibits, bounded by zero for a fully parallel architecture and four for a fully sequential architecture. Because we draw these results from the final population of the genetic algorithm, they represent the best architectures from within their subpopulation. We then conduct two-tailed *t* tests between each subpopulation to test whether the number of sequential connections is a significant predictor

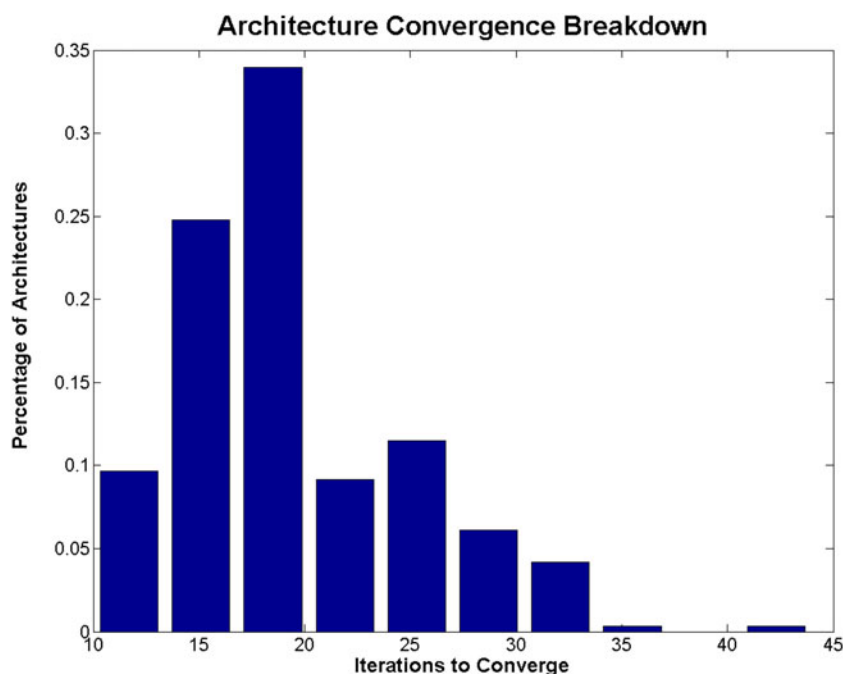


Fig. 6. Five-subsystems deterministic convergence time.

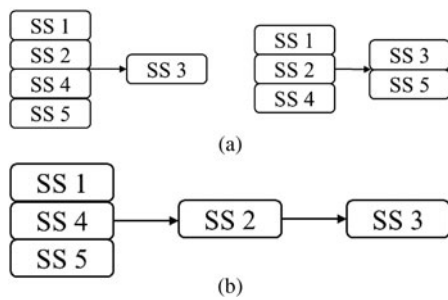


Fig. 7. (a) Fastest converging architectures for 14 iterations and (b) slowest converging architecture for 39 iterations.

of convergence time (i.e., whether the subpopulations are from the same distribution or are distinct). We summarize the paired *t* tests that yielded statistical significance in Table 3.

Overall, we see that the majority of subpopulation pairs tested uphold the parallel character hypothesis, asserting that reducing the number of sequential connections does reduce postimpulse mean convergence time for architectures that undergo dissolution and division impulses. We observe that 4-connection architectures are not present among the significance tests. It is very difficult to generate many purely sequential, 4-connection architectures using a genetic algorithm consistent with the experimental procedure of Ghosh et al. (2012) and related previous studies (Devendorf et al., 2010), because these architectures perform poorly enough to be eliminated from consideration from the final population of architectures produced by the genetic algorithm. The lack of results for 4-connection architectures is specific to this case study, because four connections represent the extreme case of a purely sequential architecture for 5-subsystem systems. Generalizing these results to systems of *n* subsystems with *n* > 5 will result in similar extreme cases when the architecture exhibits *n* – 1 connections. Thus, the lack of statistical power for tests concerning 4-connection architectures due to their underrepresentation in the final population agrees with our assertion of their poor performance relative to architectures that feature fewer sequential connections.

The remaining subpopulation paired test not represented is 0 versus 1 for division experiments. In this case, paired *t* tests demonstrated the opposite relationship, where 1 sequential connection architectures yielded a statistically significantly lower time than a purely parallel 0-connection architecture.

Table 3. Parallel character hypothesis testing for dissolution and division impulses

Impulse Type	Significant Connection Pairs	
Dissolution	0 vs. 1**	2 vs. 3**
	1 vs. 2**	
Division	1 vs. 2**	2 vs. 3*

p* < 0.01. *p* < 0.001.

Given the way division impulses are modeled in Ghosh et al. (2012), where the child subsystems are inserted in parallel at the location of the original parent subsystem, and with child subsystems sharing identical objective functions and design variables as the parent, a 0-sequential architecture under division can be thought of as a purely parallel architecture subjected to no impulses at all. Thus, because the fastest performing architectures in the case study used in Ghosh et al. (2012) yielded one sequential connection (see Fig. 7), this result agrees with our current intuition of division impulse behavior. Conducting sensitivity analysis on 0-connection architectures and their child architectures to determine whether our intuitive understanding of this behavior is correct is a topic of future study.

Having validated the parallel character hypothesis for a majority of dissolution and division impulse cases, we now turn our attention to experiments studying the simultaneous application of local and architectural impulses.

4.3. Experimental design

In this section, we describe the simulation parameters, the implementation details of local and architectural impulses, and specifics regarding the experimental design used.

4.3.1. Simulation parameters

Each architecture is evaluated as part of a genetic algorithm that determines the fastest converging architecture. We use the results of the genetic algorithm to identify the properties of the fastest converging architectures and to differentiate between the best architectures with different numbers of stages. The parameters of this genetic algorithm are shown in Table 4. We cap the maximum number of iterations at 250, which is a sufficiently large number to allow for effective observation of the system behavior. After a system reaches 250 iterations, we assume it to be divergent or an extremely slow architecture.

A critical aspect to evaluating the convergence time is the actual convergence criteria. We define convergence for these simulations to occur when all the design variables for a system have progressed to within 2% of their equilibrium value, as measured from the initial starting location. To maintain consistency across the experiments, we normalize the equilibrium point for each system to the origin and recalculate the convergence criteria when an architectural impulse occurs

Table 4. Genetic algorithm parameters

Parameter	Value
Mutation rate	5%
Elitism	Top 10% Bottom 5%
Min change in pop. mean	10%
Min change in stand. dev.	15%
Min exploration	15%

that modified the equilibrium. The stability of the studied architectures depends on the coupling between the subsystems and the independence of the design variables controlled by a single subsystem.

4.3.2. Local impulse implementation

Local impulses are implemented according to the bounded rational decision maker model discussed in Section 3.1. These impulses change the value of a given design variable based on a normal distribution centered on the true value of the design variable with standard deviation equal to a percentage of its value. Furthermore, there is a random chance that such a mistake could occur in the first place. Thus, the relevant parameters to control are *severity*, which indicates the magnitude of the standard deviation used in the mathematical model for mistakes, and *probability*, which represents the chance for an error to occur during a given iteration of the simulation.

4.3.3. Architectural impulse implementation

As noted in Section 3.3, there are three types of architectural impulses we consider in this work. To examine dissolution impulses, we remove a single subsystem from a given process architecture. For instance, a dissolution impulse applied to Subsystem 2 in the architecture $[1,2,4] \rightarrow [3,5]$ results in the new architecture $[1,4] \rightarrow [3,5]$. For this investigation we assume the design variables of the dissolved system are frozen in place when a dissolution impulse occurs. This assumption reflects the unwillingness of subsystems to change the parameters of a system they do not understand, or the inability of the subsystems to agree who should be given jurisdiction over those variables.

We implement the division of a subsystem by splitting a single subsystem into two distinct subsystems. We randomly assign the design variables to the new subsystems with the requirement that neither subsystem can control all the design variables; each subsystem maintains its original objective functions. We also preserve the relative ordering of the subsystems in the design process. For example, consider a division impulse applied to subsystem 2 in $[1,2,4,5] \rightarrow [3]$ that creates subsystems 6 and 7. The resultant architecture is $[1,6,7,4,5] \rightarrow [3]$. Similar to dissolution impulses, a division impulse does not have an impact on the stability of the system and may be applied to any arbitrary system. Division impulses do not influence stability because the design variables controlled by a subsystem are linearly independent.

Finally, combination impulses combine two distinct subsystems into a single subsystem. Because the stability of the system can be influenced by which subsystems are combined, we only combine subsystems that are decoupled from one another, in order to avoid the chance of automatic divergence. Once again, the necessity that a subsystem has linearly independent design variables precludes some combinations. We combine the objective functions for the new combined subsystem by adding them together, and the new subsystem has control over all the design variables of its constituent sub-

systems. The location in the process architecture for the combined subsystem is at the place where the original subsystems were ordered. For example, consider a combination impulse including Subsystems 3 and 5 that results in the creation of Subsystem 6. For the architecture $[1,2,4,5] \rightarrow [3]$, the postimpulse architecture is then $[1,2,4,6] \rightarrow [6]$.

A number of the assumptions we make in implementing architectural impulses in this investigation are made to preserve process stability. For instance, in Section 3.3.1, we note the advantage to freezing design variables as opposed to reassigning them for dissolution impulses, and for combination impulses we note the necessity to combine subsystems that are decoupled from one another. While these assumptions may limit the number of cases to potentially investigate, they are made to produce meaningful results to study (where generating heuristics for systems that fundamentally fail to converge is a futile exercise). Further discussion on the limitations to our modeling approach for architectural impulses can be found in Ghosh et al. (2012).

4.3.4. Experimental design

For each experiment described in the following section, 90 architectures are randomly generated in accordance with previous studies using this five-subsystem example to provide a wide range of architectures for an initial population. We then run the architectures through the genetic algorithm described in Section 4.1 for a total of 10 independent trials. During the simulations, the number of iterations is recorded, and the resulting mean and standard deviation across the 10 trials for each architecture are collected, representing our measures for system transient response in this work. Depending on the results from these experiments, we determine which experiments require further study with larger populations and statistical analysis.

In the next section, we investigate the amalgamated effects of architectural and local impulses on distributed design systems. The set of experiments investigated are summarized in Table 5. The experiment column displays the shorthand name of the experiment being run. For instance, division-(20,10) refers to an experiment with the simultaneous application of division impulses and local impulses with severity and probability parameters set to 20% and 10%, respectively, which were chosen to mirror previous studies.

4.4. Experimental results

In this investigation of amalgamated local and architectural design impulses, we use results from our prior investigation of architectural impulses in Ghosh et al. (2012) as a control and for comparative purposes to inform our conclusions in this work. The three baseline experiments that are used as a standard of comparison in this work are random dissolution, random division, and random combination. These experiments were conducted identically to what is described in Section 4.2, except without the simultaneous application of local impulses.

Table 5. *Experimental design*

Experiment Name	Architectural Impulse Applied	Local Impulse Severity	Local Impulse Probability
Dissolution-(10,10)	Dissolution	10%	10%
Dissolution-(20,10)	Dissolution	20%	10%
Division-(10,10)	Division	10%	10%
Division-(20,10)	Division	20%	10%
Combination-(10,10)	Combination	10%	10%
Combination-(20,10)	Combination	20%	10%

4.4.1. Dissolution experiments

The results from the random dissolution experiment from Ghosh et al. (2012) along with the results from the dissolution-(10,10) and dissolution-(20,10) experiments are displayed graphically in Figure 8a–c, respectively. These figures show the relationship between mean convergence time and number of sequential connections for each of the 90 architectures investigated for each experiment. The number of sequential connections is a measure for the parallel character of a system architecture, displaying a count of how many sequential steps an architecture has. For instance, while a purely parallel architecture has zero sequential connections, a purely serial one has four.

In Ghosh et al. (2012), we note that for systems that are subjected to dissolution impulses only, there is a strong preference for systems with a low number of sequential connections. This trend is evident in Figure 8a, where each of the 11 total generated architectures that have four parallel subsystems in sequence with the remaining subsystem outperforms each of the 22 total generated architectures with three sequential steps.

In Figures 8b and 8c, we see the results of the dissolution-local impulse experiments. Our first observation is that there appears to be a similar preference for parallel architectures in these experiments, albeit less resolute than in the random dissolution experiment. For instance, in dissolution-(10,10) and dissolution-(20,10), we find at least one three-sequential-step architecture among the top 10 performing architectures of both experiments: one in dissolution-(20,10) and three in dissolution-(10,10). Nonetheless, the exhibited parallel preference is evident when considering that the top performing architectures in each experiment show the least number of sequential connections among the randomly generated architectures for those experiments, where the best performing architectures for dissolution-(10,10) and dissolution-(20,10) are $[2,3,4,5] \rightarrow [1]$ and $[1,2,5] \rightarrow [3,4]$, respectively. Furthermore, the top 10% of architectures for dissolution-(20,10) all exhibit only one serial connection, the minimum encountered among the 90 randomly generated architectures.

An additional observation to make concerns the magnitude of iterations required for dissolution-(10,10) and dissolution-(20,10) architectures to converge, which are notably greater than the random dissolution experiment. This finding is con-

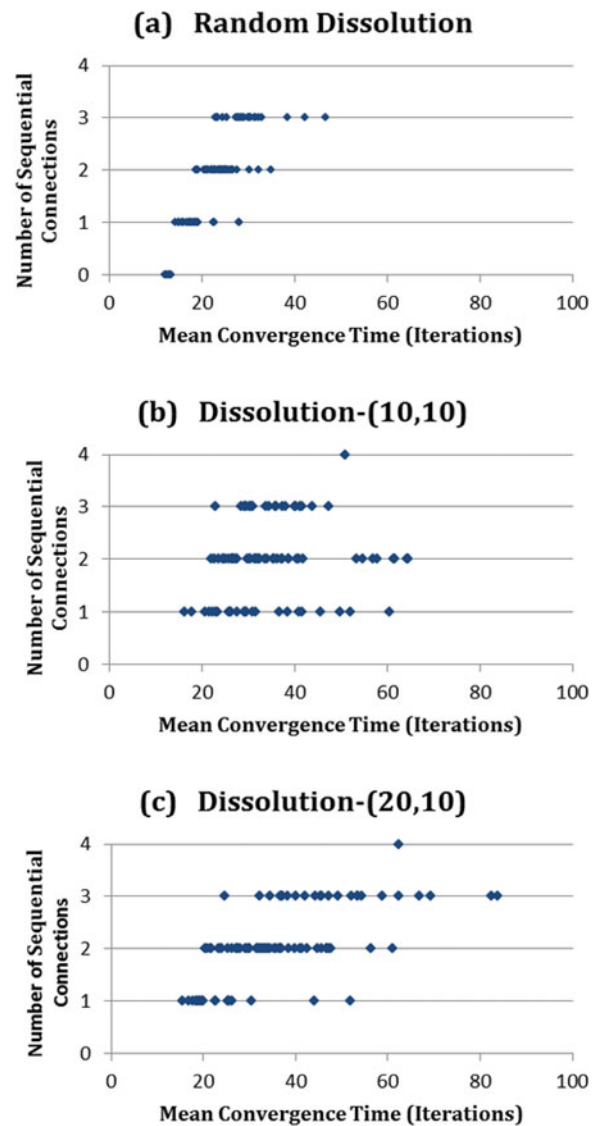


Fig. 8. Sequential connections versus convergence time for dissolution experiments.

sistent with results from Devendorf et al. (2010), where the addition of a local impulse had been shown to delay convergence time for systems.

Thus, treating the random dissolution experiment as a baseline, we find that the introduction of local impulses has produced two notable effects: to unilaterally delay the convergence of architectures and to somewhat dampen the strength of the parallel architecture preference exhibited by systems subjected to architectural impulses. These two effects can be considered somewhat complementary, because the stochastic nature of mistakes induced by local impulses not only serves to postpone system convergence but also can be somewhat disproportionate, affecting some architectures more by chance than others. This may explain the dampening of the parallel preferences for dissolution-local impulse systems, an effect whose underlying mechanics we discuss as a topic of future work.

Finally, this discussion highlights an intriguing observation: that local and architectural impulses have an additive effect on one another. In other words, it would seem that adding local impulses to dissolution experiments has not produced a novel aggregated behavior; rather, the results above would seem to suggest that the effects of local impulses act independently of the effects of architectural impulses.

4.4.2. Division experiments

The results from the random division experiment along with the results from the division-(10,10) and division-(20,10) experiments are displayed graphically in Figure 9a–c, respectively. Figure 9a displays a similar trend to dissolution, where systems subjected to division impulses prefer increasingly parallel arrangements to mitigate the effects of the impulse.

The division-local impulse results of Figure 9b and 9c exhibits similar relationships to Figure 9a as Figure 8b and 8c show to Figure 8a, though with a caveat: while there appears to be a similar delayed convergence and dampened strength of parallel architecture preferences, the effects seem to be less pronounced than when local impulses were introduced to dissolution impulses. This trend reveals itself over Figure 9a–c, where bands of architectures on each gridline for the number of sequential connections occupy roughly the same ranges of mean convergence time, whereas for the dissolution experiments, there was a visible offset between random dissolution and dissolution-(10,10) and dissolution-(20,10) in terms of mean convergence times.

Tracking differences in performance for a given architecture for the dissolution and division experiments reveals this trend. For dissolution, while the top five architecture $[1,5] \rightarrow [2,3,4]$ converged in an average of 15.00 iterations for random dissolution, the same architecture required an average of 18.38 iterations for the dissolution-(20,10) experiment.

Conversely, the top five random division architecture $[1] \rightarrow [2,3,4,5]$ required an average of 15.54 iterations to convergence, while the division-(20,10) experiment required an average of 15.67 iterations, delaying convergence only marginally (by about 0.8% of the nonlocal impulse value) when compared to the dissolution experiments (which exhibit a 22% increase in convergence time). Thus, we conclude that systems subjected to division impulses are better at handling the application of local impulses than are similar systems subjected to dissolution impulses.

4.4.3. Combination experiments

The results from the random combination experiment along with the results from the combination-(10,10) and combination-(20,10) experiments are displayed graphically in Figure 10a–c, respectively. As noted in Ghosh et al. (2012), there is no visible correlation between the parallel character of a system architecture and its mean convergence time with respect to random combination impulses, and we see a similar case for the combination-(10,10) and combination-(20,10) experiments in Figure 10b and

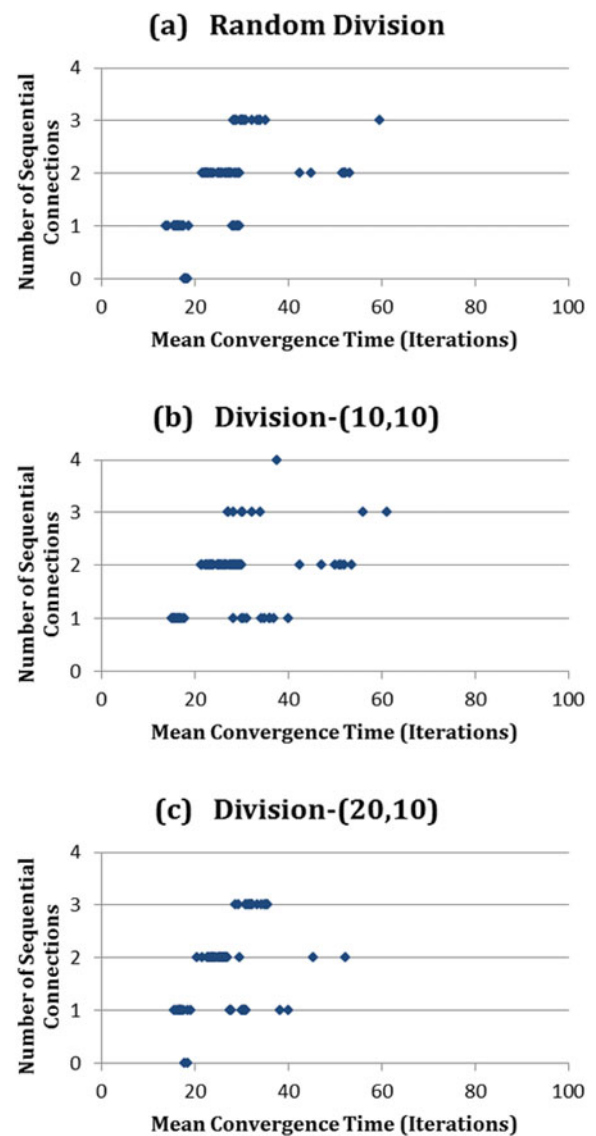


Fig. 9. Sequential connections versus convergence time for division experiments.

10c, respectively. This is in contrast to the results from dissolution and division experiments, suggesting that parallel structures are not universally better for all types of architectural impulses (with or without the addition of local impulses). This may be a symptom of the inherent stability associated with subsystem combinations (a previously noted caveat to these experiments is that subsystems to be combined must be sufficiently decoupled).

An additional observation regarding these experiments can be found in Table 6, which displays the worst performing architecture for each combination experiment considered in this work. One common feature among these systems is that they all feature the $[1,3]$ subsystem parallel pair, set in series with the remaining subsystems. The inherent instability of this subarchitecture had been well documented in Ghosh et al. (2012), and we find that the combi-

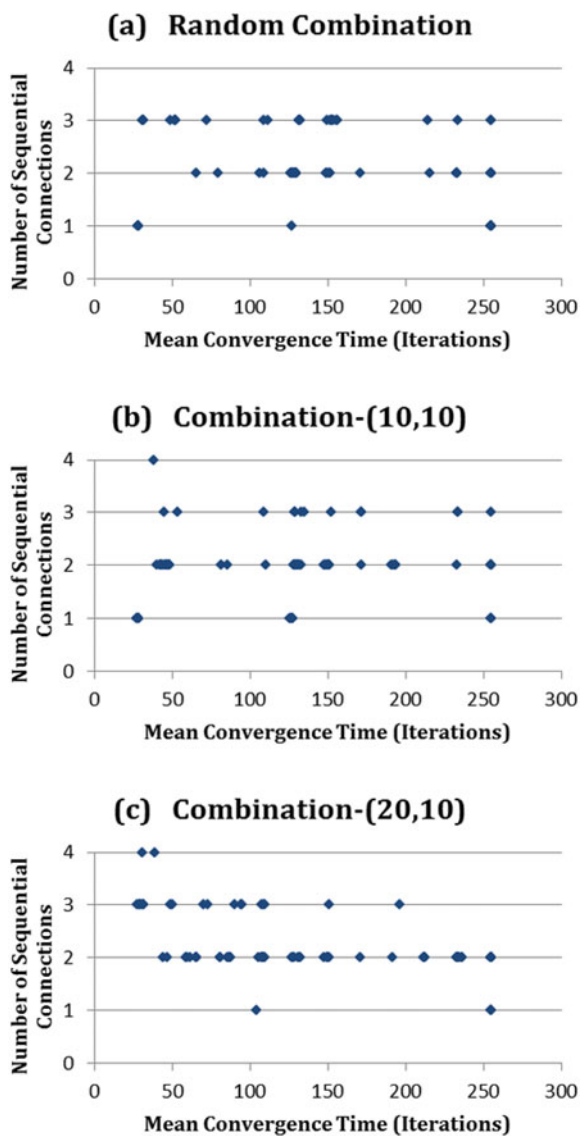


Fig. 10. Sequential connections versus convergence time for combination experiments.

nation-(10,10) and combination-(20,10) experiments support this prior finding. Thus, we find a similar trend to the dissolution and division experiments where the application of local impulses does not necessarily remove the character of behavior of systems subjected to architectural impulses alone.

4.5. Statistical testing of the parallel character hypothesis for aggregated impulses

In this section, we describe the hypothesis testing used to validate the correlations observed in Section 4.4. In particular, we examine the parallel character hypothesis in the dissolution-(10,10), dissolution-(20,10), division-(10,10), and division-(20,10) experiments. We do not conduct such hypothesis testing on the combination experiments, because the

Table 6. Worst performing architecture by experiment considered

Experiment	Worst Performing Design Architecture
Random combination	$\begin{bmatrix} 1 \\ 3 \end{bmatrix} \rightarrow [2] \rightarrow [4] \rightarrow [5]$
Combination-(10,10)	$\begin{bmatrix} 4 \\ 5 \end{bmatrix} \rightarrow \begin{bmatrix} 1 \\ 3 \end{bmatrix} \rightarrow [2]$
Combination-(20,10)	$\begin{bmatrix} 2 \\ 5 \end{bmatrix} \rightarrow \begin{bmatrix} 1 \\ 3 \end{bmatrix} \rightarrow [4]$

results of Section 4.3.3 do not present correlations that frame this investigation for statistical significance.

For these tests, we generate architectures and measure their transient response using the same procedure as in Section 4.3.4. Similar to the testing of the parallel character hypothesis conducted earlier for dissolution and division impulses alone in Section 4.2, each architecture is now run for a total of 500 trials instead of 10, enabling us to assume the mean iteration count for each architecture to be representative of that architecture's performance.

As in Section 4.2, we categorize each architecture into subpopulations according to the number of sequential connections it exhibits. As before, these results are drawn from the final population of the genetic algorithm, representing the best architectures from within their subpopulation. We then conduct two-tailed t tests between each subpopulation to test whether the number of sequential connections is a significant predictor of convergence time. We summarize the significant results of the t tests in Table 7.

From Table 7, we observe that a majority of sequential connection subpopulation pairings bear statistically significant results, thus supporting the parallel character hypothesis. Thus, we see an overall trend where an increase in the number of sequential connections for dissolution-local and division-local experiments (outlined in Table 7) is associated with statistically significant increases in mean convergence time. Similar to the analysis in Section 4.2, many of the pairings that were not statistically significant feature 4-connection, purely sequential architectures. As we note earlier, these are generally poor candidates for the final population generated for analysis by the genetic algorithm, inherently performing worse in terms of convergence time when compared to fewer sequential connection architectures. In addition, the 0-connection case for the division-(10,10) and division-(20,10) experiments remain somewhat limited in the insights they can offer based on how division impulses are currently modeled, as noted earlier in Section 4.2. Notably, however, the majority of cases support our earlier observations that the mitigation strategy for architectural impulses remains dominant, despite its direct contradiction with local impulse mitigation strategy.

Table 7. Parallel character hypothesis testing for simultaneous local and architectural impulses

Experiment	Significant Connection Pairs	
Dissolution-(10,10)	0 vs. 1**	2 vs. 3**
	1 vs. 2*	3 vs. 4**
Dissolution-(20,10)	0 vs. 1**	2 vs. 3**
	1 vs. 2**	
Division-(10,10)	0 vs. 1*	1 vs. 2**
Division-(20,10)	1 vs. 2**	2 vs. 3*

* $p < 0.01$. ** $p < 0.001$.

5. CONCLUSIONS AND FUTURE WORK

In this work, we identified design impulses as stochastic inputs during the optimal solution of decentralized design systems; these inputs have the potential to affect the transient response and equilibrium solutions of such systems. We provided statistical validation for the parallel character hypothesis, which asserts that architectures that are more parallel in character tend to handle the effects of dissolution and division impulses on convergence time best. We then studied the effects of the simultaneous application of local and architectural impulses, and examined the interactions between process architecture and the transient response of systems subjected to these impulses.

For dissolution-local experiments, we found that despite the concurrent application of local impulses, systems subjected to these pair of impulses still tend to favor architectures with parallel arrangements, the predominant mitigation strategy for the application of dissolution impulses. This is in spite of the tendency of local impulses to delay overall system convergence and to dampen the parallel preference relative to systems that only face dissolution impulses (i.e., without local impulses). For division-local experiments, we found a similar preference for parallel architectures. Finally, in the case of combination impulses, we found evidence to suggest that the simultaneous application of local impulses does not affect heuristics determined to mitigate combination impulses alone as determined in previous work. In all cases we examined architectures for the same 5-subsystem case study. These conclusions provide the basis to initiate a broader investigation that examines the applicability of the parallel character hypothesis for a general distributed design system.

Overall, the results across these experiments suggest that the simultaneous application of architectural and local impulses do not create new aggregate transient responses in a system; rather, their effects on system response appear to be somewhat independent of one another. In addition, the influence of a local impulse may be more pronounced on some architectural impulse types than others. For example, in the division-local experiments, the influence of local impulses in dampening the division-only mitigation strategy of favoring parallel architectures is less pronounced than in the case of dissolution-local experiments.

The broad applicability of these results is apparent when one considers the wide range of scenarios that architectural and local impulses can represent. Architectural impulses can represent the effects of events such as natural disasters (which may disable a subsystem, simulating a dissolution impulse, for instance) or corporate restructuring (where division and combination impulses represent the spin-off and mergers of subsidiaries, respectively). In contrast, local impulses represent one of the most common disruptive inputs: designer mistakes. Studying the amalgamated effects of these impulses in simulations can inform designers of the best mitigation strategies to pursue in response to the application of such impulses in real-world scenarios.

Areas for future work include more comprehensive large-scale statistical validation of the conclusions above for arbitrary architectures. Other areas for future study include the investigation of external impulses, which represent unintended inputs emerging from outside the design system (such as market forces or shifts in consumer demand for a type of product). Such studies, when paired with the results from this paper and earlier work on impulses, will equip the designer with the proper tools to react to a wide array of disruptive events.

ACKNOWLEDGMENTS

The authors thank the National Science Foundation (Grant IIP-1068077) and the Department of Mechanical and Aerospace Engineering at the University at Buffalo, SUNY for their support of this research. This material is also based on work supported by the National Science Foundation Graduate Research Fellowship Program (Grant 1122374).

REFERENCES

- Bloebaum, C.L., Hajela, P., & Sobieszcanski-Sobieski, J. (1992). Non-hierarchical system decomposition in structural optimization. *Engineering Optimization* 19(3), 171–186. doi:10.1080/03052159208941227
- Browning, T.R. (2001). Applying the design structure matrix to system decomposition and integration problems: a review and new directions. *IEEE Transactions on Engineering Management* 48(3), 292–306. doi:10.1109/17.946528
- Chanron, V., & Lewis, K. (2004). Convergence and stability in distributed design of large systems. *Proc. ASME Int. Design Engineering Technical Confs.*, Paper No. DETC2004/57344, Chicago, September 2–6.
- Chanron, V., & Lewis, K. (2006). A study of convergence in decentralized design processes. *Research in Engineering Design* 16(3), 133–145. doi:10.1007/s00163-005-0009-8
- Chanron, V., Singh, T., & Lewis, K. (2005). Equilibrium stability in decentralized design systems. *International Journal of Systems Science* 36(10), 651–662. doi:10.1080/00207720500219963
- Denker, S., Steward, D.V., & Browning, T.R. (2001). Planning concurrency and managing iteration in projects. *Project Management Journal* 32(3), 31–38.
- Devendorf, E., Devendorf, M., & Lewis, K. (2010). Using network theory to model distributed design systems. *Proc. 13th AIAA ISSMO Multidisciplinary Analysis and Optimization Conf.*, Paper No. AIAA-2010-9027, Fort Worth, TX, September 13–15.
- Devendorf, E., & Lewis, K. (2010). Examining interactions between solution architecture and designer mistakes. *Proc. ASME Int. Design Engineering Technical Confs.*, Paper No. DETC2010/28872, Montreal, August 15–18.
- Devendorf, E., & Lewis, K. (2011). The impact of process architecture on equilibrium stability in distributed design. *Journal of Mechanical Design* 133(10), 101001–101011. doi:10.1115/1.4004463

- Devendorf, E., & Lewis, K. (2013). Characterization of the transient response of coupled optimization in multidisciplinary design. *Mathematical Problems in Engineering*. doi:10.1155/2013/910209
- Engardio, P., & Einhorn, B. (2005, March 20). Outsourcing Innovation. *Bloomberg Businessweek*.
- Ghosh, S., Lewis, K., & Devendorf, E. (2012). Examining interactions between process architecture and architectural impulses. *Proc. 14th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conf.*, Paper No. AIAA2012/5666, Indianapolis, IN, September 17–19.
- Ghosh, S., Lewis, K., & Devendorf, E. (2013). Examining the impact of aggregated design impulses on process architecture in distributed design. *Proc. ASME Int. Design Engineering Technical Confs.*, Paper No. DETC2013-13315, Portland, OR, August 15–17.
- Gurnani, A., & Lewis, K. (2008). Using bounded rationality to improve decentralized design. *AIAA Journal* 46(12), 3049–3059. doi:10.2514/1.35776
- Martins, J.R.R.A., & Lambe, A.B. (2013). Multidisciplinary design optimization: a survey of architectures. *AIAA Journal* 51(9), 2049–2075. doi:10.2514/1.J051895
- Ogata, K. (2005). *Modern Control Engineering*. Upper Saddle River, NJ: Prentice Hall.
- Rogers, J.L. (1996). DeMAID/GA: an enhanced design manager's aid for intelligent decomposition. *Proc. 6th AIAA/USAF/NASA/ISSMO Symp. Multidisciplinary Analysis and Optimization*, Paper No. AIAA96/4157, Seattle, WA, September 4–6.
- Smith, R.P., & Eppinger, S.D. (1997). Identifying controlling features of engineering design iteration. *Journal of Management Science* 43(3), 176–193. doi:10.1287/mnsc.43.3.276
- Thurston, D. (2001). Real and perceived limitations to decision based design with utility analysis. *Journal of Mechanical Design* 123(2), 176–182. doi:10.1115/1.1363610
- Ulrich, K.T., & Eppinger, S.D. (2012). *Product Design and Development*. New York: McGraw-Hill.
- Vincent, T.L. (1983). Game theory as a design tool. *Journal of Mechanisms, Transmissions and Automation in Design* 105(2), 165–170. doi:10.1115/1.3258503
- Ward, A., Liker, J.K., Cristiano, J.J., & Sobek, D.K. (1995). The second Toyota paradox: how delaying decisions can make better cars faster. *Sloan Management Review* 36(3), 43–61. doi:10.1016/0024-6301(95)94306-J
- Wernz, C., & Deshmukh, A. (2010). Multiscale decision-making: bridging organizational scales in systems with distributed decision-makers. *European Journal of Operational Research* 202(3), 828–840. doi:10.1016/j.ejor.2009.06.022
- Wiecek, M.M., & Reneke, J.A. (2005). Complex system design decomposition under uncertainty and risk. *Proc. 6th World Congr. Structural and Multidisciplinary Optimization*, Rio de Janeiro, May 30–June 3.
- Yang, M., & Jin, Y. (2008). An examination of team effectiveness in distributed and co-located engineering teams. *International Journal of Engineering Education* 24(2), 400–408.

Sourobh Ghosh is a master's degree candidate and National Science Foundation Graduate Research Fellow in the Department of Mechanical Engineering at MIT. He earned a BS in mechanical engineering and a BA in economics from the University at Buffalo in 2013. His research interests include set-based design, multidisciplinary design optimization, and the design of complex systems.

Erich Devendorf is a Computer Engineer for the Air Force Research Laboratory Information Directorate. He earned a PhD in mechanical engineering from the University at Buffalo in 2011. Dr. Devendorf's research interests include analyzing distributed decision networks, cyber design processes, and the science of mission assurance.

Kemper Lewis is a Professor in and Chair of the Department of Mechanical and Aerospace Engineering at the University at Buffalo and a Fellow of the American Society of Mechanical Engineers. He has also served as Executive Director of the New York State Center for Engineering Design and Industrial Innovation and as the University at Buffalo Site Director of the National Center for e-Design. Dr. Lewis' research interests are in the areas of complex system trade-offs, modeling and optimization of decision networks, and adaptive energy systems.