

Lifting the Veil: Drawing insights about design teams from a cognitively-inspired computational model



Christopher McComb and Jonathan Cagan, Department of Mechanical Engineering, Carnegie Mellon University, 5000 Forbes Ave, Pittsburgh, PA 15213, USA

Kenneth Kotovsky, Department of Psychology, Carnegie Mellon University, Pittsburgh, PA 15213, USA

Novel design methodologies are often evaluated through studies involving human designers, but such studies can incur a high personnel cost. It can also be difficult to isolate the effects of specific team or individual characteristics. This work introduces the Cognitively-Inspired Simulated Annealing Teams (CISAT) modeling framework, a platform for efficiently simulating and analyzing human design teams. The framework models a number of empirically demonstrated cognitive phenomena, thus balancing simplicity and direct applicability. This paper discusses the model's composition, and demonstrates its utility through simulating human design teams in a cognitive study. Simulation results are compared directly to the results from human designers. The CISAT model is also used to identify the most beneficial characteristics in the cognitive study.
© 2015 Elsevier Ltd. All rights reserved.

Keywords: *computational model, design cognition, teamwork, engineering design*

Much cognitive research in engineering design has focused on individuals, despite the fact that most engineering design is actually performed by teams (Paulus, Dzindolet, & Kohn, 2011). This work focuses on developing a better understanding of team-based design through computationally simulating the team design process. Empirical studies are a common means for exploring design cognition and for testing new design methodologies. However, these studies can incur a high personnel cost while only returning a limited amount of data. It can also be difficult to isolate the effects of specific characteristics. This work introduces a computational framework that simulates team-based engineering design through creating software agents that directly solve engineering problems. In addition to offering a resource efficient test bed for evaluating design strategies, this framework can be used to test the conclusions from cognitive studies. It can be used to peel apart aspects of human design, and provides a succinct representation of designer behavior. The purpose of the framework is not

Corresponding author:
Jonathan Cagan
cagan@cmu.edu



www.elsevier.com/locate/destud
0142-694X *Design Studies* 40 (2015) 119–142
<http://dx.doi.org/10.1016/j.destud.2015.06.005>
© 2015 Elsevier Ltd. All rights reserved.

to replace cognitive studies, but rather to augment traditional methods of investigation, accelerating the discovery of improved design methodologies.

A significant amount of work has attempted to simulate the performance of both teams and individuals (Fan & Yen, 2004). For instance, both the Virtual Design Team model, and another model applied to teams at NASA's Jet Propulsion Laboratory, incorporated detailed descriptions of design team organization and interaction (Jin & Levit, 1996; Olson, Cagan, & Kotovsky, 2009). Both models were used to simulate complex design tasks, but were also burdened by high model complexity. For instance, the model by Olson et al. (2009) used approximately 1000 distinct variables, and required nearly 100 000 lines of code for implementation. Still other work has utilized agent-based models to explore the formation of mental models during team problem-solving with respect to both interaction structure (Dionne, Sayama, Hao, & James, 2010) and agent memory (Sayama, Farrell, & Dionne, 2011). Mental models were created by either adding noise to the true problem function, or interpolating between known function values. That work obtained results that agreed qualitatively with the literature, but only explored one- and two-dimensional continuous problem domains, and was not compared to the results of any human studies. A recent agent-based design team model also explored the effect of team structure and task complexity on the formation of transactive memory (Singh, Dong, & Gero, 2012, 2013). That work also obtained results that agreed qualitatively with the literature, but modeled the design problem as an abstract task network instead of directly solving a concrete design problem. Other work simulated with great detail the tasks involved in an integrated product development team, but did not apply the model to a real design task, or offer empirical validation (Crowder, Robinson, Hughes, & Sim, 2012).

Regarding the simulation of individuals, simulated annealing (Kirkpatrick, Gelatt, & Vecchi, 1983), a stochastic optimization algorithm, has been used as an effective model for the efforts of individual human problem-solvers (Cagan & Kotovsky, 1997). More recent work has demonstrated the potential benefit of using computational agents to rapidly test and refine rule-based search strategies that can then be provided to human designers (Egan, Cagan, Schunn, & Leduc, 2014). The rule-based search strategies included both stochastic and univariate approaches. In that work, both computational agents and human participants solved a continuous domain problem with a small number of variables, but the work was not extended to more complex problems.

When humans solve a problem, they tend to learn strategies that can be expressed in terms of the move operators that apply to the problem (Langley, 1985; Newell & Simon, 1972). Solution strategies can also be expressed in

terms of search breadth. Expert designers employ a mixture of different breadth- and depth-first search strategies during solving (Ball & Ormerod, 1995). The selection of appropriate search strategies is further impacted by the presence of known goals or targets. It is known that individuals tend to satiate, meaning that they only search the solution space until a solution that satisfies relevant targets is found (Simon, 1956). It is not uncommon for designers to have direct knowledge of goals during the design process. For instance, the widely used target costing method determines goals before design begins, and these goals are used to guide the search for solutions throughout the design process (Cooper & Slagmulder, 1997; Feil, Yook, & Kim, 2004).

Teams are composed of individuals who strive towards a common goal (Kurtzberg & Amabile, 2001). When members of a team interact while working towards the goal, they perform better than individuals working alone (Sio, Kotovsky, & Cagan, 2014; Wood, Chen, Fu, Cagan, & Kotovsky, 2012). Interaction is usually observed to occur organically, taking place at irregular intervals (Stempfle & Badke-schaub, 2002). The performance boost from interaction is caused by the ability of a team to initially diverge to explore a variety of options, but then converge at the right time, focusing the attention of the team members on a diminishing set of alternatives (Dong, Hill, & Agogino, 2004; Fu, Cagan, & Kotovsky, 2010). However, premature convergence on a single solution can be detrimental to solution quality (Jansson & Smith, 1991). For that reason, designers are typically taught to explore multiple solution concepts (Dieter & Schmidt, 2009; Osborn, 1957). Even members of high-performing teams tend to pursue slightly different solution concepts when solving well-defined problems (McComb, Cagan, & Kotovsky, 2015), indicating that members of a team don't always greedily pursue the solutions with highest apparent quality. Therefore, though team members may factor design quality into decisions, they freely pursue designs that may currently display lower quality. Interaction between members of a team is further tempered by preference for one's own designs. In particular, designers are known to largely favor their own designs, often preferring to apply numerous patches to early design concepts than explore alternatives (Ball, Evans, & Dennis, 1994; Ullman, Dieterich, & Stauffer, 1988). Designers have also been shown to preferentially evaluate their own solution concepts (Nikander, Liikkanen, & Laakso, 2014).

This work introduces the Cognitively-Inspired Simulated Annealing Teams (CISAT) modeling framework. The CISAT framework makes use of simulated annealing constructs to model several characteristics of individuals in small design teams. It differs from other simulation models because it strikes a balance between model simplicity and direct applicability, offering a succinct modeling framework that can be used to directly solve engineering design problems. Further the framework enables an analysis of which attributes of a solution strategy most impact the solution outcome. This paper first provides

a description of the characteristics that are modeled within CISAT. Next, the CISAT framework is used to simulated the results of a cognitive study (McComb et al., 2015). These results are directly compared to results derived from human designers performing an identical task. Following this validation, the CISAT model is used to evaluate which characteristics were most and least helpful to teams during the cognitive study.

1 The modeling framework

The CISAT modeling framework is an agent-based platform that is intended to simulate the process and performance of human design teams. A conceptual flowchart for the CISAT modeling framework is provided in Figure 1. Although only three agents are depicted in the flowchart, the framework is general and can model larger teams.

The CISAT framework models 8 characteristics that contribute to a description of how individuals solve problems, both independently and as part of a team. These characteristics are listed briefly below, and explained in greater detail in subsequent sections:

1. **Multi-agency:** A team is a collection of individuals with a common goal (Kurtzberg & Amabile, 2001).
2. **Organic interaction timing:** Interaction within teams occurs at irregular intervals (Stempfle & Badke-schaub, 2002).
3. **Quality-informed solution sharing:** Members of a team tend to focus on the most promising alternatives, but don't do so greedily (McComb et al., 2015).
4. **Quality bias reduction:** Individuals in a team develop multiple solution concepts to avoid premature convergence (Dieter & Schmidt, 2009; Jansson & Smith, 1991; Osborn, 1957).
5. **Self-bias:** Designers tend to be biased in favor of their own designs (Ball et al., 1994; Nikander et al., 2014; Ullman et al., 1988).
6. **Operational learning:** Individuals learn strategies over the course of solving (Langley, 1985).
7. **Locally Sensitive Search:** Designers select from a range of breadth- and depth-first search strategies as they explore the design space (Ball & Ormerod, 1995).
8. **Satisficing:** Individuals only search the solution space until a solution is found that satisfies relevant targets (Simon, 1956).

Randomized selection is employed in several of the following sections. If the selection is made from a discrete set of alternatives, a multinomial distribution is used (which can be thought of as a roll of a weighted die). If the selection involves choosing a value from within some range, a uniform distribution is employed (equal probability for all values within the range).

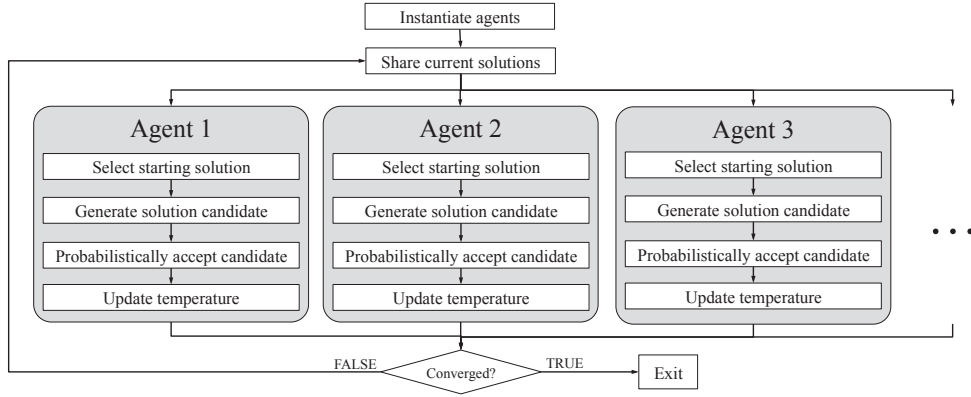


Figure 1 Conceptual flowchart for the CISAT framework

1.1 Multi-agency

The modeling framework is based upon collaboration between multiple software agents. A software agent, referred to simply as an agent in this work, is a computational routine that senses an environment and independently responds to that environment (Franklin & Graesser, 1997). For CISAT agents, the environment is the problem space, and they sense it by evaluating potential solutions. Agents then respond by creating, sharing, and refining solution concepts. Within CISAT, every human designer is modeled by exactly one agent. These agents share a common goal (the minimization of an objective function) making them a suitable proxy for members of a team (Kurtzberg & Amabile, 2001).

1.2 Organic interaction timing

The amount of inter-member communication varies between teams, and occurs at irregular intervals (Stempfle & Badke-schaub, 2002). Similarly, interaction between agents in CISAT occurs probabilistically. Agents independently and probabilistically choose whether or not to interact at the beginning of every iteration. If an agent chooses not to interact, then it continues to iteratively modify its own design. If an agent chooses to interact, it selects a design to explore from amongst the design alternatives currently being pursued by the team (it may select its own design through this process). The selection probability is imperfectly informed by the relative quality of design alternatives, and adjusted to account for self-bias.

1.3 Quality-informed solution sharing

Although a team is composed of individual problem-solvers, there is often additional benefit that is derived from interaction between the individuals (Wood et al., 2012). This arises from the ability of individuals in a team to explore a variety of options, but also to collaboratively focus their attention

on a shrinking set of the most promising alternatives (Dong et al., 2004; Fu et al., 2010). Even members of high-performing teams tend to pursue slightly different solution concepts while solving well-defined problems (McComb et al., 2015), indicating that members of a team don't always greedily pursue the solutions with highest quality. Therefore, though team members may factor design quality into decisions, they freely pursue designs that may currently display lower quality.

The CISAT selection process attempts to model the above description of interaction by allowing agents to probabilistically choose to adopt the current design of any other agent in the team. The selection probability of a design is proportional to its weight, W , which is defined as:

$$W = -F + \max(F). \quad (1)$$

The vector F contains the objective function value for each design in the set of designs currently being pursued by the agents of the team. This equation makes the selection probability of each design proportional to its quality (relative to other available designs). Once the weighting vector, W , has been computed, the agent selects a design alternative by choosing a design with probability proportional to its weight. This selection process can be visualized as the spin of a roulette wheel, or the roll of a loaded die. Once an agent has selected a design alternative to pursue using this probabilistic process, it proceeds to modify that design independently using an internal process structured similarly to simulated annealing.

1.4 Quality bias reduction

Note that in the weighting vector W , the weight placed on the worst design is 0. This means that agents are incapable of selecting the worst design when interacting, and abandon it automatically. This detail could lead to premature convergence within the team, which can be harmful to design (Jansson & Smith, 1991). Although novice designers are often taught to explore multiple solution concepts (Dieter & Schmidt, 2009; Osborn, 1957), expert designers do not generally exhibit this behavior (Cross, 2004). This implies the existence of a range of strategies that are employed along the spectrum from novice to expert. To imbue CISAT with the ability to accommodate this range, a small additional weight is added to W :

$$W \leftarrow W + k_{QBR} \cdot 1 \quad (2)$$

The variable k_{QBR} (chosen from the range $[0, 1]$) controls the strength of quality bias reduction, and 1 is the ones vector. This reduces the effect of the agents'

bias towards designs of high quality. This also places non-zero weight on the worst design alternative, meaning that agents may select any solution concept.

1.5 Self-bias

Designers tend to be biased in favor of designs that they have generated or spent substantial time working on (Ball et al., 1994; Nikander et al., 2014; Ullman et al., 1988). Therefore, CISAT agents are also made to favor their own designs. This is implemented during interaction between agents. Before an agent selects a design to pursue using Equation 2, the agent adds additional weight to the element in \mathcal{W} that corresponds to its own design:

$$W_i \leftarrow W_i + k_{SB} \quad (3)$$

The variable k_{SB} (chosen from the range $[0, 30]$) controls the strength of self-bias, and the subscript i denotes the index for the current design of the agent making the selection. This results in a higher likelihood that the agent will elect to continue working on its own design, mimicking the bias of human designers.

1.6 Operational learning

The actions that can be used to modify a solution are typically referred to as *move operators* and are inherently problem specific (Newell & Simon, 1972). Human problem-solvers tend to learn strategies in terms of these move operators (Langley, 1985). CISAT agents are also provided with a mechanism that allows them to individually learn which move operators are most helpful to design.

Within the CISAT framework all move operators initially have an equal probability of being selected and applied to the current solution (unless a prior distribution over move operators can be inferred for the specific problem). Agents learn by first selecting a move operator and using it to modify the current solution. The new solution is then evaluated using the objective function. If the application of the move operator improved the objective function, the probability of applying that move operator in the future is increased. However, if the application of the move operator yields a solution with a worse objective function value, the probability of applying the move operator is decreased.

1.7 Locally sensitive search

It is known that expert designers tend to use a mixture of depth- and breadth-first solution strategies (Ball & Ormerod, 1995), indicating the value of tailoring search strategies to local characteristics of the design space. CISAT is based on a simulated annealing methodology, so the annealing schedule controls the progressive transition from initial explorative search to final deterministic search. To mimic the locally sensitive search strategies of human

designers, every CISAT agent is given an independently-controlled Triki adaptive annealing schedule (Triki, Collette, & Siarry, 2005). This annealing schedule uses the variance of the quality of past solutions to update the temperature, helping agents respond appropriately to the local design space.

1.8 Satisficing

Decision makers tend to satisfice, meaning that they only search the solution space broadly until a solution that satisfies relevant targets is found (Simon, 1956). Further, engineers and designers tend to have access to such goals (Cooper & Slagmulder, 1997; Feil et al., 2004). Therefore, satisficing may play a crucial role in the design process. The effect of satisficing is implemented in the CISAT framework by increasing an agent's temperature if their designs are far from satisfying relevant targets. The effect of this increase is that the temperature decreases rapidly once a satisficing solution is found, making search more deterministic. However, the temperature remains high until such a solution is found, promoting broad search for a fruitful region of the design space.

2 Applying CISAT to truss design

As a means of validation, the CISAT modeling framework will be used to model the results of a cognitive study on design teams previously conducted by McComb et al. (2015). A summary of the original cognitive study will be provided, followed by a description of how CISAT was configured to model the study. The original results of the cognitive study will also be directly compared to the results from the CISAT simulations.

2.1 Summary of the truss design study

The cognitive study tasked 16 teams of 3 engineering students with the design of a truss structure. Over the course of the study the design problem was changed twice via the introduction of modified problem statements. Participants were given access to a graphical truss design program that allowed them to create, evaluate, and share truss designs within their teams.

Teams designed over the course of six 4-min design sessions. The first problem statement (PS1) provided to teams required them to design a truss structure with a factor of safety of 1.25, and a mass as low as possible. Figure 2 depicts the two loading points and three supports that were required for every design.

After working on PS1 for three of the 4-min design sessions, the participants were given the first of two modified problem statements. PS2 required participants to consider the removal of any one of the bridge supports (leaving only two supports intact at a time). This problem statement required a factor of safety of 1.0, and mass as low as possible.

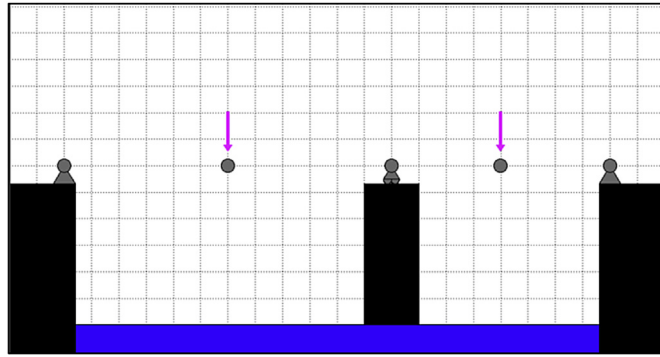


Figure 2 Diagram of problem statements 1 and 2

After working on PS2 for one 4-min design session, the participants were given the second modified problem statement (PS3). This modification required participants to design their truss around the obstacle shown in Figure 3. PS3 required a factor of safety of 1.25, and mass as low as possible. Teams worked on PS3 for the last two 4-min design sessions.

Participants were also given a mass target for each of the problem statements, thus invoking satisficing tendencies. This mass target will be incorporated into CISAT's satisficing temperature component. Each problem statement also had a constraint on the factor of safety, which will be addressed in the CISAT objective function as a penalty.

2.2 Analysis

Analysis of results from the both the cognitive study and the CISAT simulation will be performed using three metrics. The first metric, *strength-to-weight ratio (SWR) of best design*, tracks the best design produced by a team over time. This allows the quality of the team's best design to be visualized over time. The SWR for a given design is calculated as

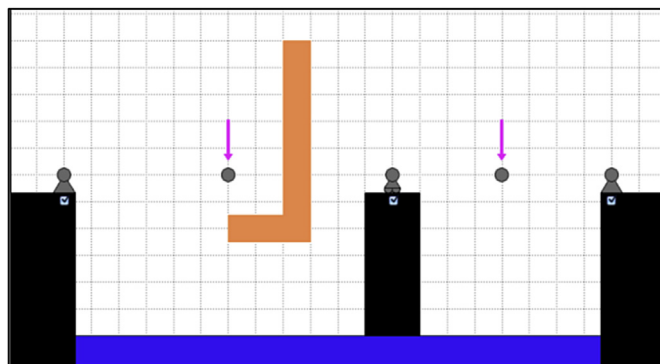


Figure 3 Diagram of problem statement 3

$$SWR(x) = \left(\frac{FOS_x}{FOS_{REQ}} \right) \left(\frac{m_x}{m_{TARGET}} \right)^{-1}. \quad (4)$$

The variables FOS_x and m_x are the factor of safety and mass of truss design x , respectively. Similarly, FOS_{REQ} is the required factor of safety, and m_{TARGET} is the required mass (per the relevant problem statement). The factor of safety of a truss is determined using standard structural analysis techniques (Hibbeler, 2008). The SWR is used only to communicate results, and is not used directly for tracking the best design. A full description of the method used to track the best design can be found in (McComb et al., 2015).

The second metric, *average pairwise distance*, is a means of quantifying divergence (or disagreement) within a team. It is computed as the distance between the designs being explored by any two members in a team at a given instant, averaged across all combinations of two team members. The distance between truss designs is measured in the number of move operators that must be applied to change one design into another (McComb et al., 2015). This is similar to the concept of average pairwise similarity used in other work (Fu et al., 2010; Wood et al., 2012).

The third metric, *frequency of topology operations*, provides a way to measure how teams change their solution strategies over time. Specifically, it tracks the proportion of topology operations in each 4-min design session. A topology operation is any operation that modifies the connectivity of the truss (adding or removing joints or members). All other operations are shape operations (changing the size of members or moving joints).

The analysis further involves comparing high- and low-performing teams. Teams are assigned to these two groups based on their cumulative performance across problem statements. The method used to rank teams is identical to that employed in (McComb et al., 2015). The top 31.25% of teams are designated as *high-performing teams*, while the lowest 31.25% are designated as *low-performing teams*. The 31.25% cut-off was chosen to match up directly with the cut-off used in the original analysis of the study.

2.3 CISAT configuration

Configuring CISAT to simulate team performance on a problem involves defining appropriate objective functions, implementing a method for instantiating a design, creating move operators to modify designs, and selecting values for other miscellaneous parameters required by CISAT.

Simulating this cognitive study requires three objective functions, corresponding to the three problem statements. Every objective function is in the form of

the mass of the design plus a penalty to resolve constraints on the solution. The first objective function only places a constraint on the factor of safety:

$$f_{PS1}(x) = m_x + g_{FOS}(x). \quad (5)$$

In Equation 4, m_x is the mass of design x , and the function $g_{FOS}(x)$ computes an appropriate penalty if the factor of safety is too low. This penalty is computed as

$$g_{FOS}(x) = 10^4 \cdot \max(0, FOS_{REQ} - FOS_x)^2. \quad (6)$$

The variable FOS_{REQ} is the factor of safety required for the current problem statement (1.00 for PS2, and 1.25 otherwise), and FOS_x is the factor of safety of truss design x .

The second objective function, f_{PS2} , applies the maximum FOS penalty across a variety of support cases, per PS2:

$$f_{PS2}(x) = m_x + \max \begin{pmatrix} g_{FOS}(x) \\ g_{FOS}(x^{-S1}) \\ g_{FOS}(x^{-S2}) \\ g_{FOS}(x^{-S2}) \end{pmatrix}. \quad (7)$$

The notation x^{-Si} represents the design x with the i^{th} support removed. Therefore, the function $f_{PS2}(x)$ applies the highest penalty, considering the same set of support conditions required in PS2.

The third and final objective function is very similar to the initial objective function, but incorporates a second penalty function, g_{OBS} , which penalizes the solution for violating the obstacle shown in Figure 3:

$$f_{PS3}(x) = m_x + g_{FOS}(x) + g_{OBS}(x). \quad (8)$$

The second penalty function, g_{OBS} , imposes a penalty based on J_x , the number of truss joints within the obstacle, and L_x , the cumulative member length within the obstacle:

$$g_{OBS}(x) = 10^4 \cdot (J_x + L_x^2). \quad (9)$$

Agents instantiate their truss designs by first determining how many joints the truss design will have by drawing a random integer. For this work the number of joints is restricted to fall between 8 and 30, inclusive. The location of each joint is chosen so that every joint is approximately equidistant from its nearest neighbors. Delaunay triangulation is then used to determine a stable pattern for connecting the joints using structural members. Only one joint or one member is added per iteration until the initial layout is completed. If an agent's design becomes statically indeterminate over the course of the simulation, it is permitted to instantiate a new truss design.

Move operators must be defined to allow agents to act upon and modify their solutions. The operators defined for this work are listed and described below:

MO1. *Add a member*: The two nearest unconnected joints are connected with a member.

MO2. *Change the size of a member*: A member is selected with probability proportional to $|FOS - FOS_{REQ}|$. If at least one member is failing, only failing members are selected from. Once a member is selected, its size is increased if $FOS < FOS_{REQ}$, and decreased otherwise.

MO3. *Change the size of all members*: If the majority of members have factors of safety greater than FOS_{REQ} , the size of every members is increased. Otherwise, the size of all members is decreased.

MO4. *Delete a member*: A member is probabilistically selected with probability proportional to its factor of safety. The selected member is then deleted.

MO5. *Move a joint*: A joint is selected at random. A greedy and deterministic search algorithm is then used to improve the location of the joint.

MO6. *Delete a joint*: A joint is probabilistically selected with probability proportional to the sum of the factors of safety of all members connecting to the joint. The selected joint is then deleted.

MO7. *Brace a member*: A member is selected for bracing from the set of members that are both in compression, and have a factor of safety less than FOS_{REQ} . A joint is inserted in the middle of the selected member. The new joint is then connected to the nearest joint that it is not already connected to. This move operator is completed over the course of 2 iterations.

MO8. *Add a joint and attach*: A joint location is selected using a procedure identical to that used in MO1. A joint is created at that location, and members are connected between this joint and the three nearest joints. This move operator is completed over the span of 4 iterations.

This set of move operators is designed to reflect the operations available to human participants in the cognitive study (MO1-6), and also to allow simple heuristic operations (MO7-8).

Participants in the cognitive study had to begin designing their truss by laying out a network of joints and members. Therefore, study participants must have

begun with a higher probability of applying move operators that would enable this layout process. To model this aspect in CISAT, the probability of MO1 (adding a member) is increased according to the number of members in the initial layout. Similarly, the probability of MO8 (add and attach a joint) is increased according to the number of joints in the initial layout.

In order to ensure further parity between CISAT simulations and the original cognitive study, agents in CISAT were made to apply move operators at the same average rate as human study participants. An analysis of the data recorded in the original cognitive study indicated that the average individual applied one operation every three seconds. Therefore, over the course of six 4-min design sessions (1440 s), the average individual applied 480 moves. Every CISAT agent was also allowed this number of moves.

2.4 Comparison of results

Of the 16 teams that took part in the original cognitive study, 5 were designated as high-performing, and 5 as low-performing, based on an evaluation of their final design solutions. In order to establish a better statistical representation, 64 teams (4 times the number of teams in the cognitive study) are simulated using CISAT. Of these teams, 20 are designated as high-performing and 20 as low-performing. A comparison between the results of the original cognitive study and those simulated using CISAT is provided in [Figure 4](#). The vertical, black lines indicate the introduction of a new problem statement.

The CISAT framework reproduces several of the main trends that are apparent in the original results from the cognitive study. For instance, the high-performing human teams showed an early divergent period, followed by a pattern of fairly constant average pairwise distance (see [Figure 4\(c\)](#)). This is echoed in the CISAT simulation (see [Figure 4\(d\)](#)). The low-performing human teams show higher average pairwise distance, and a period of divergence near the end of the study. This behavior is also evidenced in the CISAT simulation. CISAT also predicts the correct mean trend for frequency of topology operations (see [Figure 4\(e\)](#) and [\(f\)](#)). Both human teams and CISAT teams show an initial decrease in frequency of topology operations, followed by an increase after the introduction of the new problem statements.

The Pearson correlation coefficient is used to quantify the degree to which the CISAT framework reproduces the mean trends of the cognitive study. The Pearson correlation coefficient measures the linear correlation between two variables, and returns a value between -1.0 (indicating a perfect negative correlation) and $+1.0$ (indicating a perfect positive correlation). A summary of Pearson correlation coefficients for the three metrics used for comparison between human and CISAT results is provided in [Table 1](#).

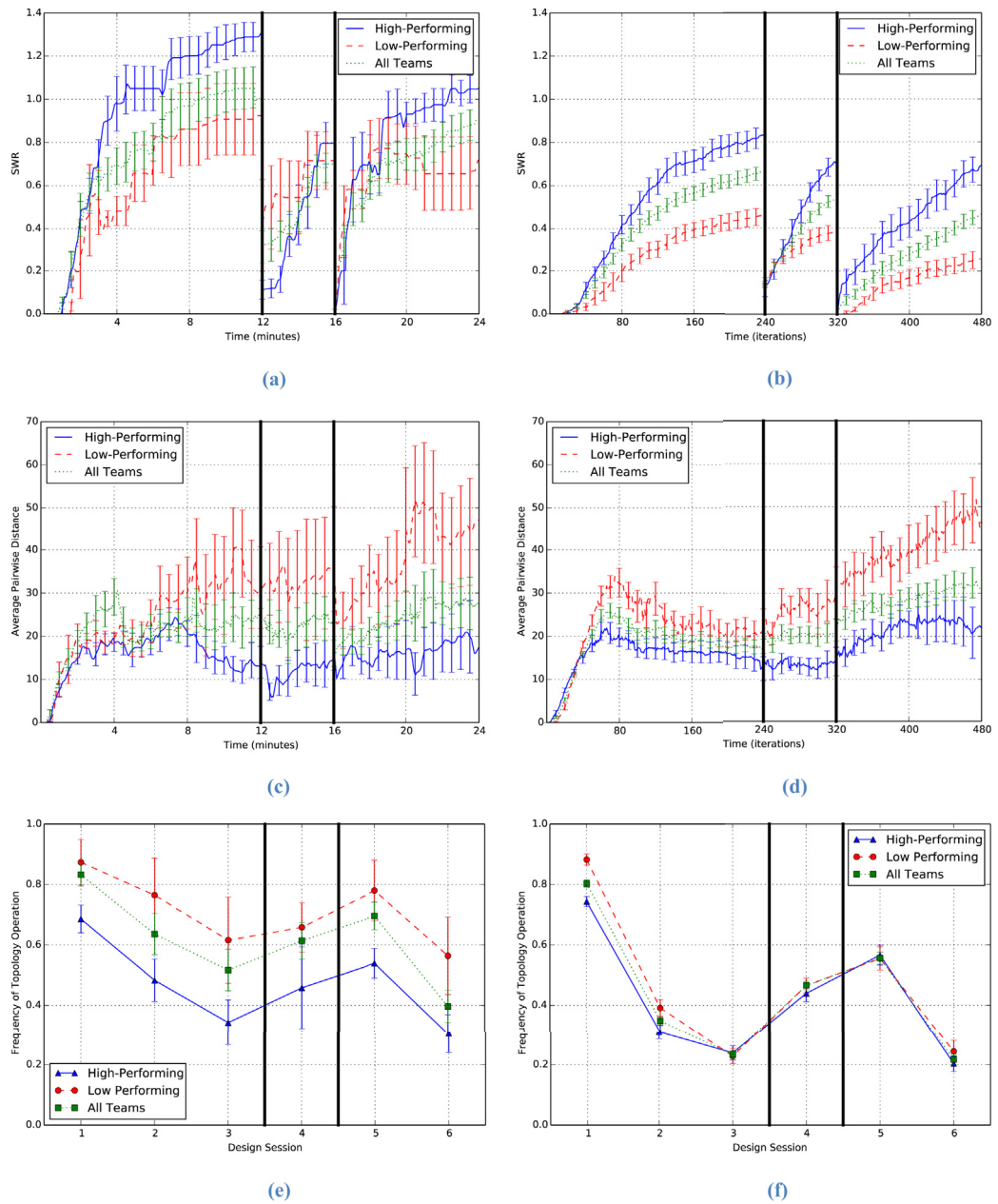


Figure 4 Comparison between cognitive study results and CISAT simulation results (all error bars show ± 1 S.E.). (a) SWR of best design (cognitive study results), (b) SWR of best design (CISAT simulation results), (c) Average pairwise distance (cognitive study results), (d) Average pairwise distance (CISAT simulation results), (e) Frequency of topology operations (cognitive study results), (f) Frequency of topology operations (CISAT simulation results)

Table 1 Summary of human versus model Pearson correlation coefficients

| <i>Metric</i> | <i>High-performing teams</i> | <i>Low-performing teams</i> | <i>All teams</i> |
|----------------------------------|------------------------------|-----------------------------|----------------------------|
| SWR of best design | 0.879 ($\rho < 10^{-5}$) | 0.786 ($\rho < 10^{-5}$) | 0.876 ($\rho < 10^{-5}$) |
| Average pairwise distance | 0.676 ($\rho < 10^{-5}$) | 0.741 ($\rho < 10^{-5}$) | 0.787 ($\rho < 10^{-5}$) |
| Frequency of topology operations | 0.951 ($\rho < 0.005$) | 0.890 ($\rho < 0.05$) | 0.936 ($\rho < 0.01$) |

The coefficients provided in Table 1 are always above 0.65, and the majority of them are above 0.85. This quantitatively reiterates a fact that is already qualitatively evident in Figure 4. Although CISAT does not perfectly reproduce the results of the cognitive study, there is a strong positive correlation between the trends displayed in the two sets of results.

The starkest difference between the CISAT simulation and the original cognitive data is found in the SWR of the best design (Figure 4(a) and (b)). It is possible that this resulted from the inability of CISAT agents to consider chains of moves. The ability of human problem-solvers to think in terms of multiple sequential moves is indicative of expert problem-solving (Pretz, Naples, & Sternberg, 2003), but has also been observed in individuals with little experience (Kotovsky, Hayes, & Simon, 1985). Therefore, although CISAT agents were applying move operators in proportions similar to those of the human truss designers, the naïve sequencing of the move operators may have had a handicapping effect. This insight indicates that implementing better models of learning and heuristic development in CISAT could lead to higher performance, and better agreement with human solvers.

3 Investigating team strengths with CISAT

In traditional human studies, it is difficult to cull which aspects of problem solving are most influential and beneficial without running multiple studies. Even if multiple studies are possible, it is usually not feasible to isolate features entirely. However, in CISAT such assessment is straightforward and informative. This section focuses on determining the characteristics that were most helpful or harmful in performing the truss design task. By assessing the final SWR of simulated teams composed of agents with and without a given characteristic, it is possible to evaluate the effect of that characteristic on overall performance. For instance, if the removal of a characteristic decreases the final SWR, it can be inferred that the presence of that characteristic is beneficial to team performance. The results of this CISAT analysis then indicate the characteristics that are most important to effective human design teams for, in this case, the truss design problem.

The above procedure is applied to quality bias reduction, operational learning, satisficing, locally sensitive search, self-bias, and organic interaction. The effect of each characteristic is evaluated with 100 simulated teams. In the interest of simplicity, these simulated teams are only used to solve PS1. Multi-agency and

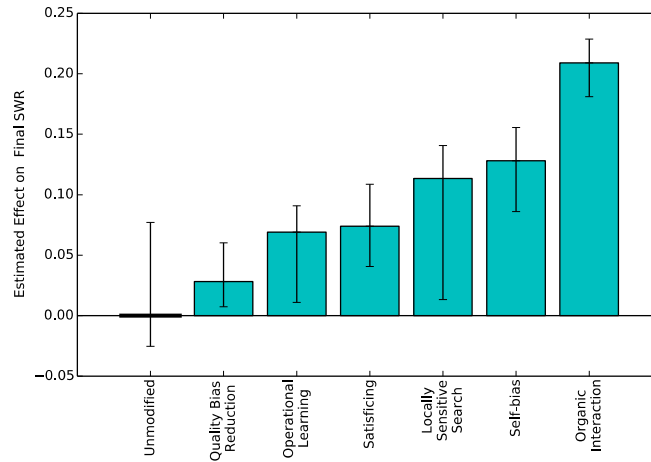


Figure 5 Estimated effect on final SWR (error bars show ± 1 S.E.)

solution sharing are not evaluated, since those characteristics enable basic team-like performance within CISAT. Figure 5 shows the estimated effect on the final SWR from each of the characteristics, relative to the median final SWR of the unmodified CISAT model. The median of the data is used to communicate these results because it is more representative of central tendency than the mean.

Figure 5 communicates the magnitude of median differences, but to further understand the impact of these differences it is necessary to report an *effect size*. Fritz, Morris, and Richler (2012) provide a more detailed description of the need for and use of effect size metrics. Although Cohen's *d* is commonly used for this purpose, Cliff's δ will be used here because of non-normality in the data (Hess & Kromrey, 2004; Leech & Onwuegbuzie, 2002). Cliff's δ is a

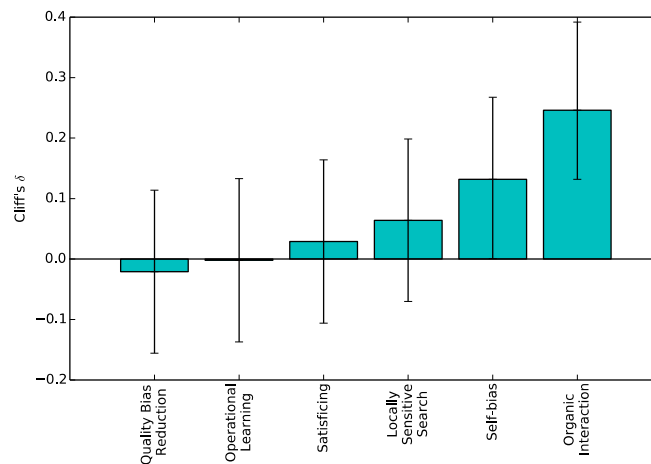


Figure 6 Cliff's δ effect size (error bars show 90% confidence intervals)

non-parametric dominance statistic that indicates the degree to which values in one set of data lie above values in a second set (Cliff, 1996). A value of -1 or $+1$ indicates no overlap between sets (complete dominance), while a value of 0 indicates complete overlap (no dominance). The effect sizes for this work are provided in Figure 6.

As discussed in previous sections of this paper, each of the characteristics analyzed in this section has been observed in humans. Figures 5 and 6 show that self-bias and organic interaction display the largest and most significant effects. Both of these characteristics specifically play a role in moderating interaction between team members or agents. These two characteristics decrease the frequency and effect of interaction when present. The absence of organic interaction increases the frequency of interaction, and the absence of self-bias increases the likelihood that an agent will abandon its current solution in favor of a solution being pursued by a teammate.

Examining the performance attributes of teams without these beneficial characteristics could offer further insight as to the cause of their poor performance. Figures 7 and 8 show the median values of the SWR of the best design and average pairwise distance during solving.

Figure 8 indicates that typical teams (teams simulated with all characteristics turned on) achieve high performance through a period of slow convergence. In contrast, the poorly performing teams converge quickly, resulting in lower final design solution quality. This indicates that both frequent interaction and low self-bias may lead to premature convergence within teams, precipitating final design solutions with lower quality. This may also imply that exploring design methods that encourage divergence may improve final design solution quality by further protecting teams from premature convergence.

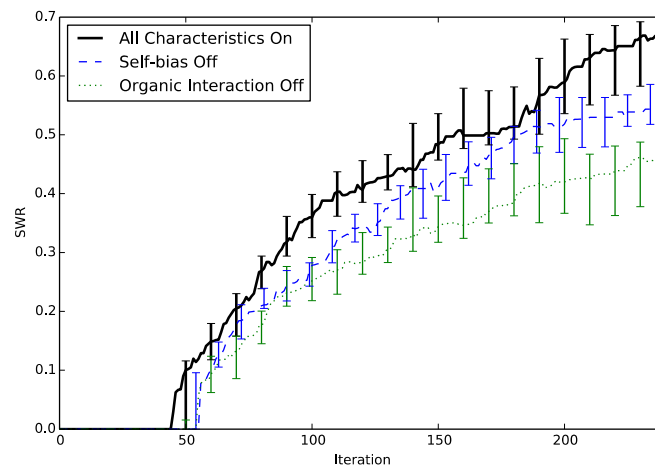


Figure 7 SWR of best design (error bars show ± 1 S.E.)

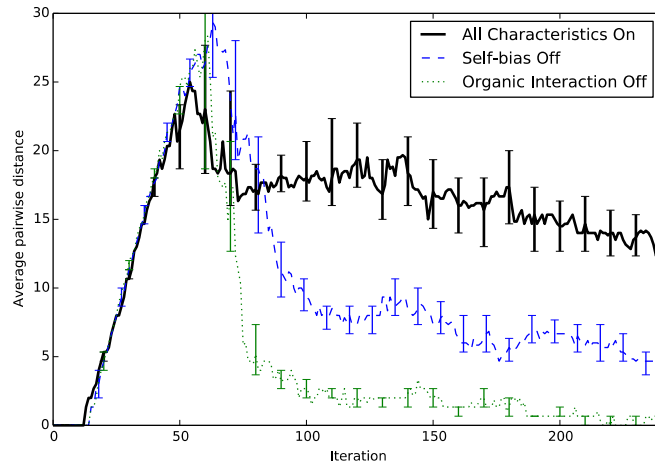


Figure 8 Average pairwise distance (error bars show ± 1 S.E.)

Additional statistical values describing the final SWR are shown in Table 2. These include the 25th percentile (or first quartile point), the 50th percentile (or median) and the 75th percentile (or third quartile point). The inter-quartile range (difference between the 75th percentile and 25th percentile) is shown to indicate the spread of the data within a group of teams, and the inter-group range is shown to indicate the range of values displayed across groups of teams. Examining these statistics can indicate the way in which the teams' final SWRs are distributed about the median. The 75th percentile is simply the median of the upper half of the data, so it is indicative of the performance of the best teams. Similarly, the 25th percentile is indicative of the performance of the worst teams. In Table 2, the value of the 25th percentile displays little difference between groups (an inter-group range of approximately 0.054). However, there is much greater variability between the upper quartiles of the groups (demonstrated by an inter-group range of 0.144). In addition, the inter-quartile range is much higher for the group of teams in which all characteristics are turned on. This indicates that removing characteristics from the CISAT simulation framework does not substantially impact the low-performing teams, but does handicap the performance of the high-performing teams. This consequentially compresses the distribution, as evidenced by the decrease in inter-quartile range.

Table 2 Descriptive statistics for final SWR

| Statistic | All characteristics On | Self-bias Off | Organic interaction Off | Inter-group range |
|-----------------------------|------------------------|---------------|-------------------------|-------------------|
| 25 th percentile | 0.400 | 0.398 | 0.346 | 0.054 |
| Median | 0.672 | 0.543 | 0.462 | 0.210 |
| 75 th percentile | 0.854 | 0.779 | 0.710 | 0.144 |
| Inter-quartile Range | 0.454 | 0.381 | 0.364 | |

Because self-bias and organic interaction play a role in moderating communication between agents, these results emphasize the crucial role of interaction in human teams (Sio et al., 2014; Stempfle & Badke-schaub, 2002; Wood et al., 2012). Self-bias encourages individuals to continue work on their current solution concept, adding significant detail and critical refinement. Organic interaction decreases the frequency of communication, therefore increasing the extent to which individuals refine their current solutions between interactions.

The benefit derived from the presence of organic interaction and self-bias can be understood by appealing to concepts used in the analysis of social networks. A social network is composed of individuals and the relationships, or ties, between them. These ties can be characterized in terms of strength (Granovetter, 1973). Strong ties exhibit frequent interaction, reciprocity, and high emotional intensity, while weak ties have infrequent interaction, little reciprocity and lower intensity (Granovetter, 1973). There is evidence that weak ties may play a crucial role by transmitting information between individuals with diverse perspectives and diverse approaches to problems (Granovetter, 1983). In the context of the truss study, the presence of organic interaction and self-bias decrease the frequency and intensity of interactions, thus forming weak ties. In turn, these weak ties likely allow team members to explore different regions of the design space, leading to higher divergence and final design solutions with higher quality.

Conversely, the absence of either organic interaction or self-bias can increase the strength or frequency of interactions, leading to the formation of strong ties. It has been demonstrated that strong ties promote group cohesion, but may do so at the expense of the group's goals (Flache & Macy, 1996; Flache, 2002). Within the context of the truss design problem, this could lead to a state in which the formation of consensus (or low divergence) between team members becomes more of a driving factor in the design process than the search for high quality solutions. Such a state is similar in many ways to groupthink, a psychological phenomena characterized by the search for consensus with little regard for critical evaluation of concepts (Janis, 1971). It has been theorized that groupthink may detrimentally affect decision-making teams (Jones & Roelofsma, 2000).

Up to this point in the paper, analyses have made comparisons between several groups of teams, where all agents within a group of teams had the same active characteristics (either all characteristics on, self-bias turned off, or organic interaction turned off). Now, trends *within* groups of teams will be examined by computing the correlation coefficient between final average pairwise distance and final SWR (both of which are team characteristics). For teams simulated with all agent characteristics turned on, the Pearson correlation coefficient between these two variables is -0.446 . For teams in which agents lacked self-bias, the correlation coefficient is -0.481 , and for teams in which

agents lacked organic interaction it is -0.301 . Similar negative correlations are observed with the removal of other characteristics, and all correlations are highly significant ($p < 0.005$). This indicates that low final average pairwise distance tends to occur in teams that also have a high final SWR. Therefore, relative to other teams in which agents have the same active characteristics, a team that shows low final divergence is likely to produce a high quality solution.

A similar trend has been demonstrated in a variety of design problems. For instance, team mental model sharedness was correlated with coordinated team performance in teams playing the open-ended Kantjil Design Game (Dong, Kleinsmann, & Deken, 2013). The correlation was also identified when the truss design problem modeled in this work was solved by human designers (McComb et al., 2015). There, the relationship was attributed to expert-like characteristics of high-performing teams. Expert designers have been shown to quickly commit to a single solution concept (Cross, 2004). The fact that expert solutions also tend to be of high quality is indicative of the fact that expert designers are capable of selecting a good initial representation of the problem, and do not need to search divergently. Although the agents created in this work are not intended to be experts, they are still created with a variety of initial representations with varying levels of quality. A team with a lower quality representation may need to search divergently in an attempt to improve that representation. However, a team with a high quality representation has no need to refine their representation through broad search. Thus, the same mechanism (selection of initial representations with varying levels of quality) may have given rise to a similar trend (negative correlation between divergence and quality) in both human- and agent-based studies.

4 Conclusions

This work introduces the Cognitively-Inspired Simulated Annealing Teams (CISAT) modeling framework, an agent-based platform for simulating team-based engineering design. This framework was used to directly simulate the results of a cognitive study in which teams of engineering students tackled a structural design problem. A comparison of the CISAT simulated results to those of the original cognitive study revealed a high degree of linear correlation between the two. This indicates that CISAT is capable of capturing the trends observed in humans solving a simplified engineering design task. Next, CISAT was used to explore the particular characteristics that were most beneficial to teams in solving this task. This analysis indicated that proper interaction (specifically self-bias and organic interaction timing) was crucial to enabling team success in the truss design task. Further analysis revealed the importance of flexible design methods that allow for sufficient, but not excessive, divergent search.

This work only models a set of cognitive phenomena that has been demonstrated within the domains of design or problem-solving. However, the

validated CISAT framework can now be used as a platform to simulate the effects of biases that have not been explicitly demonstrated in those domains. The results of such simulations could be used to formulate promising studies to be carried out with human test subjects.

The current validation study has demonstrated that the CISAT modeling framework is capable of accurately modeling small teams. However, larger teams may require a hierarchical structure amongst members, particularly if solving a complex task. Although CISAT is currently only capable of simulating flat teams (i.e. teams without overt structure), it may act as a building block for simulating larger, hierarchical organizations. For instance, an organization composed of multiple sub-teams could be modeled as a conglomeration of CISAT teams. Additional development would be necessary to define an inter-team protocol for communicating solution information, and to define specific responsibilities and sub-tasks for individual teams. In this context, biased information passing (Austin-Breneman, Yu, & Yang, 2014) between teams may emerge as an important inter-team characteristic.

Future work will seek to implement more detailed models for learning and heuristic development within the CISAT framework, allowing agents to more intelligently sequence the application of move operators. The CISAT framework will also be used to model more complex design problems to provide opportunities for further validation and refinement.

Acknowledgments

This material is based upon work supported by the National Science Foundation Graduate Research Fellowship under Grant No. DGE125252. The authors would also like to thank the AFOSR for funding this research through grant FA9550-12-1-0374. A previous version of this paper has also been submitted to the Design Theory and Methodology Conference at ASME IDETC 2015 (McComb, Cagan, & Kotovsky, 2015).

References

- Austin-Breneman, J., Yu, B. Y., & Yang, M. C. (2014). Biased information passing between subsystems over time in complex system design. In *ASME 2014 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference* (pp. DETC2014-34433). Buffalo, NY. <http://dx.doi.org/10.1115/DETC2014-34433>.
- Ball, L. J., Evans, J. S. B., & Dennis, I. (1994). Cognitive processes in engineering design: a longitudinal study. *Ergonomics*, 37(11), 1753–1786. <http://dx.doi.org/10.1080/00140139408964950>.
- Ball, L. J., & Ormerod, T. C. (1995). Structured and opportunistic processing in design: a critical discussion. *International Journal of Human-Computer Studies*, 43(1), 131–151. <http://dx.doi.org/10.1006/ijhc.1995.1038>.

- Cagan, J., & Kotovsky, K. (1997). Simulated annealing and the generation of the objective function: a model of learning during problem solving. *Computational Intelligence*, 13(4), 534–581. <http://dx.doi.org/10.1111/0824-7935.00051>.
- Cliff, N. (1996). Answering ordinal questions with ordinal data using ordinal statistics. *Multivariate Behavioral Research*, 31(3), 331–350. <http://dx.doi.org/10.1207/s15327906mbr3103>.
- Cooper, R., & Slagmulder, R. (1997). *Target costing and value engineering*. Portland, OR: Productivity Press.
- Cross, N. (2004). Expertise in design: an overview. *Design Studies*, 25(5), 427–441. <http://dx.doi.org/10.1016/j.destud.2004.06.002>.
- Crowder, R. M., Robinson, M. A., Hughes, H. P. N., & Sim, Y.-W. (2012). The development of an agent-based modeling framework for simulating engineering team work. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, 42(6), 1425–1439. <http://dx.doi.org/10.1109/TSMCA.2012.2199304>.
- Dieter, G., & Schmidt, L. (2009). *Engineering design* (4th ed.). McGraw-Hill.
- Dionne, S. D., Sayama, H., Hao, C., & James, B. (2010). The role of leadership in shared mental model convergence and team performance improvement: an agent-based computational model. *The Leadership Quarterly*, 21(6), 1035–1049. <http://dx.doi.org/10.1016/j.leaqua.2010.10.007>.
- Dong, A., Hill, A. W., & Agogino, A. M. (2004). A document analysis method for characterizing design team performance. *Journal of Mechanical Design*, 126(3), 378–385. <http://dx.doi.org/10.1115/1.1711818>.
- Dong, A., Kleinsmann, M. S., & Deken, F. (2013). Investigating design cognition in the construction and enactment of team mental models. *Design Studies*, 34(1), 1–33. <http://dx.doi.org/10.1016/j.destud.2012.05.003>.
- Egan, P. F., Cagan, J., Schunn, C., & Leduc, P. R. (2014). Cognitive-based search strategies for complex bio-nanotechnology design derived through symbiotic human and agent-based approaches. In *ASME 2014 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference* (pp. DETC2014–34714). <http://dx.doi.org/10.1115/DETC2014-34714>.
- Fan, X., & Yen, J. (2004). Modeling and simulating human teamwork behaviors using intelligent agents. *Physics of Life Reviews*, 1(3), 173–201. <http://dx.doi.org/10.1016/j.plrev.2004.10.001>.
- Feil, P., Yook, K., & Kim, I. (2004). Japanese target costing: a historical perspective. *International Journal of Strategic Cost Management* 10–19.
- Flache, A. (2002). The rational weakness of strong ties: failure of group solidarity in a highly cohesive group of rational agents. *The Journal of Mathematical Sociology*, 26(3), 189–216. <http://dx.doi.org/10.1080/00222500212988>.
- Flache, A., & Macy, M. W. (1996). The weakness of strong ties: collective action failure in a highly cohesive group. *The Journal of Mathematical Sociology*, 21(1–2), 3–28. <http://dx.doi.org/10.1080/0022250X.1996.9990172>.
- Franklin, S., & Graesser, A. (1997). Is it an agent, or just a program?: a taxonomy for autonomous agents. In *Intelligent agents III: Agent theories, architectures, and languages* (pp. 21–35), DOI: 10.1.1.52.1255.
- Fritz, C., Morris, P., & Richler, J. (2012). Effect size estimates: current use, calculations, and interpretation, 141(1), 2–18. <http://dx.doi.org/10.1037/a0024338>.
- Fu, K., Cagan, J., & Kotovsky, K. (2010). Design team convergence: the influence of example solution quality. *Journal of Mechanical Design*, 132(11), 111005. <http://dx.doi.org/10.1115/1.4002202>.
- Granovetter, M. S. (1973). The strength of weak ties. *The American Journal of Sociology*, 78(6), 1360–1380. <http://dx.doi.org/10.1086/225469>.

- Granovetter, M. S. (1983). The strength of weak ties: a network theory revisited. *Sociological Theory*, 1(1983), 201–233. <http://dx.doi.org/10.2307/202051>.
- Hess, M., & Kromrey, J. (2004). Robust confidence intervals for effect sizes: a comparative study of Cohen's d and Cliff's delta under non-normality and heterogeneous variances. In *Annual Meeting of the American Educational Research Association* (pp. 1–30), DOI: 10.1.1.487.8299.
- Hibbeler, R. (2008). *Structural analysis* (7th ed.). Prentice Hall.
- Janis, I. (1971). Groupthink. *Psychology Today*, 5(6), 43–46.
- Jansson, D., & Smith, S. (1991). Design fixation. *Design Studies*, 12(1), 3–11. [http://dx.doi.org/10.1016/0142-694X\(91\)90003-F](http://dx.doi.org/10.1016/0142-694X(91)90003-F).
- Jin, Y., & Levit, R. (1996). The virtual design team: a computational model of project organization. *Computational & Mathematical Organization Theory*, 2(3), 171–196.
- Jones, P. E., & Roelofsma, P. (2000). The potential for social contextual and group biases in team decision-making: biases, conditions and psychological. *Ergonomics*, 43(8), 1129–1152.
- Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P. (1983). Optimization by simulated annealing. *Science*, 220(4598), 671–680. <http://dx.doi.org/10.2307/1690046>.
- Kotovsky, K., Hayes, J., & Simon, H. (1985). Why are some problems hard? Evidence from Tower of Hanoi. *Cognitive Psychology*, 17(2), 248–294. [http://dx.doi.org/10.1016/0010-0285\(85\)90009-X](http://dx.doi.org/10.1016/0010-0285(85)90009-X).
- Kurtzberg, T. R., & Amabile, T. M. (2001). From Guilford to creative synergy: opening the black box of team-level creativity. *Creativity Research Journal*, 13(3–4), 285–294. http://dx.doi.org/10.1207/S15326934CRJ1334_06.
- Langley, P. (1985). Learning to search: from weak methods to domain-specific heuristics. *Cognitive Science*, 9(2), 217–260. http://dx.doi.org/10.1207/s15516709cog0902_2.
- Leech, N. L., & Onwuegbuzie, A. J. (2002). A call for greater use of nonparametric statistics. In *Annual Meeting of the MidSouth Educational Research Association* (pp. 1–24).
- McComb, C., Cagan, J., & Kotovsky, K. (2015). Studying Human Design Teams via Computational Teams of Simulated Annealing Agents. In *ASME 2015 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*. Boston, MA: ASME. <http://dx.doi.org/10.1115/DETC2015-46545>.
- McComb, C., Cagan, J., & Kotovsky, K. (2015). Rolling with the punches: an examination of team performance in a design task subject to drastic changes. *Design Studies*, 36(1), 99–121. <http://dx.doi.org/10.1016/j.destud.2014.10.001>.
- Newell, A., & Simon, H. (1972). *Human problem solving*. Prentice Hall.
- Nikander, J. B., Liikkanen, L. A., & Laakso, M. (2014). The preference effect in design concept evaluation. *Design Studies*, 35(5), 473–499. <http://dx.doi.org/10.1016/j.destud.2014.02.006>.
- Olson, J., Cagan, J., & Kotovsky, K. (2009). Unlocking organizational potential: a computational platform for investigating structural interdependence in design. *Journal of Mechanical Design*, 131, 031001-1–031001-13. <http://dx.doi.org/10.1115/1.3066501>.
- Osborn, A. F. (1957). *Applied imagination*. New York: Scriber.
- Paulus, P. B., Dzindolet, M. T., & Kohn, N. (2011). Collaborative creativity-group creativity and team innovation. In M. D. Mumford (Ed.), *Handbook of organizational creativity* (pp. 327–357). New York: Elsevier.
- Pretz, J. E., Naples, A. J., & Sternberg, R. J. (2003). Recognizing, defining, and representing problems. In J. E. Davidson, & R. J. Sternberg (Eds.), *The psychology of problem solving*. Cambridge University Press.

- Sayama, H., Farrell, D. L., & Dionne, S. D. (2011). The effects of mental model formation on group decision making: an agent-based simulation. *Complexity*, 16(3), 49–57. <http://dx.doi.org/10.1002/cplx.20329>.
- Simon, H. A. (1956). Rational choice and the structure of the environment. *Psychological Review*, 63(2), 129–138.
- Singh, V., Dong, A., & Gero, J. S. (2012). Computational studies to understand the role of social learning in team familiarity and its effects on team performance. *CoDesign*, 8(1), 25–41. <http://dx.doi.org/10.1080/15710882.2011.633088>.
- Singh, V., Dong, A., & Gero, J. S. (2013). Social learning in design teams: the importance of direct and indirect communications. *Artificial Intelligence in Engineering, Design, Analysis and Manufacturing*, 27(2), 167–182. <http://dx.doi.org/10.1017/S0890060413000061>.
- Sio, U. N., Kotovsky, K., & Cagan, J. (2014). Analyzing the effect of team structure on team performance: an experimental and computational approach. In P. Bello, M. Guarini, M. McShane, & B. Scassellati (Eds.), *36th Annual Conference of the Cognitive Science Society* (pp. 1437–1442). Austin, TX: Cognitive Science Society.
- Stempfle, J., & Badke-schaub, P. (2002). Thinking in design teams – an analysis of team communication. *Design Studies*, 23(5), 473–496. [http://dx.doi.org/10.1016/S0142-694X\(02\)00004-2](http://dx.doi.org/10.1016/S0142-694X(02)00004-2).
- Triki, E., Collette, Y., & Siarry, P. (2005). A theoretical study on the behavior of simulated annealing leading to a new cooling schedule. *European Journal of Operational Research*, 166(1), 77–92. <http://dx.doi.org/10.1016/j.ejor.2004.03.035>.
- Ullman, D. G., Dietterich, T. G., & Stauffer, L. A. (1988). A model of the mechanical design process based on empirical data. *Artificial Intelligence in Engineering, Design, Analysis and Manufacturing*, 2(1), 33–52. <http://dx.doi.org/10.1017/S0890060400000536>.
- Wood, M., Chen, P., Fu, K., Cagan, J., & Kotovsky, K. (2012). The role of design team interaction structure on individual and shared mental models. In *Design computing and cognition '12* (pp. 206–226).