

DETC2015-46545

STUDYING HUMAN DESIGN TEAMS VIA COMPUTATIONAL TEAMS OF SIMULATED ANNEALING AGENTS

Christopher McComb
Dept. of Mechanical Engineering
Carnegie Mellon University
Pittsburgh, PA, USA
ccm@cmu.edu

Jonathan Cagan
Dept. of Mechanical Engineering
Carnegie Mellon University
Pittsburgh, PA, USA
cagan@cmu.edu

Kenneth Kotovsky
Dept. of Psychology
Carnegie Mellon University
Pittsburgh, PA, USA
kotovsky@cmu.edu

ABSTRACT

Novel design methodologies are often evaluated through empirical studies involving human designers. However, such empirical studies can incur a high personnel cost. Further, it can be difficult to isolate the effects of specific team or individual characteristics. These limitations could be bypassed by employing a computational model of design teams. This work introduces the Cognitively-Inspired Simulated Annealing Teams (CISAT) modeling framework, an agent-based platform that provides a means for efficiently simulating human design teams. A number of empirically demonstrated cognitive phenomena are modeled within the platform, striking a balance between model simplicity and direct applicability to engineering design problems. This paper discusses the composition of the CISAT modeling framework and demonstrates how it can be used to simulate the performance of human design teams in a cognitive study. Results simulated with CISAT are compared directly to the results derived from human designers. Finally, the CISAT model is also used to investigate the characteristics that were most and least helpful to teams during the cognitive study.

1. INTRODUCTION

Much cognitive research in engineering design has focused on individuals, despite the fact that most engineering design is actually performed by teams [1]. 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 to replace cognitive studies, but rather to augment traditional methods of investigation, accelerating the discovery of improved design methodologies.

When solving a problem, humans tend to learn strategies that can be expressed in terms of the move operators that apply to the problem [2,3]. 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 [4]. The selection of appropriate search strategies is further impacted by the presence of known goals or targets. It is known that individuals tend to satisfice, meaning that they only search the solution space until a solution that satisfies relevant targets is found [5]. 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 [6,7].

Teams are composed of individuals who strive towards a common goal [8]. When members of a team interact while working towards the goal, they perform better than individuals working alone [9,10]. Interaction is usually observed to occur organically, taking place at irregular intervals [11]. 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 [12,13]. However, premature convergence on a single solution can be detrimental to solution quality [14]. For that reason, designers are typically taught to explore multiple solution concepts [15,16]. Even

members of high-performing teams tend to pursue slightly different solution concepts when solving well-defined problems [17], 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 [18,19]. Designers have also been shown to preferentially evaluate their own solution concepts [20].

A significant amount of work has attempted to simulate the performance of both teams and individuals [21]. For instance, both the Virtual Design Team model, and another model applied to teams at NASA's Jet Propulsion Laboratory, incorporate detailed descriptions of design team organization and interaction [22,23]. Both models were used to simulate complex design tasks, but were also burdened by high model complexity. Still other work has utilized agent-based models to explore the formation of mental models during team problem-solving [24,25]. 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 model also explored the effect of team structure on team effectiveness and the formation of transactive memory [26]. 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 [27]. Regarding the simulation of individuals, simulated annealing [28], a stochastic optimization algorithm, has been used as an effective model for the efforts of individual human problem-solvers [29]. 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 [30]. There, 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.

This work introduces the Cognitively-Inspired Simulated Annealing Teams (CISAT) modeling framework. The CISAT framework 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 [17]. 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.

2. 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 of design teams that are articulated in the literature. 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 [8].
2. **Organic interaction timing:** Interaction within teams occurs at irregular intervals [11].
3. **Quality-informed solution sharing:** Members of a team tend to focus on the most promising alternatives,

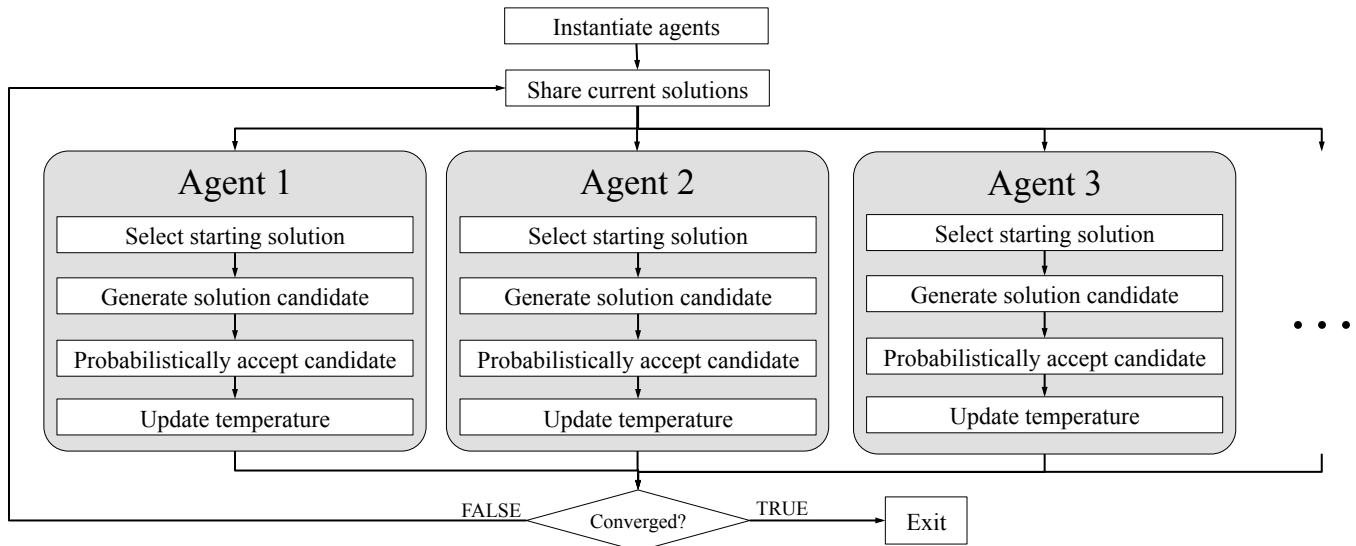


FIGURE 1. CONCEPTUAL FLOWCHART FOR CISAT FRAMEWORK

but don't do so greedily [17].

4. **Quality bias reduction:** Teams develop multiple solution concepts to avoid premature convergence [14–16].
5. **Self-bias:** Designers tend to be biased in favor of their own designs [18,19].
6. **Operational learning:** Individuals learn strategies over the course of solving [3].
7. **Locally Sensitive Search:** Designers select from a range of breadth- and depth-first search strategies as they explore the design space [4].
8. **Satisficing:** Individuals only search the solution space until a solution is found that satisfies relevant targets [5].

In the following sections, all randomized choices from a discrete set of alternatives are treated as random draws from a multinomial distribution (which can be thought of as the roll of a weighted die). All random choices that involve selecting a value from within some range are treated as random draws from a uniform distribution (equal probability for all values in the range).

2.1. 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 [31]. 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 [8].

2.2. Organic Interaction Timing

The amount of inter-member communication varies between teams, and occurs at irregular intervals [11]. 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.

2.3. 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 [9]. This arises from the ability of a team to explore a variety of options, but also focus the attention of the team members on a shrinking set of the most promising alternatives [12,13]. Even members of high-performing teams tend to pursue slightly different solution concepts while solving well-defined problems [17], 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, $\mathbf{W}$, which is defined as:

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

The vector $\mathbf{F}$ 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, $\mathbf{W}$, 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.

2.4. Quality Bias Reduction

Note that in the weighting vector $\mathbf{W}$, 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 [14]. Although novice designers are often taught to explore multiple solution concepts [15,16], expert designers do not generally exhibit this behavior [32]. 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 $\mathbf{W}$:

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

The variable k_{QBR} (chosen from the range [0, 1]) controls the strength of quality bias reduction, and $\mathbf{1}$ 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 are free to pursue any solution concept.

2.5. Self-bias

Designers tend to be biased in favor of designs that they have generated or spent substantial time working on [18,19]. 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 $\mathbf{W}$ that corresponds to its own design:

$$\mathbf{W}_i \leftarrow \mathbf{W}_i + k_{SB} \quad (3)$$

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.

2.6. Operational Learning

The actions that can be used to modify a solution are typically referred to as *move operators* and are inherently problem specific [2]. Human problem-solvers tend to learn strategies in terms of these move operators [3]. CISAT agents are also provided with a mechanism to learn which move operators are most helpful to design.

With 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 gave the solution a worse objective function value, the probability of applying the move operator is decreased.

2.7. Locally Sensitive Search

It is known that expert designers tend to use a mixture of depth- and breadth-first solution strategies [4], indicating the value of tailoring search strategies to local characteristics of the design space. CISAT is based on the 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 [33]. 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.

2.8. Satisficing

Decision makers tend to satisfice, meaning that they only search the solution space broadly until a solution that satisfies relevant targets is found [5]. Further, engineers and designers tend to have access to such goals [6,7]. 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.

3. 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. [17]. 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.

3.1. 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-minute 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.

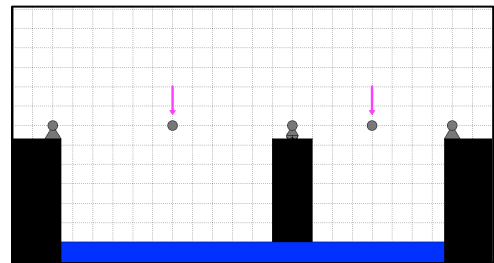


FIGURE 2. DIAGRAM OF PROBLEM STATEMENTS 1 AND 2

After working on PS1 for three of the 4-minute 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.

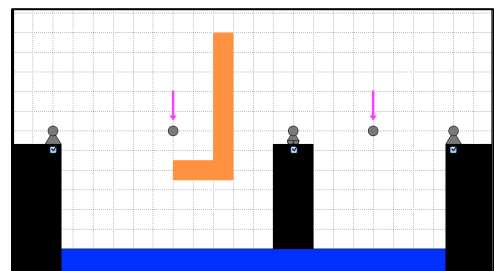


FIGURE 3. DIAGRAM OF PROBLEM STATEMENT 3

After working on PS2 for one 4-minute 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-minute 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.

3.2. 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

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

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 [34]. 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 [17].

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 [17]. This is similar to the concept of average pairwise similarity used in other work [9,13,35].

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-minute 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 [17]. 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.

3.3. 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-minute design sessions (1440 seconds), the average individual applied 480 moves. Every CISAT agent was also allowed this number of moves.

3.4. 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.

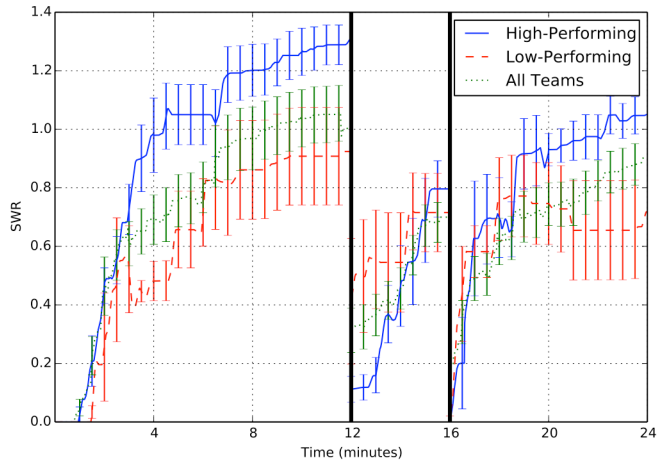
The CISAT framework reproduces several of the main trends that are apparent in the original results from the cognitive study. For instance, the high-performing human teams showed an early divergent period, followed by a pattern of fairly constant average pairwise distance (see Figure 4(c)). This is echoed 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 topology operations, followed by an increase after the introduction of the new problem statements.

The Pearson correlation coefficient (PCC) 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.

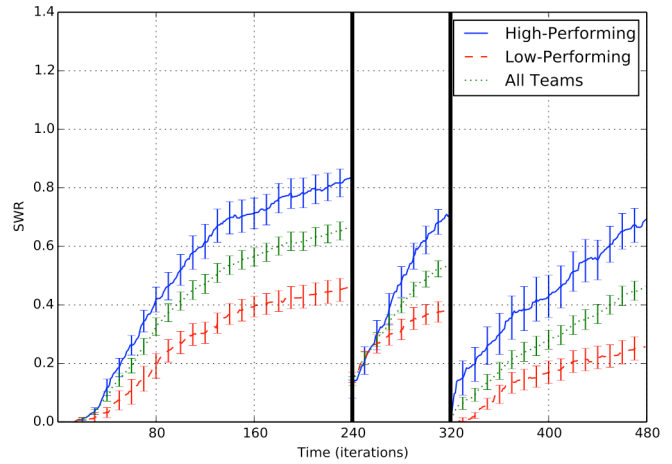
TABLE 1. SUMMARY OF HUMAN VERSUS MODEL PEARSON CORRELATION COEFFICIENTS

Metric	High-Performing Teams	Low-Performing Teams	All Teams
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 ($\rho < 10^{-5}$)
Freq. of Topology Operations	0.951 ($\rho < 0.005$)	0.890 ($\rho < 0.05$)	0.936 ($\rho < 0.01$)

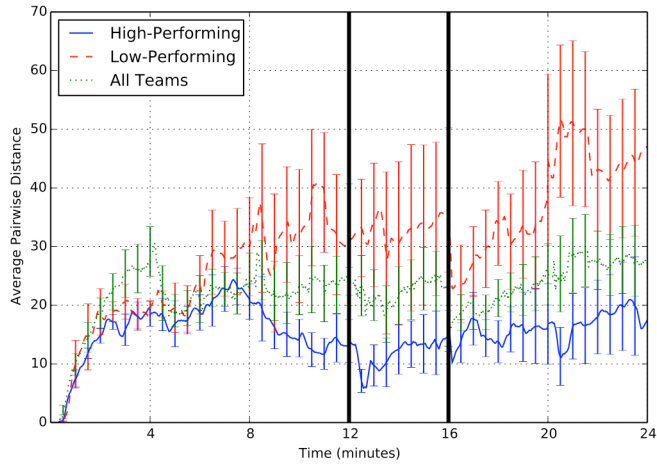
The coefficients provided in Table 1 are always above 0.65, and the majority of them are above 0.85. This quantitatively reiterates a fact that is already qualitatively evident in Figure 4. Although CISAT does not perfectly reproduce the results of the cognitive study, there is a strong positive correlation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



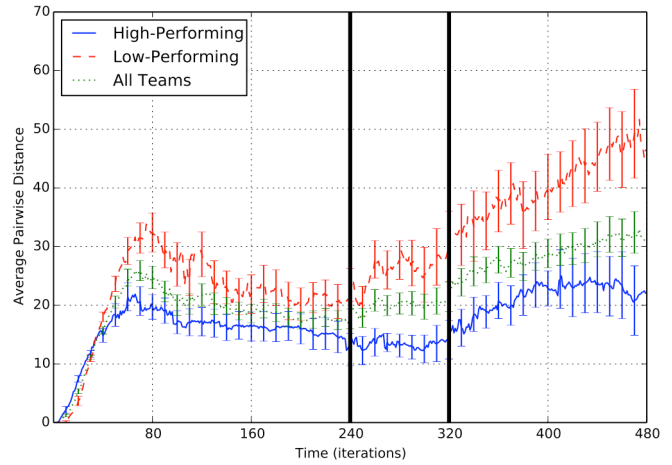
**(a) SWR of Best Design
(Cognitive Study Results)**



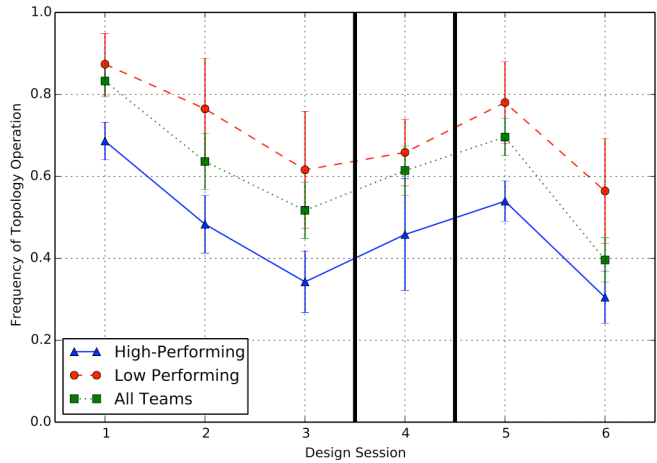
**(b) SWR of Best Design
(CISAT Simulation Results)**



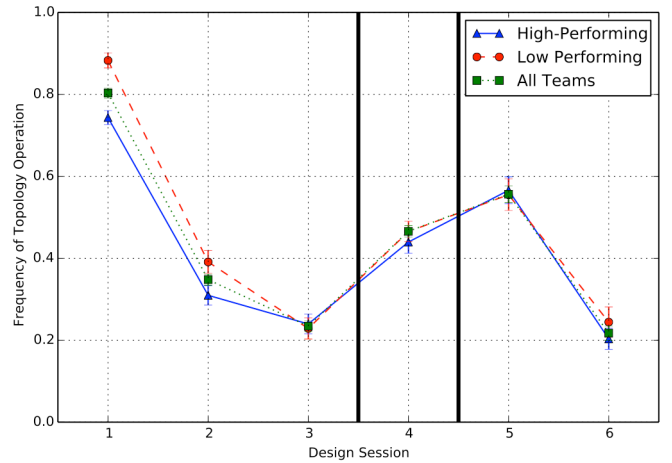
**(c) Average Pairwise Distance
(Cognitive Study Results)**



**(d) Average Pairwise Distance
(CISAT Simulation Results)**



**(e) Frequency of Topology Operations
(Cognitive Study Results)**



**(f) Frequency of Topology Operations
(CISAT Simulation Results)**

**FIGURE 4. COMPARISON BETWEEN COGNITIVE STUDY RESULTS AND CISAT SIMULATION RESULTS
(ALL ERROR BARS SHOW ± 1 S.E.)**

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 [36], but has also been observed in individuals with little experience [37]. 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.

4. 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 with and without a given feature, 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 the team. 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 solution sharing are not evaluated, since they 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 the results because it is more representative of central tendency than the mean. In addition, the non-parametric standardized effect size is reported through Cliff's δ in Figure 6. Cliff's δ measures the extent to which two distributions do not overlap, and is more robust to non-normality than Cohen's d [38].

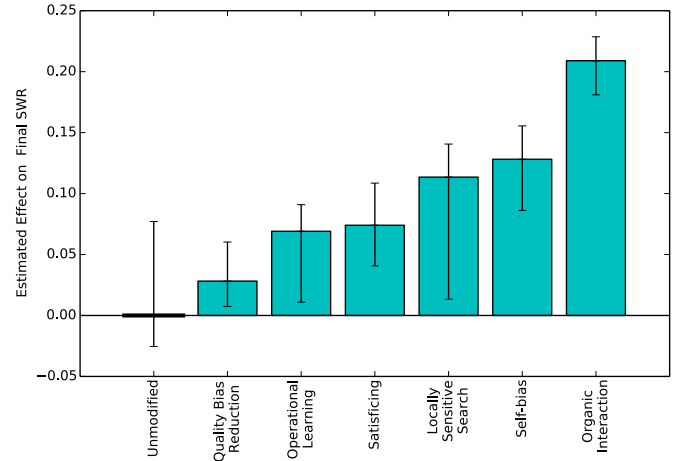


FIGURE 5. ESTIMATED EFFECT ON FINAL SWR (ERROR BARS SHOW ± 1 S.E.)

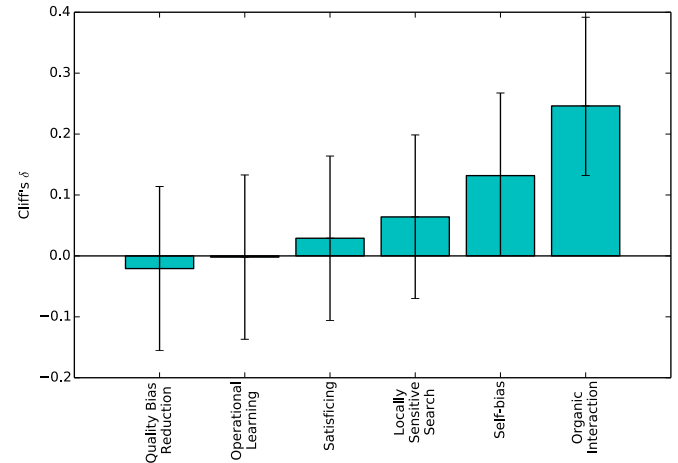


FIGURE 6. CLIFF'S δ STANDARDIZED EFFECT SIZE (ERROR BARS SHOW 90% CONFIDENCE INTERVALS)

As discussed in previous sections of this paper, each of the characteristics analyzed in Figures 5 and 6 has been observed in humans. The two characteristics with the largest and most significant effects (self-bias and organic interaction) 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.

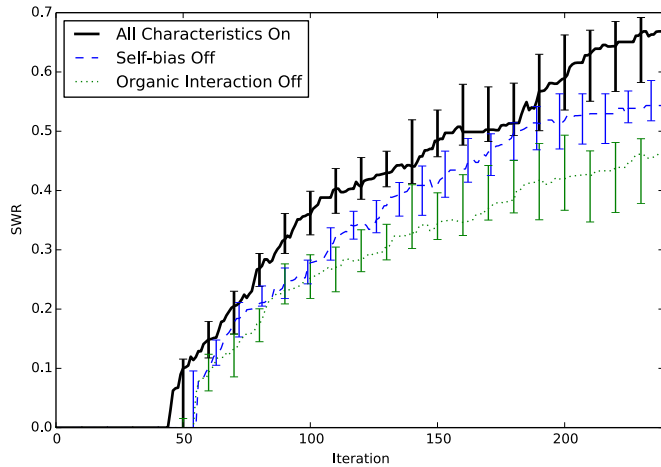


FIGURE 7. SWR OF BEST DESIGN (ERROR BARS SHOW ± 1 S.E.)

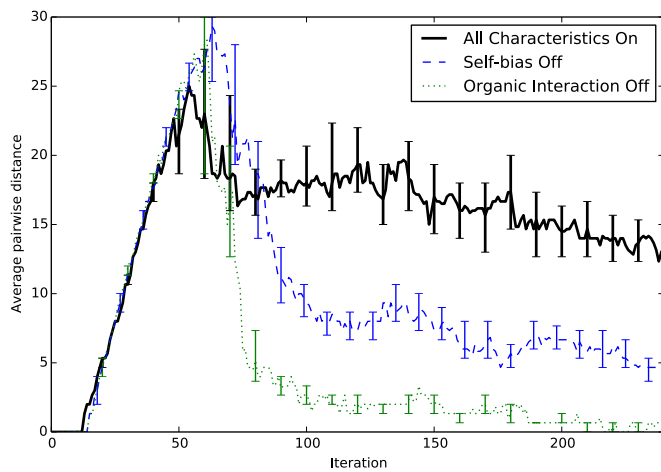


FIGURE 8. AVERAGE PAIRWISE DISTANCE (ERROR BARS SHOW ± 1 S.E.)

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 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.

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 [9–11]. 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 absence of either one of these characteristics could lead to

a state in which the creation of consensus (low divergence) becomes a driving factor in the design process. 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 [39]. It has been theorized that groupthink may detrimentally affect decision-making teams [40].

The above analysis compared the median values of several groups of teams, where all teams within a group had the same active characteristics (either all characteristics on, self-bias turned off, or organic interaction turned off). Now, trends *within* groups of teams that have the same active characteristics will be examined by computing the correlation coefficient between final average pairwise distance and final SWR. For teams simulated with all characteristics turned on, the Pearson correlation coefficient between these two variables is -0.446. For teams simulated without self-bias, the correlation coefficient is -0.481, and for teams simulated without 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 with the same active characteristics, a team that shows low final divergence is likely to produce a high quality solution.

A similar trend was identified when the truss design problem was solved by human designers [17]. 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 [32]. The fact that expert solutions 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 (variable quality of initial representation) may have given rise to a similar trend (negative correlation between divergence and quality) in both human- and agent-based studies.

5. 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.

Only cognitive phenomena that have been demonstrated within the domains of design or problem-solving were modeled in this work. 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. CISAT may also be useful for simulating larger teams with more complex, hierarchical structures. For instance, an organization composed of multiple sub-teams could be modeled as a conglomeration of CISAT teams, each working on a specific sub-task and communicating parameter values through an inter-team protocol. Further work would likely be necessary to extend CISAT for such simulations, specifically with respect to coordinating work between teams.

Future work will seek to implement more detailed models for learning and heuristic development within the CISAT framework. The CISAT framework will also be used to model additional 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 and the United States Air Force Office of Scientific Research through grant FA9550-12-1-0374.

REFERENCES

- [1] Paulus, P. B., Dzindolet, M. T., and Kohn, N., 2011, "Collaborative creativity-Group creativity and team innovation," *Handbook of organizational creativity*, M.D. Mumford, ed., Elsevier, New York, pp. 327–357.
- [2] Newell, A., and Simon, H., 1972, *Human Problem Solving*, Prentice Hall.
- [3] Langley, P., 1985, "Learning to Search: From Weak Methods to Domain-Specific Heuristics," *Cogn. Sci.*, **9**(2), pp. 217–260.
- [4] Ball, L. J., and Ormerod, T. C., 1995, "Structured and opportunistic processing in design: a critical discussion," *Int. J. Hum. Comput. Stud.*, **43**(1), pp. 131–151.
- [5] Simon, H. A., 1956, "Rational Choice and the Structure of the Environment," *Psychol. Rev.*, **63**(2), pp. 129–138.
- [6] Cooper, R., and Slagmulder, R., 1997, *Target Costing and Value Engineering*, Productivity Press, Portland, OR.
- [7] Feil, P., Yook, K., and Kim, I., 2004, "Japanese target costing: a historical perspective," *Int. J. Strateg. Cost Manag.*, pp. 10–19.
- [8] Kurtzberg, T. R., and Amabile, T. M., 2001, "From Guilford to Creative Synergy: Opening the Black Box of Team-Level Creativity," *Creat. Res. J.*, **13**(3-4), pp. 285–294.
- [9] Wood, M., Chen, P., Fu, K., Cagan, J., and Kotovsky, K., 2012, "The Role of Design Team Interaction Structure on Individual and Shared Mental Models," *Design Computing and Cognition '12*, pp. 206–226.
- [10] Sio, U. N., Kotovsky, K., and Cagan, J., 2014, "Analyzing the Effect of Team Structure on Team Performance: An Experimental and Computational Approach," 36th Annual Conference of the Cognitive Science Society, P. Bello, M. Guarini, M. McShane, and B. Scassellati, eds., Cognitive Science Society, Austin, TX, pp. 1437–1442.
- [11] Stempfle, J., and Badke-schaub, P., 2002, "Thinking in design teams - an analysis of team communication," *Des. Stud.*, **23**(5), pp. 473–496.
- [12] Dong, A., Hill, A. W., and Agogino, A. M., 2004, "A Document Analysis Method for Characterizing Design Team Performance," *J. Mech. Des.*, **126**(3), pp. 378–385.
- [13] Fu, K., Cagan, J., and Kotovsky, K., 2010, "Design Team Convergence: The Influence of Example Solution Quality," *J. Mech. Des.*, **132**(11), p. 111005.
- [14] Jansson, D., and Smith, S., 1991, "Design Fixation," *Des. Stud.*, **12**(1), pp. 3–11.
- [15] Osborn, A. F., 1957, *Applied imagination.*, Scriber, New York.
- [16] Dieter, G., and Schmidt, L., 2009, *Engineering Design*, McGraw-Hill.
- [17] McComb, C., Cagan, J., and Kotovsky, K., 2015, "Rolling with the punches: An examination of team performance in a design task subject to drastic changes," *Des. Stud.*, **36**(1), pp. 99–121.
- [18] Ullman, D. G., Dieterich, T. G., and Stauffer, L. A., 1988, "A model of the mechanical design process based on empirical data," *Artif. Intell. Eng. Des. Anal. Manuf.*, **2**(1), pp. 33–52.
- [19] Ball, L. J., Evans, J. S. B., and Dennis, I., 1994, "Cognitive Processes in Engineering Design: a longitudinal study," *Ergonomics*, **37**(11), pp. 1753–1786.
- [20] Nikander, J. B., Liikkanen, L. a., and Laakso, M., 2014, "The preference effect in design concept evaluation," *Des. Stud.*, **35**(5), pp. 473–499.
- [21] Fan, X., and Yen, J., 2004, "Modeling and simulating human teamwork behaviors using intelligent agents," *Phys. Life Rev.*, **1**(3), pp. 173–201.
- [22] Jin, Y., and Levit, R., 1996, "The Virtual Design Team:

- A Computational Model of Project Organization,” *Comput. Math. Organ. Theory*, **2**(3), pp. 171–196.
- [23] Olson, J., Cagan, J., and Kotovsky, K., 2009, “Unlocking Organizational Potential: A Computational Platform for Investigating Structural Interdependence in Design,” *J. Mech. Des.*, **131**, pp. 031001–1–13.
- [24] Dionne, S. D., Sayama, H., Hao, C., and James, B., 2010, “The role of leadership in shared mental model convergence and team performance improvement: An agent-based computational model,” *Leadersh. Q.*, **21**(6), pp. 1035–1049.
- [25] Sayama, H., Farrell, D. L., and Dionne, S. D., 2011, “The Effects of Mental Model Formation on Group Decision Making: An Agent-Based Simulation,” *Complexity*, **16**(3), pp. 49–57.
- [26] Singh, V., Dong, A., and Gero, J. S., 2013, “Social learning in design teams: The importance of direct and indirect communications,” *Artif. Intell. Eng. Des. Anal. Manuf.*, **27**(2), pp. 167–182.
- [27] Crowder, R. M., Robinson, M. A., Hughes, H. P. N., and Sim, Y.-W., 2012, “The Development of an Agent-Based Modeling Framework for Simulating Engineering Team Work,” *IEEE Trans. Syst. Man, Cybern. - Part A Syst. Humans*, **42**(6), pp. 1425–1439.
- [28] Kirkpatrick, S., Gelatt, C. D., and Vecchi, M. P., 1983, “Optimization by simulated annealing,” *Science*, **220**(4598), pp. 671–680.
- [29] Cagan, J., and Kotovsky, K., 1997, “Simulated Annealing and the generation of the objective function: a model of learning during problem solving,” *Comput. Intell.*, **13**(4), pp. 534–581.
- [30] Egan, P. F., Cagan, J., Schunn, C., and Leduc, P. R., 2014, “Cognitive-based search strategies for complex bio-nanotechnology design derived through symbiotic human and agent-based approaches,” *ASME 2014 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, pp. DETC2014–34714.
- [31] Franklin, S., and Graesser, A., 1997, “Is it an Agent, or just a Program?: A Taxonomy for Autonomous Agents,” *Intelligent Agents III: Agent theories, architectures, and languages.*, pp. 21–35.
- [32] Cross, N., 2004, “Expertise in design: an overview,” *Des. Stud.*, **25**(5), pp. 427–441.
- [33] Triki, E., Collette, Y., and Siarry, P., 2005, “A theoretical study on the behavior of simulated annealing leading to a new cooling schedule,” *Eur. J. Oper. Res.*, **166**(1), pp. 77–92.
- [34] Hibbeler, R., 2008, *Structural Analysis*, Prentice Hall.
- [35] Wood, M., 2013, “Problem Representation and Team Mental Model Development in Individual and Team Problem Solving Performance,” *Carnegie Mellon University*.
- [36] Pretz, J. E., Naples, A. J., and Sternberg, R. J., 2003, “Recognizing, Defining, and Representing Problems,” *The Psychology of Problem Solving*, J.E. Davidson, and R.J. Sternberg, eds., Cambridge University Press.
- [37] Kotovsky, K., Hayes, J. ., and Simon, H. ., 1985, “Why are some problems hard? Evidence from Tower of Hanoi,” *Cogn. Psychol.*, **17**(2), pp. 248–294.
- [38] Hess, M., and Kromrey, J., 2004, “Robust confidence intervals for effect sizes: A comparative study of Cohen’sd and Cliff’s delta under non-normality and heterogeneous variances,” *Annu. Meet. Am. Educ. ...*, pp. 1–30.
- [39] Janis, I., 1971, “Groupthink,” *Psychol. Today*, **5**(6), pp. 43–46.
- [40] Jones, P. E., and Roelofsma, P., 2000, “The potential for social contextual and group biases in team decision-making: biases, conditions and psychological,” *Ergonomics*, **43**(8), pp. 1129–1152.