Modelling the Dynamics of Influence on Individual Thinking During Idea Generation in Co-design Teams

Harshika Singh, Christopher McComb, and Gaetano Cascini

Abstract Social influence is not evenly distributed in teams. Some individuals, referred to here as influencers, become more influential than others. Consequentially, these influencers play a significant role in shaping project performance. The current work simulates the presence of influencers during idea generation in co-design teams to better understand emergent socio-cognitive phenomena. Besides providing, a novel approach for modelling learning in concept generation the model highlights the results related to individual cognition during idea generation. Idea quality and exploration of design space are affected by the presence of influencers in design teams. Teams with no well-defined influencers produce solutions with high general exploration but less quality. In contrast, the agents in the teams with only one influencer produce solutions high quality than those teams with no influencers.

1 Introduction

Co-design is beneficial for innovation because it employs teams of individuals with different skills and ideas [1]. More specifically, co-design is a process used for creating products and services by involving different viewpoints and stakeholders. Implementing co-design has become a trend over traditional practices of design [1]. Research has been conducted to study the factors that hinder or facilitate these collaborative team activities [2]. This has been done either by studying the design outcome or the design process. Currently, more emphasis is being given to explore the factors from individual to project level [2], which influence design process, and consecutively affect final performance. Moreover, examining the design process is a potentially time-consuming practice [3]. Therefore, this research aims at providing a

H. Singh (✉) · G. Cascini
Politecnico di Milano, Milan, Italy
e-mail: harshika.singh@polimi.it

C. McComb
Carnegie Mellon University, Pittsburgh, USA

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2022

J. S. Gero (ed.), *Design Computing and Cognition'20*,
https://doi.org/10.1007/978-3-030-90625-2_3

quicker approach to study co-design team dynamics while keeping in mind the relevant socio-cognitive features.

The work presented in this paper deals with an agent-based model approach for simulating idea-generation in co-design teams. Besides providing a novel approach to model learning abilities in agents, it also investigates the effect of the influencers on individual thinking during idea generation. The effect of influencers during idea generation is potentially significant but has been unaccounted in the literature. The early results obtained in this work show how solution quality and exploration of the design space are affected by changing the number of influencers in a team during idea generation. The wider focus of the research work is to simulate the entire co-design process based on the framework provided by [4] for assisting future researchers and practitioners. Specifically, the model would provide them with a faster digital approach to understand how people interact in design teams by varying the input parameters to fulfil the purpose of their work.

The structure of the paper is as follows; the following section is about the past literature in this domain. It also identifies a research gap and highlights the main contribution of the paper. This section is followed by a detailed methodology on which the model is built and some early results of the simulation are mentioned. In the end, a conclusion provides a summary of the paper along with the limitations and the future goals.

2 Background

Agent-based modelling is a relatively new computational approach to model a dynamic phenomenon. It is a quicker, convenient approach for modelling individual behavior and heterogeneous interactions, where individuals (agents) exhibit characteristics such as memory, learning and adaptation [5]. These agents behave according to a set of rules assigned to them to fulfil the purpose of the model [5]. Agent-based modelling has been used in many different domains, from social sciences to medical fields [6]. Here, it is being used to model co-design team dynamics. Simulating a co-design activity involves many parameters [7] and it is unfeasible and often difficult to consider all the parameters in the model. Some authors have investigated problem-solving in design teams [8, 9], while others have proposed models based on team expertise, team experience on task performance [10, 11]. Individual attributes, such as the choice of partners or cognitive style [12, 13] and social attributes have also been modelled in the past [14].

Various former models were focused on simulating project-based design teams [10]; thus, a phenomenon of the same designers working on a certain task for a long period is still underexplored. The above-mentioned trend is most common in small and medium enterprises that have the same set of individuals working on similar problems for years. As a result, the model in this paper is based on this scenario where the same designers work on a complex routine design task as the solutions vary from person to person based on their competence [15].

The type of design task alters teams' performance; hence, it is crucial to define a task fitting the purpose of the model. Design tasks in some computational models mentioned above have been demonstrated as binary functions [16] or have been decomposed into sub-design tasks to serve their purpose [14]. Often these design task representations have extreme solution values, (i.e. immediately next to the best solution, there is the worst solution) which in real world is an inaccurate representation of a more stable design task with robust solutions i.e. they have less variations or a gradual slope (of intermediate values) between the best and the worst solution.

Agent-based modeling often involves imbuing agents with the ability to learn about their environment. Some researchers have modelled social learning by means of mental models [15] or learning from doing the task [13, 17]. While simulating learning, it is often assumed that agents know the design solution space and therefore pursue optimal solutions. However, in the proposed model the agents do not directly know any value of the solutions in the design space, but at the same time, they are aware of the design variables and the boundary conditions. This is similar to real situations where the design solution space is not completely known to the team of individuals working on a design problem, and they learn from their previous solution results and from others in the team. Learning from doing or experience is modelled by considering several individual characteristics (see the section on Model description). The rationale behind why and how learning from others in the team or social learning is explained below.

Whenever there is human interaction, there is social influence. Social influence may cause individuals to modify their opinions, attitudes and behavior to be similar to the others they are interacting with [18]. This imitation is embedded in human nature and is referred to here as social learning [19]. However; this social influence is not equally distributed in the team. Some individuals are more influential; the capacity to persuade others is not necessarily spread evenly among team members [20]. There may be some individuals who are regarded as more influential in their views and judgements than others [20]. In this work, these relatively more influential individuals are referred as influencers [21]. These influencers play a significant role in shaping the project performance. Many researchers have tried to study the traits, attitudes and behavior makes someone more influential than others. As proposed by Baker (2015), an individual's personality, skills and communication could result in such phenomenon. In this work, individuals' self-efficacy is taken as one of the traits that could affect their personality, skills and communication [22]. Self-efficacy, itself depends on intrinsic and extrinsic motivation [24]. Furthermore, how well the two individuals have known each other previously or the '*strength of ties*' between them [23], is one of the factors accountable for trust is also considered in this model for determining the degree of influence. Trust depends on the interacting individual's familiarity and reputation [25, 26] (see Fig. 9).

Social dynamics within a group can influence individual performance, hence affecting team performance [41]. Thus, the main significance of the paper lies in its contribution towards providing (1) a novel approach to simulate learning ability in agents when the solution space is unknown, and (2) simulating the 'effect of influencers' during idea generation. Overall, the research aims to improve the understanding of design teams and intervene in the team dynamics as necessary for better project outcomes.

The objective of the work presented in the paper is to assess the effect of the influencers on individual thinking during idea generation in design teams. Hence, based on this research objective, the paper describes a suitable methodology that forms the foundation of a simulation along with some results showing the functioning model.

3 Model Description

As mentioned above, the wider purpose of the research is to generate a computational model for an entire co-design activity, the framework of which can be seen from Fig. 1.

The co-design activity consists of a project on which teams work. Usually, a project consists of a number of design sessions of idea generation and selection before the final project outcome is reached. In each idea generation activity, an agent takes a certain number of steps (explores multiple solutions) before being ready with one which it shares with the team. The work presented in this paper only simulates idea generation sessions where individuals work on a robust design task without considering the gaps between the sessions. These gaps are the pauses or breaks between the two ideation sessions, essential for the incubation effect (not considered in the scope of this paper). The results stated are related to individual thinking during brainstorming in idea generation where agents at this point generate, explore, evaluate and select solutions [27]. The collective teamwork on idea selection is out of the scope of this paper.

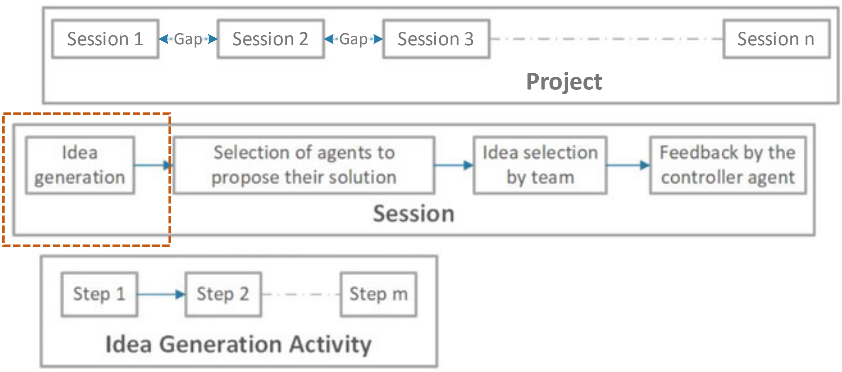


Fig. 1 The focus of the study presented in the paper

3.1 The Design Task

Like any co-design activity, the simulation starts with a design task. The controller agent (real-world equivalent to a project leader, project manager or a professor) gives the agents (individual humans who work in a team) a task and the agents have to find solutions to the design problem. In computational terms, the design problem is n-dimensional that consists of a landscape function *f(x)* (given below in Eq. 1) constructed from the pre-defined best solution points, and agents aim to find the best solutions. For initiation, simplification and visualization purposes, a 2D function is representing the design problem; however, it could be extended to multiple dimensions for the upcoming articles. The 2 dimensions in a design space represent two notional design variables to explore and the values perpendicular to the design variables define the quality of the solutions. The values of *f(x)* represent the solution space, which has a maximum value of 1 (lightest hue) and minimum 0 (darkest hue), as shown in an example in Fig. 2 with several local maxima and minima.

$$f(x) = \frac{1}{\left(1 + e^{\left(\left(\frac{1}{\sqrt{N}}\right)d - 2\right)}\right)} \tag{1}$$

where N is the number (size) given to represent the solution space in 2D matrix. In this case, N = 100, such that the solution space was represented as a 100-by-100 matrix. Here, d represents the distance between the random point *(x,y)* and the nearest best solutions.

While mathematically, representing the solution space it was taken into consideration, the real-life design problem-solving, where there is a low probability of having immediate extreme values. This means that there is a low chance of

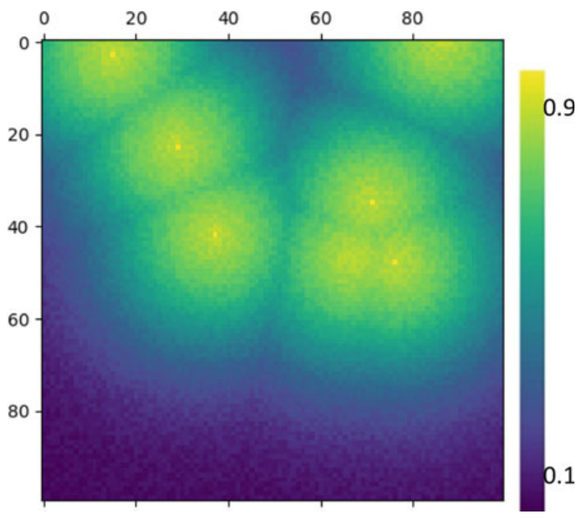


Fig. 2 An example of a design solution space with a side bar showing solution values

having the best and the worst solutions placed next to each other. The design problem resembles a case where its solutions have less variation among its immediate surrounding solutions. In other words, the agents are solving a design problem where in most cases the solution generated by them will not extremely vary from the values of its neighboring cells. Likewise, the solution space with this landscape function $f(x)$, is modelled such that there is a gradual decrease in the hue around the best solutions. Some noise is added to the design space so that the probability of having maximum and minimum solution values right next to each other is not completely eliminated. Like any other design problem, this design problem representation also has multiple best solutions.

3.2 Agent Generating Solutions

After the design task is given, the agents start generating solutions. Similar to brainstorming, the agents first individually generate solutions then communicate with the team to further build on them. The paper is only focusing on reporting the method and results related to the state of agents when they are individually thinking during ideation (while the entire process described in Fig. 1 is operating in the backend).

An agent generates solutions based on the characteristics given below. These are related to general human behavior related to thinking in idea generation. Certainly, there are many cognitive and social factors influencing individual brainstorming that are complicated to mimic, however, as identified in a review of the literature, these are most salient.

- Their way to explore solution space
- Memory to store past experiences
- Recall capability
- Ability to learn from failure and successful past experience
- Influence of the influencer(s) (as explained in the Background section above)

3.3 Agent's Desire to Explore Solution Space

Exploration of the design solution space is based on the fact that individuals during the initial ideation phase are slower in exploring the solutions as they get warmed up in by triggering memory search. This is followed by more exploration by recalling input from their memory. However, at some point, this recalling process becomes tiring, and exploring the solution space drops towards the end of the session [28] (see Fig. 3). This behavior of individuals is modelled in the agents as in Eq. 2 below.

Changing the shape parameter of the curve (σ), makes it possible to generate different exploration styles, assignable to different agents.

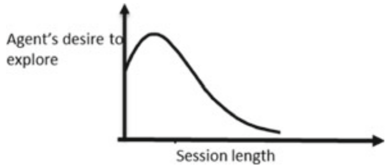


Fig. 3 An example plot of the curve O(x′)

$$O(x') = \frac{1}{x'\sigma\sqrt{2\pi}} e^{\left(-\frac{ln(x')}{2\sigma^2}\right)} + c \tag{2}$$

In the given equation, x' is $x/7$ ($x \geq 0$). The value of σ lies between $0 < \sigma \leq 1$, it represents the shape parameter which affects the overall shape of the curve. c is the value of exploration when the session starts and it varies from agent to agent.

3.4 Memory

The simulation setup as defined at the beginning of the paper imitates situations found in many small and medium enterprises, where the agents work on a similar design problem for many sessions. Accordingly, an agent has a different memory capacity and stores results from the sessions in the past experience from working on these design problems. These experiences are in the form of success or failure encountered in the past when doing a design task. From the memory of an agent, the stored element is forgotten when it is not recalled for a long time. This forgetting is based on the Decay Theory of Forgetting where it suggests that if there was no attempt to recall an event, the greater the time since the event the more likely it would be to forget the event. Thus, the agents exhibit the behaviour that suggests that memories are not permanent [29].

3.5 Recall Capability

Recalling, on the other hand, refers to the act of bringing a past event back to one's mind. In real situations, an individual might not be able to recall any similar experience from the past while approaching a problem. Similarly, in the model, an agent has its successes and failures in its memory but it might not be able to recall while solving the problem. In addition, if an agent recalls events from the past that might alter the way it approaches the solution. The model takes into account free recall, where individuals can recall events in any order [30]. The recalling power is different from agent to agent and depends on the intensity of the solution value and the time of recall as explained by [31]. In the real world case, where individuals recall their worst and best events results more clearly than their mediocre outcomes.

The events are also recalled based on the recency and primacy effect [30] which means that the recent and the first events are recalled more frequently than the middle ones. In real world, the mediocre results of events that happened a long time ago are more difficult to recall than recalling the best/worst result that happened at the same time. Moreover, recalling the recent best/worst events is easier than past events with similar results. Hence, the model is based on similar features where agents have a higher probability to recall their recent best/worst solutions than old ones.

3.6 *Learning from Experience*

The agents in the model have the capability to learn from their past success or failure events, which had occurred in the previous sessions, it could be seen from the example shown in Fig. 4. Learning from past success and failure are different as they have a different impact on the current situation. This implementation of learning from the success of an agent on its current solution depends on the following factors (Fig. 5):

1. Similarity between the current solution 'in mind' and recalled successful solution. If the recalled success is similar (closer on solution space) to the solution 'in mind', the agent is more influenced by its previous success than the success that is far in distance (not so similar) [32]. However, if the recalled success is too close or same as the solution 'in mind', an agent is less influenced by it.

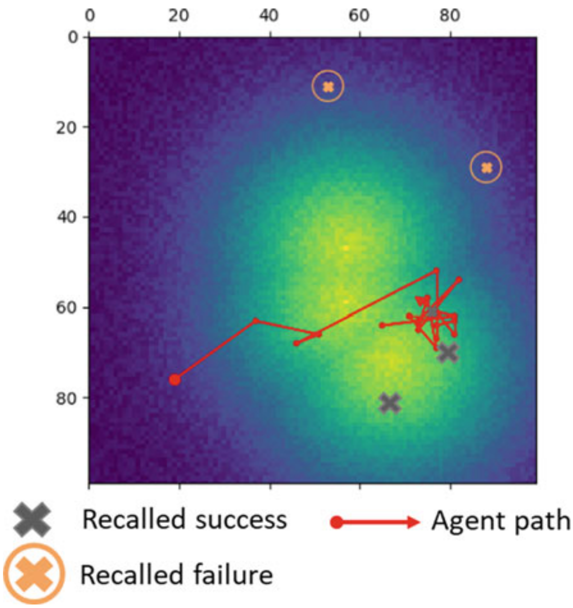


Fig. 4 An example showing an agent recalling events while exploring solutions

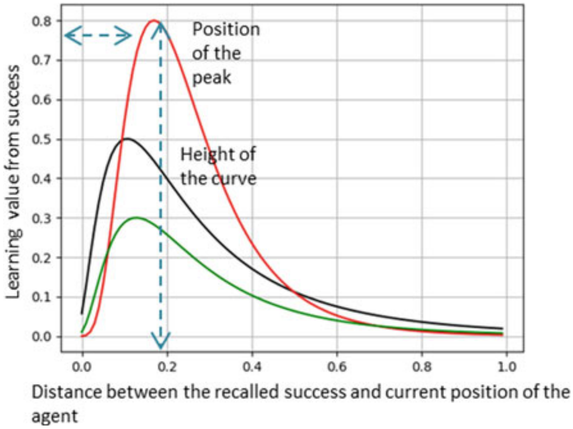


Fig. 5 Different amount of learning from one's own success

2. The amount of learning the previous success depends on the (i) experience of an agent, i.e. the position of the peak of the learning curve. It means that when an agent is more experienced, it will learn faster therefore a steeper slope than the agent who has lesser experience [33]. (ii) The time when the successful event occurred, i.e. the height of the learning curve from success i.e. more height/ intensity when the success was recent.

The amount of learning from success recalled (magnitude of the success vector as shown in Fig. 6) can be represented by *S(d')* in Eq. 3.

$$S(d') = \tau \left(\frac{\frac{1}{d'\alpha\sqrt{2\pi}} e^{\left(-\frac{(\ln(d'))}{2\alpha^2}\right)}}{0.7} \right) \tag{3}$$

$d' = 4.0 \times d + 0.1$. Here d' is the adjusted value of d such that $0 \leq S(d') \leq 1$.

d is the similarity between the current design task and recalled success experienced task. In computational terms, *d is the* distance between the current agent (solution) position in session n and recalled success (solution) position of session S_n. (In Eq. 3 *S(d')* is divided by 0.7 to get the desired value within 0–1).

The other variables in the above equation are explained below:

$$\alpha = 0.8 - (0.2 \times \text{agent's experience } (E)) \tag{3.1}$$

$$\tau = 1 - (0.7 \times \Delta t) \tag{3.2}$$

$$\Delta t = \mathrm{n} - S_n/N, \tag{3.3}$$

where n is the current session number of an agent and S_n is the session when the recalled success occurred. N = number of sessions.

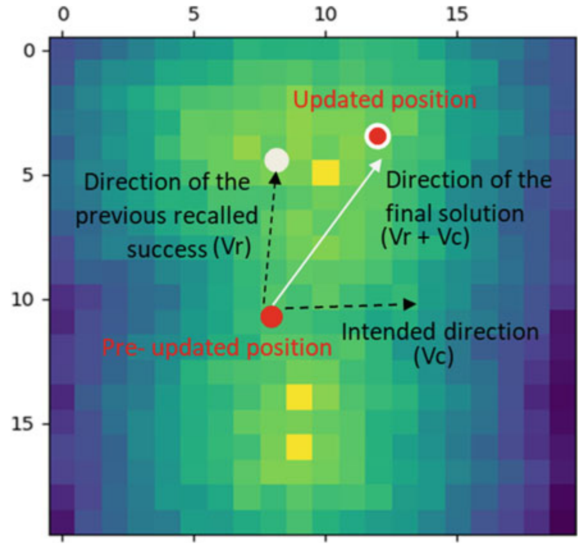


Fig. 6 The updated position on an agent is the sum of the vectors of the intended direction and the direction of the recalled success (20 × 20 Solution space only for the zoomed in visualization)

Success and failure have a different impact on the current situation. For example, humans try to avoid the failures they have committed in the past and tend to follow the path that led to previous success [34]. Unlike learning from success, learning from failure is done by forming circles of a certain radius r around the failed solution value. The circle is constructed by the agent around the failed solution whose radius varies from agent to agent. Similar to the real scenario where an individual remembers the failure zones on the solution space while exploring new solutions. The learning from failure depends on the recalled failure value (Fig. 7), where an agent learns maximum when the failure was severe. The radius or the size of the circle denotes the learning capacity from a failure of an agent and it will avoid the circle area around the recalled failure (Fig. 8).

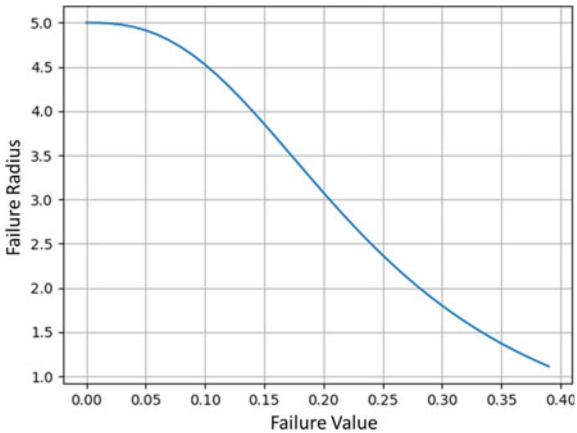


Fig. 7 Failure radius depends on the value of the recalled failure (where 5 units is the max radius for a 100 × 100 units of solution space)

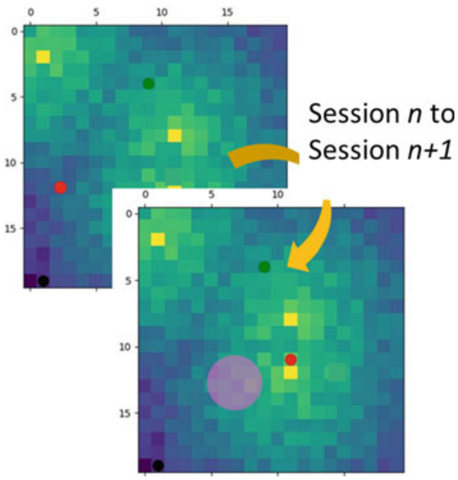


Fig. 8 An example where an agent (in red) encounters a failure at session n which is being recalled in session n + 1, an area around the failure is avoided. (20 × 20 Solution space only for the zoomed-in visualization)

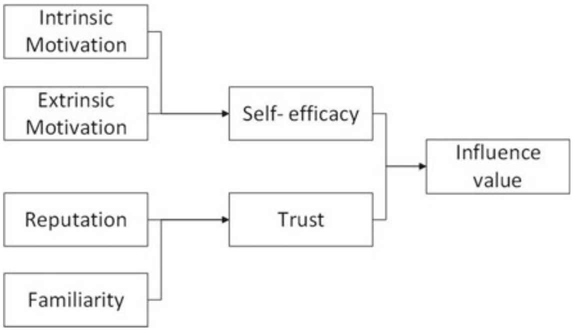


Fig. 9 Determining influence value

3.7 *Effect of the Influencers*

As mentioned in the previous section, some individual(s) in teams are regarded more influential in their views and judgements than others [20]. Therefore, in order to model this 'influencing effect', each agent has an influencing value from other agents in the team and it depends on the factors shown in Fig. 9. The influence value I (magnitude of the vector), for an agent i (in an example in Fig. 10 as agent A4) of agent j (as agent A2) is computed as Eq. 4:

$$I_i^j(\Delta SE, T) = \frac{(\Delta SE^{1.5} + T_i^j)}{2} \tag{4}$$

where, ΔSE = difference in self-efficacy of agent i and agent j, T is the trust of agent i on agent j.

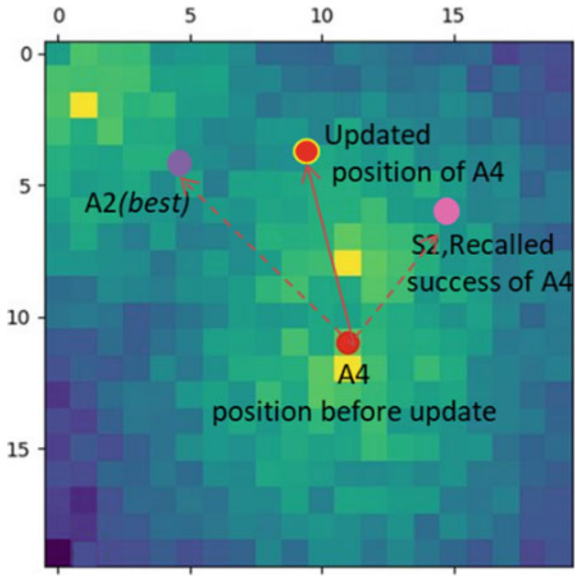


Fig. 10 The updated position on an agent is the sum of the vectors of the intended direction, the direction of its own recalled success and the influence value vector (20X20 Solution space only for the zoomed-in visualization)

$$T(R,F)_i^j = (R^j + F_i^j)/2 \tag{4.1}$$

Trust, T of an agent i on agent j depends on R and F [25]. R is the reputation of an agent j and F is the familiarity of an agent i with agent j. Familiarity, F between two agents, is the number of session agents i and j have worked together, therefore familiar with each other.

$$R = N_a/N_p \tag{4.2}$$

where, N_a is the number of solutions that are accepted by the controller agent and N_p is the total number of the solutions proposed by an agent.

The increase in extrinsic motivation occurs when the controller agent accepts the final solution of the team. An increase in intrinsic and extrinsic motivation increases self-efficacy [24]. Other studies state that individuals contribute more in idea generation when the team accepts ideas from them [35]. This is due to their increase in intrinsic motivation that increases their self-efficacy. It was also found that individuals with high self-efficacy are less likely to be demotivated when other team members do not select their ideas [35]. Taking these findings into account, the agents in the model demonstrates similar behavior, where an increase/decrease in an agent self-efficacy depends on their current state of self-efficacy. Agents who have the highest and the lowest self-efficacy get a gradual boost in their existing state of self-efficacy than the agents having a moderate amount of self-efficacy. Moreover, highly motivated agents have a more steady decrease in their self-efficacy than the lowest self-efficacy agents who are more rapidly demotivated.

4 Methodology

In order to address the research objective, which is to find out the effect of the influencers on individuals' idea generation, the team self-efficacy distribution was varied while keeping other parameters (team size, team familiarity and design task) constant. To check the functionality of the model, two scenarios were framed and tested. The first scenario tested the situation when the team has high variance in the self-efficacy of its agents, i.e. some agents have high self-efficacy and others low when they start working on a design task. Three sub-scenarios here were:

1. One agent with high self-efficacy and others with low
2. Two agents with high self-efficacies and others with low
3. Three agents with high self-efficacies and others with low

The second scenario tested the situation when the team has low variance in the self-efficacy of its agents, i.e. all agents either have high or low self-efficacy when they start working on a design task. Two sub-scenarios here were:

1. All with low self-efficacies (i.e. no influencer)
2. All with high self-efficacies (i.e. all influencers)

These two scenarios would help in understanding the team dynamics that affect design output due to the presence of unequal social influence. Thus, in order to see the functionality of the model, the findings are related to (i) difference in learning, (ii) quality of the solutions and (iii) exploration of design space for measuring the design task outcome [36, 36]. The quality of the solution is the value of a 2D point on a design solution space, in order words; it is the value of the design task *f(x)* defined earlier in the paper. On the other hand, the exploration values are calculated in three different ways:

4.1 Exploration Quality Index (EQI)

$$EQI = \frac{\text{no. of solutions} > t}{\text{total soln present in solution space} > t} \tag{5}$$

Exploration quality index is the ratio of the number of the explored solution above a certain threshold, t (in this case t is above 0.5, where 0 is min and 1 is max solution quality value) on a reduced solution space (i.e., by a factor of 5 units hence, 20 $\times$ 20) to the total number of solutions available on the design solution space greater than the threshold value. This means that if an agent explores neighboring solution cells, the mean of the solution values of these cells is taken. It was done to avoid having an inaccuracy in the exploration quality that could arise, e.g. when an agent explores 5 immediate neighbor cells to an agent exploring 5 cells at a larger distance

4.2 Exploration Index (EI)

$$EI = \frac{\text{solutions explored on a reduced solution space}}{\text{reduced solution space area}} \tag{6}$$

Exploration index is the number of solutions explored on a reduced solution space to the area of the reduced solution space. Reduced solution space is the original solution space (100 × 100) is decreased in size by a factor (5 in this case) so that the resultant is a smaller space (20 × 20). This means that if an agent explores neighboring solution cells, it is counted as one unit exploration. It was done to avoid having an inaccuracy that could arise, e.g. when an agent explores 5 immediate neighbor cells to an agent exploring 5 cells at a larger distance.

Dispersion of the solution values is the mean of the dispersion of the solutions from the centroid of the solutions. Spread or the dispersion of the solutions obtained was calculated to see how different the solutions are from each other (variety of the solutions).

The above-mentioned model description and methodology show how details of socio-cognitive characteristics of individuals and teams were taken into consideration during simulating idea generation. The goal of the work is not to get the optimal solution values, but to understand how individual thinking in design teams are affected by the presence of influencers during idea generation. Thus, the model is based on theoretical and empirical findings to mimic 'real-world' idea generation. Some of the early findings of the model are stated in the next section.

5 Results and Discussion

Figure 11 shows the average change in solution quality over 1000 simulations, comparing agents that learn from their success and failure (with the effect of the influencer) with those that do not. The increase in the quality of ideas with each session could be due to recall, which is correlated with the number of ideas generated [42]. However, the way an agent with high self-efficacy (but lesser than the self-efficacy of an influencer) behaves during idea generation, is different from an agent with low self-efficacy in a team where there is an influencer. Figure 12 shows the distance between the solutions of a low and high self-efficacy agent with respect to an influencer (here the maximum sessions were 20). It can be seen that an agent with high self-efficacy explore solutions different from the influencer while the low self-efficacy agent generates solutions closer to that of an influencer. This aligns with expectations on the nature of influence in design teams.

Since another parameter familiarity (*f*, as given in above the equation for trust *(T))* that contributes to the degree of influence was constant, agents defined with high self-efficacy controlled the team processes (influencers). Figures 13 and 14 below show teams with a varying number of influencers learn from their success and failure. Learning from the past events in the form of success and failure (as

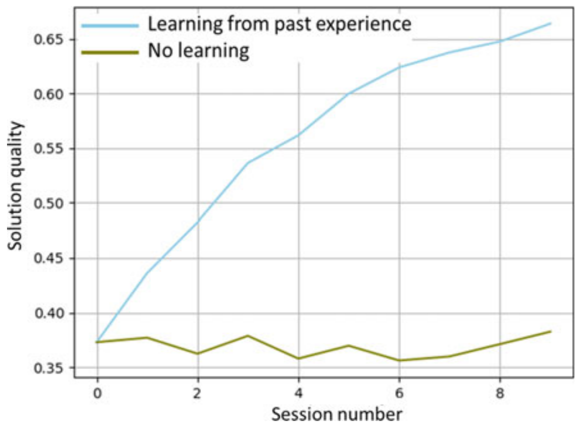


Fig. 11 Agent learning from past experience

Fig. 12 Distance between low and high self–efficacy agents from the influencer (for maximum sessions = 20)

explained in the section on Model description), where agents avoid the failures they have committed in the past and tend to follow the path that led to previous success. The curves shown in these figures are similar to those proposed empirically and theoretically in other works [38]. The teams in which all agents start at high self-efficacy ('All influencers') have a greater ability to learn from failure than the other combinations tested. With respect to learning from success, all the agents in the team with 'No influencer' or all agents with low self-efficacy, learn least from their own success.

Social influence, which leads to the imitation in individuals to modify opinions, attitudes and behavior similar to the others they are interacting with, is referred to as social learning. As the influence of individuals is unevenly distributed in a team, consequently is social learning. The amount of social learning as expected is maximum in the teams of 3 influencers, while minimum when all agents have low self-efficacy (Fig. 15).

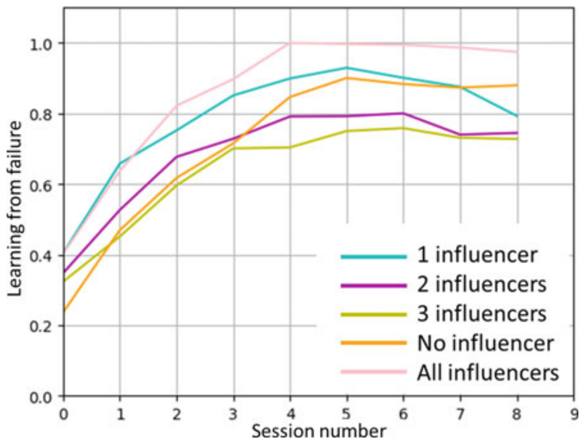


Fig. 13 Learning from failure

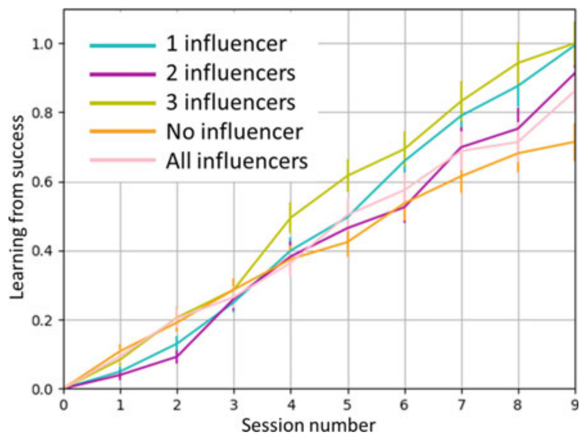


Fig. 14 Learning from Success

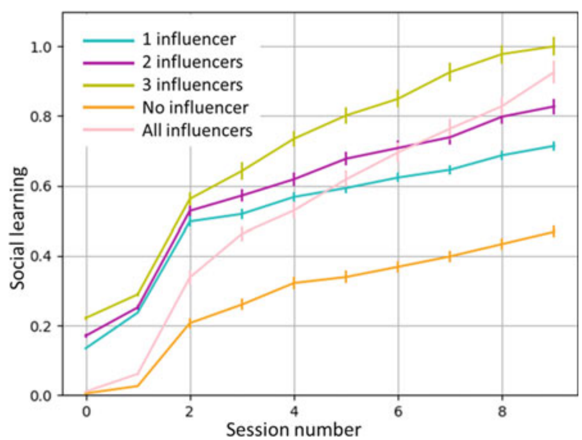


Fig. 15 Social learning

Computationally, solution quality is the function value of a specific point in a design solution space. Figure 16 shows the mean solution quality of each agent in a team over a project of 10 sessions. There was no significant difference in the mean solution quality of each agent in a team however, some minor differences can be observed. As shown in Fig. 16, when all the agents in a team start working with low self-efficacy, they maintain consistently low performance. Interestingly, teams with only one influencer have the highest mean solution quality from session 2 onwards. Agents in teams of all the remaining combinations tested (2, 3 and all high self-efficacy), have similar mean solution quality at the end of the project (last session). The quality results of the model are consistent with the study done by [43], where it was shown that exposure to others' ideas, may increase the quality of ideas generated. In contrast to [43], the results (normalized values) related to the exploration of the design solution space are shown in Fig. 17. It can be noted teams with small/no variations in their self-efficacy, have the highest exploration index while the least global quality exploration index, as found in [9] where teams who diverge less and focus on certain areas on design space perform better.

The agents in these teams explore more areas of design solution space but encounter fewer above average quality solutions. This could mean that due to their low self-efficacies, they are less influenced by their own success; hence, they keep exploring new areas on solution space without producing a higher quality of exploration index. The agents in teams with 1 influencer have less exploration and more global quality exploration index, which decreases as the number of influencers are increased. The solutions of the agents in the teams of one influencer are most dispersed (different from each other) than other team compositions.

The exploration rate, i.e. the number of solutions in a design space explored during a session, without considering the ones in the previous session is shown in Fig. 18. It can be noticed that the teams with well-defined influencers have relatively similar exploration rates over a different session and the exploration rate during sessions 3–6 is lower than in other sessions. It could be deduced that the influencers affect the exploration of design space somewhat in the middle of the project.

Fig. 16 Quality of solutions

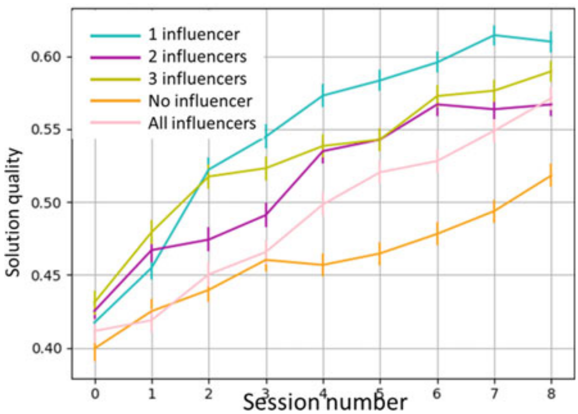


Fig. 17 Normalized exploration values

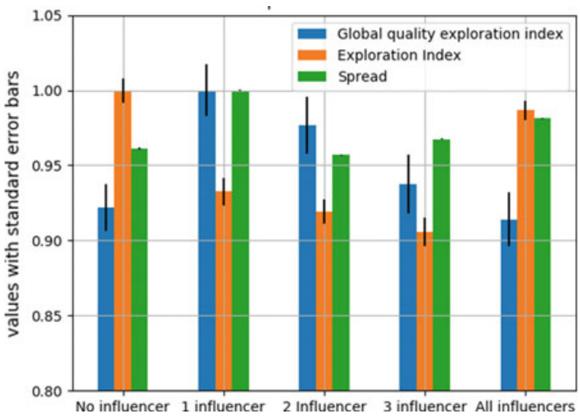
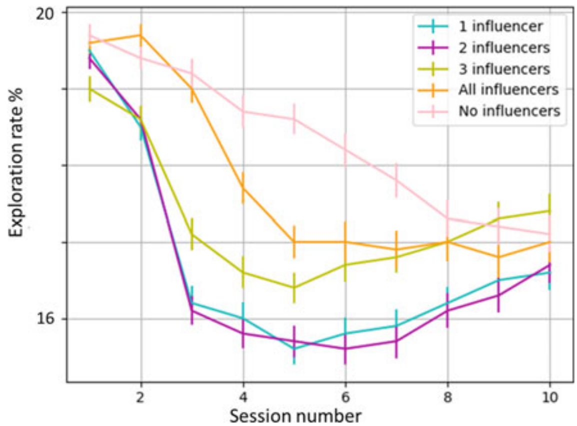


Fig. 18 Session-wise exploration rate



It can also be seen in Fig. 18, where the teams are with no defined influencers, i.e. all the agents are either with low or high self-efficacy, have significantly different exploration rates as compared to the previous scenario. In the case of teams with all low self-efficacy, the exploration rate decreases gradually towards the end of the project. On the other hand, the exploration rate in the teams of all high self-efficacy agents abruptly drops after mid-project as these agents were more confident in their own solutions than others initially. It could be inferred that as the agents in the teams approach the end of the project, some agents among them might have started to emerge like influencers, hence affecting others in the exploration.

It is known that social influence affects creativity. Some authors say that it restricts creativity by limiting variety while others say that it enhances quality [39]. Group-level creativity is a "*function of the extent to which social influences affect individuals within the group at earlier stages*" [39]. This behavior of individuals in design teams was clearly seen in the results during ideation sessions in the teams of uneven distribution of influence. Undoubtedly, more work needs to be done to see how influencers in the design team affect team and organization creativity. On the other hand, the length of a session (time allotted for generating ideas) and the number of sessions also influence the exploration rate of design space [40]; as well, needs more consideration.

6 Conclusion

This work investigates the unequal distribution of social influence within teams, where the agents with high influence on others, are referred to as influencers. This research specifically investigated the effect of influencers on individual team members' exploration and quality as mediated by influence. The team of agents works on a project, which consists of multiple co-design sessions with idea generation and selection. Like any other co-design session, the simulation starts with a design task given to a team of agents who have to produce solutions. Similar to the output of a real world idea generation, solution quality and exploration of the design space were considered as parameters to determine idea creativity in the model. The summary of the results related to quality and exploration are as follows:

- The results show that the agents in the teams without influencers have the least social learning and the least learning from their failure. Consequently, they also the lowest solution quality during all the sessions. However, agents in teams composed entirely of influencers learn the most from their failures. Interestingly, the agents in teams with only a single influencer have the highest solution quality at the end of a project. As the influencer(s) in the team also influence other agents' solution quality during brainstorming as stated in [43].
- Coherent to the past literature, which support that the quality and exploration are negatively related, the teams in scenario 2 (teams without well-defined influencers) explore design space more than the ones in scenario 1 (teams with well-defined influencers), while having a lower global quality index. Similar to the above quality result, teams with one influencer have the highest global quality exploration index and the spread (dispersion) of the solutions.
- The session-wise exploration rate was lowest during the middle and gradually increased towards the end of the project for teams in scenario 1. However, significantly different behavior was observed in teams in scenario 2 where the exploration rate decreased towards the end of the project. This suggests that the effect of the influencers is most prominent in the middle of a project.

6.1 Limitations

Although, the model provides insights related to different team compositions and ideation output, the ability of agent-based modelling to mimic human behavior is fundamentally limited. The results provided here are based on a simplified two-dimensional representation of the design space, which is not necessarily equivalent to actual design problems. The interactions and relations between different agent behaviors may have confounding effects and thus, the validation of the simulation framework is needed. A direct validation study should be a subject of future work. In addition, observational studies will be conducted on idea generation to further tune the model.

Although this representation was adequate for the results produced here, future work should expand the design space to multiple dimensions. Moreover, future work should perform real human experiments to validate the results of the model and provide experimental feedback to improve the model. Ultimately, the inference from the model could provide a faster approach to study co-design activities. The outcome of the research could also assist in suitable people management strategies by project managers, leaders, tutors, facilitators and other leaders of problem-solving teams, making it more feasible to obtain near-optimal project outcomes.

References

1. Steen M, Manschot M, De Koning N (2011) Benefits of co-design in service design projects. Int J Des 5:53–60
2. Kleinsmann M, Valkenburg R (2008) Barriers and enablers for creating shared understanding in co-design projects. Des Stud 29:369–386
3. Becattini N, Cascini G, O'Hare JA, Morosi F (2019) Extracting and analysing design process data from log files of ICT supported co-creative sessions. In: Proceedings of the 22nd International Conference on Engineering Design (ICED19), Delft
4. Singh H, Cascini G, Casakin H, Singh V (2019) A computational framework for exploring the socio-cognitive features of teams and their influence on design outcomes. In: Proceedings of the 22nd International Conference on Engineering Design (ICED19), Delft
5. Bonabeau E (2002) Agent-based modeling: methods and techniques for simulating human systems. Proc Natl Acad Sci USA 99:7280–7287
6. Abar S, Theodoropoulos GK, Lemarinier P, O'Hare GMP (2017) Agent based modelling and simulation tools: a review of the state-of-art software. Comput Sci Rev 24:13–33
7. Salas E, Guthrie JW, Wilson-Donnelly KA, Priest HA, Burke CS (2005) Modeling team performance: the basic ingredients and research needs. In: Organizational Simulation. Wiley Hoboken, NJ, pp 185−228
8. McComb C, Cagan J, Kotovsky K (2017) Optimizing design teams based on problem properties computational team simulations and an applied empirical test. J Mech Des 139:041101-1–041101-12
9. McComb C, Cagan J, Kotovsky K (2015) Lifting the veil: drawing insights about design teams from a cognitively-inspired computational model. Des Stud 40:119–142
10. Gero JS, Kannengiesser U (2004) Modelling expertise of temporary design teams. J Des Res 4:1–13
11. Perišić MM, Štorga M, Gero JS (2018) Exploring the effect of experience on team behavior: a computational approach. In: Design Computing and Cognition DCC'18, Lecco, Italy

12. Hinds PJ, Carley KM, Krackhardt D, Wholey D (2000) Choosing work group members: balancing similarity, competence, and familiarity. Organ Behav Hum Decis Process 81:226–251
13. Lapp S, Jablokow K, McComb C (2019) KABOOM: an agent based model for simulating cognitive styles in team problem solving. Des Sci 4:1–34
14. Singh V, Dong A, Gero JS (2011) How important is team structure to team performance? In: International Conference on Engineering Design, ICED11. Technical University of Denmark
15. Singh, V (2009) Computational studies on the role of social learning in the formation of team mental models. Doctor of Philosophy thesis. Design Lab Faculty of Architecture, Design and Planning, The University of Sydney, Sydney
16. Schreiber C, Singh S, Carley KM (2004) Construct – a multi-agent network model for the co-evolution of agents and socio-cultural environments. CASOS – Center for Computational Analysis of Social and Organizational Systems, Carnegie Mellon University, Pittsburgh, USA
17. McComb C (2016) Designing the Characteristics of Design Teams via Cognitively Inspired Computational Modeling. Carnegie Mellon University, Pittsburgh
18. Myers DG (1982) Polarizing Effects of Social Interaction. Group Decision Making. Academic Press, London
19. Akers RL, Krohn MD, Lanza-Kaduce L, Radosevich M (1979) Social learning and deviant behavior: a specific test of a general theory. Am Sociol Rev 44:636–655
20. Brown R, Pehrson S (2019) Innovation and changes in groups: minority Influence. Group Process: Dyn Between Groups 3:85–100
21. Pei S, Morone F, Makse HA (2018) Theories for influencer identification in complex networks. In: Complex Spreading Phenomena in Social Systems. Springer, pp. 125–148
22. Bandura A (1977) Self-efficacy: toward a unifying theory of behavioral change. Psychol Rev 84:191–215
23. Granovetter MS (1973) The strength of weak ties. Am J Sociol 78:1360–1380
24. Ryan RM, Deci EL (2000) Intrinsic and extrinsic motivations: classic definitions and new directions. Contemp Educ Psychol 25:54–67
25. Costa AC (2003) Work team trust and effectiveness. Pers Rev 32:605–622
26. Mui L, Mojdeh M, Halberstadt A (2002) A computational model of trust and reputation. In: Proceedings of the 35th Hawaii International Conference on System Sciences, Hawaii. IEEE
27. Stempfle J, Badke-Schaub P (2002) Thinking in design teams - an analysis of team communication. Des Stud 23:473–496
28. Goucher-Lambert K, Moss J, Cagan J (2019) A neuroimaging investigation of design ideation with and without inspirational stimuli, understanding the meaning of near and far stimuli. Des Stud 60:1–38
29. Oberauer K, Lewandowsky S (2008) Forgetting in immediate serial recall: decay, temporal distinctiveness, or interference? Psychol Rev 115:544–576
30. Murdock BB (1962) The serial position effect of free recall. J Exp Psychol 64:482–488
31. Banaji RM (1986) Affect and memory: an experimental investigation. The Ohio State University
32. Gentner D (1989) Analogical Learning. Similarity and Analogical Reasoning. Cambridge University Press, Cambridge
33. Novice versus Expert Learning. How People Learn, September 2010. https://serc.carleton.edu/econ/quantitative_writing/noviceVexpert.html
34. Wimmer EG, Shohamy D (2017) Preference by association: how memory mechanisms in the hippocampus bias decisions. The American Association for the Advancement of Science, New York, pp 270–273
35. Singh H, Cascini G, Mccomb C (2020) Analyzing the effect of self-efficacy and influencers on design team performance. In: 16th International Design Conference DESIGN 2020, Dubrovnik, Croatia
36. Shah JJ, Smith SM, Vargas-Hernandez N (2003) Metrics for measuring ideation effectiveness. Des Stud 24:111–134

37. Dorst K, Cross N (2001) Creativity in the design process: co-evolution of problem–solution. Des Stud 22:425–437
38. Leibowitz N, Baum B, Enden G, Karniel A (2010) The exponential learning equation as a function of successful trials results in sigmoid performance. J Math Psychol 54:338–340
39. Agars MD, Kaufman JC, Locke TR (2008) Social influence and creativity in organizations: a multi-level lens for theory, research, and practice. In: Multi-Level Issues in Creativity and Innovation (Research in Multi-Level Issues, vol 7. Emerald Group Publishing Limited, Bingley, pp 3-61
40. Tsenn J, Atilola O, McAdams DA, Linsey JS (2014) The effects of time and incubation on design concept generation. Des Stud 35:500–526
41. Paulus PB, Dzindolet MT (1993) Social influence processes in group brainstorming. J Pers Soc Psychol 64:575–586
42. Dugosh KL, Paulus PB (2005) Cognitive and social comparison processes in brainstorming. J Exp Soc Psychol 41:313–320
43. Paulus PB (2000) Groups, teams, and creativity: the creative potential of idea-generating groups. Appl Psychol: Int Rev 49:237–262