ISSN 1330-3651 (Print), ISSN 1848-6339 (Online)
https://doi.org/10.17559/TV-20170912134413
Subject review

Agent-Based Modelling and Simulation of Product Development Teams

Marija Majda PERIŠIĆ, Mario ŠTORGA, Vedran PODOBNIK

Abstract: The success of product development highly depends on the quality of cooperation among members of a team involved in the process. Thus, a tool capable of simulating product development team may be beneficial for researchers interested in teamwork, as well as useful for managers struggling with team formation during process planning phase. This work aims at providing a detailed overview of agent-based simulators of product development teams. Specifically, the scientific databases Web of Science, Scopus, ACM DL, and IEEE were searched to extract relevant agent-based models of teamwork in mechanical engineering and aerospace context and obtained models were reviewed to identify their key advantages and limitations.

Keywords: agent-based modelling; complex socio-technical systems; product development; teamwork simulation

1 INTRODUCTION

By definition provided in [1], a *product* is "something sold by an enterprise to its customers", and *product development* is "a set of activities beginning with the perception of a market opportunity and ending in the production, sale, and delivery of a product". In the light of given definition, a product can be physical, like a machine or a tool, but the definition also appoints software or service as a product. Ulrich and Eppinger [1] state that, except in a few specific contexts, a minority of the products are developed by a single individual. Rather, products are an outcome of a joint effort of a group of people with the common goal - a *team*.

Benefits of well-coordinated teamwork in terms of reduction of time required for development process completion and improvement of the quality of the resulting product are widely recognised in the literature [2]. However, since there are numerous factors influencing team performance and mutual effect of these factors depends on time and context, predicting team performance is extremely challenging. Additionally, in the field of product development, studies on collaboration and interactions between team members are seldom. All of this leads to the conclusion that project managers have a difficult task while forming a team.

Organisations performing research and development tasks with their teams and individuals interacting within them exhibit each of the characteristics of complex systems [3] such as dynamism, the interdependence of elements, emergent/self-organising behaviour, and non-linearity. Similarly, Oyama et al. [4] argue that organisations can be seen as a complex socio-technical system. Complex systems display behaviour, which cannot be predicted by observing elements in isolation. One of the methods used to overcome this problem by enabling exploration of possible outcomes of interactions between system elements is a *simulation*, i.e. implementing and analysing a *model* - physical, virtual or mathematical representation of the observed system. Model is an approximation of the system - simplified portray of reality whose manipulation and analysis helps in gaining a deeper understanding of the problem and enables drawing conclusions which afterwards can be applied in practice.

Running a simulation enables not just observation of behaviour the modelled system is likely to display in certain conditions but also serves as a tool for researchers to change the input parameters and observe the effect the change has on the dynamics of the simulated system and simulation outcome. In the context of teamwork, a researcher can use simulations to observe the probable performance of the given team, examine the impact various factors have on the performance and, in the same manner, compare different team compositions. These, however, are not the only reasons one can find models and simulation useful.

In fact, Epstein [5] has listed 16 reasons to develop models and run simulations among which is discovering new questions and data collection. Latter proves especially important in the context of examining collaboration in product development context since longitudinal studies are costly and troublesome to conduct.

Altogether, a product development team forms a system whose behaviour cannot be easily predicted by observing the behaviour of its elements, and study of which is hampered by a lack of data. Given these shortcomings, employing computer simulations in studies of product development teams could be particularly useful [6]. A *product development teamwork simulator* can be seen as a research and experimental tool which provides support for researchers and project managers, enables detail examination of the team performance and serves for evading time costs and resources of longitudinal studies by enabling hypothesis testing and scenario analysis [7].

However, to build a useful model, the complexity of the system has to be reduced to the point where it can facilitate examination of the problem and provide practical guidance. Consequently, one has to be careful to ensure a desired behaviour of the simulated system is well-captured within the simulation. Due to the complexity of product development teamwork, this is a challenging task and to achieve all of the potential benefits of the simulation, one has to put a lot of time, effort, caution and knowledge.

The study of state of the art practices in the simulation of product development teamwork which is presented in this work commences with a short introduction of various modelling techniques and their comparison. The capability of agent-based modelling to capture the behaviour and the overall complexity of the socio-technical systems arising from the internal and external interactions is emphasised, and its suitability for the development of the desired teamwork simulation tool is described in Section 2. Section

3 presents an overview of agent-based models of teamwork developed and used in a research context of product development. The insights drawn are presented in Section 4, where several open issues are identified, and limitations and advantages of listed models are discussed. Finally, the work is concluded by identification of the possible future research directions in Section 5.

2 COMPLEX SYSTEMS AND TECHNIQUES FOR THEIR MODELLING AND SIMULATION

Despite the fact that there is no generally accepted definition of complex systems, researchers agree the essential characteristic of a complex system is a large number of elements whose interconnections over time give rise to the collective behaviour that cannot be easily predicted by observing parts in isolation [8]. As noted, a team, as a system of interacting individuals (human and non-human) situated in working environment, can be seen as a complex socio-technical system. This can be shown in Fig. 1, which represents design process model (after Hubka and Eder [9]).

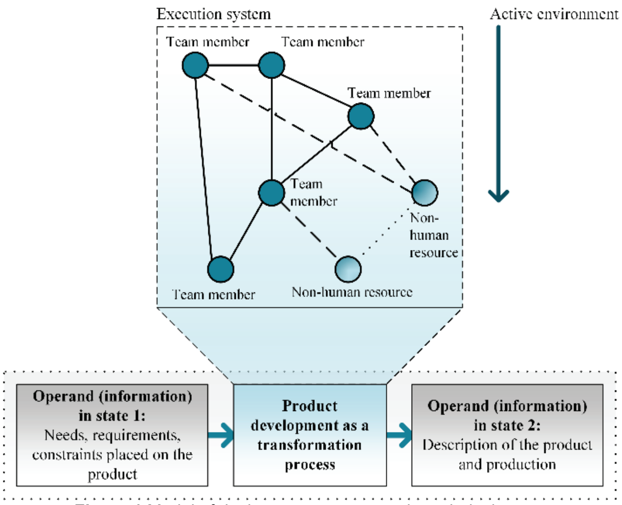

Figure 1 Model of design process as a socio-technical system (after work of Hubka and Eder [9])

Fig. 1 demonstrates transformation system in which individuals interact to produce the description of the product and production [9]. One can note that transformation process necessary includes cooperation and interactions between team members and non-human resources. Thus, team members (human), non-human resources and work environment create a socio-technical system whose complexity lies in number and type of hardly predictable interactions. Consequently, when deciding on the most suitable technique to obtain a model for simulation of teamwork as presented in Fig. 1, common techniques for modelling and simulation of complex systems can be considered. Balestrini Robinson [3] presented the following list of most common modelling and simulation techniques: Network Simulation, Dynamical Systems Simulation, System Dynamics Simulation, Discrete Event Simulation, Markov Simulation, Petri Net Simulation, Poisson Simulation, Cellular Automata, and Agent-based Simulation.

The line between these approaches is not clearly defined. For example, in [3], the author states that Markov and Petri Net Simulation can be seen as special cases of Discrete Event Simulation. Brailsford [10] suggests that some known Discrete Event models have a lot of agent-based model's properties. However, despite the described examples of ambiguity and the lack of criterion which would enable precise classification of models, each of the listed techniques uses a different approach when modelling interactions, entities and environment and, thus, they differ in terms of suitability for problems - a technique should be chosen (or dismissed) depending on the problem requirements [11].

Method / Evaluation criteria	Network Models	Discrete Event Models	System Dynamics Models	Agent-based Models
Nonlinearity – ability to model disproportionate causes-to-effects	Good	Very Good	Very Good	Excellent
Interactions – ability to model the effect of interdependencies between entities	Very Good	Good	Poor	Very Good
Intelligent Agents – ability to model sentience of the entities	Very Poor	Poor	Poor	Excellent
Represent Hierarchies – ability to represent the organisational hierarchy of the entities	Very Good	Good	Good	Excellent
Emergent Behaviour – ability to provide insight into the macroscopic behaviours that cannot be elucidated from the analysis of individual entities in isolation	Good	Poor	Poor	Excellent
Adaptation – ability to model capability to change of the individual entities, closely related to the ability to model sentience	Very Poor	Poor	Poor	Excellent
Dynamic Behaviour – ability to model time-dependent effects and changes of state	Poor	Very Good	Very Good	Very Good
Ease of Creation – how much time and effort must be devoted to developing the model	Excellent	Poor	Good	Very Poor
Ease of Validation – how much time and effort must be expanded in validating the model	Very Good	Good	Good	Very Poor

Figure 2 Comparison of simulation methods (recreated from [3])

Teams display dynamic, non-linear behaviour driven by the actions of intelligent individuals whose behaviour changes over time. Additionally, relations among team members are subject to change over time. Interactions within product development teams depend on various organisational, psychological and social factors and give rise to behaviours, which cannot be predicted by observing individuals in isolation – i.e. teams display emergent behaviour. Therefore, the evaluation criteria listed in Fig. 2 are chosen to compare the simulation techniques based on their capability to present features relevant to the study of product development team's performance and behaviours. Fig. 2 compares Network, Discrete Event, System Dynamics and Agent-based Simulations as these techniques can be used for a broad range of modelling needs [3]. It can be noticed that Agent-based Simulation (ABS) technique possesses great flexibility and expressiveness, enables natural representation of individuals and is superior to others in the ease of implementing desired properties like intelligent agents, various interactions, non-linearity, dynamism and hierarchy. However, as a consequence, ABS models are difficult to create and to validate. In the rest of the paper, the overview and analysis of the ABS models for the product development will be provided.

3 AGENT-BASED MODELS OF PRODUCT DEVELOPMENT TEAMWORK

3.1 Research Methodology Applied

The research methodology used to create an overview of agent-based models of teamwork in a context of product development was an exploration of the scientific publications databases Web of Science, Scopus, ACM DL, and IEEE with the following query:

agent AND (model* OR simulat*) AND (team* OR group*) AND ("product development" OR "product design" OR "engineering design"),*

or its equivalent for each database. Only the content written in English was analysed while no additional filters (nor regarding publication type, nor time frame) were selected. Articles where a) agents were used as human avatars intended to display realistic behaviour, b) the model is used to study product development team behaviour and performance, and c) sufficient details were provided to gain understanding of the model (i.e. agents, environment and interactions are described, although the implementation may not be performed) were included.

Since the goal of this work is to present models which aim at realistic simulations of individual and team behaviour, models where agents are employed as tools for supporting teamwork and collaborative activities, e.g. [12], or in which agents replace humans in teams in order to automate processes and increase efficacy, are omitted from this study. In addition, it is important to highlight that only the models developed and used in the mechanical engineering and aerospace context are presented here due to the space limitations of the paper, even though the full study included agent-based models for mass collaborative product development and open collaboration, construction engineering and management, and software development contexts. Nevertheless, in most cases, the models presented here apply to the broader research area. Of 92 articles in IEEE database, 254 in Web of Science, 271 in Scopus and 72 in ACM DL resulting from the stated query, 8, 17, 18 and 6 respectively were found to be fitting the described criteria, and their lists of references and citations were used to identify additional relevant models. Overall, the search resulted in the identification of twenty one distinct agent-based models which are presented in the next section.

3.2 Results of the Analysis

One of the most prominent agent-based models of teams is the Virtual Design Team (VDT) model developed at Stanford University over the course of 20 years. Its primary purpose is to enable analysis and exact design of a project organisation by implementing agents as individuals with defined skills, experience and position in the team [13]. VDT has been extensively validated and was used for simulating teamwork in product development, e.g. [14]. It provides an estimation of project's duration, cost and quality, and simulates the impact of different communication tools and task uncertainties on project execution. However, it neglects team member's social behaviour and affective states which influence team performance and climate. Another example of the general purpose model that has been used for simulating teamwork in product development is TEAKS [15]. TEAKS was developed in Java-based framework JADE and has been verified and validated on an industrial project. Contrary to the VDT, TEAKS focuses on social and emotional states of team members but fails to include important technical aspects of socio-technical systems, as resources allocation, rework, failures and exception handling have not been modelled. Further, an agent in TEAKS model cannot reason about task importance, urgency or uncertainty, nor decide on the sequence of tasks.

Similar to VDT, model developed by Yang et al. [16] has a purpose of assessing team performance concerning project duration and effective work time. This model is implemented in Visual C++ and has been further refined over the course of the years [17-21]. Tasks are described in terms of resource requirements, expected duration, input and output information (normal, temporary and/or feedback), failure probability, exception probability and collaboration probability. Similar to VDT, tasks are prescribed to agents who try to process them and, if an exception is encountered during the task execution, an agent sends the report to its superior and waits for the instructions. Thus, agents have protocols for exception report, iteration rework, design revision, and interrupt and error reaction. Additional protocols included in the first versions of Yang et al. model are collaborative behaviour, partner selection, task scheduling and resource selection protocols which have been detailed over the course of the years. In the extension proposed by Wang et al. [18], the "planned waiting time" parameter for each agent is introduced, indicating agent's willingness to wait for collaboration or exception report feedback. If waiting time exceeds the planned waiting time, the agent cancels the collaboration request/ignores the exception, and continues to execute the task on its own, thus increasing the design risk, which results in a decrease of the process and product quality.

In Zhang et al. model [20], the partner selection algorithm was refined to take into account skill requirements, availability and organisational type. Another novelty is the introduction of recovery time parameter, which represents the time needed for the agent to recover from the interruption and concentrate on its work. The same model was later extended by introducing task importance, urgency, task recovery cost and agent's workload preferences, and used them to equip agents with the ability to dynamically schedule their tasks based on the utility function defined [21]. Listed models assumed unlimited resources, but in Li et al. model [17] resources are restricted, and arbitration agent responsible for the resolution of resource conflicts is introduced, thus enabling the study of the effects of different resource resolution strategies on the project duration and quality of the outcome. Finally, Zhang and Li [19] introduced the partner selection function based on matching degree algorithm which takes into account potential partner's technique ability, innovation ability, collaboration ability and character attributes based on Myers-Briggs indicator.

These extensions of Yang et al. model were aimed to increase the simulation accuracy in terms of project duration and task allocation. Collaboration realism was increased by refinement of the partner selection algorithm, but other social, cognitive and affective aspects are not included. For example, motivational and affective states of the individuals, goal preferences, agent's perception of the team and the objectives, and team states (or, what Levitt [22] refers to as "chemistry" between team members) have not yet been modelled, although the latest extension [19] has included parameters indicating collaborative and innovation ability of an individual, as well as the parameters indicating personality traits. However, unlike earlier versions of this model, the validation of this extension was not reported.

Crowder et al. [23] presented another model where working time, total project time and total quality are estimated based on the agents' processing of predefined, ordered and preassigned tasks. Similar to [21] an agent with insufficient knowledge can contact other agents for help. Each agent can reject the help request, or decide to accept it. If the request is accepted, helper and help-seeker spend some time working together on the task, while help seeker's competence level increases. In contrast to Zhang et al. model [20], agents in [23] do not select partner based on skill or personal preference. Rather, help-seeker sends a request to every agent, and others respond based on their response rate parameter. However, Crowder at al. model introduced the notion of trust each agent holds for a team. Trust changes depending on whether agent's requests for help are rejected or accepted, and influences the formation of a shared mental model, which further influences agents' motivation and competencies.

Similar to Crowder et al., Dutta et al. [24] developed their model in JADE. Dutta et al. model extended Crowder et al. model by enabling an agent to work on two tasks in parallel, as well as enabling several agents to collaborate on the same task. Authors also introduced a composite measure of team capability and implemented different motivational and learning behaviours. However, neither of [23] and [24] models considers failures, rework or exceptions. Further, change of trust and shared mental model in [23] are described with equations whose coefficients are derived through regression performed on collected data. No additional validation of the model has been reported, and thus the equations may not prove suitable for products, projects or companies different than the one used for data collection.

Another model where insufficient knowledge triggers help-seeking behaviour is Zhang and Thomson [25] model. As in previous models, an agent can learn from its helper, but in this model if no other agent responds to help request, an agent performs the task on its own. Further, the parameter guiding the communication efficacy is introduced to moderate the learning effect. In contrast to [23], rework due to insufficient experience, effort or deficit in communication efficacy is modelled. However, Zhang and Thomson model was not reported to be validated.

What further differentiates the Zhang and Thomson model from all previously mentioned models is the task representation. While all of the previously listed models take as an input workflow structure, Zhang and Thomson [25] have represented a product as a group of knowledge-intensive, interdependent functions characterised by function complexity and integration complexity, and each agent is given a function to work on. Several other agent-based models of product development teamwork implement design task as a search over an abstract, rugged landscape where the height of the landscape indicates the quality of the design solution.

One such model is developed by Mihm et al. [26] for studying how project size impacts coordination between team members. Their simulation showed that, even if every component is simple (e.g. agents are searching for an optimum of a quadratic local performance function) and interdependencies between components are simple, rugged landscape representing team performance arises. The results' robustness to parameter change has been tested.

Herrmann [27] has modelled design space as solution space divided into sets which represent groups of similar solutions (i.e. concepts). Each solution is characterised by its value, while solution space is characterised by its difficulty which indicates the probability of finding a high-value solution. Agents roam the space while following simple behavioural, collaboration and search rules.

Ambler [28] modelled a design landscape as Kauffmann's NK model [29] where each point in space represents a distinct design concept. This model was developed in NetLogo. It focuses on simulation of the long-term performance of the team and is used to examine methods that incentivise beneficial team formation dynamics and minimise structural design complexity. New agents are entering the simulation, while others leave due to the ageing process or insufficient performance. An agent can return to the team if another team member perceives its fitness as sufficient. Agents explore neighbouring locations, jump to distant concepts, provide jumping-off locations for future newcomers, and share information about their relative fitness through collaborative linkages. In contrast to previously described models, agents in Ambler's model have a perception of the current situation, the team, other team members and themselves, and can decide on their next steps.

Similarly, in CISAT model [30] eight theory-based characteristics are implemented in agents to adequately represent problem-solving behaviour displayed by the product design team. Namely, agents have a similar goal, interact in irregular intervals, tend to be biased in favour of their designs, focus on most promising alternatives, learn, develop multiple solutions to avoid premature convergence and, by using breadth- and depth-first search strategies, they search solution space until a satisfying solution is found. CISAT model has been validated and is used to analyse team's processes and performance achieved on problem-solving tasks.

Martynov and Abdelzaher [31] presented the model of the influence of knowledge overlap, search width and problem complexity on the team performance. Authors implemented the model in Delphi 7 and modelled the problem space as Kauffmann's NK model [29]. The agents in [31] model are characterised with knowledge of the fixed number of sub-problems, i.e. the area of expertise, and at every step, agents communicate, search for the solutions and propose them, and form proposal ratings based on their own and other team member's evaluations. Additionally, this work considers the effect of noise in the communication.

In contrast to abstract representations of agent's environment, models like [32] and [33] describe specific environment in great detail to enable simulations of teams in particular work spaces. Christian [32] studied information exchange and team coordination by simulating agents' geographical movement and communication behaviour during the design process. The model is implemented in C++. It has been validated and enables simulation of meeting rooms and offices populated by agents characterised by short-term memory, knowledge, role, work and communication efficacy, and environmental awareness.

Saoud and Mark [33] built virtual collaboration environment for evaluation of different cooperation scenarios based on the data collected during the space mission design team's sessions. This model is developed in Swarm and is used to explore the level of noise in war room of predefined height and width, within which agents are positioned in specific locations. Sidebars, characterised by a number of participants, initiator, beginning time and duration, are prescheduled and create the noise in the room, but also provide a source of information for agents which can hear the conversation. This model is built for a specific cause and requires large amounts of data as an input which limits its application.

Another model built on the findings from data was presented by Olson et al. [34]. Authors developed (and implemented in Java), an agent-based simulator which was used to simulate behaviour observed in design group at NASA's Jet Propulsion Laboratory called Team X. In the model, collaboration between two or more agents is enabled, and two collaborative strategies are implemented: direct negotiation and indirect negotiation, as observed in Team X. It is important to note that the objective of Olson at al. model was not to simulate cognitive processes but to identify patterns of organisational problem-solving strategies and implement them on the platform to enable further experiments.

Models of product development teams listed are used either for examination of the team performance in terms of project duration, estimated quality, number of effective work hours, and cost, or focus on modelling of specific teams and environments. Although, as seen, these models vary in representation and richness of agent's environment, coordination mechanisms, and agent's parameters and complexity level, most of the listed models do not study any of the emergent team properties and states. More precisely, aside to TEAKS and VDT, of listed models only Crowder et al. and Dutta et al. models include elements like team motivation, shared mental models, or trust.

However, there are several models developed specifically for studies of emergent team properties. One such model is Singh et al. [35], which is used to study differences in a dynamic formation of a transactive memory system in flat, distributed and functional teams, and to observe the effect of transactive memory system on activity coordination and team effectiveness.

Similarly, while studying temporary design teams, Singh and Gero [36] proposed the architecture of a situated, cognitive and affective agent, which is creating a mental model of its team members by describing them in terms of their function, behaviour and structure. The agent uses formed mental model to create generalisations and expectations, thus deriving the hypotheses about other member's characteristics (based on the previous experiences with others) when insufficient information is present. The agent refines its models through subsequent interactions, enabling the new generalisations to emerge. In Gero and Kannengiesser [37] model, such mental models of an agent were used to study expertise formation in temporary design teams where each agent has to adapt to new team configuration and find common ground with others in order to successfully collaborate.

Singh and Casakin [38] further make use of agent's mental models of others. In their work, authors proposed a model for studying how the use of between- and inter-domain analogies in design team influences the team cohesion and collaboration. In this model, agents are characterised by the number of domains the agent is familiar with and their expertise in these domains, and "who knows what" mental models are used to communicate analogies between team members successfully. Singh and Casakin model is also related to another concept, which is of great importance for design team - creativity. Computational experiments can be used to study impact agents have on each other's thought process, leaning, number of ideas and value system.

Dehkordi et al. [39] studied the likelihood of creativity in teams under the different levels of workload pressure, relevant knowledge distribution and personal factors. This model considered many aspects not covered in other models, for example: what is the effect of deadlines on agent's stress level, how does motivating and challenging environment help team innovativeness, how do team members (de)motivate each other and what are "good-group features". However, no validation of the model was provided.

Sosa and Gero [40] examined the adopter's impact on the perception of designers' creativity. In their work, authors modelled both, adopters and designers, as adaptive, interactive entities. Adopters interact with each other to learn by exchanging opinions on the designs. As a consequence, their preferences constantly change. Designers, on the other hand, try to produce the best scoring designs and learn either by themselves or by imitating others. This work demonstrates how, even if all designer agents start as equally creative, interactions between adopters cause certain designers to stand out or, in other words, be perceived as more creative than others.

It is important to note that designers modelled in [40] do not form a team. Rather, each agent develops a product on its own, interacts with others only through imitation of their products, and competes with others for a market share. However, since it explicitly models the behaviour of individual designers and their mutual influence on the development process, Sosa and Gero's model has been included in this overview.

Another model by Sosa and Gero [41] is used to examine the impact of group influence, measured as the ratio of ideas available to agents, on brainstorming groups. This way, the authors wanted to inspect effect of different team structures on idea generation. Their agents develop new shapes by combining shapes from the initial set, and task difficulty is measured by the number of forms in the initial set and their number of sides. Agents can explore by drawing random shape and transformation, evaluate by forming a concept from topology relationships of shapes, and exploit by applying learned concepts. Availability of concepts generated by other agents and exploration length were modelled as parameters and varied in simulation runs to study their effect on quality and quantity of ideas. However, this model has several limitations as diversity between agent's capabilities, leadership style influence, compliance to group majority and group agreement to adjust idea influence have not been modelled.

4 DISCUSSION OF THE FINDINGS

The overview presented in the previous section enables several insights. Most of the models listed focus either on simulation of the technical performance of the team regarding project time, quality of the product, or collaboration and production cost, e.g. [21], or focus on studying social phenomena such as the formation of trust and transactive memory, e.g. [35]. Models whose purpose is to estimate the technical performance of the team usually neglect important social processes such as trust, or participant's transactive memory systems and other forms of mental models. On the other hand, models focused on the exploration of social processes and emerging team properties, such as cohesion or team creativity, omit task, resource or project details. However, a model which would enable studies and measurement of both, intangible individual and team level aspects and tangible (e.g. time and cost) aspects, could provide a more comprehensive view of the team performance [42].

Regarding the modelling of interactions between team members, it can be noticed that all of the models implement collaborative behaviour in the form of either exchange of the current design solutions, e.g. [28, 27, 31, 30], or exchange of relevant information due to task interdependencies, e.g. [23, 22]. Further, helping behaviour initiated due to knowledge deficiency of some agent has been implemented in several models, e.g. [23], [24], but for example, back-up behaviour intended to equal the distribution of workload between team members has not been modelled. Finally, only a few models include informal communication, e.g. [32], or indirect interactions, such as learning by observation, e.g. [35], and imitation, e.g. [40, 41].

Interactions between team members are necessary for team properties to emerge from the simulation. Models that enable simulation of emergent team properties have been utilised to study trust, team cohesion, shared mental models, team creativity, team expertise, team capability, team motivation and team innovation, as listed in Tab. 1.

Table 1 Emergent team properties simulated by the analysed models

Model and references	Emergent team properties
VDT [13, 22]	Trust Transactive memory system
TEAKS [15]	Trust
Crowder et al. [23] model	Trust Shared mental models
Dutta et al. [24] model	Team motivation Team capability
Singh et al. [35] model	Transactive memory system
Singh and Gero [36]model	Transactive memory system Trust Shared mental models Team adaptability
Gero and Kannengiesser [37] model	Team expertise
Singh and Casakin [38] model	Team cohesion Team collaboration
Dehkordi et al. [39] model	Team motivation Team creativity
Sosa and Gero [41] model	Team creativity

Agents, as implemented in most of the models, are (implicitly or explicitly) assumed to have a mental model of tasks, team, equipment, goal, and team interactions. For example, in each model where an agent contacts more knowledgeable team member for help, it is implicitly assumed that the agent "knows" the exact level of team members' knowledge. An example of a model where biased, human-like behaviour is explicitly implemented is CISAT [30] where agents have a personal bias when rating design solutions. Further, human mental models constantly change. Some of the models implement change in agent's abilities as an increase in competence parameter, e.g. [23], [24], but details on the content learnt are omitted.

However, what is learnt and communicated, in what phase of the new product development process, and in which manner, matters for the success of product development. Thus, for some applications, more detail representation of the mental models and their evolution could be beneficial for studies of teamwork. Ideally, one would implement a system where agents are capable of dynamically updating their mental models based on the situations encountered during the simulation, consequently enabling them to change their behaviour with respect to the situational context and knowledge formed based on previous experiences. Several listed models include some of the aspects required for providing such functionality. For example, agents in [35] dynamically change their beliefs, learn, and update their transactive memory system, while in [40] agents are continuously creating new shapes, learning and refining their understanding of adopters' preferences (which themselves are changing). Such mental models enable simulation of various human behaviour. For instance, in [37] agents develop generalised representations of others and use them to create expectations of newcomers joining the team, which could be understood as forming stereotypes and using associations. Similarly, in [36] agents could be utilised for studying the effects of turnover within a team.

Another aspect of human behaviour which is influencing team behaviour is emotionality. For example, Schaub [43] states that designers necessarily have to deal with stress and time pressure. Nevertheless, only [15, 36] included influences of emotions on the team performance and processes, while in [23] team member's motivational state is modelled.

Finally, as it is presumably the biggest limitation of agent-based modelling [3] the validity of listed models needs to be explored. Out of all presented models which can be used for simulation of product development teams, eight models [25, 27, 31, 36, 37, 38, 39, 41] reported no verification or validation details, findings of [26, 28, 34, 40] models are reported to be tested for statistical robustness by employing sensitivity analysis, [23, 24, 35] models are validated by comparison of the model's outcomes with the assumptions made based on the literature or by face validation, and remaining models [13, 15, 20, 30, 32, 33] are reported to be verified and validated by comparison with the empirical data. However, some of the validated models used the same data for calibration of the model's parameters and validation purposes. These results are likely due to the difficulty of collecting data on teams. Typical team-related data collection methods include self-reporting, most commonly through work-diaries or questionnaires. However, these methods are retrospective, subjective, insufficiently detailed and/or report on the intangible aspects of teamwork which are difficult to translate into parameter values. All of the listed characteristics highly influence precision and accuracy of the data. Although there is no commonly accepted solution to this challenge, an impact of subjectivity in some cases can be reduced by collecting answers from multiple sources (e.g. 360 performance assessment), while retrospection can be avoided by using work-sampling applications [44].

5 CONCLUSIONS

The quality of cooperation between team members plays a great role in the success of product development. Team performance is largely influenced by the complementarity of team members' knowledge, their mutual understanding, trust and compatibility of their goals. However, numerous factors are complicating prediction of characteristics a team will have: features of a team cannot be easily predicted by studying individuals in isolation, research on team characteristics is seldom and findings are occasionally contradictory, data are hardly obtainable and context-dependent, and many intangible aspects are influencing the team performance. Agent-based models provide assistance in overcoming listed shortcomings by enabling simulation of controlled, repeatable and extensible experiments, and, as a consequence, facilitate understanding and direct further research.

A literature review revealed the growing trend of utilising agent-based simulations for studying different aspects of teamwork in product development. Broad applicability and expressiveness of agent-based modelling technique enabled researchers to simulate likely project duration and to estimate project cost, but also to study intangible features, simulate team processes and explore emerging properties of the team. However, the potential of employing agent-based modelling in product development teamwork studies has yet to be fulfilled, and there are several possible directions for further research:

- *Simulation of emerging team properties and team processes.* For example, models studying the emergence of shared mental models, situational awareness, trust, knowledge grounding or other affective, social or cognitive constructs and their relation to known design phenomena such as fixation and design patterns could be developed.
- *Models' validation and development of methods for their comparison*. Additional experiments are needed to validate findings of most of the listed models. However, lack of adequate data hinders this process. Alternative validation techniques such as docking (i.e. model-to-model analysis) are encouraged. Further, model developers are encouraged to provide detail documentation and/or code of their models to facilitate understanding and enable replication of their results.
- *Refinement and extension of current models.* The example of VDT model shows that building upon existing, validated models by carefully enriching its assumptions can lead to powerful, veridical models. Apart from refining, existing models can be extended to include additional organisational or market factors thus detailing the context in which design team operates.

The summary and critique of the literature presented herein represents the first review of agent-based models used for simulation of product development teamwork and can serve not only as the starting point for those interested in simulating the behaviour of a product development team, but also as a guidance for managers and practitioners in search of simulation models that would ease the planning of future projects. Despite various challenges and shortcomings identified, this review demonstrated the wide applicability of agent-based technique in team behaviour simulation and outlined several possibilities for future research. With the growth of empirical research on individual and team behaviour, and with the increase of computational power and advances in computational studies, the reliability of described simulators could be significantly advanced. In the words of Raymond E. Levitt, "There is much exciting work to be done!" [22].

Acknowledgement

This work was supported by Croatian Science Foundation (http://dx.doi.org/-10.13039/-501100004488) MInMED project (grant number 6430, www.minmed.org).

Notice

Parts of this paper were submitted by Marija Majda Perisic to the Council of the Faculty of Electrical Engineering and Computing, University of Zagreb, in partial fulfilment of the requirements for the doctoral qualifying exam.

6 REFERENCES

[1] Ulrich, K. T. & Eppinger, S. D. (2003). *Product design and development*. Tata McGraw-Hill Education.
[2] Hoegl, M. & Gemuenden, H. G. (2001). Teamwork quality and the success of innovative projects: A theoretical concept and empirical evidence. *Organ. Sci., 12*(4), 435-449. https://doi.org/10.1287/orsc.12.4.435.10635
[3] Balestrini Robinson, S. (2009). *A modeling process to understand complex system architectures*. Georgia Institute of Technology.
[4] Oyama, K., Learmonth, G., & Chao, R. (2015). Applying complexity science to new product development: modelling considerations, extensions, and implications. *J. Eng. Technol. Manage., 35*, 1-24. https://doi.org/10.1016/j.jengtecman.2014.07.003
[5] Epstein, J. M. (2008). Why Model? *JASSS, 11*(4), 12.
[6] Louie, M. A. & Carley, K. M. (2008). Balancing the criticisms: Validating multi-agent models of social systems. *Simulat. Modell. Pract. Theory, 16*(2), 242-256. https://doi.org/10.1016/j.simpat.2007.11.011
[7] Helbing, D. (2012). *Social self-organization: Agent-based simulations and experiments to study emergent social behavior*. Springer-Verlag Berlin Heidelberg. https://doi.org/10.1007/978-3-642-24004-1
[8] Bar-Yam, Y. (2002). *General features of complex systems. Encyclopedia of Life Support Systems (EOLSS)*, UNESCO, EOLSS Publishers, Oxford, UK.
[9] Hubka, V. & Eder, W. E. (1992). *Engineering design*. Zürich: Heurista.
[10] Brailsford, S. (2014). Discrete-event simulation is alive and kicking! *J. Simul., 8*(1), 1-8. https://doi.org/10.1057/jos.2013.13

[11] Siebers, P.-O., Macal, C. M., Garnett, J., Buxton, D., & Pidd, M. (2010). Discrete-event simulation is dead, long live agent-based simulation! *J. Simul.,* 4(3), 204-210. https://doi.org/10.1057/jos.2010.14

[12] Bojic, I., Podobnik, V., & Kusek, M. (2010). *Agent-enabled collaborative downloading: towards energy-efficient provisioning of group-oriented services*. KES-AMSTA, 62-71.

[13] Jin, Y. & Levitt, R. E. (1996). The Virtual Design Team : A Computational Model of Project Organizations. *Comput. Math. Organiz. Th., 2*(3), 171-196. https://doi.org/10.1007/BF00127273

[14] Cardinal, L. B., Turner, S. F., Fern, M. J., & Burton, R. M. (2011). Organizing for product development across technological environments: Performance trade-offs and priorities. *Organ. Sci., 22*(4), 1000-1025. https://doi.org/10.1287/orsc.1100.0577

[15] Martínez-Miranda, J. & Pavón, J. (2012). Modeling the influence of trust on work team performance. *Simulation, 88*(4), 408-436. https://doi.org/10.1177/0037549711404714

[16] Yang, Y., Song, L., & Zhang, X. (2007), Organization-oriented simulation of collaborative product development process based on designer's agent model. *In 11th International Conference on Computer Supported Cooperative Work in Design, CSCWD 2007*, 309-314.

[17] Li, Y., Zhang, X., & Zhang, S. (2011). Arbitration strategy analysis of resource conflict in collaborative product development process using agent-based simulation. *In 15th International Conference on Computer Supported Cooperative Work in Design*, 324-328.

[18] Wang, Y., Xu, Y., & Zhang, X. (2009). Optimization of product development process based on multi-agent simulation. *In International Conference on Cooperative Design, Visualization and Engineering*, 351-358. https://doi.org/10.1007/978-3-642-04265-2_53

[19] Zhang, S. & Li, Y. (2014). Multi-agent simulation of partner selection behaviour based on matching degree in collaborative product development process. *Computer Modelling and New Technologies, 18*(10), 25-30.

[20] Zhang, X., Luo, L., Yang, Y., Li, Y., Schlick, C. M., & Grandt, M. (2009). A simulation approach for evaluation and improvement of organisational planning in collaborative product development projects. *Int. J. Prod. Res., 47*(13), 3471-3501. https://doi.org/10.1080/00207540802356770

[21] Zhang, X., Zhang, S., Li, Y., & Schlick, C. (2012). Task scheduling behaviour in agent-based product development process simulation. *Int. J. Comput. Integ. M., 25*(10), 914-923. https://doi.org/10.1080/0951192X.2011.596223

[22] Levitt, R. E. (2012). The Virtual Design Team: Designing project organizations as engineers' design bridges. *J. Organiz. Des., 1*(2), 14-41.

[23] Crowder, R. M., Robinson, M. A., Hughes, H. P. N., & Sim, Y. W. (2012). The development of an agent-based modeling framework for simulating engineering team work. *IEEE T Syst. Man. Cy. A, 42*(6), 1425-1439. https://doi.org/10.1109/TSMCA.2012.2199304

[24] Dutta, P., Pepe, C., & Xi, H. (2015). Analyzing Organization Structures and Performance through Agent-Based Socio-technical Modeling. *In Proceedings of the 18th Asia Pacific Symposium on Intelligent and Evolutionary Systems, Vol. 2*, 39-53.

[25] Zhang, X. & Thomson, V. (2016). The Impact and Mitigation of Complexity during Product Design. *Int. J. Design & Nature and Ecodynamics, 11*(4), 553-562. https://doi.org/10.2495/DNE-V11-N4-553-562

[26] Mihm, J., Loch, C., & Huchzermeier, A. (2003). Problem-solving oscillations in complex engineering projects. *Manage. Sci., 49*(6), 733-750. https://doi.org/10.1287/mnsc.49.6.733.16021

[27] Herrmann, J. W. (2015). Predicting the Performance of a Design Team Using a Markov Chain Model. *IEEE T. Eng. Manage., 62*(4), 507-516. https://doi.org/10.1109/TEM.2015.2456833

[28] Ambler, N. P. (2015). *Design in the Modern Age: Investigating the Role of Complexity in the Performance of Collaborative Engineering Design Teams*. Virginia Polytechnic Institute and State University.

[29] Kauffman, S. A. (1993). *The origins of order: Self-organization and selection in evolution*. Oxford University Press, USA.

[30] McComb, C., Cagan, J., & Kotovsky, K. (2015). Lifting the Veil: Drawing insights about design teams from cognitively inspired computational model. *Design Stud., 40*, 119-142. https://doi.org/10.1016/j.destud.2015.06.005

[31] Martynov, A. & Abdelzaher, D. (2016). Iterative problem solving in teams: insights from an agent-based simulation. *Team Perform. Manag., 22*(1/2), 2-21. https://doi.org/10.1108/TPM-04-2015-0023

[32] Christian, A. D. (1995). *Simulation of information flow in design*. Massachusetts Institute of Technology.

[33] Saoud, N. B.-B. & Mark, G. (2007). Complexity theory and collaboration: An agent-based simulator for a space mission design team. *Comput. Math. Organiz. Theor., 13*(2), 113-146. https://doi.org/10.1007/s10588-006-9002-7

[34] Olson, J., Cagan, J., & Kotovsky, K. (2009). Unlocking organizational potential: a computational platform for investigating structural interdependence in design. *J. Mech. Des., 131*(3), 31001-31013. https://doi.org/10.1115/1.3066501

[35] Singh, V., Dong, A., Gero, J. (2013). Social learning in design teams: The importance of direct and indirect communications. *AIEDAM, 27*(2), 167-182. https://doi.org/10.1017/S0890060413000061

[36] Singh, V. & Gero, J. S. (2007). Computational modeling of temporary design teams. *In Proceedings of CAADRIA07*, 49-56.

[37] Gero, J. S. & Kannengiesser, U. (2004). Modelling Expertise of Temporary Design Teams. *J. Design Res., 4*(3), 1-13. https://doi.org/10.1504/JDR.2004.009840

[38] Singh, V. & Casakin, H. (2015). Developing a computational framework to study the effects of use of analogy in design on team cohesion and team collaboration. *In Proceedings of ICED 15*, 101-110.

[39] Dehkordi, F. M., Thompson, A., & Larsson, T. (2012). Impacts of Project-Overload on Innovation inside Organizations: Agent-Based Modeling. *International Journal of Social, Behavioral, Educational, Economic, Business and Industrial Engineering, 6*(11), 2808-2813.

[40] Sosa, R. & Gero, J. S. (2005). A computational study of creativity in design: *The role of society. AIEDAM*, 19(4), 229-244. https://doi.org/10.1017/S089006040505016X

[41] Sosa, R. & Gero, J. S. (2012). Brainstorming in Solitude and Teams : A Computational Study of Group Influence. *In Proceedings of ICCC 2012*, 188-194.

[42] Škec, S., Cash, P., & Štorga, M. (2017). A dynamic approach to real-time performance measurement in design projects. *J. Eng. Design, 28*(4), 255-286. https://doi.org/10.1080/09544828.2017.1303665

[43] Schaub, H. (2007). The importance of the characteristics of the task to understand team mental models. *CoDesign, 3*(1), 37-42. https://doi.org/10.1080/15710880601170800

[44] Škec, S., Štorga, M., & Tečec-Ribarić Z. (2016). Work Sampling of Product Development Activities. *Technical Gazette, 23*(6), 1547-1554.

Contact information:

Marija Majda PERIŠIĆ, mag. math.
(Corresponding author)
University of Zagreb,
Faculty of Mechanical Engineering and Naval Architecture
Ivana Lučića 5, 10000 Zagreb, Croatia
mperisic@fsb.hr

Mario ŠTORGA, PhD.
University of Zagreb,
Faculty of Mechanical Engineering and Naval Architecture
Ivana Lučića 5, 10000 Zagreb
mario.storga@fsb.hr

Luleå University of Technology,
Department of Business Administration, Technology and Social Sciences
97187 Luleå, Sweden
mario.storga@ltu.se

Vedran PODOBNIK, PhD.
University of Zagreb,
Faculty of Electrical Engineering and Computing
Unska 3, 10000 Zagreb, Croatia
Vedran.Podobnik@fer.hr