Design of modular generation of human behaviour in a collaborative context

Tristan de Blauwe, Domitile Lourdeaux, Nicolas Sabouret

To cite this version:

Tristan de Blauwe, Domitile Lourdeaux, Nicolas Sabouret. Design of modular generation of human behaviour in a collaborative context. 25th IEEE International Conference on Computer Supported Cooperative Work in Design (CSCWD 2022), May 2022, Hangzhou, China. pp.269-274, 10.1109/CSCWD54268.2022.9776258 . hal-03752147

HAL Id: hal-03752147
https://hal.science/hal-03752147v1

Submitted on 16 Aug 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Design of modular generation of human behaviour in a collaborative context.

1st Tristan de Blauwe
Alliance Sorbonne University
UTC, CNRS, Heudiasyc, UMR 7253
Compiègne, France
tristan.de-blauwe@hds.utc.fr

2nd Domitile Lourdeaux
Alliance Sorbonne University
UTC, CNRS, Heudiasyc, UMR 7253
Compiègne, France
domitile.lourdeaux@hds.utc.fr

3rd Nicolas Sabouret
Université Paris-Saclay
LISN, CNRS UMR 9015
Orsay, France
nicolas.sabouret@universite-paris-saclay.fr

***Abstract*—Training people for facing dangerous situations has always been expensive and difficult or dangerous to organize. New approaches use virtual environments and simulation in which the role of collaborative people is played by cognitive agents, which allows making training sessions when they were not previously possible, or else decreasing their cost. Two problems arise: firstly, it is necessary to simulate teams, composed of virtual agents and the learner, interacting in a collaborative context. Secondly, we need to support the definition of a variety of explainable and professional behaviours (*i.e.* representative of the expected behaviours of an expert in a given profession). It is extremely complex and time consuming. To facilitate the design of tools allowing to build adequate agents model, we propose a meta-model of human behavior simulation. The originality of this meta-model is to make the cognitive models, used for explainability, and the agent model completely independent, to avoid repeated implementations of various behaviours. We show how task selection works on a simple agent model, by means of a graph of influences and preferences. This means has the advantage of being intelligible by its visual nature and of facilitating the addition of new behaviours impacting on the selection. A preliminary evaluation has shown that human users perceive the variability of behaviour and that the provided explanations are relevant.**

***Index Terms*—simulation for training, virtual collaborative agents, behaviours, cognitive model, task selection, influence graph, preferences, independence.**

I. Introduction

New training environments use cognitive agents to simulate collaborating humans [1]–[3]. We are interested in the training of socio-technical skills during collaborative activities in crisis situations. For this purpose, we use agents to simulate team members whose behaviour varies according to several aspects, such as stress and/or personality. These behaviours can impact on the collaborative activity and require the learner to adapt to their team [4].

These agent are Autonomous Virtual Characters, abbreviated as AVCs. They are embodied virtual agents evolving within a virtual environment (VE). They interact with the elements of the environment, whether they are artefacts, other AVCs, or human users. The activity of AVCs within the VE is defined by an agent model. Classically, the model follows repeated perceive-reason-act cycle [5]. An agent model is thus composed of a set of information processing mechanisms, from perception to action. We call them *processes*.

Our objective is to incorporate different sorts of *behaviours* in the model that impact the processes so as to create an illusion of life by imitating some psychological mechanisms. For example, Driskell and Salas [6] proposed "a limited number of cognitive, emotional, and social mechanisms through which stress impacts performance." One of these behaviours could impact perception as "Stress increases distraction and decreases attentional focus." Another could impact task selection as "under stress people tend to be less likely to help others [...]." Cognitive models in the literature like this one or others like [7] or [8] propose different ways of implementing such behaviours within an agent model. In this paper, we propose a meta-model allowing to combine different behaviours from different cognitive models within one single agent model, in a modular way.

To evaluate this model, we place ourselves on a case study in the field of emergency medicine [2]. We show how our model allows us to simply define identifiable, explicable and modular behaviours in relation to the domain (here, stress management and communication).

Our first goal is to be able to reproduce behaviours that are representative of those observed in the field. We call it *situated behaviours* [4].

Our second goal is to obtain explainable behaviours. While the representativity requires some variability in the expressed behaviours, we want that the actions of the agent make sense for the observer and that its apparent decisions are always explainable (especially since we are in a learning context). According to [9], explainability includes taking into account the observer's model to provide the relevant explanations. However, in this paper, we limit explainability to providing an explanation of the causes of a behaviour, through the cognitive models and the strategies used. We aim at using cognitive models to explore the variability in an explainable way. An observer may not understand at first the observed behaviours, but they should make more senses after an explanation is provided.

Our third goal is to propose a modular approach to behaviour simulation through cognitive models integration. Indeed, we want to be able to add or to remove behaviours as required by the application domain. Building such a domain-independent model requires the model to be also independent from the cognitive models, so that different aspects of different models can be integrated. The model we present in this paper supports such a modularity.

The next section positions our work w.r.t. other models in the literature. Section 3 presents the basics of our model. Section 4 shows how we implement behaviours during task selection. Section 5 present a preliminary experiment to assess whether users can perceive the differences in AVCs' behaviours.

II. Problem statement

We are interested in operationnalizing a set of cognitive models about collective work, within an agent model. We first started to look at relevant computational approaches.

A. Related-works

Several approaches are conceivable to simulate our agents and generate behaviours in line with their cognitive models.

On one hand, we could use techniques present in video games [10], such as Behaviour Trees (BT) or Finite State Machine (FSM). However, scaling up with the necessary variability takes us away from such approaches.

We could also consider planning approaches such as Hierarchical Task Network (HTN) [11], but on their own they lack the functionality needed for an agent model such as communication, perception, etc.

On the other hand, other possibilities would be to use heavy cognitive architectures, like SOAR [12], or ACT-R [13]. An approach frequently adopted in simulation [14] is to use BDI [15] architectures. However, their complexity and their cognitive models are not necessarily what we desire. Another difficulty is to integrate other cognitive models within such architectures, hence the large number of extensions to BDI [14], like PEP-BDI [16] or more recently BEN [17], a normative social emotional agent architecture.

We do not seek to challenge these approaches. Rather, we see them as tools that we could use/build upon to add required cognitive models. A question is how we can modularly add cognitive models and dynamically modify the agent's model without having to do extensive work. We are therefore moving towards what we call an agent meta-model. To illustrate our work in a training context, we build upon an already existing agent model that takes into account some cognitive models.

B. Agent model

For the reasoning part of the agent model, an option is to consider hybrid approaches, like Replicants [4]. The model reasons on an activity model, called ACTIVITY-DL. The formalism allows to model human activity in a hierarchical way, close to the mental representation of experts. It is particularly relevant for describing collaborative work, as it allows the modeler to define agent roles or qualifications for each task, number of agents, and the preconditions that impact the subtasks in the hierarchy. Moreover, this model inherently supports the generation of variable behaviours that achieve the procedure, by allowing agents to fulfill only a subset of the conditions.

The model already takes into account some cognitive models like Demary [7]. However, this model does not meet our requirements (like modularity) and must therefore be adapted.

C. Cognitive models

We are interested in cognitive models applicable to a team in a crisis situation, particularly those that may impact the collective aspect. Therefore, we consider the following cognitive models:

Demary [7] studied behaviours of followers in a hierarchical team, including the difference in behaviour between a proactive follower and a passive follower. It is based on the followership models of [18], [19].

Driskell [20] has conducted numerous studies on teams in extreme environments and the impact of stress, among other factors, on their behaviours. An example of such behaviour would be the "flexibility in the team's status structure, such that one team member may take the lead for some tasks and another team member for other tasks." The model is of particular interest to us because of the multiple impacts of stress on different processes, such as emotions, cognition and sociability. It represents a challenge in terms of the variety of behaviours to be accounted for.

A first observation is the difficulty of operationalising one but also several cognitive models within an agent model. Thus, we would like to facilitate this step.

III. Presentation of the Meta-model

In this section, we will first present the basics of the meta-model. Then, we will give some details on the use of influence graphs and preferences to select tasks.

A. Meta-model to simulate AVC

Our meta-model consists of programming an agent model to simulate AVCs. We do not seek to define yet another agent model. The objective is to facilitate the operationalisation of behaviours, whatever the agent model used, according to the needs of the application. This is why we propose a meta-model and not a specific model.

1) AVC*:* An AVC is characterised at least by an internal state and an agent model, as shown by Fig.1.

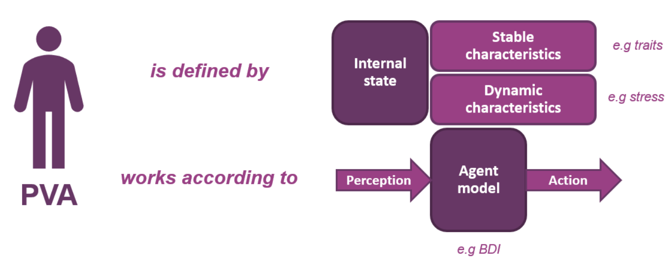

Fig. 1. Structure of AVCs. The exact definition depends on the application.

The internal state contains what is called a *profile*, *i.e.* a set of characteristics. We differentiate between stable characteristics (*e.g.* personality traits) and dynamic characteristics (e.g.

stress). The difference lies in the way these characteristics are assessed. A stable characteristic can be determined in an arbitrary way. This characteristics may change during the simulation, but overall it is expected to remain the same. In contrast, dynamic characteristics are expected to change regularly during the simulation. A priori, its value can only be evaluated during the simulation, by a process which is responsible for updating its value.

The agent model takes as input a set of percepts, *i.e.* information that it can perceive. Its operation is the result of a set of processes, whether or not they are chained together. It is inspired by flow-based programming [21]. Finally, the agent model outputs actions to be performed. Whether it is the agent model or the internal state, we cannot specify the contained data more precisely. Indeed, this depends on the needs of the application [5].

2) Process: A first contribution is to standardise the definition of a process. The goal is to combine different processes whatever the models from which they are derived. A process is a function with any inputs and outputs. The process is characterised by a strategy that indicates how a set of behaviours, noted C, from relevant cognitives models, will alter the process.

3) Behaviour: The definition of a behaviour is therefore dependent on the strategy. We can identify that a behaviour has, as a minimum, the inputs of the process it affects. The operation and output of the behaviour depend on the strategy adopted by the process.

As Faur [8] points out, a behaviour depends on both the agent's profile and his evaluation of the situation. This evaluation depends on the agent's profile and its knowledge. For example, an introverted agent might behave extrovertly in the presence of friends, but introvertly in the presence of co-workers. Therefore, each behaviour is associated with an activation function. It indicates when this behaviour should be taken into account according to the profile of the agent and his evaluation of the situation. This function takes the agent's state as input, evaluates the situation and returns $true$ or $false$ according to the indications of the cognitive model.

B. Operationalising a cognitive model

Operationalising a cognitive model consists in implementing it within the agent model so that it can exhibit the behaviours induced. The implementation can be done in 4 steps, ordered by increasing difficulty:

1) **Adding stable characteristics** to the internal state (e.g *adding personality traits*).
2) **Adding dynamic characteristics** to the internal state. A process must be responsible for updating it. If such process is not defined, see step 4. (e.g *adding stress, constantly updated by a process*).
3) **Adding behaviour** so the characteristics are taken into account in the appropriate processes (e.g *"high stress reduces cooperation"*).
4) **Adding processes** in case where the agent model does not have an adequate process to update a dynamic characteristic or to take into account certain behaviours (e.g *an appraisal process is responsible for evaluating stress.*).

IV. Influences graph and preferences

To illustrate our model, we consider the Replicants cognitive model [4] that we extend with a strategy called "choice by influence and preference graph" (abbreviated as I&P graph) to operationalise behaviours during task selection. An example of influence and preference graph is given on Figure 2.

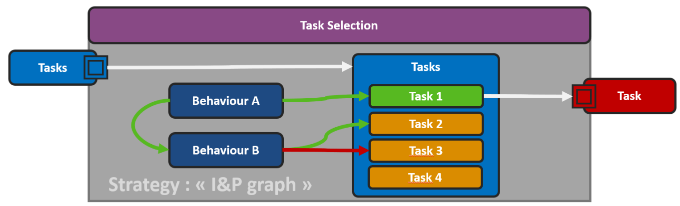

Fig. 2. Task selection process is using a strategy of "choice by I&P graph".

A. Overview

A "task selection" process (see algorithm 1) takes as input a set of candidate tasks T and returns a selected task $t \in T$. The influence graph is a bipartite graph with, in this context, behaviours indicating which tasks are suitable or not, symbolised by positive or negative influences. The score of a task is the sum of the positive $(+1)$ and negative (-1) influences. The task with the highest score indicates the one that most satisfies the behaviours. It is therefore selected and corresponds to the output of the process. Tasks are identified by a name and a set of indicators. This information allows behaviours to reason and decide whether or not they should be favored. The set of indicators depends on the application. We can assume that at least the cost will be present. This graph is automatically constructed from the behaviours taken into account and the candidate tasks.

Algorithm 1 Task-selection process

Input a set of candidate tasks T, a set of behaviours C.
Output $t \in T$

```
influence_graph ← {}
for all c ∈ C do
    for all influence ∈ compute(c, T) do
        add_edge(influence_graph, influence)
highest_tasks ← highest_score(influence_graph)
if size_of(highest_tasks) == 0 then
    return t_0
else
    return random_element(highest_tasks)
```

B. Influence function

Each behaviour is defined as an influence function which defines a list of positive or negative influences towards some elements $t \in T$. The influence value depends on ad-hoc variables in the behaviour (*e.g.* give select the shortest task) and, possibly, the other elements $t' \in T$ (*e.g.* select the cheapest task).

C. Selection

The task with the best score is selected. If several tasks have the best score, a task is randomly selected. If we had a reason to select one of the tasks, then this reason would be a high-level rule, *i.e.* an influence function.

D. Preferences

Also, on its own, influences are not sufficient to describe certain behaviours. This is why we propose to couple them with a preference graph. This graph allows us to refine the choice of the behaviour to select if there is an indecision. The principle of the preference graph is to partially order behaviours, either according to a justification from a cognitive model: *"in a situation of intense stress, a reduction in communication is observed"*, or if the scenario favours a behaviour. For example, they could indicate a preference on one of the tasks. We will therefore select this one, rather than selecting randomly. Like the influence graph, this graph is deduced automatically according to the preference rules established and the behaviours taken into account.

The combination of these two graphs improves the expressiveness of the strategy, which ultimately allows us to operationalise behaviours more finely.

We could have used weights instead of preferences, but this would have been at odds with our goals of modularity and genericity. In particular, instead of specifying preferences between only the behaviours concerned, we would have to adjust the weights of all the behaviours. Also, using preferences is more intelligible than weights.

E. Explainable

As mentioned at the beginning of the paper, we relax the explainable property [9]. This goes beyond our scope. We just try to provide an explanation of the causes of a behaviour, through the cognitive models and the strategies used, during debrief.

To do this, we propose the influence and preference graph mechanism. It is a visual and intelligible mechanism to identify the factors (i.e rules from cognitives models) that led to this behaviour, against other possibilities.

Another mechanism, if the visualisation of the graph is not possible, is to provide a textual explanation that takes the following form: *"The agent did X, because Y, he does Z."* X is the realised behaviour. Y is the information about the internal state (i.e. the profile and the evaluation of the situation) relevant for the rule Z. Z is the high-level rule that led to X. If several cognitive models are linked to the behaviour selection, each one provides their Y and Z.

F. Example

Fig.3 is an example of an influence graph. On the top, there are three behaviours and on the bottom are some tasks. Before detailing how this example works, we will detail the data.

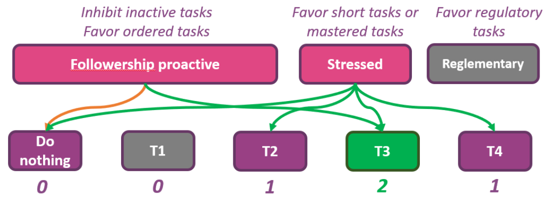

Fig. 3. Diagram of the "choice by influence graph" strategy. In this example, task T3 has the maximum score. It is therefore the one selected.

1) Tasks: Let us assume the following indicators:

- **cost B**: base cost to execute this task
- **cost R**: additive regulatory cost if following the rules
- **Tags**: set of symbols to describe the tasks. Here we use:
  - SCX, where X is the skill level of the PVA. The higher the number, the better.
  - *Ordered*, indicates an order from their leader.

The more advanced the indicators, the more advanced the influence functions can be. Figure IV-F1 describes the tasks used by Fig. 3.

Name	cost B	cost R	Tags
Do nothing	*0*	*0*	*None*
T1	*1*	*0*	*SC3*
T2	*2*	*0*	*SC1*
T3	*3*	*0*	*SC2 - Ordered*
T4	*2*	*5*	*SC2 - Regulatory*

Fig. 4. Set of tasks.

2) Behaviours: Figure IV-F2 describes the behaviours used by Fig. 3.

Behaviour	Activation function	Influence function
Followership Proactive	*is proactive*	$-$ *inactive tasks* $+$ *ordered tasks*
Stressed	*is stressed*	$+$ *mastered tasks* ($\leq$ *to skill*) *or shortest tasks*
Regulatory	*situation is* $\neg$ *urgent* ***xor*** *It is urgent and agent is regulatory*	$+$ *regulatory tasks*

Fig. 5. Set of behaviours.

3) Scenario: Let's say for this example that we have the following parameters. Our agent is: **proactive**, **not regulatory** and **stressed**. The situation is considered to be **urgent**.

4) Description: With this data and scenario, we obtain the influence graph shown in Fig.3. According to the activation rules, the two behaviours "followership proactive" and "stressed" are activated. The "regulatory" behaviour is not activated because the situation is urgent and he is not regulatory. The example is simple enough, not to detail everything. We specify the operation only for the two following tasks:

- "Do nothing" is influenced negatively by "followership proactive" but positively by "stressed" as a short task.
- "T3" is influenced positively by "followership proactive" as this task has been ordered to him and also positively by "stressed" because it is a task that it masters.

G. Discussion

The use of an influence graph is justified by its intelligibility and modularity. Indeed, we can trace which behaviours are responsible for the selection. Moreover, we can add other behaviours, without having to modify the existing behaviours and the impacted process. But, on its own, it is not sufficient to describe certain behaviours. This is why we propose to couple it with a preference graph, to refine the choice of the behaviour to adopt if there is an indecision.

A priori, if we do not take into account the preferences, but rather the influences in the first place, it is because we wish to privilege what the agent prefers, rather than the preferences of external users such as the scenario. However, the opposite is quite possible and could be another strategy.

V. User Experimentation analysis

Before testing the impact of agent behaviour on training for collaborative activities and the ease of use of the tool, we set up a preliminary experiment. We tested an application of our model. We wish to ensure that generated behaviours remain relevant when multiple cognitive models are combined (H1). We also hypothesised that the users would be able to differentiate the generated profiles (H2), and that an explanation brought from the models we used helps to understand the presented behaviour (H3).

The experiment was based on the following written scenario: a doctor diagnoses a patient, concludes s/he need an injection of adrenaline and asks a nurse to prepare the injection. The nurse realises that the patient actually needs morphine, reacts to the doctor's orders, prepares an injection and reports their actions.

The nurse's decision-making and actions vary depending on their attributed profile, determined by a combination of models we would use for the AVCs.

A. Protocol

Subjects were randomly given an alternate version of the scenario, each of these corresponding to a different behaviour for the nurse, characterised by four keywords: passive/proactive, (un)communicative, (un)stressed and (un)skilful. All behavioural profiles were defined as skilful in this experiment, so we had eight variations matching the remaining combinations. After reading the scenario, the subjects were asked to evaluate the believability of the nurse's behaviour and the correspondence to the three variable behavioural keywords. Then, subjects were given explanations on the behaviours, either justified by the chosen models, or simple and naive. The subjects are asked to evaluate the relevance of the given explanation.

All questions were presented as Likert scales, and the subjects could justify every answer they gave.

The experiment was held as an online survey. Participants were automatically allocated to one of the sixteen forms (eight profiles and two types of explanations).

We used G*Power [22], [23] to estimate the needed number of participants among our sixteen groups, with a medium effect size.

B. Results

The subjects were recruited from a pool of students and lab staff. 174 entries have been registered, with at least ten participants per group. We attributed integer values ranging from -2 to 2 on the Likert scales, -2 being complete disagreement and 2 being complete agreement.

a) H1: The mean over all groups concerning the believability of the nurse's behaviour is -0.16. We actually observe the only positive means in the groups corresponding to a communicative & proactive profile, with a global 1.45, against the rest being at -0.70.

b) H2: Regarding the evaluation of the behavioural keywords, we observed very polarised results for the (un)communicative and passive/proactive traits, the absolute value of the median being over 1 for all groups, and actually at 2 for most of them. All these answers match with the defined behaviour, except for the communicative & passive trait combination, where the (un)communicative trait has been evaluated as uncommunicative. However, the (un)stressful trait has a mean of zero with a wide standard deviation of 1, with most results being 0.

c) H3: On average, the relevance of the model-based explanations was rated 0.82, and the naive ones were rated 0.40. The difference proved to be significant enough with our statistical sample ($p = 0.031 < 0.05$).

C. Interpretations

a) H1: First, in terms of perceived behaviour, the majority of people did not find the behaviour of the passive scenarios convincing. However, this is the case for the scenarios where the nurse was proactive&communicative. One explanation would be that we misrepresented the urgency of the situation and the context, which is difficult without VE and introducing biases. According to user feedback, less extreme behaviour would have been preferable.

b) H2: Then, at the level of the perception of the behaviours, the values obtained show that the users perceive the differences in behaviours between the profiles (criteria 6.1), except for the stress characteristic, which is difficult to capture in text, without giving too much information and introducing a bias. A VE would have been more appropriate, as we discuss in the next section.

c) H3: Finally, as far as explanations are concerned, they have a better overall acceptance than naive explanations (H3). However, the difference doesn't seem significant enough to conclude.

D. Discussions

For this first experiment, we were mainly interested in whether the difference in behaviour was well perceived (criteria 6.1), which is the case. Once the work is more advanced, we would like to make the experiment more complex and test it in a more meaningful way. In particular, we wish to extend the duration of the simulation to observe long-term behaviour and increase the number of possible behaviours. Ideally, we would like to run the experiment from a VE. The user could interact with the simulation and take part in the scenario. A observation from this experiment, is that the urgency of the situation and behaviours (especially stress) could have been better conveyed with an VE than just text, as the means of comprehension would have been closer to reality, especially regarding perception of the different details of the scenario.

VI. Conclusions & Perspectives

Developing training environment using cognitive virtual agents in a collaborative context is a complex and time consuming task. To facilitate the work of modelling these agents, we proposed a meta-model to generate an agent model tailored to the needs of an application, *i.e* a set of behaviours and cognitive models relevant to the domain. The originality of this meta-model is to make the cognitive models, used for explainability, and the agent model completely independent, to avoid repeated implementations of various behaviours. We formalised the processes that compose this model, by introducing the notion of strategy. A strategy defines how to take into account several behaviours. This reinforces the genericity but also the modularity depending on the strategy used.

We have applied this approach to generalise the operationalisation of behaviours during the selection of tasks, using a strategy called "choice by influence and preference graphs". This strategy has the advantage of being modular and explainable, at least by its intelligibility from its visual nature or by explanations that can be constructed from the graphs. A requirement for our agents is the ability to express various explicable behaviours, induced from several factors from cognitive models, such as stress or personality. A preliminary evaluation showed that users perceived the difference in behaviour between different agents in the same context. It also showed that the explanations provided by the cognitive models gave a better understanding of the agents' behaviours.

Our future work will consist in applying this approach to operationalise behaviours from Driskell's studies [20]. In particular, we would like to operationalise stress, flexibility and adaptability behaviours. Before that, we need to continue to work on the meta-model, especially on the sequence of processes. Afterwards, the next step is to test with designers whether the tool facilitates this work and whether these agents improve the training of socio-technical skills.

Acknowledgment

Work funded by the ORCHESTRAA DGA RAPID project. We thank Adrien CHARANNAT for his help in setting up the experiment.

References

[1] L. Callebert, D. Lourdeaux, and J.-P. Barths, "Trust-based decision-making system for action selection by autonomous agents," in *2016 IEEE 20th International Conference on Computer Supported Cooperative Work in Design (CSCWD)*, May 2016, pp. 4–9.

[2] L. Huguet, D. Lourdeaux, and N. Sabouret, "Prsentation du projet VICTEAMS," in *Workshop affects, compagnons artificiels et interaction (WACAI 2016)*, Brest, France, 2016.

[3] B. G. Silverman, D. Pietrocola, B. Nye, N. Weyer, O. Osin, D. Johnson, and R. Weaver, "Rich socio-cognitive agents for immersive training environments: case of NonKin Village," *Autonomous Agents and Multi-Agent Systems*, vol. 24, no. 2, pp. 312–343, Mar. 2012.

[4] L. Huguet, D. Lourdeaux, and N. Sabouret, "Moteur de slection de tches pour des personnages virtuels autonomes non omniscients," in *Workshop affects, compagnons artificiels et interaction (WACAI 2018)*, Porquerolles, France, Jun. 2018.

[5] B. Parasumanna Gokulan and D. Srinivasan, "An Introduction to Multi-Agent Systems," in *Studies in Computational Intelligence*, Jul. 2010, vol. 310, pp. 1–27, journal Abbreviation: Studies in Computational Intelligence.

[6] T. Driskell, J. E. Driskell, and E. Salas, "Mitigating Stress Effects on Team Cohesion," in *Team Cohesion: Advances in Psychological Theory, Methods and Practice*, ser. Research on Managing Groups and Teams. Emerald Group Publishing Limited, Jan. 2015, vol. 17, pp. 247–270.

[7] G. Demary, "valuation cognitive du leader dans une dyade hirarchique : des comportements non verbaux du suiveur aux comportements de leadership," *Bulletin de psychologie*, vol. Numro 569, no. 5, pp. 271–275, Nov. 2020.

[8] C. Faur, "Approche computationnelle du regulatory focus pour des agents interactifs : un pas vers une personnalit artificielle," These de doctorat, Universit Paris-Saclay (ComUE), Oct. 2016.

[9] T. Chakraborti, A. Kulkarni, S. Sreedharan, D. E. Smith, and S. Kambhampati, "Explicability? Legibility? Predictability? Transparency? Privacy? Security? The Emerging Landscape of Interpretable Agent Behavior," *Proceedings of the International Conference on Automated Planning and Scheduling*, vol. 29, pp. 86–96, Jul. 2019.

[10] I. Millington, *AI for Games*, 3rd ed. Boca Raton: CRC Press, Mar. 2019.

[11] K. Erol, J. Hendler, and D. Nau, "HTN Planning: Complexity and Expressivity," *Proceedings of the National Conference on Artificial Intelligence*, vol. 2, May 1994.

[12] J. E. Laird, A. Newell, and P. S. Rosenbloom, "SOAR: An architecture for general intelligence." Standford, Tech. Rep., 1986.

[13] J. R. Anderson, "ACT: A simple theory of complex cognition." *American psychologist*, vol. 51, no. 4, p. 355, 1996, publisher: American Psychological Association.

[14] C. Adam and B. Gaudou, "BDI agents in social simulations: a survey," *The Knowledge Engineering Review*, vol. 31, no. 3, pp. 207–238, Jun. 2016.

[15] M. Bratman, *Intention, Plans, and Practical Reason*. Center for the Study of Language and Information, 1987.

[16] H. Jones, J. Saunier, and D. Lourdeaux, "Personality, Emotions and Physiology in a BDI Agent Architecture: The PEP-> BDI Model," in *Proceedings of the 2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology-Volume 02*, 2009, pp. 263–266.

[17] M. Bourgais, "Vers des agents cognitifs, affectifs et sociaux dans la simulation," phdthesis, Normandie Universit, Nov. 2018.

[18] I. Chaleff, "The Courageous Follower: Standing Up to And for Our Leaders," *Nassp Bulletin*, vol. 81, pp. 119–119, Feb. 1997.

[19] R. Kelley, "In praise of followers," *Harvard Business Review*, vol. 66, pp. 142–148, Jan. 1988.

[20] T. Driskell, E. Salas, and J. E. Driskell, "Teams in extreme environments: Alterations in team development and teamwork," *Human Resource Management Review*, vol. 28, no. 4, pp. 434–449, Dec. 2018.

[21] J. P. Morrison, "Flow-based programming," in *Proc. 1st International Workshop on Software Engineering for Parallel and Distributed Systems*, 1994, pp. 25–29.

[22] F. Faul, E. Erdfelder, A.-G. Lang, and A. Buchner, "G*Power 3: A flexible statistical power analysis program for the social, behavioral, and biomedical sciences," *Behavior Research Methods*, vol. 39, no. 2, pp. 175–191, May 2007.

[23] F. Faul, E. Erdfelder, A. Buchner, and A.-G. Lang, "Statistical power analyses using G*Power 3.1: Tests for correlation and regression analyses," *Behavior Research Methods*, vol. 41, no. 4, pp. 1149–1160, Nov. 2009.