




# Artificial intelligence moving serious gaming: Presenting reusable game AI components

Wim Westera<sup>1</sup>  · Rui Prada<sup>2,3</sup> · Samuel Mascarenhas<sup>2</sup> · Pedro A. Santos<sup>2,3</sup> · João Dias<sup>2,3</sup> · Manuel Guimarães<sup>2,3</sup> · Konstantinos Georgiadis<sup>1</sup> · Enkhbold Nyamsuren<sup>1</sup> · Kiavash Bahreini<sup>1</sup> · Zerrin Yumak<sup>4</sup> · Chris Christyowidiasmoro<sup>4</sup> · Mihai Dascalu<sup>5</sup> · Gabriel Gutu-Robu<sup>5</sup> · Stefan Ruseti<sup>5</sup>

Received: 13 December 2018 / Accepted: 5 July 2019 / Published online: 30 July 2019  
© The Author(s) 2019

## Abstract

This article provides a comprehensive overview of artificial intelligence (AI) for serious games. Reporting about the work of a European flagship project on serious game technologies, it presents a set of advanced game AI components that enable pedagogical affordances and that can be easily reused across a wide diversity of game engines and game platforms. Serious game AI functionalities include player modelling (real-time facial emotion recognition, automated difficulty adaptation, stealth assessment), natural language processing (sentiment analysis and essay scoring on free texts), and believable non-playing characters (emotional and socio-cultural, non-verbal bodily motion, and lip-synchronised speech), respectively. The reuse of these components enables game developers to develop high quality serious games at reduced costs and in shorter periods of time. All these components are open source software and can be freely downloaded from the newly launched portal at [gamecomponents.eu](http://gamecomponents.eu). The components come with detailed installation manuals and tutorial videos. All components have been applied and validated in serious games that were tested with real end-users.

**Keywords** Serious games · Artificial intelligence · Software reuse · Game development · Component-based architecture · Intelligent tutoring systems

## 1 Introduction

Computer games have been linked with artificial intelligence (AI) since the first program was designed to play chess (Shannon 1950). The challenge to defeat human expert players in rule-based strategy games such as Chess, Poker and Go has greatly

---

✉ Wim Westera  
wim.westera@ou.nl

advanced the domain of AI research, affecting breakthroughs in e.g. computational intelligence, algorithms, machine learning, and combinatorial game theory (Fujita and Wu 2012). In turn, such new AI methods have been used in computer games, for instance to enhance graphical realism, to generate levels, sceneries and storylines, to establish player profiles, to balance complexity or to add intelligent behaviours to non-playing characters (NPC; Yannakakis and Togelius 2015, 2018).

Over the years, however, various authors (Champanand 2004; Bourassa and Massey 2012; Yannakakis 2012; Yannakakis and Togelius 2018) have pointed at the marginal penetration of academic game AI methods in industrial game production. This limited uptake has been attributed to 1) research projects largely focusing on advanced, but non-scalable projects of little commercial or practical value, and 2) game studios reluctant to adopt and include promising but risky AI techniques (such as neural networks) rather than established, fully scripted technologies in their games. The game industry's reticence to embrace advanced AI may partly be explained by the manifest failure of AI during the 1980's and 1990's to live up to their promises of enabling expert systems and intelligent dialogue. Ever since, the marriage between AI and gaming has appeared brittle, which is readily attributed to the limited interconnection and exchange between research and industry.

Research policy makers and politicians both at national and international levels have recognised that the transfer of knowledge and technologies from research and development organisations to societal sectors in order to create economic and social value, is a fundamental problem that should be urgently addressed. This failure is generally known as the “knowledge paradox” (European Commission 1995), referring to the fact that in many countries increased public investments in science and technology do not translate into economic benefits and job creation, while leaving many scientific findings unused. The process of knowledge valorisation often fails, which is painful as such in the case of games for learning, because of their dual role in both innovating the domain of education and contributing to raised skills levels in other content domains. This failure has been manifestly the case for the game industry, in particular the serious game industry (developing games for serious purposes rather than entertainment), since the serious game industry sector is composed of a large number of small independent studios (Stewart et al. 2013) that lack the scale and capacity to easily access new research knowledge and technologies and include these in their projects. Having recognised the potential of games for teaching and training and other societal sectors, the European Commission has stimulated diverse collaborations between game research and game industry. This article presents the main outcomes of the RAGE project, which has been the principal and most sizable research and innovation project in the European Horizon 2020 funding programme addressing serious games. Its goal and research assignment has been to investigate how a framework of reusable, intelligent game software components should be devised to structurally accommodate technology transfer from game research to the game industry, to assess and validate the outcomes, and propose measures for sustained societal impact. Rather than addressing AI per se, the research essentially focuses on the opportunities of the practical application of AI in serious games.

This article is the very first aggregate publication of the RAGE research programme carried out by over 130 researchers from all over Europe. We will first summarise related work of recent advances in game AI. Next, we will briefly explain the proposed

game software component architecture. Then, we will describe and explain a selection of AI-driven game software components. These address functionality for player modelling (real-time facial emotion recognition, automated difficulty adaptation, stealth assessment), natural language processing (sentiment analysis and essay scoring on free texts), and believable non-playing characters (emotional and socio-cultural, non-verbal bodily motion, and lip-synchronised speech), respectively. In conclusion, key results from the various application pilots will be summarised and discussed in the light of the anticipated knowledge and technology transfer mechanism for the serious gaming community.

## 2 Related work in game AI

Game design is essentially about creating valuable interactive experiences for the players. These experiences are effected by a variety of orchestrated game elements, including narratives, challenges, graphical representations, sounds, timing of events and phenomena, and the entities that directly interact with the player, be it opponents, allies, or other objects in the game environment. AI techniques will become indispensable to coordinate the ever-growing complexity and dynamics of games. From a pragmatic perspective, game developers are happy to use ad hoc cheats that offer players the illusion of intelligence, instead of any deep intelligence (Rabin 2017). This can work well until extended interaction reveals the tricks used breaking down the game experiences. As hardware capabilities improve, new types of interaction will emerge that would need better AI. In recent years, AI in games has improved appreciably (Lewis and Dill 2015). Below, we refer to example usage of advanced AI techniques in mainstream commercial games, mostly applied to control NPC behaviours. Artificial Neural Networks with 3 layers were used in the real-time strategy game *Supreme Commander 2* [Gas Powered games, 2010] to control platoons' reaction to encountering enemy units (Robbins 2013). From robotics research, systems for collision avoidance based on Reciprocal Velocity Obstacle (RVO) techniques (Van den Berg et al. 2011) have been made available as libraries and found their way into *Warhammer 40,000: Space Marine* [Relic, 2011] and many other games. The game *Guild Wars 2: Heart of Thorns* [ArenaNet, 2015] used an advanced utility-based decision architecture to solve problems related to tactical movement and skill use selection for its Non-Playing Characters (NPC) (Lewis 2017). *Forza Motorsport 5* [Turn 10 Studios, 2013] and its successors gather data about how players drive, that is then processed using Machine Learning techniques. This allows for the creation of “drivatars” that mimic a specific player driving style, and can then be used to play against. A similar goal was attained in the *Killer Instinct* fighting game [Iron Galaxy Studios, 2014] using case-based reasoning. For other uses of Machine learning techniques in games, the survey by Nguyen et al. (2015) provides a good start. Multi-agent systems have been suggested as powerful solutions to intelligent NPCs (Dignum et al. 2009). However, real-time synchronisation of many agents acting autonomously, for instance in battlefield games such as the *Call of Duty* series [Infinity Ward, from 2003], easily produce performance problems. Several advanced techniques for optimisation such as flow field and congestion concepts (Pentheny 2013, 2015), context steering (Fray 2015), and even robotics-inspired Velocity Obstacle techniques (Guy and

Karamouzas 2015) have been applied. The blockbuster game Grand Theft Auto V [Rockstar North, 2013] uses multi-agent based architectures for the simulation of subsystems, as do most of the real-time modern strategy games, like those based on the Clausewitz Engine from Paradox Development Studios.

Adaptive gameplay has been accommodated with various algorithms used for matching two human players, such as TrueSkill (Herbrich, Minka and Graepel 2006) and variations of Elo (Elo 1978), or matching game task difficulty to players' skill, such as the Computerized Adaptive Practice algorithm (Klinkenberg et al. 2011). Hierarchical Task Networks (HTN) were used in the planner implemented for the third person shooter Transformers: Fall of Cybertron [HighMoon, 2012] (Humphreys 2013).

Advances in Natural Language Processing (NLP) have opened up new opportunities to support natural dialogues with NPCs, either companions or enemies, and to support interactive storytelling (Yannakakis and Togelius 2018). In the multiplayer online battle arena League of Legends [Riot Games] NLP trained models have been used to recognise and remove toxic behaviour from the player chat channels (Maher 2016).

Generally, these applications of AI are proprietary solutions bound and tuned to a particular game and not accessible and reusable by other parties. Also, their application in serious games has been quite limited.

### 3 Platform-independent game AI

#### 3.1 Lightweight game software reusability framework

Given the diversity of software platforms, programming languages, browsers and operating systems, favourable conditions for the reuse of software by game developers should be accommodated by a shared architectural framework. The main starting points for the architecture include: 1) Extensibility (The architecture should be robust over extending the set of components with new software components, 2) Addressing platform and hardware dependencies (Direct access to the operating system should be avoided; a conservative approach as to avoid browser version issues as much as possible), 3) Portability across game engines and programming languages, 4) Avoiding dependencies on external software frameworks and libraries (such as jQuery or MooTools for JavaScript), 5) Neutrality with respect to different software design methodologies (the development process), 6) Neutrality with respect to game genre, design and style (avoiding direct access to the interface; components just provide smart functionality under the hood), and 7) Truly lightweight (easy to use in different operational contexts). In close collaboration with game industry representatives, a component-based design framework (Bachmann et al. 2000) has been developed (Van der Vegt et al. 2016a, b). Although the architecture is self-contained and supports component-to-component communication its application context is generally driven by a game engine that can access the component's functionality once the component or its service is declared and integrated in the engine. Client-side plug-in characteristics of software components are created by relying on well-established coding practices and software patterns that procure abstraction, viz. decoupling functionality and its implementation. Most notable software patterns used for communication with the game engine are the Bridge pattern and the Broadcast/Publish/Subscribe pattern (Gamma

et al. 1994); Birman and Joseph 1987.). Remote communications with server-side components are covered by web services. The architecture has been extensively tested and validated in connection with a wide diversity of development tools, target platforms and programming languages that are being used in practice (Van der Vegt et al. 2016a).

### 3.2 The gamecomponents.eu portal

The recently launched gamecomponents.eu portal funded by Horizon2020 is the technical platform for exchanging advanced game technologies and associated resources: it accommodates an open marketplace, which is driven by the RAGE Foundation. Notably, the portal offerings are fully platform independent, while – in contrast - existing game portals are either driven by commercial game platform vendors (e.g., Unity, Unreal from Epic, CryEngine from CryTek), by vendors of other creative software tools (e.g., Adobe), or by general media stock asset marketplaces (e.g., [graphicriver.net](http://graphicriver.net)). Moreover, existing portals focus mostly on media assets (e.g. 3D objects, textures, sounds) rather than software. Also, the gamecomponents.eu portal specifically targets serious games, while other platform primarily address leisure games. Nevertheless, leisure games could also benefit from the technologies exposed on our portal. Figure 1 shows a screenshot of a software catalogue page, revealing taxonomy-based filtering, keyword search and a results section displaying available game components. Software developers can describe and submit their contribution through a component-authoring widget that provides a stepwise guidance through the submission process, allowing for entering the software or software references (e.g.

Fig. 1 A screenshot of the software catalogue at the marketplace portal (gamecomponents.eu)

Github), its metadata (a specific game component metadata schema was designed for this (Georgiev et al. 2016)), and supporting artefacts such as technical documents (installation guides), training materials (e.g. video tutorials), and marketing materials. In addition, it offers an interactive stakeholder map, a set of tools for taxonomy management, training course creation and eCommerce management, and it uses open-ID user management, offers social API for the exchange with social networks (e.g. Slideshare, Mendeley) and includes a rating system based on scores by end-users. The portal is available at [gamecomponents.eu](http://gamecomponents.eu).

### 3.3 First batch of public game AI components

An initial set of about 40 game components are exposed on the portal. The components exposed at the portal cover a wide range of AI-based functionalities that are relevant for serious game development, including personalisation, game difficulty balancing, assessment, player analytics, competence modelling, social gamification, language technologies and affective computing, among other topics. All components are open source and free of charge. Most client-side components, which need to be directly integrated with the game engine, are available in C#, while some also offer versions in JavaScript (TypeScript), or C++. Additional conditions to promote the adoption and reuse of the software have been met: 1) successful integration has been demonstrated with various game development environments (e.g. Unity, Xamarin, Cocos, Mono), 2) the integration in games is easy, 3) all components have been used and tested in real games with real end-users to provide empirical evidence of practicability; 4) the components have been enriched with ample documentation, tutorials, demos, research articles and evaluation results; 5) they use the highly flexible Apache 2.0 license (white label software), which allows for reuse by third parties both for commercial and non-commercial purposes, either under open source or closed source conditions. To further the viability and sustainability of the marketplace portal and to attain critical mass of relevant game software, third-party providers – either game research projects or IT-oriented companies - are expressly being invited to port their game software, whether or not compliant with the component-based design framework, onto the portal.

## 4 Selected game AI components

### 4.1 AI key areas

This section presents a selection of reusable game AI components that have been made available at the [gamecomponents.eu](http://gamecomponents.eu) portal. The selection focuses on Player Experience Modelling (PEM), Natural Language Processing (NLP), and advanced Non-Playing Character modelling (NPC), respectively, all of which are among the flagships of game AI research (Yannakakis 2012; Yannakakis and Togelius 2015). Their relevance for serious games is readily explained by the pedagogical frame of teaching, which assumes a teaching agent (cf. NPC) that frequently probes and assesses the learner's mental states (cf. PEM) and, when

needed, engages in a supportive dialogue with the player (cf. NLP) to provide guidance or feedback.

#### 4.1.1 PEM: Player experience modelling

PEM can be based on a variety of player data, including the player's behavioural and performance data from the game (e.g. speed, score, decisions) and multiple player-related input modalities such as speech data (intonation, text), images (pupillometry, gaze tracking, gesture and bodily movement tracking), or physiological signals (EEG, respiration, blood volume pulse, skin conductance). So far, however, the capturing of physiological signals has been problematic since it requires hardware (sensors) that are often too obtrusive and unpractical for continuous application. In this article three unobtrusive PEM-related AI components will be presented:

- Real-time facial emotion recognition
- Adaptation and assessment
- Stealth assessment

#### 4.1.2 NLP: Natural language processing

Natural language processing is the field of AI focused on the understanding, interpretation and manipulation of human language by computers. It allows the computer to assess any textual messages or documents sent by the player, and thereby it allows to respond to these automatically in a meaningful way. So far, NLP has scarcely been used in games. The following NLP services will be presented:

- Natural language processing: sentiment analysis
- Natural language processing: automated essay grading

#### 4.1.3 NPC: Non-playing characters

Game AI for NPCs has a longstanding history, in particular focused on navigation and other low levels of control (Yannakakis 2012). Recent research, however, has been focusing on a variety of high level NPC behaviours that should effect more flexible, believable, knowledgeable, human-like, and intelligent behaviours, including realistic bodily motion, NPC emotion modelling, and compliance with socio-cultural conventions. This reflects a more holistic perspective on the NPC capable of flexible responses, as opposed to fully scripted applications. The following NPC components will be presented:

- Role play character: emotion appraisal and social importance dynamics
- Nonverbal bodily motion: behaviour mark-up language
- Nonverbal bodily motion: lip-synchronised speech



## 4.2 PEM: Real-time facial emotion recognition

### 4.2.1 Emotion recognition

Artificial Emotional Intelligence (AEI), which is also known as emotion recognition or emotion detection, is a technology that extracts human emotions from displayed behavioural or physiological features (Schuller and Schuller 2018). Human facial expressions have demonstrated to produce the most informative data for computer awareness of emotions (Sebe 2009), outperforming approaches that make use of either speech and vocal intonations, physiological signals, body gesture and pose, text, or combinations of two or more of these approaches (Bahreini et al. 2016). So far, the use of emotion recognition functionality has not been a feasible option well within reach of game developers, because of the complexity of the implementation involved, limited accuracy, problems with facial hair and glasses, specific requirements with respect to lighting conditions, extensive post-processing and some more (Pantic et al. 2005). The real-time facial emotion recognition component, created by the Open University of the Netherlands, solves many of these problems (Bahreini et al. 2018).

### 4.2.2 Relevance for learning and teaching

Emotions are a significant influential factor in the process of learning, as they affect memory and action (Pekrun 1992). Any classroom teacher would take into account the emotional states of learners during the lessons. In computer-based learning, however, the learner's emotion has been systematically neglected as a learner model variable, because it was hard, if not impossible to detect. Now that emotion recognition technology is becoming available and accessible, learner models in serious games can include the learners' emotions and thereby improve the quality of personalised guidance and feedback. Also, the players' emotions can become part of the learning content or game scenario, for instance in games for communication training, conflict management, or actor training (Bahreini et al. 2017). Finally, emotion recognition can be used to collect emotion data during play testing.

### 4.2.3 AI approach

This software component uses artificial emotional intelligence to unobtrusively cover unbiased facial expressions of emotion from any image, either from a still, a video file, a video stream or a webcam. The technology uses a combination of fuzzy logic rules and machine learning. The fuzzy logic AI algorithm uses unordered fuzzy rule induction (FURIA algorithm; Hühn and Hüllermeier 2009), which is trained with a reference set of recorded emotions. It detects happiness, sadness, surprise, fear, disgust, anger and the neutral face, with accuracy above 80%, which compares or even outperforms human experts. Alternative machine learning approaches, such as neural networks, Bayesian networks, and decision trees are less practical for real-time operation as they require extensive processing, while offering weaker performance.



#### 4.2.4 Application cases

A usage example of the real-time facial emotion recognition component would be the Jobquest game, which offers a job application interview training (Gutu et al. 2018). During the job interview, the players should control their manifest emotions and never display anger, fear or disgust. During the exercise, they receive direct on-screen feedback about the displayed emotion through the player's webcam shot and the associated emoticon (upper left corner in Fig. 2).

Emotion recognition has also been used for communication training in the Communication Advisor game (Bahreini et al. 2017). This game places the players in a variety of real life situations to which they have to respond via a natural dialogue. Feedback to players is based on their facial expressions.

#### 4.2.5 Technical considerations

The real-time facial emotion detection is a client side software component that is to be integrated in the game engine. While using the player's personal webcam, it detects emotions in real-time. It returns a string value representing the seven basic emotion classes, which can be used for further processing in the game. It can also process a single image file, or a recorded video file. Also, presence of multiple players in one shot can be accommodated as it can detect multiple faces and interpret their emotions at the same time. It can easily be integrated in many game engines, including, for instance Unity3D.

### 4.3 PEM: Adaptation and assessment (TwoA)

#### 4.3.1 Game balancing

The TwoA software component offers a dynamic game difficulty balancing mechanism, which automatically matches the difficulty of the player's task to the player's skill



Fig. 2 Direct emotion feedback in the Jobquest application interview game (upper left corner)

(Nyamsuren et al. 2017). Game difficulty balancing is deemed an essential mechanism to preserve player engagement, improve player motivation, and improve the overall gameplay experience. Gee (2003) suggests that the secret of a video game is not in the fancy, high quality, immersive 3-D graphics, but it is in the underlying mechanism that balances the challenges offered to the player with the players' abilities "...seeking at every point to be hard enough to be just doable". Dynamic game difficulty balancing avoids both frustration of the player (when tasks are too complex) and boredom (when tasks are too easy). This also holds for teams: difficulty balancing is commonly applied in online multiplayer games such as the first-person shooter series Halo and multiplayer online battle arena games like League of Legends to ensure that opposing teams are evenly matched in terms of skills (Claypool et al. 2015). While the adaptation mechanism is often mistaken and confused with a simple if-then-else structure or a level closure, it should incorporate a sophisticated self-adjusting optimisation algorithm that frequently reiterates both task difficulty and skills mastery. The implementation and testing of such algorithm is anything but straightforward. The adaptation and assessment component created by the Open University of the Netherlands offers a fully automated, self-adjusting balancing algorithm that exposes superior reliability and stability. It comes as an easy to use software component that can be readily integrated in various game engines.

#### 4.3.2 Relevance for learning and teaching

The engaging capabilities of games are to be largely attributed to the process of game balancing. By avoiding frustration and boredom and offering doable challenges a well-balanced serious game enhances and preserves learner motivation, which is a principal determinant of learning. In accordance with the Zone of Proximal Development theory (Vygotsky 1978), a real-time adaptation of the game difficulty enables a smoother learning experience. The AI algorithm controls difficulty so that the player is challenged to improve a skill or acquire new knowledge without facing overly difficult tasks beyond player skill level. As a result it produces an optimised learning curve, since it iteratively re-assesses the player's skills mastery and continually adjusts task difficulty to the appropriate level. This means that the learning process becomes highly efficient: progression is optimised, while no time is wasted on tasks that do not contribute to learning. Continued assessment of the player's skill serves as a form of formative learning analytics that can be used by the player or the teacher to monitor learning progress and identify potential learning barriers (Hofman et al. 2018). Finally, game difficulty assessment also enables the analysis and optimisation of the game's learning content (Nyamsuren et al. 2018a).

#### 4.3.3 AI approach

The AI algorithm developed for this adaptation software component is strongly rooted in the Elo rating system that was originally developed to assess chess players' skills (Elo 1978). This also holds for one of the widely-known examples of such balancing algorithms, namely TrueSkill (Herbrich et al. 2006) developed by Microsoft. However, TrueSkill was designed specifically to assess and match players in large-scale commercial online games. Another example would be the Computerized Adaptive Practice

system (CAP), which was specifically developed to assess player skill in a serious game (Klinkenberg et al. 2011). It extends the Elo algorithm with methods from Item Response Theory (IRT; Lord and Novick 1968). The methods from the Elo system enable CAP to (re)estimate both the player skill and the game difficulty based on the player's real-time performance. In turn, the methods from IRT enable CAP to adapt the game difficulty based on the player skill using the previously estimated ratings. The Adaptation and Assessment component presented here adds several theoretical and practical improvements to CAP. TwoA considerably improves CAP's adaptive capabilities by minimising selection bias that may be present while choosing an appropriate difficulty level. This is achieved by expanding the IRT methods with fuzzy logic (Hühn and Hüllermeier 2009). Multiple selection criteria can mitigate the selection bias, and fuzzy logic allows to combine these criteria into a single selection rule. As a result, the improved algorithm is more robust and accurate than CAP especially during the calibration period when true skill and difficulty ratings are not well approximated (Nyamsuren et al. 2018b).

#### 4.3.4 Application case

The effectiveness of CAP was extensively demonstrated with its application in Math Garden ([www.mathsgarden.com](http://www.mathsgarden.com)), a popular Dutch serious gaming platform addressing primary school children (Van der Maas and Nyamsuren 2017). Publicly available data from a Math Garden game collected from over 1500 Dutch schools featuring 87,000 unique players were reused to validate the improved performance of the algorithm of the adaptation and assessment component. The TwoA component has also been used in an entrepreneurial skills training game (Hatch) at Hull College.

#### 4.3.5 Technical considerations

From a practical perspective, this adaptation and assessment component offers an open-source, highly portable, and easy-to-use implementation of the AI algorithm. As a reusable component compliant with the RAGE architecture, it can be easily integrated with the most modern game engines (Van der Vegt et al. 2016a). The component hides all the complexities of the algorithm behind a simple interface. Apart from the management of player and game data, its operation requires only two method calls from the game to the component.

### 4.4 PEM: Stealth assessment

#### 4.4.1 Using log game data for assessment

Stealth assessment is a promising methodology for applying formative assessments in serious games to unobtrusively assess the players' knowledge or skills mastery based on the player's behaviours and decisions in the game (Shute 2011). This means that behavioural data (e.g. log files) are analysed at a certain point in time to determine the player's mastery without the need for explicit tests, e.g. multiple choice questions. In practice, however, the application of stealth assessment in serious games is a complex and time-consuming process (Moore and Shute 2017). Therefore, its uptake has been

below par as yet. The generic tool provided by the Open University of the Netherlands removes many of the practical barriers for applying stealth assessment, as it has largely automated the many data processing steps that so far need to be handled manually (Georgiadis et al. 2018).

#### 4.4.2 Relevance for learning and teaching

Games are expressly suited for the acquisition of highly contextualised, tacit knowledge and action-bound skills, which are notably hard to capture in formal tests and exams. Cases in point would be social skills, communication skills, group moderation skills, but also competencies such as persistence, creativity, self-efficacy, teamwork and the wider collection of twenty-first century skills, all of which are deemed essential for successful future careers and presupposing a strong link with concrete action (Dede 2010). Given this tacit knowledge dimension, the assessments should not be administered (solely) as separate oral or written assignments, but instead should be directly based on the activities displayed. Stealth assessment provides an attractive alternative to the existing de-contextualised assessment methods by linking the assessment directly to the practical use of knowledge and skills in relevant situations. Moreover, these situations should entail scenarios that require the application of various competencies at the same time. This is exactly what serious games are capable of providing.

#### 4.4.3 AI approach

Stealth assessment uses machine learning technology to provide probabilistic reasoning over the learners' knowledge and skills levels by exploiting meaningful data which is collected during gameplay. Stealth assessment combines two main ingredients: 1) the Evidence-Centered Design (ECD; Mislevy 2011), and 2) machine-learning (ML) algorithms. ECD is a conceptual assessment framework that can be used to express the statistical relationships between competency constructs, in-game observables, and in-game tasks. As for the machine learning algorithms, originally Bayesian Networks were used (Shute 2011) although alternative solutions have also been examined (Decision Trees, Support Vector Machines, and Deep Learning) (Sabourin et al. 2013; Min et al. 2015). The new, generic application for stealth-assessment presented here allows the user to 1) define and configure ECDs, 2) import numerical data from log files deriving from any serious game, and 3) declare desirable machine learning optimisations (e.g. select the preferred machine learning algorithm type and its inner options). Thereby the need for specific machine learning expertise is minimised as the tools cover machine learning functions automatically. The tool produces detailed output for both students' performances and the machine learning algorithms' performances for evaluation purposes.

#### 4.4.4 Application case

So far, stealth assessment has been proven to be robust for assessing several competencies in serious games, such as qualitative physics (Shute et al. 2013), persistence (Ventura et al. 2014), and problem-solving skills (Shute et al. 2016). The application has been rigorously tested and validated with a large volume of simulation data sets

including different competencies, in particular with respect to stability, accuracy and robustness under conditions of normality violation. All accuracies are well in the 95% range. As a next step practical validation with authentic game data is anticipated.

#### 4.4.5 Technical considerations

The stealth assessment component is currently available as a console application. It was coded in C# using the .NET framework and it functions as a stand-alone client-side console application. It includes various data reformatting procedures. It makes use of ML libraries from the Accord.NET framework. On top of the console application a graphical user interface is being developed, including a wizard that supports the workflow and assists the user (e.g. serious game developer, educator) at tuning and optimising assessment settings.

### 4.5 NPC: Sentiment analysis

#### 4.5.1 ReaderBench sentiment analysis

The ReaderBench framework (Dascalu et al. 2013), developed by University Politehnica of Bucharest, is a multi-lingual, advanced text analysis framework that offers a wide variety of NLP functionalities. Sentiment analysis (also referred to as opinion mining) consists of the automated extraction of subjective information related to human feelings and opinions from natural language texts (Liu 2012). It provides insights towards users' perceptions by interpreting information about the polarity of a text (i.e., how positive or negative) and by identifying emotions expressed within it.

#### 4.5.2 Relevance for learning and teaching

In the context of serious games, sentiment analysis can be used, for instance, in dialogues, commonly available either in multi-player communication or in discussions with a virtual character. The arising insights about how people feel and interact during these interactions can then be fed back to the game for further usage in the game scenario, or can be provided as feedback for the game development team. Alternatively, sentiments can be extracted from written free text or spoken assignments in the game, such as reports, pitches or answers to open-ended questions.

#### 4.5.3 Approach

The state of the art in sentiment analysis is represented by deep learning with either Convolutional Neural Networks (CVN; Kim 2014), Recurrent Neural Networks (Socher et al. 2013), the Dependency Tree Long Short-Term Memory Networks (Tree-LSTM; Tai et al. 2015) or Bi-directional LSTM (BiLSTM; Graves and Schmidhuber 2005). The LSTM networks are probably the most used type of text encoder for the majority of tasks involving text comprehension. *ReaderBench's* sentiment analysis service is based on a BiLSTM network. In a comparative study using data from a corpus of 201,552 games reviews crawled from Metacritic (<http://www.metacritic.com/game>), the BiLSTM network achieved an overall accuracy of 74%,

thereby outperforming various Dependency Tree Networks, Support Vector Machine approaches, Universal Sentence Encoder (USE; Cer et al. 2018), and Multinomial Naive Bayes models with 3% to 7%.

#### 4.5.4 Application case

A practical example of using sentiment analysis in a serious game is the Jobquest game, referred to above. Users are requested to prepare and optimise their Curriculum Vitae (CV) in French language, in view of a specific job opening. The textual content of the uploaded CV is then analysed with *ReaderBench* services, which returns specific feedback, including sentiment valence scores, indicators of emotions, textual complexity factors and general statistics related to visual or contents quality (Gutu et al. 2018). A French corpus consisting of a collection of articles published by the Le Monde newsarticle was used to train the system. The CV model was trained with a training set of around 100 CVs, manually assessed on a set of characteristics that define a good commercial CV, which then produced an accuracy of 67%.

#### 4.5.5 Technical considerations

The sentiment analysis service provided by *ReaderBench* can be accessed as a remote web service through a dedicated endpoint exposed within the *ReaderBench* API. The service is open and does not require authentication. As the *ReaderBench* framework is provided as an open-source framework, developers can install it on their servers and can develop their own services by extending the facilities of the framework, which can be cloned from a GitLab server (<https://git.readerbench.com/ReaderBench/ReaderBench>). In terms of semantic models, developers can either use pre-trained corpora, or they can train a custom model for their specific scenarios. The sentiment analysis service currently supports multiple languages, namely English, French, and Dutch.

### 4.6 NPC: Automated essay scoring

#### 4.6.1 ReaderBench essay scoring

The *ReaderBench* framework also incorporates an essay scoring functionality which is capable of assigning comprehension scores to open text inputs, for instance students' assignments (or reports), or answers to open-ended questions.

#### 4.6.2 Relevance for learning and teaching

In serious games writing assignments and open-ended questions are scarce, because of the intensive manual effort needed for assessing the learner productions. Also, writing assignments may be considered too schoolish as opposed to the fun of playing games and they are deemed a potential disruption of the player's flow (Shute 2011). Writing assignments, however, accommodate deeper knowledge processing since they require explicit consideration of learned concepts, principles and their relationships, reflection about the significance and appraisal of the experiences, and the creative synthesis of

argumentation (Westera et al. 2018). In addition, writing assignments would provide an excellent diagnostic tool for detailed assessment of learning progress. That is why schools and universities often require students to write reports or theses as proofs of mastery. Also, most professions require excellent writing skills, for instance in journalism, health, education, marketing, business consultancy and many other areas. Now that automated processing is becoming available, writing assignments need no longer be omitted in serious games. The very method of essay scoring can also be used to inform the game development team about the complexity of instructional texts and other textual learning materials exposed in the game, which allows their adjustment for a better fit with the player characteristics and needs.

#### 4.6.3 AI approach

For various languages a separate NLP pipeline model was created, using language specific dictionaries, stop words elimination, word lemmatisation, and part-of-speech tagging. Latent Semantic Analysis (LSA; Landauer and Dumais 1997), Latent Dirichlet Allocation (LDA; Blei et al. 2003) and word2vec (Mikolov et al. 2013) models were trained on extensive corpora adapted to specific scenarios. In addition, the WordNet lexical ontology was used to identify lexical chains (Budanitsky and Hirst 2006). A dedicated essay scoring model is then used by feeding a training set of example essays and their assigned scores to the system. *ReaderBench* services provide various textual complexity indices such as (Dascalu et al. 2018): the length of sentences and paragraphs in word and character counts, statistics with regard to the use of different parts of speech and syntactic dependencies, semantic cohesion scores, and discourse structure. After training the essay scoring model, the testing and validation of additional student essays can be performed. Accuracy scores strongly depend on the text volumes and number of example documents used for training.

#### 4.6.4 Application case

An example of essay scoring in French language involves the classification of documents from primary school manuals into five complexity classes (Dascalu et al. 2014). A successful example of essay scoring in a serious game would be the VIBOA game (Westera et al. 2008). In this video game, master students adopt the role of an environmental policy consultancy charged with the investigation of authentic environmental problem cases. As part of the game scenario, students have to summarise and explain their findings, obtained from a variety of legal documents, calculations and (simulated) stakeholder interviews (cf. Fig. 3) in scientific reports and upload these to the game server. Teachers are supposed to manually assess these reports and return the outcomes to the students in the game. In practice, the manual assessment of many reports generates an unacceptably high teacher workload. In this game, the ReaderBench essay scoring software has demonstrated to offer an excellent replacement, offering high accuracy and considerable workload reduction. It was shown that the teachers' work load reduces to 68%, while a lower limit of 90% precision is preserved.





Fig. 3 Screenshot of attending a meeting with experts and stakeholders in the VIBOA game

#### 4.6.5 Technical considerations

The essay scoring service provided by *ReaderBench* can be accessed in a similar manner as the sentiment analysis service. Developers can either use pre-trained corpora and textual complexity models, or they can create their tailored models specific for their learning requirements. The essay scoring service provides a wide range of textual complexity indices, freely available for English, French, Dutch, Spanish, Romanian, and Italian languages.

### 4.7 NPC: Role play character (FAtiMA)

#### 4.7.1 The FAtiMA toolkit

The FAtiMA Toolkit is an emotion engine for AI Characters (Dias et al. 2014; Mascarenhas et al. 2018). It is a collection of open-source tools that help researchers, game developers and roboticists to incorporate a computational model of emotion and decision-making in their projects. In particular, it enables developers to easily create Role Play Characters. These are socially intelligent characters with detailed AI modules that makes them autonomous regarding social interactions.

#### 4.7.2 Relevance for learning and teaching

The added value of socially intelligent characters in a serious game is twofold. First, game characters that expose believable emotional responses give the illusion of interactions with real human participants, which deepens the (learning) experiences. This is especially relevant for games that aim to address social skills and communication skills. In recent years, these skills have been re-established as crucial generic skills for meeting the demands of the digital age: the so-called twenty-first century skills

(Dede 2010). Another most promising application area is the therapy and training of people with special social needs, for example, children with autism, who can use games with artificial social characters as a safe environment to mitigate the anxiety associated with social interactions (Bernardini et al. 2014).

#### 4.7.3 AI approach

The FATiMA Toolkit facilitates the inclusion of a dynamic model of emotions that affects not just how the character looks and acts but also how the player's responses are evaluated. For this, it follows a character-centred approach rather than a plot-centred approach. The authoring is focused on defining general profiles (a set of rules) of how characters should respond emotionally in their games across different scenarios and contexts. The main advantage of this approach is that the characters' behaviours are consistent across different contexts and no elaborate hard-coding is needed. Reasoner components are used in conjunction to augment the capabilities of the decision making and emotional responses of each agent in different socio-cultural contexts. An example is a reasoner addressing social importance; it allows to create groups of agents that would act and feel according to different cultural values (Mascarenhas et al. 2016). A second reasoner, which is named CiF-CK, is based on a model that describes different social exchanges and its consequences within a social environment (Guimaraes et al. 2017). The toolkit is modular allowing other types of reasoners to be easily added to the system.

#### 4.7.4 Application case

The FATiMA Toolkit has been used in various case studies. In the Space Modules Inc. game, which addresses customer communication skills, the player takes on the role of a customer services representative working at the helpdesk of a spaceship parts manufacturer (Mascarenhas et al. 2018). Customers with a variety of starting moods and emotional dispositions get in touch with the helpdesk about problems they are experiencing. The player has to manage diverse situations and has to decide how best to respond. The FATiMA Toolkit is used to model the decisions and emotional reactions of the diverse (virtual) customers, the outcomes of which can be used to change their on-screen appearances (Fig. 4).

A similar application is the Sports Team Manager game, which is about composing and managing the best performing sailing team. The player first interviews the various virtual characters to identify their skills and personalities and then must communicate with the team, deciding which members are placed into each position per race and resolve conflict situations as they arise. Again, the FATiMA Toolkit is used to model the characters' emotions. Other usage examples are in a Virtual Reality experience designed as police interrogation exercise, and in robotics: controlling the decisions of two social robots playing the card game Sueca with two human players, while exposing group-based emotions (having each robot appraising both its own actions and the actions of its partner).

#### 4.7.5 Technical considerations

The FATiMA Toolkit does not require installation. To facilitate its integration with game engines it works as a C# library. Each of the toolkit's component is able to



Fig. 4 A screenshot from the Space Modules Inc. game, showing one of the customers

fully load and save its internal state to a JSON file, which may be used for further processing by the game. Although any text editor of choice can be used for authoring, each component included in the role play character comes with a dedicated editor, providing a graphical user interface, syntax error detection and the capability to edit the complex intertwined data structures needed for covering the characters emotions, autobiographical memory, and appraisal rules, among other things.

## 4.8 NPC: Behavior mark-up language realizer

### 4.8.1 Nonverbal bodily motion: Behaviour mark-up language

The Behavior Mark-up Language (BML) Realizer created by Utrecht University defines and controls the on-screen representation of virtual characters, in particular their non-verbal behaviours: facial expressions, body movements, gestures, and gaze,

respectively. The importance of non-verbal behaviours either from avatars or non-playing virtual characters should not be underestimated. For inducing intense, realistic game experiences the challenge is not only to make virtual characters just look like humans but also to make them behave like humans. The behaviours should provide an illusion of realism, firstly by demonstrating responsiveness to the actions of players and other agents in the game, secondly by taking into account the context of operation, and thirdly, by securing that the behaviours are meaningful and interpretable. In other words, the displayed behaviours should reflect the inner state of the artificial character (Thiebaut et al. 2008). Thus, virtual characters should be equipped with properties such as personality, emotions and expressive non-verbal behaviours in order to engage the users to the game.

#### 4.8.2 Relevance for learning and teaching

As many serious games rely on experiential learning, which means they aim to provide intense and meaningful learning experiences and allow the active participation of players in contexts that in many cases mimic professional practice, a large degree of realism or authenticity is indicated (Westera et al. 2018). Moreover, the realism supports the acquisition of tacit, implicit knowledge bound to the experiences and helps to promote successful transfer to the real world situations. In this respect, the believability of virtual characters is evident, either as personas in realistic game scenarios (for instance in a job interview training) or as virtual tutors that guide students during their game sessions.

#### 4.8.3 AI approach

There are two main approaches for modelling non-verbal behaviours and animations: rule-based (procedural) and machine learning (Beck et al. 2017; Yumak and Magnenat-Thalmann 2015), respectively. Rule-based approaches are based on findings from social sciences and biomechanics. These rules are typically obtained through empirical analysis of human behaviour. The disadvantage of such methods is that they might not capture the full complexity of the motion trajectories. However, they provide greater level of control, while keeping the realism at a sufficient level. Machine learning approaches automate this process and find regularities and dependencies between factors using statistics, and they learn from a larger amount of data to cover various cases. However, obtaining good annotated data is problematic. Moreover, these data typically apply to the specific conditions of the context where they were collected, but do not necessarily generalise well. Therefore, a rule-based (procedural) approach was chosen for the realisation of non-verbal behaviours, providing maximum control in various application contexts. The rule-based coding approach of the BML Realizer allows to efficiently define a controlled set of the non-verbal behaviours, while avoiding the laborious job of separately coding the animations of all non-verbal behavioural attributes. Behavior Mark-up Language (BML; Kopp et al. 2006) is an XML based language that is used to model and coordinate speech, gesture, gaze and body movements (cf. Fig. 5). Each behaviour is divided into six animation phases bounded with seven synchronisation points: start, ready, stroke-start, stroke, stroke-end, relax, and end, respectively. Synchrony is achieved by assigning the sync-point of one



**Fig. 5** BML-modelled virtual human capable of speech, gaze and gesture control

behaviour to the sync-point of another. The behaviour planner that produces the BML also gets information back from the behaviour realizers about the success and failure of the behaviour requests.

#### 4.8.4 Application cases

The Virtual Human Controller has been successfully used in various applications. An example would be the Job-Quest game, which is a full-3D application interview training game. Also, it has been used for controlling the Virtual Receptionist character at the entrance of the computer science building at Utrecht University. This set-up includes a microphone to capture the visitors' speech and a Kinect camera to capture their behaviours.

#### 4.8.5 Technical considerations

The BML realizer can be used in the Unity 3D game engine and allows to define speech, gaze and gesture animation for a conversational character. The animation pipeline includes the following steps: 1) Importing a 3D character that supports animation from an .fbx-file editor, for instance the DAZ3D Editor (or other tools that can export .fbx files), together with blendshapes for speech and facial animation and adding it to the Unity project, 2) adding separate animation controllers for speech, facial animation, gestures and gaze, 3) linking the separate controllers to the BML Realizer, and 4) writing a BML script to generate multi-modal synchronised animations. Beyond these functionalities, we have added Google speech recognition and chatting functionalities using AIML Pandorabots. Furthermore, we developed a novel autonomous gaze control module based on Kinect to drive the “look at” behaviour of the virtual character in group-based interactions using a data-driven approach (Yumak et al. 2017). The BML realizer has been successfully integrated with the Communicate! dialogue manager from Utrecht University, allowing for a direct connection between dialogue authoring and non-verbal expression, and with the Emotion Appraisal component, which is part of the FATiMA toolkit described above.

## 4.9 NPC: LipSync generator

### 4.9.1 Lip-synchronised speech

The LipSync Generator produces lip-synchronised speech animation. This is an important element of believable NPCs and contributes significantly to the illusion of realism and to accommodating more natural human-computer dialogues.

### 4.9.2 Relevance for learning and teaching

As explained before, the believability of virtual characters is a relevant contribution to provoking authentic and effective learning experiences. Lip-synchronised speech thus readily enhances the quality of either virtual tutors or any virtual characters in the game scenario.

### 4.9.3 AI approach

Different approaches to lip-synchronised speech animation have been proposed over the years. Procedural approaches are a better choice in terms of the control of the animation, although they may not reach the level of naturalness obtained by performance-capture and data-driven approaches (Edwards et al. 2016). So far, none of the proposed procedural approaches (Taylor et al. 2017) explicitly takes into account the effect of emotions on the mouth movement. Our current contribution entails an audio-driven speech animation method for interactive game characters where the control aspect is high priority. While doing this, we aim to push the boundaries of naturalness by introducing the effect of emotions. The work includes an expressive speech animation model that takes into account emotional variations in audio and a co-articulation model based on dynamic linguistic rules varying among different emotions. The component takes as input text, sends it to a text-to-speech (TTS) system, parses the phonemes, maps them to visemes (visual counterparts of phonemes) and finally blends them for smooth speech animation (Fig. 6).

### 4.9.4 Application cases

The LipSync Generator is mostly used in conjunction with the BML Realizer to setup a joint Virtual Human Controller. Application cases are the Job-Quest game, mentioned before, and the Virtual Receptionist character at the entrance of the computer science building at Utrecht University.



Fig. 6 Phoneme-to-viseme mapping: “hello” -> h @ l @ U -> GK AHH L OHH UUU



#### 4.9.5 Technical considerations

The LipSync Generator is implemented as a Unity 3D plug-in. It currently uses an off-the-shelf text-to-speech (CereVoice) library. It is also possible to link it with other text-to-speech systems. Since Unity 3D is multi-platform, the component can be deployed on different platforms including Web and mobile applications. Similar to the BML Realizer it directly works with the Communicate! dialogue manager from Utrecht University, allowing for a direct connection between dialogue authoring and non-verbal expression, and with the Emotion Appraisal component, which is part of the FAtiMA toolbox described above.

### 5 Assessment

As was explained in the previous chapter, a set of real-world application pilots were arranged to accommodate empirical evaluation of the overall concept of game software reuse. The application pilots were centred around various real-world games created by professional game studios. The designs of these games were on the one hand guided by the specific educational contexts, target groups and learning objectives of the cases, but were also informed by the new functionalities that RAGE components offer. The evaluations covered all critical aspects of the process, with a main focus on experiences, usability and (learning) outcomes. Various detailed publications of these studies are being prepared, while background information and some key results can already be found in two recent, technical reports (Bazzanella et al. 2018; Steiner et al. 2018). Feedback on the set of components was collected from game developers and external users from academia and game industry, demonstrating good overall usability and confirming their usefulness, perceived benefits and cost effectiveness for applied game development. Benefits-costs ratios for the components were found to be in the 10–100 range (detailed analysis available at [www.gamecomponents.eu/content/604/cost-benefit-analysis-of-the-rage-case-studies](http://www.gamecomponents.eu/content/604/cost-benefit-analysis-of-the-rage-case-studies)): while development from scratch would take weeks or months –if possible at all-, the integration of existing components is a matter of days or even hours. Both component developers and game developers (23 subjects) appreciated the usability of the component-based architecture and indicated to keep using the architecture in future projects: the architecture is lightweight indeed, allowing for the easy conversion of software into reusable components and making the integration of components with the game engine an easy job. The application pilots indicated for each component in chapter 4 involved over 1,500 end-users in various contexts. Overall, game experiences obtained positive feedback from teachers and players involved; the games' potential to support learning was well recognised. Significant evidence of learning gains was demonstrated. The evaluation of search, collaboration, and course authoring features within the gamecomponents.eu portal among game developers showed overall positive results and provided useful inputs for the provisional launch of the portal (2018). Online course authoring, which is supported by the portal, was even rated as being easier and more comfortable than course authoring in existing learning management systems. Overall, the RAGE



research programme to support the serious game development community by accommodating the reuse of intelligent software components has been demonstrated to be fit for purpose. After its launch, traffic to the `gamecomponents.eu` portal steadily grew, doubling every two months up to over 6000 visitors by January 2019 and leading to many hundreds of component downloads. Sustainable exploitation of the system is carried by the recently launched RAGE Foundation, which is a not-for-profit alliance of serious game stakeholders, including key players from industry, education and academia.

## 6 Discussion and conclusion

This article presented a comprehensive overview of AI advances for serious games. It described a set of concrete game AI software artefacts that have been made available on the platform-independent `gamecomponents.eu` portal. For each component, a brief explanation was provided, including technical considerations, benefits for serious games, and a description of one or more application cases. Three game AI key areas covered are: Player Experience Modelling, Natural Language Processing and Non-Playing Characters, respectively.

With respect to Player Experience Modelling, it is important to refer to the ethical issues that may arise when creating detailed user profiles and having these analysed with smart algorithms that may uncover sensitive personal traits, behaviours, preferences, capabilities, opinions. Although player modelling is a key activity for any teacher in a classroom situation, continually checking how pupils are doing and whether or not they would need specific support, it takes place in a natural, largely implicit or even intimate way, based on the direct personal relationships and trust that are developed during the interactions between teachers and pupils. In contrast, computer traces of pupils' behaviours are not based on personal relationships and trust, but reflect the relentless recording of events, indelibly available for mining algorithms and computerised judgements, the outcomes of which are easily mistaken for absolute truths. As a consequence, qualitative aspects of teaching and learning are likely to be overlooked in favour of quantifiable aspects. From a psychological or pedagogical perspective, the idea of being under permanent computer-surveillance may induce unwanted player behaviours, for instance risk avoiding behaviours or reduced initiative, which may affect the quality of learning. Important concerns have been raised because analytics could severely disempower and demotivate learners when they are provided with continuous feedback about their weak performances as compared with other students (Westera et al. 2014). Consequently, ethical, legal and pedagogical considerations draw the line to the practical application of player experience modelling.

Natural Language Processing applications have demonstrated to greatly enhance the quality of assessing the significance of the player's verbal and written utterances. Still, as Siri, Google, Assistant, Alexa and other popular virtual assistants demonstrate, NLP is not without flaws. In addition, NLP applications may require substantial pre-processing or post-processing that assume specific

NLP expertise, for instance for the cautious training, testing and validation of machine learning models, which may pose a barrier for adoption.

Realistic, believable Non-Playing Characters procure a natural fluency in human-computer interaction, which may lead to engaging learning experiences and a sense of authenticity offered by the learning environment. But this pursuit of realism should be put in perspective. Experiments based on media equation theory (Reeves and Nass 1996) have demonstrated that human individuals respond socially and naturally to a variety of non-human objects such as robots and avatars, but also to computers or any graphical objects (cf. the simple ghost characters in Pacman, which may still raise exciting, dramatic, if not thrilling experiences). This tendency toward anthropomorphism is explained by the fact that the human brain simply cannot distinguish between inter-human and mediated or symbolic interactions, whereby it is not capable of suppressing natural interpersonal responses in any interaction. A complementary explanation at the level of conscious thought would be the “willing suspension of disbelief”, which is the well-considered acceptance of unrealistic hypotheses, present in many fictional works in literature, cinema, and games. Accepting the (unrealistic) presupposition that Superman can fly, pays off with the rewarding experience of being carried away by the adventures in the Superman movie, while its rejection would have spoiled the experience. Likewise, in Pacman we are ready to accept the idea that a handful of pixels are real ghosts. These considerations suggest that the level of realism is not always critical. It should be decided upon at design time, given the specific context, content and purpose of the game.

For practical reasons, this study has been restricted to the game AI fields of PEM, NLP and NPC's. From the wider domain of artificial intelligence various additional high-potential game AI areas have been identified (Yannakakis and Togelius 2015), which would open up many new possibilities. These areas include search and planning (for pathfinding, adaptation and computer playing strength), procedural content generation (creating design-tailored game contents, e.g. cities, furnished rooms, people), computational narrative (optimisation procedures for game storytelling, event generation, generating sequences of game events, deciding about camera angles), and AI-assisted game design (smart tools to support creative game design and development, e.g. level design, simulating playthroughs, game rule design). Altogether, it seems the era of AI is just starting. New AI concepts and technologies will continue to foster and innovate the domain of serious gaming.

While focusing on game AI, the initiative described in this article has been a substantial attempt to bridge the gap between research and industry, resulting in a knowledge and technology valorisation and transfer mechanism instantiated by the gamecomponents.eu marketplace portal that has the potential to become a favourable example of structural research-industry collaboration.

**Acknowledgements** This work has been partially funded by the EC H2020 project RAGE (Realising an Applied Gaming Eco-System); <http://www.rageproject.eu/>; Grant agreement No 644187 and by national funds through Fundação para a Ciência e a Tecnologia (FCT-UID/CEC/500 21/2013).

**Open Access** This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

## References

- Bachmann, F., Bass, L., Buhman, C., Comella-Dorda, S., Long, F., Robert, J., Sea-Cord, R., & Wallnau, K. (2000). *Technical concepts of component-based software engineering, volume II*. Pittsburgh: Carnegie Mellon University, Software Engineering Institute.
- Bahreini, K., Nadolski, R., & Westera, W. (2016). Data fusion for real-time multimodal emotion recognition through webcams and microphones in E-learning. *International Journal of Human Computer Interaction*, 32(5), 415–430. <https://doi.org/10.1080/10447318.2016.1159799>.
- Bahreini, K., Nadolski, R., & Westera, W. (2017). Communication skills training exploiting multimodal emotion recognition. *Interactive Learning Environments*, 25(8), 1065–1082. <https://doi.org/10.1080/10494820.2016.1247286>.
- Bahreini, K., Vegt, W. van der & Westera, W. (2018). A fuzzy logic approach to reliable real-time recognition of facial emotions. *Multimedia Tools and Applications*. (Accepted).
- Bazzanella, B., Casagrande, M., Molinari, A., Humphreys, S., Sleightholme, G., Lepoivre, O., Halbeher, M., Romana, A., Falua, C., Soeiro, C., Guerra, R. & Kommeren, R. (2018). *D5.4 – Pilots quality report round 2*. RAGE project. <https://research.ou.nl/en/publications/d54-pilots-quality-report-round-2>. Accessed 12 July 2019.
- Beck, A., Yumak, Z. & Magnenat-Thalmann, N. (2017) Body movement generation for virtual characters and social robots. In: J. K. Burgoon, N. Magnenat-Thalmann, M. Pantic & a. Vinciarelli. *Social Signal Processing* (pp. 273–286). Cambridge, MA: Cambridge University press. <https://doi.org/10.1017/9781316676202.020>.
- Bernardini, S., Porayska-Pomsta, K., & Smith, T. J. (2014). ECHOES: An intelligent serious game for fostering social communication in children with autism. *Information Sciences*, 264, 41–60.
- Birman, K., & Joseph, T. (1987). Exploiting virtual synchrony in distributed systems. In *Proceedings of the eleventh ACM symposium on operating systems principles (SOSP '87)* (pp. 123–138). New York: ACM.
- Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3(4–5), 993–1022.
- Bourassa, M. A. J., & Massey, L. (2012). *Artificial intelligence in games. A survey of the state of the art*. Ottawa: Defence R&D Canada Retrieved from [http://cradpdf.drdc-rddc.gc.ca/PDFS/unc120/p536670\\_A1\\_b.pdf](http://cradpdf.drdc-rddc.gc.ca/PDFS/unc120/p536670_A1_b.pdf). Accessed 12 July 2019.
- Budanitsky, A., & Hirst, G. (2006). Evaluating WordNet-based measures of lexical semantic relatedness. *Computational Linguistics*, 32(1), 13–47.
- Cer, D., Yang, Y., Kong, S.-Y., Hua, N., Limtiaco, N., St John, R., Constant, N., Guajardo-Cespedes, M., Yuan, S., Tar, C., Sung, Y.-H., Strophe, B. & Kurzweil, R. (2018). Universal Sentence Encoder. Retrieved from <https://arxiv.org/pdf/1803.11175.pdf>. Accessed 12 July 2019.
- Champanard, A. J. (2004). *AI Game Development*. San Francisco: New Riders Publishing.
- Claypool, M., Decelle, J., Hall, G. & O'Donnell, L. (2015). Surrender at 20? Matchmaking in league of legends. In: Proceedings of the IEEE Games Entertainment Media Conference 2015, pp. 1–4.
- Dascalu, M., Dessus, P., Trausan-Matu, S., Bianco, M., & Nardy, A. (2013). ReaderBench, an environment for analyzing text complexity and reading strategies. In H. C. Lane, K. Yacef, J. Mostow, & P. Pavlik (Eds.), *16th Int. Conf. On artificial intelligence in education, AIED 2013* (pp. 379–388). Memphis, USA: Springer.
- Dascalu, M., Stavarache, L. L., Trausan-Matu, S., Dessus, P., & Bianco, M. (2014). Reflecting comprehension through French textual complexity factors. In: *26th International Conference on Tools with Artificial Intelligence, ICTAI 2014*, Limassol, Cyprus (pp. 615–619). New York: IEEE. <https://doi.org/10.1109/ICTAI.2014.3>.
- Dascalu, M., Crossley, S., McNamara, D. S., Dessus, P., & Trausan-Matu, S. (2018). Please ReaderBench this text: A multi-dimensional textual complexity assessment framework. In S. Craig (Ed.), *Tutoring and intelligent tutoring systems* (pp. 251–271). Hauppauge, NY, USA: Nova Science Publishers, Inc..
- Dede, C. (2010). Comparing frameworks for 21st century skills. In J. Bellanca & R. Brandt (Eds.), *21st century skills: Rethinking how students learn* (pp. 51–75). Bloomington, IN: Solution Tree Press

- Retrieved from [http://sttechnology.pbworks.com/f/Dede\\_\(2010\)\\_Comparing%20Frameworks%20for%2021st%20Century%20Skills.pdf](http://sttechnology.pbworks.com/f/Dede_(2010)_Comparing%20Frameworks%20for%2021st%20Century%20Skills.pdf). Accessed 12 July 2019.
- Dias, J., Mascarenhas, S., & Paiva, A. (2014). Fatima modular: Towards an agent architecture with a generic appraisal framework. In T. Bosse, J. Broekens, J. Dias, & J. van der Zwaan (Eds.), *Emotion modeling* (pp. 44–56). Cham: Springer.
- Dignum, F., Westra, J., Van Doesburg, W. A., & Harbers, M. (2009). Games and agents: Designing intelligent gameplay. *International Journal of Computer Games Technology*, 837095, 1–18. <https://doi.org/10.1155/2009/837095>.
- Edwards, P., Landreth, C., Fiume, E. & Singh, K. (2016). JALI: An animator-centric viseme model for expressive lip synchronization. *ACM Trans. Graph.* 35(4), article 127, 1–11.
- Elo, A. E. (1978). *The rating of chess players, past and present*. New York: Arco Pub.
- European Commission (1995). EC green paper on innovation. Brussels: European Commission. Retrieved from [http://europa.eu/documents/comm/green\\_papers/pdf/com95\\_688\\_en.pdf](http://europa.eu/documents/comm/green_papers/pdf/com95_688_en.pdf). Accessed 12 July 2019.
- Fray, A. (2015). Context steering - behavior-driven steering at the macro scale. In: *Game AI Pro*, Vol. 2 (pp. 183–193). New York: CRC press.
- Fujita, H., & Wu, I.-C. (2012). A special issue on artificial intelligence in computer games: AICG. *Knowledge-Based Systems*, 34, 1–2. <https://doi.org/10.1016/j.knosys.2012.05.014>.
- Gamma, E., Helm, R., Johnson, R., & Vlissides, J. (1994). *Design patterns: Elements of reusable object-oriented software*. London: Pearson Education.
- Gee, J. P. (2003). What video games have to teach us about learning and literacy? New York: Palgrave MacMillan. <https://doi.org/10.5040/9781628927924.ch-013>.
- Georgiadis, K., van Lankveld, G., Bahreini, K., & Westera, W. (2018). Accommodating stealth assessment in serious games: Towards developing a generic tool. In: *10th IEEE International Conference on Virtual Worlds and Games for Serious Applications (VS-Games 2018)*, pp. 1–4.
- Georgiev, A., Grigorov, A., Bontchev, B., Boytchev, P., Stefanov, K., Bahreini, K., Nyamsuren, E., Van der Vegt, W., Westera, W., Prada, R., Hollins, P. & Moreno, P. (2016). The RAGE Software Asset Model and Metadata Model. In: Tim Marsh, Minhua Ma, Manuel Fradinho Oliveira, Jannicke Baalsrud Hauge and Stefan Göbel (Eds.), *Serious Games, Proceedings of the Second Joint International Conference, JCSG 2016*, Brisbane, QLD, Australia, September 26–27, Lecture Notes in Computer Science 9894 (pp. 191–203). Cham, Switzerland: Springer International Publishing AG. [https://doi.org/10.1007/978-3-319-45841-0\\_18](https://doi.org/10.1007/978-3-319-45841-0_18).
- Graves, A., & Schmidhuber, J. (2005). Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Networks*, 18(5), 602–610. <https://doi.org/10.1016/j.neunet.2005.06.042>.
- Guimaraes, M., Santos, P. & Jhala, A. (2017). CiF-CK: An architecture for social NPCs in commercial games. In: *Proceedings of the IEEE Conference on Computational Intelligence and Games, CIG 2017* (pp. 126–133).
- Gutu, G., Paraschiv, I. C., Dascalu, M., Cristian, G., Trausan-Matu, S., & Lepoivre, O. (2018). Analyzing and providing comprehensive feedback for French CVs with ReaderBench. *Scientific Bulletin, University Politehnica of Bucharest, Series C*, 80(2), 17–28.
- Guy, S. J. & Karamouzas, I. (2015). Guide to anticipatory collision avoidance. *Game AI Pro*, Vol. 2 (pp. 195–208). New York: CRC Press.
- Herbrich, R., Minka, T., & Graepel, T. (2006). TrueSkill™: A Bayesian skill rating system. In B. Schölkopf, H. C. Platt, & T. Hoffman (Eds.), *Advances in neural information processing systems* (pp. 569–576). Cambridge, MA: MIT Press.
- Hofman, A. D., Jansen, B. R., de Mooij, S. M., Stevenson, C. E., & van der Maas, H. L. (2018). A solution to the measurement problem in the idiographic approach using computer adaptive practicing. *Journal of Intelligence*, 6(1), 14.
- Hühn, J., & Hüllermeier, E. (2009). FURIA: An algorithm for unordered fuzzy rule induction. *Data Mining and Knowledge Discovery*, 19(3), 293–319.
- Humphreys, T. (2013). Exploring HTN planners through example. In *Game AI Pro* (pp. 149–167). New York: CRC Press.
- Kim, Y. (2014). Convolutional Neural Networks for Sentence Classification. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014* (pp. 1746–1751). Stroudsburg, PA: ACL. 21, 1120, 1142 <https://doi.org/10.1109/LSP.2014.2325781>.
- Klinkenberg, S., Straatemeier, M., & van der Maas, H. L. (2011). Computer adaptive practice of maths ability using a new item response model for on the fly ability and difficulty estimation. *Computers & Education*, 57(2), 1813–1824.

- Kopp, S., Krenn, B., Marsella, S., Marshall, A. N., Pelachaud, C., Pirker, H., Thorisson, K. R., & Vilhjalmsson, H. (2006). Towards a common framework for multimodal generation: The behavior markup language. In *Proceedings of the 6th international conference on intelligent virtual agents, IVA'06* (pp. 205–217). Berlin/Heidelberg: Springer.
- Landauer, T. K., & Dumais, S. T. (1997). A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction and representation of knowledge. *Psychological Review*, *104*(2), 211–240.
- Lewis, M. (2017). Choosing effective utility-based considerations. *Game AI Pro*, Vol. 3 (pp. 167–178). New York: CRC press.
- Lewis, M. & Dill, K. (2015): Game AI appreciation, revisited. In: *Game AI Pro*, Vol. 2 (pp. 3–17). New York: CRC press. Retrieved from [http://www.gameaipro.com/GameAIPro2/GameAIPro2\\_Chapter01\\_Game\\_AI\\_Appreciation\\_Revisited.pdf](http://www.gameaipro.com/GameAIPro2/GameAIPro2_Chapter01_Game_AI_Appreciation_Revisited.pdf). Accessed 12 July 2019.
- Liu, B. (2012). Sentiment analysis and opinion mining. *Synthesis Lectures on Human Language Technologies*, *5*(1), 1–167. <https://doi.org/10.2200/S00416ED1V01Y201204HLT016>.
- Lord, F. M., & Novick, M. R. (1968). *Statistical theories of mental test scores; with contributions by Allan Birnbaum*. Reading, MA: Addison-Wesley.
- Maher, B. (2016). Can a video game company tame toxic behaviour? *Nature*, *531*(7596), 568–571.
- Mascarenhas, S., Degens, N., Paiva, A., Prada, R., Hofstede, G. J., Beulens, A., & Aylett, R. (2016). Modeling culture in intelligent virtual agents. *Autonomous Agents and Multi-Agent Systems*, *30*(5), 931–962.
- Mascarenhas, S., Prada, R., Dias, J., Santos, P. A., Star, K., Hirsh, B., Spice, E., & Kommeren, R. (2018). A virtual agent toolkit for applied game developers. In C. Browne, M. Winands, J. Liu, & M. Preuss (Eds.), *Proceedings of the IEEE Conference on Computational Intelligence and Games, CIG 2018, Maastricht* (pp. 1–7). New York: IEEE.
- Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representation in vector space. In *Workshop at ICLR*. Scottsdale, AZ.
- Min, W., Frankosky, M. H., Mott, B. W., Rowe, J. P., Wiebe, E., Boyer, K. E., & Lester, J. C. (2015). DeepStealth: Leveraging deep learning models for stealth assessment in game-based learning environments. In C. Conati, N. Heffernan, A. Mitrovic, & M. Verdejo (Eds.), *Artificial Intelligence in Education. AIED 2015. Lecture notes in computer science, Vol 9112* (pp. 277–286). Cham: Springer.
- Mislevy, R. J. (2011). *Evidence-Centered Design for Simulation-Based Assessment. CRESST Report 800*. Los Angeles, CA: National Center for research on evaluation, standards, and student testing (CRESST).
- Moore, G. R., & Shute, V. J. (2017). Improving learning through stealth assessment of conscientiousness. In A. Marcus-Quinn & T. Hourigan (Eds.), *Handbook on digital learning for K-12 schools* (pp. 355–368). Cham: Springer. <https://doi.org/10.1007/978-3-319-33808-8>.
- Nguyen, T. D., Chen, Z. & El-Nasr, M. S. (2015). Analytics-based AI techniques for a better gaming experience. *Game AI Pro*, Vol. 2 (pp. 481–500). New York: CRC Press.
- Nyamsuren, E., van der Vegt, W., & Westera, W. (2017). Automated adaptation and assessment in serious games: A portable tool for supporting learning. In M. Winands, H. van den Herik, & W. Kusters (Eds.), *Advances in Computer Games, ACG 2017. Lecture notes in computer science, Vol. 10664* (pp. 201–212). Cham: Springer. [https://doi.org/10.1007/978-3-319-71649-7\\_17](https://doi.org/10.1007/978-3-319-71649-7_17).
- Nyamsuren, E., van der Maas, H. L. J. & Maurer, M. (2018a). Set-theoretical and combinatorial instruments for problem space analysis in adaptive serious games. *International Journal of Serious Games*, *5* (1), 5–18. <https://doi.org/10.17083/ijsg.v5i1.219>.
- Nyamsuren, E., van der Vegt, W. & Westera, W. (2018b). A fuzzy rule for adaptive gaming to reduce problem-selection Bias and improve automated difficulty rating. *IEEE Transactions on Learning Technologies* (accepted for publication).
- Pantic, M., Sebe, N., Cohn, J. F., & Huang, T. (2005). Affective multimodal human-computer interaction. In H. Zhang, T.-S. Zua, R. Steinmetz, M. Kankanhalli, & L. Wilcox (Eds.), *Proceedings of the 13th Annual ACM International Conference on Multimedia, Hilton, Singapore* (Vol. 5, pp. 669–676). New York: ACM.
- Pekrun, R. (1992). The impact of emotions on learning and achievement: Towards a theory of cognitive/motivational mediators. *Journal of Applied Psychology*, *41*, 359–376.
- Pentheny, G. (2013). Efficient crowd simulation for Mobile games. In *Game AI Pro* (pp. 317–323). New York: CRC Press.
- Pentheny, G. (2015). Advanced techniques for robust, efficient crowds. In: *Game AI Pro*, Vol. 2 (pp. 173–181). New York: CRC press.
- Rabin, S. (2017): The illusion of intelligence. In: *Game AI Pro*, Vol. 3 (pp. 3–9). New York: CRC press.
- Reeves, B., & Nass, C. (1996). *The media equation: How people treat computers, television, and new media like real people and places*. Cambridge: Cambridge University Press.



- Robbins, M. (2013): Using neural networks to control agent threat response. In: *Game AI Pro* (pp. 391–398). New York: CRC press. Retrieved from [http://www.gameaipro.com/GameAIPro/GameAIPro\\_Chapter30\\_Using\\_Neural\\_Networks\\_to\\_Control\\_Agent\\_Threat\\_Response.pdf](http://www.gameaipro.com/GameAIPro/GameAIPro_Chapter30_Using_Neural_Networks_to_Control_Agent_Threat_Response.pdf). Accessed 12 July 2019.
- Sabourin, J. L., Shores, L. R., Mott, B. W., & Lester, J. C. (2013). Understanding and predicting student self-regulated learning strategies in game-based learning environments. *International Journal of Artificial Intelligence in Education*, 23(1–4), 94–114.
- Schuller, D., & Schuller, B. W. (2018). The age of artificial emotional intelligence. *Computer*, 51(9), 38–46. <https://doi.org/10.1109/MC.2018.3620963>.
- Sebe, N. (2009). Multimodal interfaces: Challenges and perspectives. *Journal of Ambient Intelligence and Smart Environments*, 1(1), 23–30.
- Shannon, C. E. (1950). Programming a Computer for Playing Chess. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 41(314), 256–275.
- Shute, V. J. (2011). Stealth assessment in computer-based games to support learning. *Computer games and instruction*, 55(2), 503–524.
- Shute, V. J., Ventura, M., & Kim, Y. J. (2013). Assessment and learning of qualitative physics in newton's playground. *Journal of Educational Research*, 106(6), 423–430.
- Shute, V. J., Wang, L., Greiff, S., Zhao, W., & Moore, G. (2016). Measuring problem solving skills via stealth assessment in an engaging video game. *Computers in Human Behavior*, 63, 106–117.
- Socher, R., Perelygin, A., Wu, J. Y., Chuang, J., Manning, C. D., Ng, A. Y. & Potts, C. (2013). Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank. *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing* (pp. 1631–1642). Stroudsburg, PA: ACL. Retrieved from <http://www.aclweb.org/anthology/D13-1170>. Accessed 12 July 2019.
- Steiner, C., Gaisbachgrabner, K., Nussbaumer, A., Mertens, J., Hemmje, M., Nadolski, R. J., Westera, W., Bazzanella, B., Casagrande, M., Moinari, A., Humphreys, S., Mascarenhas, S., Guimarães, M., Prada, R. & Santos, P. A. (2018). *D8.4 – Second RAGE Evaluation Report*. RAGE project. <https://research.uu.nl/en/publications/d84-second-rage-evaluation-report>. Last accessed June 24, 2019.
- Stewart, J., Bleumers, L., Van Looy, J., Mariën, I., All, A., Schurmans, D., Willaert, K., De Grove, F., Jacobs, A. & Misuraca, G. (2013). *The Potential of Digital Games for Empowerment and Social Inclusion of Groups at Risk of Social and Economic Exclusion: Evidence and Opportunity for Policy*. C. Centeno (Ed.), Joint Research Centre, European Commission. Retrieved from <http://ipts.jrc.ec.europa.eu/publications/pub.cfm?id=6579>. Accessed 12 July 2019.
- Tai, K. S., Socher, R. & Manning, C. D. (2015). Improved semantic representations from tree-structured Long short-term memory networks. *Proceedings of ACL* (pp. 1556–1566). Stroudsburg, PA: ACL. <https://doi.org/10.1515/popets-2015-0023>.
- Taylor, S., Kim, T., Yue, Y., Mahler, M., Krahe, J., Rodriguez, A. G., Hodgins, J. & Matthews, I. (2017). A deep learning approach for generalized speech animation. *ACM Trans. Graph.* 36(4), article 93.
- Thiebaut, M., Marsella, S., Marshall, A. N. & Kallmann, M. (2008). Smartbody: Behavior realization for embodied conversational agents. In: *Proceedings of the 7th International Joint Conference on Autonomous Agents and Multiagent Systems - Volume 1, ser. AAMAS '08, Estoril, Portugal*. Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems (pp. 151–158). Retrieved from <https://pdfs.semanticscholar.org/2ba6/7705aa5d45ea3cff74104d0cb9082dd7add7.pdf>. Accessed 12 July 2019.
- Van den Berg, J., Guy, S. J., Lin, M. & Manocha, D. (2011). Reciprocal n-body collision avoidance. In: C. Pradalier, R. Siegwart & G. Hirzinger (Eds.), *Robotics Research. Proceedings of the 14th International Symposium of Robotic Research, ISRR 2009 - Lucerne, Switzerland*. Springer tracts in advanced robotics, Vol. 70 (pp. 3–19). [https://doi.org/10.1007/978-3-642-19457-3\\_1](https://doi.org/10.1007/978-3-642-19457-3_1). Retrieved from <http://gamma.cs.uic.edu/ORCA/publications/ORCA.pdf>
- Van der Maas, H. L. J., & Nyamsuren, E. (2017). Cognitive analysis of educational games: The number game. *Topics in Cognitive Science*, 9, 395–412. <https://doi.org/10.1111/tops.12231>.
- Van der Vegt, W., Nyamsuren, E., & Westera, W. (2016a, 6–7 June). RAGE reusable game software components and their integration into serious game engines. In: G. M. Kapitsaki & E. Santana de Almeida (Eds.), *Bridging with Social-Awareness, 15th International Conference, ICSR 2016, Limassol, Cyprus*. Proceedings, lecture notes in computer science, Vol. 9679 (pp. 165–180). Cham: Springer.
- van der Vegt, W., Westera, W., Nyamsuren, E., Georgiev, A., & Martínez Ortiz, I. (2016b). RAGE architecture for reusable serious gaming technology components. *International Journal of Computer Games Technology*, Article ID, 5680526, 1–11. <https://doi.org/10.1155/2016/5680526>. Accessed 12 July 2019.
- Ventura, M., Shute, V., & Small, M. (2014). Assessing persistence in educational games. Design recommendations for adaptive intelligent tutoring systems. *Learner Modeling*, 2, 93–101.

- Vygotsky, L. S. (1978). Interaction between learning and development. In L. S. Vygotsky (Ed.), *Mind and society* (pp. 79–91). Cambridge MA: Harvard University Press.
- Westera, W., Nadolski, R., Hummel, H., & Wopereis, I. (2008). Serious games for higher education: A framework for reducing design complexity. *Journal of Computer Assisted Learning*, 24(5), 420–432.
- Westera, W., Nadolski, N., & Hummel, H. (2014). Serious gaming analytics: What students' log files tell us about gaming and learning. *International Journal of Serious Games*, 1(2), 35–50.
- Westera, W., Dascalu, M., Kurvers, H., Ruseti, S., & Trausan-Matu, S. (2018). Automated essay scoring in applied games: Reducing the teacher bandwidth problem in online training. *Computers & Education*, 123, 212–224. <https://doi.org/10.1016/j.compedu.2018.05.010>.
- Yannakakis, G. N. (2012). Game AI revisited. *Proceedings of the 9th Conference on Computing Frontiers, Cagliari* (pp. 285–292). New York: ACM. <https://doi.org/10.1145/2212908.2212954>.
- Yannakakis, G. N., & Togelius, J. (2015). A panorama of artificial and computational intelligence in games. *IEEE Transactions on Computational Intelligence and AI in Games*, 7(4), 317–335.
- Yannakakis, G. N. & Togelius, J. (2018). Artificial intelligence and games. Berlin: Springer. Retrieved from <http://gameaibook.org/>. Accessed 12 July 2019.
- Yumak, Z., & Magnenat-Thalmann, N. (2015). Multi-modal and multi-party social interactions. In N. Magnenat-Thalmann, J. Yuan, D. Thalmann, & B. You (Eds.), *Context aware human-robot and human-agent interaction* (pp. 275–298). Cham: Springer International Publishing.
- Yumak, Z., van den Brink, B., & Egges, A. (2017). Autonomous social gaze model for an interactive virtual character in real-life settings. *Computer Animation and Virtual Worlds*, 3(4), e1757. <https://doi.org/10.1002/cav.1757>.

**Publisher's note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

## Affiliations

Wim Westera<sup>1</sup> · Rui Prada<sup>2,3</sup> · Samuel Mascarenhas<sup>2</sup> · Pedro A. Santos<sup>2,3</sup> · João Dias<sup>2,3</sup> · Manuel Guimarães<sup>2,3</sup> · Konstantinos Georgiadis<sup>1</sup> · Enkhbold Nyamsuren<sup>1</sup> · Kiavash Bahreini<sup>1</sup> · Zerrin Yumak<sup>4</sup> · Chris Christyowidiasmoro<sup>4</sup> · Mihai Dascalu<sup>5</sup> · Gabriel Gutu-Robu<sup>5</sup> · Stefan Ruseti<sup>5</sup>

Rui Prada  
rui.prada@tecnico.ulisboa.pt

Samuel Mascarenhas  
samuel.mascarenhas@gaips.inesc-id.pt

Pedro A. Santos  
pedro.santos@tecnico.ulisboa.pt

João Dias  
joao.dias@gaips.inesc-id.pt

Manuel Guimarães  
manuel.m.guimaraes@tecnico.ulisboa.pt

Konstantinos Georgiadis  
Konstantinos.Georgiadis@ou.nl

Enkhbold Nyamsuren  
e.nyamsuren@gmail.com

Kiavash Bahreini  
kbsocialnetwork@gmail.com

Zerrin Yumak  
Z.Yumak@uu.nl



Chris Christyowidiasmoro  
C.Christyowidiasmoro@uu.nl

Mihai Dascalu  
mihai.dascalu@cs.pub.ro

Gabriel Gutu-Robu  
gabriel.gutu@cs.pub.ro

Stefan Ruseti  
stefan.ruseti@cs.pub.ro

- <sup>1</sup> Open University of the Netherlands, PO Box 2960, 6401 DLHeerlen, The Netherlands
- <sup>2</sup> INESC-ID, Rua Alves Redol, 9, 1000-029 Lisbon, Portugal
- <sup>3</sup> Instituto Superior Técnico, Universidade de Lisboa, Av. Rovisco Pais 1, 1049-001 Lisbon, Portugal
- <sup>4</sup> Game Research, Department of Information and Computing Sciences, Utrecht University, P.O. Box 80.089, 3508 TBUtrecht, The Netherlands
- <sup>5</sup> University Politehnica of Bucharest, Splaiul Independenței nr. 313, Sector 6, RO-060042 Bucharest, Romania