Report from Dagstuhl Seminar 12191

Artificial and Computational Intelligence in Games

Edited by

Simon M. Lucas[1], Michael Mateas[2], Mike Preuss[3], Pieter Spronck[4], and Julian Togelius[5]

1 **University of Essex, GB,** `sml@essex.ac.uk`
2 **University of California – Santa Cruz, US,** `michaelm@cs.ucsc.edu`
3 **TU Dortmund, DE,** `mike.preuss@tu-dortmund.de`
4 **Tilburg University, NL,** `p.spronck@uvt.nl`
5 **TU Dortmund, DE,** `julian@togelius.com`

Abstract

This report documents the program and the outcomes of Dagstuhl Seminar 12191 "Artificial and Computational Intelligence in Games". The aim for the seminar was to bring together creative experts in an intensive meeting with the common goals of gaining a deeper understanding of various aspects of artificial and computational intelligence in games, to help identify the main challenges in game AI research and the most promising venues to deal with them. This was accomplished mainly by means of workgroups on 14 different topics (ranging from search, learning, and modeling to architectures, narratives, and evaluation), and plenary discussions on the results of the workgroups. This report presents the conclusions that each of the workgroups reached. We also added short descriptions of the few talks that were unrelated to any of the workgroups.

Seminar 6.–11. May, 2012 – www.dagstuhl.de/12191
1998 ACM Subject Classification I.2.1 Applications and Expert Systems.
Keywords and phrases Artificial Intelligence, Computational Intelligence, Computer Games
Digital Object Identifier 10.4230/DagRep.2.5.43
Edited in cooperation with Nicola Hochstrate

1 Executive Summary

Simon M. Lucas
Michael Mateas
Mike Preuss
Pieter Spronck
Julian Togelius

The video game industry is the largest of the entertainment industries and growing rapidly. The foundations of this industry are techniques from computer science. New developments within video games pose fresh challenges to computer scientists. Around the world, the number of dedicated study programs producing the workforce of the game industry is increasing steadily, as is the number of computer science academics dedicating their careers to solving problems and developing algorithms related to video games. Such problems often require domain knowledge from various research domains, such as psychology and the arts, leading to an inherently interdisciplinary research field.

Artificial and Computational Intelligence in Games, *Dagstuhl Reports*, Vol. 2, Issue 5, pp. 43–70
Editors: Simon M. Lucas, Michael Mateas, Mike Preuss, Pieter Spronck, and Julian Togelius.
DAGSTUHL REPORTS Dagstuhl Reports
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Artificial intelligence (AI) and computational intelligence (CI), in one form or another, can be found at the heart of almost any video game, controlling the non-player characters (NPCs) as well as many aspects of the game world. They are also used throughout the game design and development process. Academic research within these domains in games aims to solve problems and enable innovation, pertaining to game design, game development, and gameplay. A main focus is on solving algorithmic problems to make game mechanisms more intelligent and efficient, thus making games more immersive, interesting, and entertaining. In the context of serious and educational games, such improvements enable these games to fulfill their societal objectives better.

Artificial intelligence seeks to simulate intelligent behavior in any possible way with human intelligence as a paradigm. Computational intelligence is an umbrella term for nature-inspired computational methods for optimization, learning and controlling. The main methods are evolutionary algorithms, artificial neural networks, fuzzy logic, swarm intelligence, and artificial immune systems. Nowadays, the borders between both disciplines are blurred, and state-of-the-art solutions use hybrid techniques combining elements of symbolical AI systems, CI algorithms and methods from statistical machine learning.

The aim for the *Dagstuhl Seminar on Artificial and Computational Intelligence in Games* was to bring together creative experts in an intensive meeting with the common goals of gaining a deeper understanding of various aspects of games, and of further improving games. It was meant to enforce the communication of different communities and the collaboration with the games industry. The exchange of different views and competencies was to help identify the main challenges in game AI research and the most promising venues to deal with them. This could lead to a common vision of what kind of games could be made possible in the future.

The Seminar was held from Sunday, May 6, 2012, until Friday, May 11, 2012. Over 40 researchers came together at Schloß Dagstuhl, many of them highly-respected and well-known researchers in their field, but also several talented young researchers and even a few representatives from the AI specialists of the game industry. In contrast to what is common for such gatherings, very little time was spent on plenary talks. Instead, the focus was on workgroups which discussed particular topics. However, several plenary sessions were held in which the workgroups reported on their results, and new topics for discussion were brought up. To allow researchers to present their recent work, a poster session was held during the second day of the Seminar, and the posters remained up until the end.

The topics of the workgroups, in alphabetical order, were the following:

- AI Architectures for Commercial Games
- AI Clearing House
- AI for Modern Board Games
- Computational Narratives
- Evaluating Game Research
- Game AI for Mobile Devices
- General Game Playing
- Learning in Games
- Pathfinding
- Player Modeling
- Procedural Content Generation
- Search
- Social Simulation Games
- Video Game Description Languages

In the remainder of this report, abstracts for all the workgroups, describing their discussions and results, are given. Several researchers wrote a short report on their poster, which are included too. We aim to bring a full reports of all the workgroups in the form of proceedings later.

As organizers we are really pleased with how the Seminar turned out. It proved to be the stimulating and inspirational environment that we had hoped for. We found that most, if not all participants agreed with us on that. A lot of this success is due to the excellent facilities provided by the people of Schloß Dagstuhl. We are highly grateful for having had the opportunity to be their guests for the Seminar. We definitely hope to return in the future.

Simon M. Lucas
Michael Mateas
Mike Preuss
Pieter Spronck
Julian Togelius

2 Table of Contents

Overview of Short Talks

3 Overview of Workgroups

This section contains an overview of the results of each of the 14 workgroups.

3.1 Pathfinding in Games

Adi Botea, Christian Bauckhage, Bruno Bouzy, Michael Buro, and Dana Nau

Pathfinding remains one of the most important applications of artificial intelligence in commercial games. The problem has received a significant attention in recent years, both from academic researchers and game developers. In the past, A* search, combined with a heuristic function such as the Manhattan distance, and performed on grid-based map representations, used to be a common approach to computing paths in video games.

In recent years, there has been significant progress in this area: much faster path computation methods based on hierarchical abstractions, more informed, memory-based heuristic functions, symmetry reduction, triangulation-based map representations, and compressed databases with all-pairs shortest paths information. Some of the novel techniques, such as contributions to hierarchical abstraction and triangular map encoding, have already been adopted in video games.

Despite these impressive advances, there are many challenges to be addressed in pathfinding in video games. Previous work often made a number of restrictive assumptions, such as a fully known, static environment in which units are often treated as independent from each other. Such a narrow, single-agent view shows its limits in both collaborative multi-agent pathfinding, and adversarial pathfinding. For example, units can be trapped in deadlocks or collide with each other during path execution. Also, taking physical constraints, such as the speed and minimal turning angle of units, into account, can greatly increase the realism of generated paths. Moreover, ever growing map sizes and speed and memory limitations of game consoles demand more efficient pathfinding algorithms.

Another challenge is to increase academia–industry collaboration. Progress in this direction could be facilitated by ensuring that benchmark problems that are used to evaluate academic research overlap more with problems that are truly relevant to game developers. An example is multi-agent pathfinding, where recent academic contributions have focused on either providing optimal solutions or scaling suboptimal methods to hundreds or even thousands of units. While important on their own, such objectives are not necessarily the most critical ones in current games.

The pathfinding workgroup at the Dagstuhl seminar 12191 has focused on acquiring a snapshot of current pathfinding approaches and identifying important topics for a future research roadmap. In this abstract we have summarized the main conclusions of the workgroup.

3.2 Search in Real-Time Video Games

Peter Cowling, Michael Buro, Bruno Bouzy, Dana Nau, Martin Butz, Philip Hingston, Adi Botea, Moshe Sipper, Hector Muñoz-Avila, and Michal Bida

It is fascinating to speculate as to whether the search approaches that have been so successful for Chess, Go, Checkers, etc. can be used to create strong / interesting / robust players for video games. This large and very active / lively group met several times during the Dagstuhl seminar week to investigate the current state of the art, the benefits and challenges in this area, and promising techniques and directions for future research. This report provides a brief overview of the main points of our discussion, which will soon be extended to a much more detailed summary and road map for future research.

The benefits of using search in creating AI players for turn-based board and card games are readily apparent – with chess programs running on personal computers not losing against world-champions anymore, and Go programs playing at expert level. Real-time video games, however, provide a range of new challenges, and effective search-based players for these games could provide a new dimension in gameplay. For example, a new generation of AI players may develop for complex, real-time strategy and first-person games. Restricting the information available to the AI player to the information a human player may be able to gather in the game, and restricting the actions of the AI player to human player actions (executed in time and space) will make AI players more human-like and also more appealing to human players. Developing search techniques, in such game scenarios, yielding AI players that continue to be effective and interesting against any strategy adopted by a human opponent currently poses a fundamental challenge to the community.

In addition to the potential of transferring search techniques from game to game, greatly reducing development time, new game components may be developed in which lower-level game tasks are accomplished by AI-based search techniques. For instance, using a hierarchy of search levels, a command hierarchy could be simulated for games where a player controls many agents, giving high-level strategy, and recognizable tactics to work alongside effective and realistic low-level behaviors.

In general, the issue for real-time strategy and first-person games is to provide “strong” non-cheating players, and the questions whether a player is "strong" and whether gameplay is “interesting” are closely linked (by the fact that strong players can be adapted to lower playing strength). Hence, the focus of the group was on maximizing AI player strength – which may also provide an easier set of metrics for judging progress in AI in games research. Techniques developed for strategic video games also have much potential for serious games and military simulations.

The use of search allows strategy to emerge from reusable components:

- search algorithms
- knowledge (such as tactical notions) embodied in state evaluation functions and scripted policies
- tuning approaches such as evolutionary algorithms, reinforcement learning, and data mining

Search Techniques

The state of the art in search algorithms includes Monte Carlo Tree Search (MCTS), which shows tremendous promise for games where it is difficult to estimate the value of an intermediate state, such as Go and Hex, as well as showing great promise for hidden information card and board games. Alpha-Beta minimax search has yielded world-champion level players for a range of perfect-information board games including Chess, Checkers, and Othello, as well as high level play in games with hidden information such as Bridge. A* search is the algorithm of choice for a range of pathfinding and other planning problems. Heuristics and Evolutionary Algorithms have used analogies with human and natural problem solving approaches to achieve excellent results for optimization problems and single-player games.

A number of approaches have been developed to capture strategy and knowledge, and to deal with the real-time nature of decision making. These include Hierarchical Task Networks (HTNs), which show much promise as they capture strategy in a form within which search algorithms can be applied. Time slicing is a simple approach for dealing with real-time decision making, which has shown effectiveness in simple real-time domains. Hierarchical and level-of-detail approaches allow the computational budget to be allocated where most needed. Dynamic Scripting and "Black Box" approaches, which use learning and search techniques to choose between a range of scripted approaches, also show promise.

Challenges

The challenges in this area are manifold. Particularly in comparison with turn-based games, the most obvious challenge is the massive branching factor and depth that comes from simulating a continuous environment, where decisions can be made at any time (or at least at 20 frames per second). Manually capturing an abstraction of states and actions in such a space is difficult, and automatically finding abstractions poses even a greater challenge. Any abstractions should ideally be "continuous", i.e., preserving the property that nearby intermediate states lead to similar outcomes. In order to use search, abstractions must also allow simulations forward in time. Moreover, most strategic video games have multiple players and asymmetric hidden information, requiring the investigation of new search algorithms. It is also highly desirable that the decisions made by search are robust (in terms of varying opponents) and explainable to convince video companies to integrate search-based techniques in their products. Finally, the search must use hardware that, although becoming increasingly powerful, provides stringent limits on how complex the applied abstractions may be and how deeply/broadly the game tree may be searched while still adhering to the real-time game constraints.

A number of existing techniques represent promising directions for future research. Hierarchical Task Networks (HTNs) can be used to reason and represent high-level decisions while other representational/reasoning methods can be used for low-level decision-making. HTNs can be obtained using a combination of manual approaches and machine learning. The use of search trees with simple atomic actions (such as Group() and Move()) provides a framework for thinking about these lower level problems as well as providing a controllable level of complexity for pathfinding and tactics. Considering game theoretic techniques at an appropriate level of complexity allows for an effective way of handling hidden information and opponent models. Maintaining the complete tree (at least at an abstract level) but partitioning it into "similar" states provides a promising general- purpose direction for dealing with complexity and hidden information. Maintaining trees of strategic choices between other "black box" strategies (e.g. manually generated rule-based ones) also shows much promise.

Further investigations in the areas discussed in this paragraph are likely to indicate many other promising directions in turn.

Outlook

In the future we can imagine video games where AI strategies based on search provide strong, interesting gameplay against any possible strategy, removing the possibility of uninteresting tactics typically used to exploit AI limitations. These general-purpose approaches are expected to be easily transferable from game to game, providing bottom-up abstractions to find high-level strategies, given only information about game rules and physics. For success, however, hybrid and hierarchical approaches will likely be needed. Fast simulations of game states will need to be available as part of this general-purpose approach as well, yielding diverse controllable strategies through parametrization, and explainability through information about the part of the tree searched and simulation outcomes. Machine learning, particularly mining the data from games played between humans, should allow the automatic capture of strategies using tools such as Hierarchical Task Networks. Evolutionary, machine learning, and heuristic approaches will allow opponent strategies to be modeled and exotic new strategies to be found – also in order to extend human gameplay interest. Rather than AI players that follow rather predictable strategies and that were designed in advance, search will allow for flexible AI whose decisions will be as interesting and varied as human expert players. Difficulty levels will be tuned using varying search time and abstraction properties.

To achieve research success, the development and widespread adoption of competitions and benchmarks of increasing complexity will be necessary. There are promising developments here – competitions such as the current Starcraft AI competition [1] provide a platform for researchers to interact with a commercial video game. It is hoped that more competitions will be developed and more video games will allow open access to their game information.

There are links here with other groups at Dagstuhl: the modern board games group (which is investigating highly complex, multiplayer, hidden information games, but with discrete turns), the General Video Game Playing group (which is investigating a platform for studying general- purpose techniques for video games), and the Learning group (which focuses even more on the learning challenge in game environments).

3.3 AI and CI Games for Mobile Devices

Philip F. Hingston, Clare Bates Congdon, and Graham Kendall

This group considered the possibilities of creating new and innovative games that are targeted at mobile devices, such as smart phones and tablets, and that showcase AI (Artificial Intelligence) and CI (Computational Intelligence) approaches. Such games might take advantage of the sensors and facilities that are not available on other platforms, or might simply rely on the "app culture" to facilitate getting the games into users' hands. While these games might be profitable in themselves, the focus of our discussion was on the benefits and challenges of developing AI and CI games for mobile devices.

[1] http://webdocs.cs.ualberta.ca/ cdavid/starcraftaicomp/

We recognize that mobile games are easier to bring to market than commercial (large scale) video games. This makes them a practical choice for development and study in an academic environment, using relatively small teams of academics and students, who are able to work on relatively low budgets. For example, the relatively small screen size and lack of powerful graphics hardware typical of mobile devices means that simple graphics, often only 2 or 2.5 dimensional are expected, so that large teams of highly skilled artists and 3D modelers are not required. Furthermore, mobile devices usually provide a wider variety of input data (touch, location, images, video, sound, acceleration, orientation, personal data, data from/about other users etc.) and offer more output options (images, video, animation, sound, vibration, wireless, bluetooth, infrared) than is normally available on a desktop or laptop computer. In addition, the popularity of mobile devices provides a means to recruit large numbers of casual users, which provides another potentially large data source. Novel game mechanics and interaction methods might be made possible by processing these input data using AI and CI methodologies.

Computational power and battery life present two potential obstacles to intensive AI/CI-based games, and some potential designs will require offloading some of the computation to servers. It might also be difficult to implement large-scale, complex game worlds due to the limited resources that are available. There are also significant challenges in developing AI/CI libraries that can work with low memory, limited battery power etc., adapting or developing AI/CI methods to work effectively in games that are played in short bursts, using unreliable communications, and providing real-time responses. However, these constraints provide significant research opportunities.

In summary, our group felt that mobile devices are still "young" enough to provide opportunities for developers to implement innovative products without having to employ large specialist teams (e.g. graphic designers, musicians etc.) This opportunity will not last long (perhaps less than two years) so those who are interested in this area might want to explore it now. Moreover, CI/AI both provide significant opportunities both in terms of research challenges and also to make the games more interesting and more fun to play. We would like to see the research community take up the challenge to showcase what can be done with the limited resources available on mobile devices, but also utilizing the larger number of sensors (e.g. movement detection) and other options (e.g. location awareness) which are not available on traditional "living room" game consoles.

3.4 Generalized Video Game Playing

John Levine, Clare Bates Congdon, Michal Bida, Marc Ebner, Graham Kendall, Simon Lucas, Risto Miikkulainen, Tom Schaul, and Tommy Thompson

One of the grand challenges of AI is to create general intelligence: an agent that can excel at many tasks, not just one. In the area of games, this has given rise to the challenge of General Game Playing (GGP). In GGP, the game (typically a turn-taking board game) is defined declaratively in terms of the logic of the game (what happens when a move is made, how the scoring system works, how the winner is declared, and so on). The AI player then has to work out how to play the game and how to win. In this work, we seek to extend the idea of General Game Playing into the realm of video games, thus forming the area of

General Video Game Playing (GVGP). In GVGP, computational agents will be asked to play video games that they have not seen before. At the minimum, the agent will be given the current state of the world and told what actions are applicable. Every game tick the agent will have to decide on its action, and the state will be updated, taking into account the actions of the other agents in the game and the game physics. We envisage running a competition based on GVGP playing, using arcade-style (e.g. similar to Atari 2600) games as our starting point. These games are rich enough to be a formidable challenge to a GVGP agent, without introducing unnecessary complexity. The competition that we envisage could have a number of tracks, based on the form of the state (frame buffer or object model) and whether or not a forward model of action execution is available. We propose that the existing Physical Traveling Salesman (PTSP) software could be extended for our purposes and that a variety of GVGP games could be created in this framework by AI and Games students and other developers. Beyond this, we envisage the development of a Video Game Description Language (VGDL) as a way of concisely specifying video games. For the competition, we see this as being an interesting challenge in terms of deliberative search, machine learning and transfer of existing knowledge into new domains.

3.5 Artificial Intelligence Architectures for Games

Daniele Loiacono, Michal Bida, and Alex Champandard

Despite the differences in technology and the unique designs of specific games, the Artificial Intelligence architectures for commercial titles are based on similar principles. Although in the literature several efforts have been done to define a standard AI architecture for games, the field suffers from a lack of consensus about the best approach to follow. There may not even be a single best approach to follow, but by identifying underlying patterns we can make progress nonetheless.

In particular, two major approaches can be identified when it comes to defining AI architectures for games: an input-output taxonomy and a design based on different abstraction layers. Both the approaches involve modular and layered architectures. In the first approach, the components are typically organized with (i) an input layer, which processes the data from the game environment; (ii) a middle layer (or more middle layers), which elaborates processed data and takes decisions; (iii) an output layer, which map decisions into actions in the game environment. In contrast, the second approach consists of organizing the components of the architectures in different abstraction layers: (i) the bottom layer deals directly with the game engine; (ii) the middle layer(s) provides a convenient abstraction of the game environment; (iii) the top layer deals with reasoning and high-level decisions.

Rather than introducing an alternative approach to design AI architectures for games, we propose a taxonomy of the typical components they involve. To this purpose, we combined the underlying principles of the approaches previously described: components of the architecture are classified with respect to both their level of abstraction and their role in the input-output pipeline. Our aim is to support the analysis and the understanding of the existing middleware, commercial games and academic frameworks. The proposed taxonomy might be used to understand which components are provided by a given middleware, which is the standard technology to implement a specific component, how the gameplay affects the design of the AI architecture, etc. In addition, it might also allow to identify the major challenges in specific areas and to discover opportunities for the game research.

As an example, we selected a set of typical components involved in an AI architecture for games and discussed about the major challenges they provide; then, we identified the most promising areas in our taxonomy not yet fully exploited in existing frameworks; finally, we used our taxonomy to classify and to compare popular middleware and games.

3.6 Believable Agents and Social Simulations

Michael Mateas, Elizabeth André, Ruth Aylett, Mirjam P. Eladhari, Richard Evans, Ana Paiva, Mike Preuss, and Michael Young

The topics for this workgroup were believable agents and social simulations in digital games. The environments discussed were interactive short form dramas and massively multi-player online (MMO) games. For the MMOs our vision was to populate them with rich characters, massive interaction affordances, and drama in every situation. When envisioning short form dramas we saw massively repayable dramatic situations as well as many actions available at any moment. We discussed aspects of the experience, and games where the player has rich, near human bandwidth (symmetric interaction). When discussing affective engagement we focused on ways to evoke a much broader range of emotions.

Our discussions addressed three main topics: scaling up, believable characters, and social modeling. For each of these topics we identified the main challenges and opportunities.

Scaling up – The Authoring Wall

When using traditional game development methods the authoring effort becomes unmanageable if the game world becomes very big or complex as illustrated in Figure 1. It becomes difficult to control the dynamism so that the players' experiences are enjoyable and make sense. An important challenge for research in AI is to find ways to overcome the authoring wall, allowing increased dynamism in future games. A challenge for researchers in this area is that, for only moderately dynamic experiences, traditional game development techniques (finite state machines, scripting, etc.) take less authoring effort than using generative AI techniques. So in building research games, it can be difficult to demonstrate that the AI is enabling a previously impossible game experience except by building a highly dynamic, fully playable experience, something many research teams lack the resources and experience to do.

Opportunities for moving beyond the authoring wall include developing techniques for crowd-sourcing and supporting user-generated AI content. Authoring tool for non-experts is another approach, with Inform 7 serving as an existence proof that a highly declarative, rule-based authoring paradigm can be made accessible to writers without computer science training. We noted that declarative representations scale more readily than procedural ones. There is the opportunity to learn lessons from Knowledge Engineering tools in the 1980s, which encoded domain structure to guide knowledge acquisition (authoring).

Believable Characters

We identified the current state of the art in the development of believable characters to be:

- use of planning models – reactive and traditional;

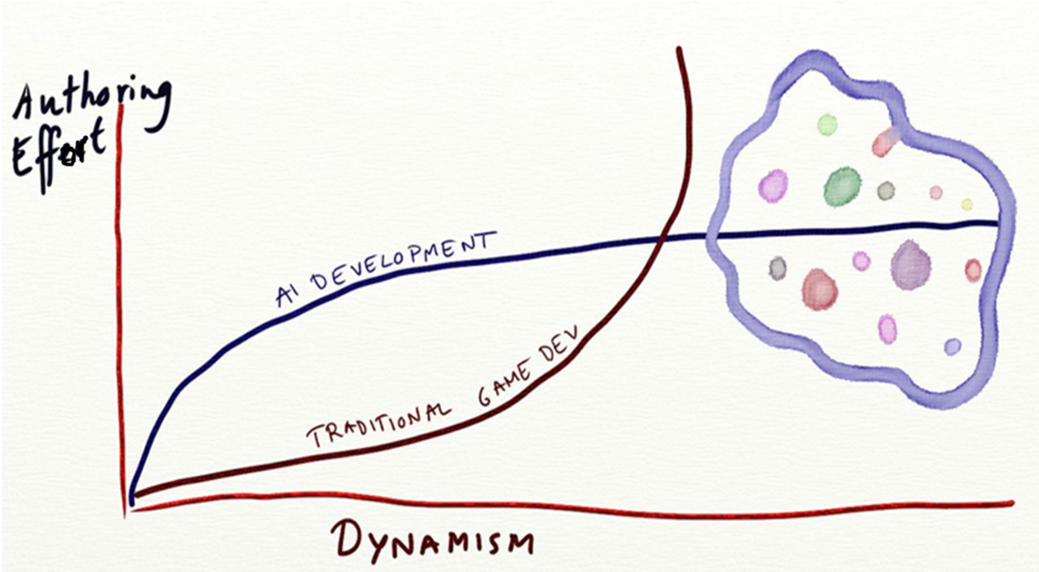

Figure 1 The Authoring Wall

- use of plan-based representations of physical, social and language actions (domain specific hierarchical planning);
- emotion modeling with convergence on cognitive appraisal architectures;
- FSMs (returning for use in dialogue- moving towards performance of interactive dialogue);
- personality captured through traits that modulate behavior and expression;
- personality-trait models less directly tied to expression but strongly correlated with multiple cognitive processes (so are more generally applicable across capabilities);
- mood modeled as a function of emotions (sums, averages) and with decay;
- modeling mood through spreading activation. When a character is reminded of something that is emotionally significant, there is an emotional echo; and
- episodic memory supporting temporal queries, evaluative labeling, and generalization.

We noted that most agent learning focuses on easy-to-evaluate criteria. For believable characters, significant challenges remain for making learning personality-specific and maintaining believability while online learning engages in exploration.

Opportunities for future development of believable characters include:

- to combine statistical language model for style with semantic or symbolic content models;
- dynamically generated character soundtrack communicating character and social state;
- two directions for more information-rich signals from:
  - player: high-bandwidth naturalistic interaction (gestures, gaze, prosody); new communication modes (biometrics, out-of-band interactions like music selection, player modeling);
  - purposefully set up choices for characters to allow them
    - to express personality through choice, and
    - conversely set up player choices that give information about player;
- take advantage of low-cost motion capture to correlate prosodics and gesture features in generative models (has to be parameterized by emotion and personality and social context); and
- using explainable AI to drive interface elements (text explanations) or thought bubbles.

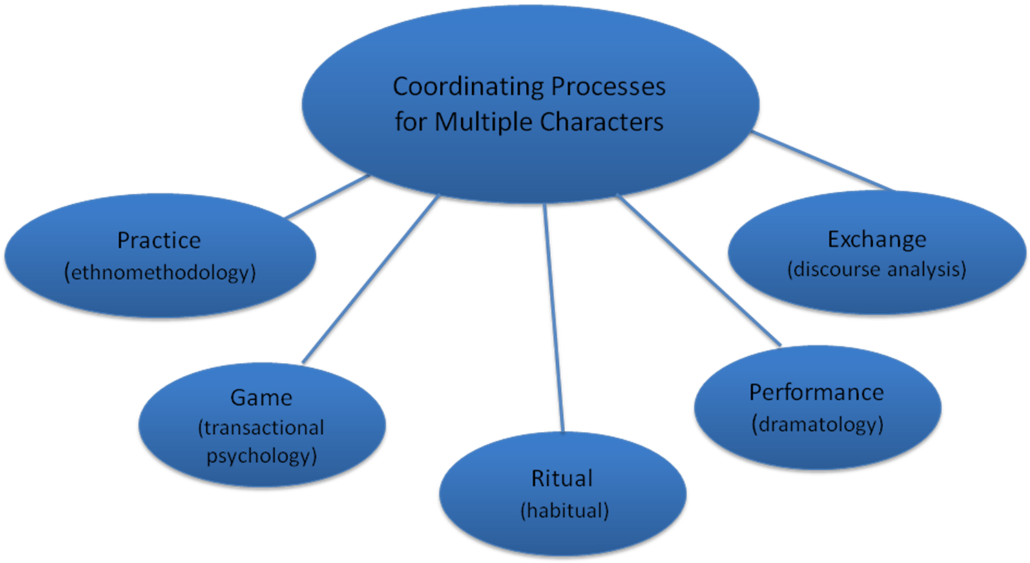

Figure 2 Coordinating Processes of Multiple Characters

Regarding expression of believable agents we recognized several limitations:
- the rich emotional models that are computed reside in the side-scenes and are not currently being expressed to players;
- multi-modal expression often results in inconsistencies between modes (uncanny); and
- handcrafted art- and audio assets currently work best (as opposed to procedurally created content), but create an authoring bottleneck.

Opportunities we recognized regarding expression were:
- to go beyond naturalistic world simulation affordances to express character state;
- reifying state in characters and objects;
- behavior explanation (HUD); and
- use of stylized comic-inspired forms, music, abstract visuals.

Social Modelling

In the current state of the art of social modeling we are converging on architectures with multiple social practices above the agent level, but using different approaches of which some are noted in Figure 2. In general, work on social modeling is far less developed than work on believable characters who engage in limited social interactions.

Challenges in the field of social modeling include addressing that we are unable to share social practices between systems and that there is no common theoretical basis for social practices (and no common representational formalism).

Opportunities for social modeling include:
- make connections with agent-based social simulations;
- make a detailed comparison between different social models to compare how they represent the same social practice; and
- use a situation which has been tagged with social-annotations as a common target for comparing the different formalisms.

3.7 Learning and Game AI

Hector Muñoz-Avila, Michal Bida, Graham Kendall, Christian Bauckhage, and Clare Bates Congdon

We focused on learning aspects in current computer games, challenges, and opportunities for future applications. Unlike the General Video Game Playing group, which focused on an unified environment for a class of games, we looked at the broader issues with learning independently of languages and platforms. Unlike the planning and search group, we considered learning even in the event in which no state-action transition models are given including topics related to data mining and pattern recognition of playing logs.

State of the art

In our discussions we observed that there are very few applications of machine learning to commercial games. Those of us who teach courses or give lectures on the topic, tend to focus on the few well- documented success stories (e.g., the use of induction of decision trees in the commercial game Black and White). There is a number of noncommercial applications of machine learning to game (e.g., the use of reinforcement learning to play Backgammon). The use of machine learning to find patterns from network and log data has demonstrated significant potential (insert- Christian-paper-here). Also there is significant research, demonstrating the use of Monte Carlo techniques and evolutionary computation to evolve rules for high-performance for arcade games (cite-Clare-paper(s)-here), to learn

Challenges

We identify five kinds of challenges.
1. Need to explain decisions. It will be desirable for machine learning algorithms to explain its decisions. Whereas in non-commercial applications such as determining if a credit application to be granted or not, it is fine if the algorithm doesn't explain itself (because a generic explanation to the customer suffices), in the gaming context, poor performance in a small number of situations will be magnified through social media and player-to-player communication. In addition, assessing quality is very difficult because the search space in many commercial games (e.g., a real-time strategy game) is too large.
2. Selecting machine learning technique. Another challenge is that there is no simple answer if a game developer asks the question about which machine learning to use. In other areas (e.g., pathfinding), the capabilities and limitations of some techniques relative to others is well understood.
3. Obtaining the input data. Getting the data for input to test the machine learning algorithms can be difficult. There is no clear value added for a commercial company to gather and share the data.
4. Need to demonstrate value added. If a gaming company has money to invest in a game, aspects such as graphics will get prioritized simply because it is unclear what the benefit from investing on machine learning is.
5. In some situations adaptable AI might not be desirable. In games that for reasons such as commercial considerations, the expected time the game will be played is bounded to, say, 20 hours, having adaptable AI can make it replayable for a long time and hence it might be undesirable to have those capabilities.

Opportunities

We identify a number of opportunities for machine learning techniques including:
1. Balancing gaming elements. Many games have different elements such as factions in a real-time strategy games (e.g., humans versus orcs) or classes in a role-playing game (e.g., mages versus warriors). Machine learning could help with balancing these elements.
2. Balancing game difficulty. Games, particularly those that are open- ended such as massive multiplayer online (MMO) games. A difficulty is how to tailor the game simultaneously towards dedicated players (e.g., players who play 20+ hours per week) and casual players (e.g., players who play 10 hours or less a week).
3. Finding loopholes in games. Pattern recognition techniques can be used to first detect usual patterns in game logs and then use these patterns to detect outliers. Such techniques will enable to detect anomalies (e.g. exploits in MMOs) much faster than it is currently done, which is manually for the most part.
4. In some situations adaptable AI is highly desirable. In open ended games such as MMOs there is a need to extend the playtime. Providing adaptable AI can be an important contributing factor to extend the game AI.

3.8 AI for Modern Board Games

Pieter Spronck, Peter Cowling, Alex Champandard, Pier Luca Lanzi, and Ana Paiva

Modern board games are recently developed board or card games, which usually involve more than 2 players, and often contain non-deterministic elements and/or imperfect information. Examples of such games are Monopoly, Poker, Settlers of Catan, Dominion, Puerto Rico, Agricola, and Power Grid. In the last decade, tens of thousands of new board games have been published. Information on all those games, including rulesets, can be found on such websites as boardgamegeek.com and pagat.com.

Research into AI for classic, 2-player, deterministic, perfect information games such as Chess and Go in the last 50 years has lead to highly capable computer players which outrank human grandmasters in many games. The techniques developed for this research (in particular, tree-search techniques) have not been applied to AI for modern computer games, as they are considered to be unsuitable. Instead, AI for modern computer games is mostly based on heuristics and probabilistic reasoning.

However, modern board games, which contain elements from both classic board games and modern computer games, can function as a stepping stone to discover new applications for the classic techniques, so that they may find applications in modern computer games as well.

To investigate AI for modern board games, some typical games must be chosen as research subjects. The five requirements we identified for these game selections are:
1. The social aspect should be crucial in the gameplay, even if limited to tactical aspects such as temporary alliances.
2. The game should be sufficiently popular, and should have a relatively simple ruleset, so that it can be easily explained.
3. The game should support 3 or more players, as a major objection to the use of tree-search techniques is that they generally do not support more than two players.

4. The game should contain some imperfect information, so that no "perfect move" can be identified.
5. Non-determinism should have little influence on the outcome of the game, to limit the number of trials that must be run to get statistically significant results.

We have not yet identified a game that meets all these requirements. We have, however, organized a one-day workgroup in which multiple programmers developed an AI for the game "The Resistance". These AIs entered into a competition, in which various teams played the game over at least 10,000 rounds.

"The Resistance" is a game in which players get assigned a role: each player is either a Spy or a Resistance fighter. The roles are secret but the Spies get to know each other at the start of the game. One player is the leader, who has the responsibility to propose a team to go on a mission. The size of the team is determined by the rules. Everybody gets to vote on the team, and if the majority agrees with the leader's selection, the team gets to execute the mission. Regardless of the outcome of the vote, the leader role moves to the next player. Moreover, if the proposed team for a mission is rejected five times, the Spies automatically win the game. During the mission, the members of the team each play a card that indicates whether they support or sabotage the mission. These cards are played in secret, shuffled, and revealed. If one of the cards says that the mission is sabotaged, the mission fails. Otherwise it succeeds. As soon as three missions have succeeded, the Resistance fighters win. If, however, before that time the Spies manage to sabotage three missions, the Spies win.

During the programming workgroup, diverse ideas for the implementation of AIs were investigated. While it is difficult to draw solid conclusions from the experiments, several things were noted:

1. It seems that the Spies have a much better chance at winning than the Resistance fighters. The best Spy AIs won the game almost 50% of the time, while the best Resistance fighter AIs won the game less than 20% of the time.
2. The proportion of victories of Spies compared to Resistance Fighters highly depends on the types of AIs included in the competition. Especially the inclusion of a randomly playing strategy had a high impact on the reported results.
3. Simple strategies tended to work better than more complex strategies. This may no longer hold when new strategies are built.

We now intend to expand the game engine with extra communication possibilities, and organize the competition on a larger scale.

3.9 Player Modeling

Pieter Spronck, Georgios Yannakakis, Christian Bauckhage, Elisabeth André, Daniele Loiacono, and Günter Rudolph

Player modeling concerns the capturing of characteristic features of a game player in a model. Such features may encompass player actions, behaviors, preferences, goals, style, personality, attitudes, and motivations. Player models can be used to let the game adapt automatically to be better able to achieve its goals with respect to the player. Games may adapt to a player without constructing a player model by simply responding to changes to the game world or to biometric data of the player; however, creating a player model as an intermediate step has at least two advantages: (1) it creates an understanding of who the player is, and therefore an argument for making specific adaptations; and (2) a player model allows generalization of adaptations to other games.

Player models can be constructed upon three data domains: (1) gameplay data, which are directly extracted from the game and player interactions with the game world; (2) subjective data, which are collected by means of a questionnaire (e.g. psychometrics, demographics, emotional states, personality tests); and (3) objective data which can be extracted from e.g. biometrical observations.

Two key approaches to developing player models can be distinguished. The first, theory-driven, approach is derived from experimental psychology and the social sciences, which consists of proposing a model based on literature and domain experience, and then validating that model empirically. The second, data-driven, approach is derived from computer science and the natural sciences, which consists of collecting a large dataset of measurements, and then using computational methods to automatically (or semi-automatically) derive models. Comparing the two approaches, we note that the first inherently contains argumentation and understanding for the choices of the model, which the second lacks. However, practice shows that such models often fail to encompass relevant features because of a lack of insight of the model builders. The second approach has the advantage of automatically detecting relevant features; however, it is also prone to detecting meaningless relationships between user attributes, game context and user experience.

In computer games an extensive set of features of player behavior can be extracted and measured. At the same time there is, usually, lack of insight in what these features actually mean, at least at present. Therefore, in the current state of research, the second approach seems most suitable. A typical technique employed for creating player models using the second approach is data clustering. This technique cannot be applied well without sufficient insight into the structure of the data space, however; depending on this structure, different clustering algorithms may give widely different results. Domain-specific knowledge, feature extraction and feature selection are necessary to achieve results that make sense.

Experience has shown that diligent application of clustering techniques may provide insight into group behaviors. However, it remains hard to make predictions about individuals. As such, player models can usually only give broad and fuzzy indications on how a game should adapt to cater to a specific player. One possible solution is to define several possible player models and classify an individual player as one of them. Then, when gameplay is going on, the model can be changed in small steps to fit the player better. I.e., the player model is not determined as a static representation of the player, used to determine how the game should be adapted; rather it is a dynamic representation of a group of players, that changes to highlight the general characteristics of a specific player, and drives the game adaptation dynamically.

Note that player characteristics within a game environment may very well differ from the characteristics of the player when dealing with reality. Thus, validated personality models such as psychology's Five Factor Model might not fit well to game behavior. An interesting research direction in player modeling research is to determine a fundamental personality model for game behavior; such a model will have some correspondence with the Five Factor Model, but will also encompass different characteristics. Moreover, the behavioral clues that can be found in game behavior will be considerable different from those that can be found in reality.

Regardless of the line of research in player modeling chosen, the biggest obstacle right now is a lack of data. What is required is a rich multimodal corpus of gameplay and player data as well as player descriptions. Such a corpus must include detailed gameplay data for several games for a large number of players, including actions, events, locations, timestamps as well as biometrical data that are trivial to obtain in large scales (e.g. camera images and eye-tracking). Demographic data for the players must be available, as well as player information in the form of several questionnaires and structured interview data. Not all this data needs to be available for every subject in the database; several large datasets of gameplay data already exist, and it would be beneficial to include those in the database too.

3.10 Evaluating AI in Games Research

Kenneth O. Stanley, Michal Bida, Paolo Burelli, Risto Miikkulainen, and Georgios N. Yannakakis

An important task for any researcher in artificial intelligence (AI) is to formulate the right experimental setup to evaluate research results. However, this task is particularly challenging when AI is combined with video games because the goal of the research is often unstated and the results may be in part subjective. Nevertheless, the general issue of the proper course for evaluation in this area is rarely discussed, which has sometimes led to confusion when authors and reviewers hold differing assumptions on how evaluation should be conducted. The goal of our group session was to begin to address this issue by surveying available types of evaluation and offering some recommendations for future authors.

We identified two key categories of goals for research in AI and games. The first is "games as AI arenas." In this type of research, the game acts mainly as a benchmark for testing the performance of AI algorithms. The second category is "AI for better games," wherein the goal is to improve the game experience itself through innovative AI technology. An important conclusion of our session was that it is critical that authors should identify into which category their research falls because the proper form of evaluation will depend necessarily upon this distinction.

Participants also explored several other dimensions of research evaluation in this area, all of which ultimately tie into the question of the primary goal of the research. Among the other issues discussed were different types of evaluation (i.e. researchers studies versus evaluation by a reviewer), the expectations of different audiences (i.e. academia, industry, or players), methods of evaluation (e.g. objective versus subjective and quantitative versus qualitative), and evaluation metrics (e.g. human subjects studies, competitions, expert evaluations, benchmarks, statistical analysis, etc.). A general challenge for many studies is to obtain sufficient player data to draw conclusions.

Our main conclusion is a recommendation for authors in this area, particularly those submitting to the IEEE Transactions on Computational Intelligence and AI in Games (TCIAIG) journal: It would help in the evaluation of research by reviewers if authors would specify (1) their research goal (one of the two primary goals) and (2) how their evaluation matches their goal. Our recommendation is to ask authors to make clear their answers to both these questions in any paper that they submit in this area.

3.11 The AIGameResearch.org AI Game Clearinghouse

Kenneth O. Stanley, Alex J. Champandard, Clare B. Congdon, Philip F. Hingston, Graham Kendall, Pier Luca Lanzi, Daniele Loiacono, and Risto Miikkulainen

Many researchers in the AI-in-Games community build their own video games to demonstrate or test new technologies. These games often exhibit innovative features that would appeal to other researchers, the game industry, and even the general public. It is also often the case that the more players an experimental game attracts, the more informative the experimental analysis can be. Yet at present there is no simple or common solution for attracting a sufficient audience to an experimental game. The aim of our session was to begin to organize an "AI-Game Clearinghouse" that will allow the community to pool its collective products to attract a significant audience to the work of the community in general. The result will be a new website, AIGameResearch.org, where anyone on the internet can find a wide range of innovative games based on cutting-edge AI technologies.

Creating such a website can yield several other benefits for the research community. Not only can it attract more players to our games and thereby enhance our research and publicity, but it can also help to demonstrate the significance of our work to the game industry and to funding agencies concerned with relevance and impact. Furthermore, in addition to games, the website can provide a pathway to active experiments, source code, announcements, publications relating to the game collection, and discussion forums for users. Thus it can serve as a clearinghouse for research in this area. Another interesting facet of such a site is that because the public is genuinely interested in playing innovative games, it is possible that the site could attract revenue, which might someday help to support research in AI and games. Research groups might also someday sell their games through the site.

A site offering games to the general public will require moderation to ensure an acceptable level of quality. Otherwise, the public will not ultimately trust the site, jeopardizing its mission. It is likely that many individuals and groups (including non-researchers) will eventually want to submit their games to the site once it becomes widely known. Therefore, we propose to establish an "editor in chief" who makes final decisions on whether to include individual games that are submitted based on the recommendation of reviewers from a permanent "editorial board." Another aim to maximize the site's impact is to establish ties to IEEE TCIAIG, AIGameDev.com, GPEM, and other relevant venues and resources. However, the site will remain a community effort independent of any specific organization.

Participants resolved to continue this effort by establishing the site and editorial board after the conclusion of the Dagstuhl workgroup.

3.12 Video Game Description Language

Tommy Thompson, Simon Lucas, Tom Schaul, John Levine, Marc Ebner, and Julian Togelius

As participants in this Dagstuhl session address the challenge of General Video Game Playing (GVGP), we have recognised the need to create a Video Game Description Language (VGDL).

Unlike General Game Playing, we have envisioned GVGP will not require a prescribed language to facilitate understanding of the logic of the game: requiring the computational agents to ascertain these facts for itself. However, we would still require means to define the wide range of problems the GVGP agents may face for the purposes of classification and categorization. Not only would such a language provide means to encapsulate the features and mechanics of a game for the purposes of human understanding, but also provide context for the evaluation of GVGP agents having completed playing.

Outside of the issues of classification, there is also the opportunity for automatic game generation. Given the intent of the GVGP group to work within the Physical Traveling Salesman Problem (PTSP) framework, we aim to attach a code-base to the VGDL compiler that derives implementations of these games from the definition that can be used in conjunction with GVGP. Implementing such a compiler could provide numerous opportunities; users could modify existing games very quickly, or have a library of existing implementations defined within the language (e.g. an Asteroids ship or a Mario avatar) that have pre-existing, parameterized behaviors that can be customized for the users specific purposes. Provided the language is fit for purpose, automatic game creation could be explored further through experimentation with machine learning algorithms, furthering research in game creation and design.

In order for both of these perceived functions to be realized and to ensure it is suitable for a large user base we recognize that the language carries several key requirements. Not only must it be human-readable, but retain the capability to be both expressive and extensible whilst equally simple as it is general. In our preliminary discussions, we sought to define the key requirements and challenges in constructing a new VGDL that will become part of the GVGP process. From this we have proposed an initial design to the semantics of the language and the components required to define a given game. Furthermore, we applied this approach to represent classic games such as Space Invaders, Lunar Lander and Frogger in an attempt to identify potential problems that may come to light.

In summary, our group has agreed on a series of preliminary language components and are now keen to experiment with forms of implementation for both the language and the attached framework. In future we aim to realize the potential of the VGDL for the purposes of Procedural Content Generation, Automatic Game Design and Transfer Learning and how the roadmap for GVGP can provide opportunities for these areas.

3.13 Procedural Content Generation

Julian Togelius, Alex J. Champandard, Pier Luca Lanzi, Michael Mateas, Ana Paiva, Mike Preuss and Kenneth O. Stanley

The work group on procedural content generation (PCG) focused on the future potential of this young subfield of game AI research. Three ambitious goals for PCG research were stated, and eight medium-term research challenges were identified. The solution to these challenges would constitute good progress towards ultimately solving the grand research goals. Further, six actionable steps were identified; these are concrete research projects which one could start work on immediately and which would contribute to solving some of the research challenges. The three goals stated were:

1. high-quality multi-level, multi-content PCG
2. PCG-based game design and
3. generating complete games from scratch.

The eight research challenges were:
1. non-generic, original content
2. representing designer style
3. general content generators
4. search space construction
5. usable and powerful interfaces for PCG systems
6. overcoming the animation bottleneck
7. interaction and opportunistic control flow and
8. establishing a comprehensive theory and taxonomy of PCG systems.

The six actionable steps were:
1. generating complete Atari 2600 games
2. procedural animation for simple generated creatures (e.g. a fish)
3. co-generating quality quests and maps
4. generating music modulated by game events
5. competent generation of Super Mario Bros levels including macro-scale structure and progression and
6. player-directed generation with model-based selection.

For each of the goals, challenges and actionable steps, the work group identified the state of the art in terms of published papers and games.

3.14 Computational Narrative

R. Michael Young, Ruth S. Aylett, Paolo Burelli, Mirjam P. Eladhari, Richard Evans, and Ana Paiva

The use of automatic generation methods to create narrative elements in games has the potential to create genres of game play that have only been hinted at to date in commercial titles. In pursuit of this capability, a growing number of research efforts are targeting critical representational and algorithmic problems in the area of computational narrative. Current challenges for scientists working in this area range across many problem areas. Some of the challenges we consider both significant and near at hand include:
- the creation of dynamic characters (inducing their dialog and internal personalities)
- the generation of novel quests and story sequences
- the adaptation of a game's story in response to player activity and
- the development of sharable systems for realization. Specifically, the construction of research systems in computational narrative would be accelerated by access to game environments/toolkits that can be used to address multiple research problems, that can be accessed by distinct research sub-systems and that and can be shared across many research groups.

Near-term research opportunities for this community include:
- systems that build stories drawn from MMO game play logs,

- systems that generate tailored story-based support for face-to-face role playing used in corporate training and simulation and
- collaborative support for machinima generation used as pre-visualization for game developers, cinematographers and video/film producers.

We also see the internet as a significant resource to address a number of research problems, including aspects of narrative content creation, crowd-sourced data collection for evaluation and other aspects.

- Longer-term research goals for this community include:
- Systems that, like Star Trek's holodeck,create entire story worlds and dynamically drive them through a user's interactive experience,
- games that blend alternate reality game mechanics, augmented reality capabilities and automatically generated narrative elements,
- Long-lived, drop-in/drop-out narrative-based games that last for months or years and
- Systems that automatically drive hybrid human-robot systems, where human players interact with robots as NPCs

4 Overview of Short Talks

During the seminar about 15 short talks were given. The contents of most of these are incorporated in the workgroup reports (Section 3). The contents of the remaining ones are given below.

4.1 Using Computer Games to Close the Loop in Artificial Visual Information Processing

Marc Ebner (Universität Greifswald, DE)

The human brain is highly complex. Even though significant advancements have been made in recent years, it is still unknown how the brain works as a whole. We are currently unable to simulate an artificial brain which is able to mimic human performance in its entirety. This is in part due to a lack of understanding how various parts interact. In order to create fully autonomous individuals which can interact with human players in computer games or virtual realities we need to be able to simulate a human individual in a way such that it is not apparent to the player whether this individual is artificial or not. Computer games, viewed through screen captures, are an ideal tool to perform research on closing the loop from visual input to the action of a virtual player.

4.2 Social Simulation Games

Mirjam P. Eladhari, Richard Evans, and Michael Mateas

The purpose of this talk was to introduce the topic of social simulation games for discussion at the Dagstuhl Seminar 12191, "Artificial and Computational Intelligence in Games". We wanted to identify why we would want to simulate social interactions in games, and to open the seminar-discussion for identifying the main research challenges in the field.

Well-known examples of social simulation games include Little Computer People[2], The Sims[3], PsychSim [7], Façade [5], and recently, Prom Week [6]. The type of play in social simulation games can be compared to children's play with dollhouses, as with chamber plays and improvisational theater. While the former help children's learning about practices in social behavior, the latter may help us, in later stages of life, to better understand the human condition. This is also the case for well-crafted social simulation games.

Generally, the term 'social' refers to the interaction of organisms with other organisms and to their collective co-existence. In terms of simulating these interactions we always, when building social simulations, use different models of mind for the system design. In the area of social simulation games, these can be divided into three types. One type of model can be an underlying theory of social behaviors, such as Wittgenstein's notion of social practices [3, 8, 9] that is used in The Sims 3[4], or the use of Goffman's [4] theories in the design of Prom Week. A second type of model for design is that of game genre. Prom Week have similarities with puzzle games, and The Sims with dollhouse play. In the research prototype Pataphysic Institute [2] the conventions of combat in the genre massively multiplayer online role playing games are used as a model for interaction in certain play modes. Yet a third type of model of mind affecting the design of social simulations may derive from the use of an existing or for the purpose invented implementation technique. For example, much of the game design in the Pataphysic Institute revolves around the fact that spreading activation networks are used in the implementation of both autonomous and semi-autonomous entities, while Prom Week and The Sims 3 both use forward chaining rules. Social simulation games are created for various reasons. It can be for their own sake, taking a l'art pour l'art perspective as in the artistic tradition, or for exploring the human condition as in the humanist tradition. It can be for pure entertainment, and the selling of the same, or it can be for a practical purpose such as for teaching about conflict resolution or cross-cultural understanding. No matter the purposes of individual projects, the creation of social simulation games is likely to lead to new advances within the areas of social believable agents as well as in complex systems.

Challenges for future work, identified at the seminar, include:
- defining believability [1];
- negotiating boundaries of the research areas addressing social simulation and believable agents respectively in order to allow technology transfer;
- a single implementation approach that work in both single and multiplayer, for both player- and non player characters;
- further development for use of social practices by:
  - defining an operational language for specifying social norms and practices;

[2] David Crane and Rich Gold, Activision, 1985
[3] Will Wright, Electronice Arts, 2000
[4] Electronic Arts, 2009

  - modelling examples of full breath of social practices;
  - creation of agents who learn new social practices by adapting to an inhabited world or environment;
  - solve how to allow agents to participate in simultaneous, concurrent social and hierarchical practices;
- addressing authoring by:
  - creating authoring tools for balancing casts of characters;
  - creating in-character action sequences, or drama management, producing coherent story units;
  - integrating multiple capabilities for coherent agent-performance;
- increasing expression and believability of agents/characters by:
  - facilitation of expression of character by actions and adverbial modifiers;
  - widening "the expression bottle-neck", currently production of audiovisual content is costly even if procedural;
  - increased usage of non naturalistic forms of expression;
  - create believable agents that evolve and change, i.e. with increased persistence, and;
  - create characters who behave as if they have a history.

References

1 J. Bates, The Role of Emotions in Believable Agents, Technical Report CMU-CS-94-136, Carnegie Mellon University, 1994.

2 M. Eladhari, “The Pataphysic Institute”, in Proceedings of the Sixth AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment (AIIDE-10), The AAAI Press, Menlo Park, CA, 2010.

3 H. Garfinkel, Studies in Ethnomethodology (Social & political theory), Polity, 1991.

4 E. Goffman, The Presentation of Self in Everyday Life, Anchor, 1959.

5 M. Mateas and A. Stern, “Procedural Authorship: A Case-Study of the Interactive Drama Façade”, in Digital Arts and Culture: Digital Experience: Design, Aesthetics, Practice (DAC 2005), 2005.

6 J. McCoy, M. Treanor, B. Samuel, B. Tearse, M. Mateas, and N. Wardrip-Fruin, “Authoring Game-based Interactive Narrative using Social Games and Comme il Faut”, in Proceedings of the 4th International Conference & Festival of the Electronic Literature Organization: Archive & Innovate (ELO 2010), 2010.

7 D. Pynadath, V. David, and S. Marsella, “PsychSim: Modeling Theory of Mind with Decision-Theoretic Agents”, in Proceedings of the IJCAI05 Conference, 2005, pp. 1181–1186.

8 L. WittgenStein, Philosophische Untersuchungen / Philosophical Investigations, Basil Blackwell, New York, 1953.

9 L. Wittgenstein, “Preliminary studies for the Philosophical Investigations”, in The Blue and Brown Books, Harper & Row, New York, 1958, pp. 1961–1961.

4.3 Believable Agents for Games

Philip F. Hingston (Edith Cowan University, AU)

Humans are interacting more and more with robots, bots, and other agents. Our thesis is that if these agents are more believable/human-like, our interactions are likely to be more successful. For example, in the computer game context, human-like bots or NPC's (non-player characters) are often more fun to play against (see, for example, [1]).

We ask the reader to imagine themselves playing a game against a robot (e.g. tennis) or an NPC (e.g. a fast-paced action game). Suppose you know (or suspect) that your opponent is not human — would this make the experience more or less fun, or would it make no difference to you? Aside from the initial novelty value, there is quite a bit of evidence to suggest that non-humanlike bots may to be less fun to play because: they are too hard to beat (too fast, too accurate etc); or too easy to beat (too stupid, non-adaptive etc); or perhaps because there is no shadenfreud – if you know your opponent is a bot, then there is no joy in causing it pain (for example, Weibel et al. showed that players prefer to play opponents that they believe to be human, even if they are not in fact human [2]).

Suppose then we that we want to make more believable/human-like bots, or more generally, agents. How can we go about it? There are at least three different kinds of approaches: ad-hoc (these usually use some kind of hand-crafted rules, perhaps with randomness added); learning-based (these use various methods to learn competence, or to learn by imitation to be human-like, or to adapt to the opponent/environment); cognitive/psycho-social models (these are the most ambitious, attempting to model human behavior). The last of these is the most recent innovation, and it will be interesting to see how successful it will be going forward.

Whatever means we use to create them, how can we tell if the agents we create are believable? One answer is to design a suitable Turing test. For example, I organize an annual NPC Turing test competition based around the commercial game Unreal Tournament 2004 (a FPS or first-person shooter). In this competition, competitors create and enter AI-based NPC's, and human judges try to decide which of their opponents is a human and which is a bot. To date, the competition has been run in 2008, 2009, 2010 and 2011, and although the NPC's are improving, judges still reliably rate human opponents as more human than NPC opponents (see [3] for a description and analysis of the results up to 2009). Competitors have used all of the approaches listed above, and combinations of them. At present, there is theorizing but no clear understanding of how the judges are able to make this distinction, even in the very limited context of a FPS, where the available actions mainly consist of frenetically running and jumping, shooting at the opponent with various deadly weapons (only in the virtual game world, of course).

The purpose of this talk was to engage the listener to start to consider believability, or the related concept of human-like-ness in intelligent agents. Some seminar participants were keen to disagree with the views put forward in this talk, and that is a good starting point for interesting discussion, argument and questioning!

To conclude, I'd like to put a couple of more philosophical questions:

- should agents merely appear human-like, or should they BE human- like?
- would it be better in some cases to make them unmistakably NOT human?

References

1 B. Soni and P. Hingston, "Bots Trained to Play Like a Human are More Fun", in Proc. of IEEE International Joint Conference on Neural Networks, IJCNN 2008, Hong Kong, June 2008.

2 D. Weibel, Bartholomäus. Wissmath, S. Habegger, Y. Steiner and R. Groner, "Playing online games against computer- vs. human-controlled opponents: Effects on presence, flow, and enjoyment", Computers in Human Behavior, 24(5), pp. 2274–2291, 2008.

3 P. Hingston, "A Turing Test for Computer Game Bots", IEEE Transactions on Computational Intelligence and AI In Games, Vol. 1, No. 3, September 2009.

4.4 Adaptive Artificial Intelligence in Games

Pieter Spronck (Tilburg University, NL)

Joint work of Spronck, Pieter; Ponsen, Marc; Sprinkhuizen-Kuyper, Ida; Postma, Eric
Main reference P. Spronck, M. Ponsen, I. Sprinkhuizen-Kuyper, E. Postma. "Adaptive game AI with dynamic scripting," Mach. Learn. 63, 3 (June 2006), 217–248, 2006.
URL http://dx.doi.org/10.1007/s10994-006-6205-6

Adaptive Game AI concerns artificial intelligence (AI) in computer games which adapts to dynamic circumstances. In particular, the AI adapts in response to behavior of human players. The three main goals of adaptive game AI are (1) self-correction, the ability to recover from mistakes; (2) creativity, the ability to invent new tactics; and (3) scalability, the ability to exhibit behaviors appropriate for the human player's skill level.

Commercial game developers have included adaptive game AI in only a very small number of games. Some of the reasons for this lack of interest are the high complexity of modern games, the perceived lack of efficiency of adaptive techniques, and fear of AI learning the wrong lessons.

For game developers to accept adaptive techniques in their games, it is essential that these techniques meet four computational and four functional requirements. The computational requirements are: (1) Speed: the AI must be fast as there is little processing power available for adaptation; (2) Effectiveness: the AI cannot tolerate (much) ineffective behavior, even during learning; (3) Robustness: the AI must be able to deal with the inherent non-determinism that exists in most modern games; and (4) Efficiency: the AI must use as many learning opportunities as possible, to finish learning long before the game is over. The functional requirements are: (1) Clarity: game developers wish to understand what adaptive game AI is actually doing; (2) Variety: the AI should not always exhibit the same behavior; (3) Consistency: the AI should finish learning in a predictable period of time; and (4) Scalability: the AI should take the human player's skills into account.

While traditional adaptation techniques seldom meet all these requirements (e.g., most of them are breaking either the effectiveness or the efficiency requirement), several techniques for adaptive game AI exist which are suitable for commercial modern games – in particular, techniques based on optimization (e.g., hill-climbing), imitation (e.g., case-based reasoning), and reinforcement (e.g., dynamic scripting).

With the ever increasing complexity and realism of virtual game world, the player's freedom to express behavior in games increases as well. The consequence is that the AI has to take into account and interpret an increasing variety of player behaviors. This means that AI that worked well last year, is no longer sufficient for newly released games. We can actually observe a decline in effectiveness of game AI that is developed with classic methods. Adding adaptation to game AI will allow it to become more effective automatically. Therefore, it is not a question if, but when game developers will give their AIs adaptation capabilities by default.

Participants

- Elisabeth André
Universität Augsburg, DE
- Ruth Aylett
Heriot-Watt University Edinburgh, GB
- Christian Bauckhage
Fraunhofer IAIS – St. Augustin, DE
- Michal Bida
Charles University – Prague, CZ
- Adi Botea
IBM Ireland – Dublin, IE
- Bruno Bouzy
Paris Descartes Unversité, FR
- Paolo Burelli
IT Univ. of Copenhagen, DK
- Michael Buro
University of Alberta, CA
- Martin V. Butz
Universität Tübingen, DE
- Alex J. Champandard
AiGameDev.com KG, AT
- Clare Bates Congdon
University of Southern Maine, US
- Peter Cowling
University of Bradford, GB
- Marc Ebner
Universität Greifswald, DE
- Mirjam P. Eladhari
University of Malta, MT
- Richard Evans
Little Text People at Linden Lab, GB
- Philip F. Hingston
Edith Cowan University, AU
- Graham Kendall
University of Nottingham, GB
- Pier Luca Lanzi
Politecnico di Milano, IT
- John M. Levine
University of Strathclyde, GB
- Daniele Loiacono
Politecnico di Milano, IT
- Simon M. Lucas
University of Essex, GB
- Michael Mateas
University of California – Santa Cruz, US
- Risto Miikkulainen
University of Texas – Austin, US
- Hector Munoz-Avila
Lehigh Univ. – Bethlehem, US
- Dana S. Nau
University of Maryland – College Park, US
- Ana Paiva
IST – TU of Lisbon & INESC-ID
- Mike Preuss
TU Dortmund, DE
- Günter Rudolph
TU Dortmund, DE
- Tom Schaul
New York University, US
- Moshe Sipper
Ben Gurion University – Beer Sheva, IL
- Pieter Spronck
Tilburg University, NL
- Kenneth O. Stanley
University of Central Florida – Orlando, US
- Tommy Thompson
The University of Derby, GB
- Julian Togelius
IT Univ. of Copenhagen, DK
- Georgios N. Yannakakis
IT Univ. of Copenhagen, DK
- R. Michael Young
North Carolina State Univ., US