A Survey of Adaptive Game AI: Considerations for Cloud Deployment

Gabriel Iuhasz, Victor Ion Munteanu, and Viorel Negru

Abstract. Modern video games have become an important part of AI research in the past years, largely thanks to the characteristics of their environment and the challenges they pose to AI researchers. This paper is a a survey of the current game AI state of the art and highlights important achievements in this field. An adaptive multi-agent system that can be deployed on a cloud infrastructure to solve computational constraints of advanced machine learning methods is also presented.

Keywords: Artificial Intelligence,Video Games, Machine Learning, Multi-Agent Systems, Cloud Computing.

1 Introduction

In recent years, researchers have become ever more active in using games as a test bed for advancing artificial intelligence (AI) techniques, especially due to the fact that games provide a controlled environment while still maintaining properties similar to real-life.

Games are populated with entities called non-player characters (NPCs) with which the player can interact and which can be considered agents. In most games there is a heterogeneous distribution of agent types, some having only reactive agents while others needing proactive ones as part of there AI.

As games usually encompass one or more NPCs, we can infer that games are multi-agent systems (MASs) which are structured hierarchically based on the tasks each agent has to solve. While commercial games have a tendency to stick with simple static methods for their AI systems, academic AI systems do not and mostly

Gabriel Iuhasz · Victor Ion Munteanu · Viorel Negru
West University of Timişoara, Timişoara, Romania
e-mail: {iuhasz.gabriel,vmunteanu,vnegru}@info.uvt.ro

Victor Ion Munteanu · Viorel Negru
Institute e-Austria Timişoara, Timişoara, Romania

F. Zavoral et al. (eds.), *Intelligent Distributed Computing VII*,
Studies in Computational Intelligence 511,
DOI: 10.1007/978-3-319-01571-2_36,

utilize adaptive machine learning (ML) techniques. There are two types of ML techniques in game AI: offline and online.

This research is important as lessons learned from developing MAS for game AI can be transferred to real life situations that have similar task distribution and environmental constraints. This paper presents a survey on adaptive agent and multi-agent based game AI solutions. The focus is on strategy games and massive multiplayer online (MMO) as these genres provide a greater challenge and encompass the majority of AI challenges present in other titles. Furthermore, a novel multi-agent solution for deploying ML techniques in games on a cloud infrastructure is presented.

2 Major Trends in Game AI

Generally speaking most modern video games have extremely complex environments for NPCs to interact in [1]. These environments posses incomplete information, each NPC knows only parts of the game state. The level of interaction each NPC has with the environment is also an important characteristic as some modern games each player has unlimited opportunity to interact with the environment, concurrent moves being permitted.

In strategy games there is also a level of asymmetry as one player may have some form of advantage, be it in resources or unit numbers, over the opposing players. Games have more in common to real life scenarios such as military command or disaster management than traditional board games. It should be noted that game environments are also stochastic.

Because of these characteristics, researchers have highlighted the potential to develop "human-level" AI [2]. In his paper Michael Buro [3] identifies some of the major AI challenges present in games, giving good insight into the complexity of the modern game domain. These include resource management, decision making under uncertainty, spatial and temporal reasoning, collaboration, opponent modeling, learning and adversarial real-time planning. Some of these challenges have been actively pursued by the AI community while others have been sadly neglected. The field of game AI collaboration has received little attention.

One important distinction between the academic definition of AI and the gaming industry definition of AI is the fact that game AI has to be entertaining. This means that most NPC behaviour present in games is not based on academic AI techniques but rather on rule of thumb. Sophisticated ML techniques are rarely used. The gaming industry prefers static techniques such as Finite State Machines (FSM), simple Decision Trees, scripting and Rule Based systems.

Hard-coded techniques have one major drawback: they become predictable after a relatively short amount of time and are hard to maintain and debug when they are used to model more complex behaviours. For example, FSMs transitions and the number of states (s) grow exponentially with the number of events (e): $s = 2^e$, consequently increasing the number of transitions (a) even faster: $a = s^2$.

Rule-base Systems are brittle when faced with a problem that is out of bounds with their knowledge base. They are unable to rely on past experience to select a similar rule or update their rule base. Expert knowledge is used in their creation which, once in place, can not be modified and requires substantial effort to maintain and debug. The A* pathfinding algorithm and FSM for decision making make up more than 2/3 of current game AI systems [4].

It is common for AI systems and gameplay elements to be designed and implemented in parallel. This means that if a gameplay mechanic is changed in the latter stages of development it has a catastrophic effect on the AI systems. This situation can be remedied by adaptive and ML techniques which are far more flexible and would require little to no modification. An ML framework could be used in several game titles thus lowering development time and cost.

2.1 Game Genres

Strategy games such as Real-Time Strategy games (RTS) are one of the more challenging genres when it comes to their AI systems. One of the main factors when designing the AI for RTS games is the extremely large state space and environment which are: partially observable, deterministic, sequential, dynamic and continuous. It is also important to note that these types of games posses a massive state space approximated to 10^{11500} in the case of RTS [5]. Also, the number of actions available to the player is also superior in the case of RTS as it has approximately 1 million possible actions while chess has 30. Many sofisticated methodologies an techniques have been used to create AI systems for RTS games. These include: Cognitive architecture [6], Goal-driven autonomy [7], Reactive Planning [8] and Case-based reasoning [9].

FPS (First person shooter) are at the cutting edge when it comes to graphics technologies. It has also benefited from some extremely interesting AI solutions like the AI Director in the game Left 4 Dead[1]. Its role is to dynamically adjust gaming difficulty and story pacing based on current player stress levels. Other commercially available titles such as FEAR and Killzone have adopted interesting adaptation mechanisms to their game AI. FEAR uses a STRIPS-planner while Killzone 2 uses HTN (Hierarchical Task Network) planner [10].

ML based systems for FPS are also present in the academic literature as the GDA system which was used Choi [11]. Genetic algorithms and neural networks were shown to manage the complexity of FPS environment [10]. There are still a lot of open research problems: dynamic terrain analysis, fast tactical path finding, efficient combat reasoning, opponent modeling and squad coordination. Some effort has been made in these directions, of particular note being the architecture developed by Hartley [12] which uses incremental dual-state representation and k-d tree-based techniques.

RPG (Role playing game) gameplay involves controlling one or more characters in order to solve quest-based challenges. In contrast to other genres, neutral and

[1] http://aigamedev.com/premium/article/procedural-director/

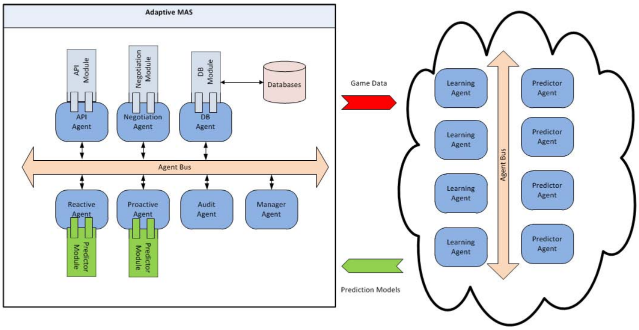

Fig. 1 Proposed Cloud deployment for Game AI MAS

even friendly NPCs are common. Due to the idiosyncrasies of RPG games, content generating techniques have been widely used, research being focused on procedural content generation. In [13] Grey J. and Bryson J. present an AI system capable of dynamically generating dialog with context generated from episodic memory and emotional bias towards past social interactions.

3 Cloud/Server Side AI

Massively multiplayer online (MMO) games feature support for a large number of players which simultaneously interact in a persistent environment. In order to play these games, a player has to log into a server which hosts the environment of the game and its the AI, meaning that on the client side little to no persistent information is stored.

Some MMOs have millions of subscribers and, because of this, the game world is split up into realms or shards, each of them representing a particular instance of a world. This is not the only solution as some MMOs have one cluster of servers that runs the entire game world. Such is the case with EVE online which can host up to 60000 player on a single game world instance.

When it comes to the AI methods used in MMO games these are not sophisticated. Utilizing hard-coded static methods such as those presented in Section 2. The work of Claybool M. [14] shows that the impact of network latency and the degree of lag has varying impact on the gameplay. Games that need only the first level of reasoning (micromanagement or fine unit control) are extremely susceptible to game lag. Other games such as RTS games where micromanagement is only a small part of the AI job are more resilient.

There is a wide variety of MMO sub-genres, the most prevalent one being RPGs (MMORPG). Research done in this area is focused on rule based systems [15] and on Bayesian modeling for human players [16]. Aranda G. proposes another approach that defines a MMOG layer where all the games logics and mechanics are implemented and must be solved at run-time [17].

3.1 Cloud Game AI Deployment Considerations

The major shortcoming of ML techniques when applied to game AI is that they require a substantial amount of computational resources which are limited on normal gaming hardware. Some data mining approaches for strategy prediction and army clustering have also been done [18]. They used game replays to extract expert human domain knowledge to create predictive models.

We propose a MAS framework for deploying adaptive game AI on a cloud platform. More and more single player games require perpetual internet connection in order to be played. This is a security measure that is part of digital rights management (DRM) system. This represents an opportunity to use existing DRM technologies to benefit gameplay by deploying adaptive agents on a cloud computing infrastructure.

Our system has 9 agent types: API agents that handle interface with game instances, reactive and proactive agents which execute tasks depending on the game genre, audit and manager agents that handle MAS deployment and stability, negotiation agents that handle all inter-MAS negotiations and database agents that handle all database queries, databases varying from shared knowledge ones to game event logging ones.

The most important part of our proposed framework is the two agent types that are the learner agent and the predictor agent deployed in the cloud. The learner agent receives data in the form of past game states and produces a viable predictive model. Once the learning agent finishes, it exports the model to the predictor agent which then proceeds to validate the predictive model. Once this is done the model can be incorporated into the AI subsystem. It is important to note that most ML methods can be encapsulated into learner and predictor agents. By scaling these agents we can address the high computational cost of such methods while still maintaining an adequate QoS.

4 Conclusions

This paper presents a survey on advances and current research in ML and MAS solutions for computer games. This survey focused mainly on RTS and MMO game genres as the authors feel that these represent the biggest challenges when it comes to game AI.

Open research questions have also been highlighted such as inter-agent negotiation and the lack of a game domain ontology to aid with interoperability. A cloud-based MAS solution is proposed in order to alleviate some of the computational constraints of ML techniques applied to game AI systems.

In a field where agent modeling is more mature, standard ontologies have been developed in order to formalize the relevant domain knowledge. There has been little to no work for creating a viable game domain ontology which would be extremely useful for MAS interoperability and reusability.

Acknowledgements. This work was partially supported by the Romanian national grant PN-II-ID-PCE-2011-3-0260 (AMICAS), FP7-REGPOT-CT-2011-284595 (HOST) and by the strategic grant POSDRU/CPP107/ DMI1.5/S/78421, Project ID 78421 (2010), co-financed by the European Social Fund - Investing in People, within the Sectoral Operational Programme Human Resources Development 2007–2013. The views expressed in this paper do not necessarily reflect those of the corresponding projects' consortium members.

References

1. Lewis, J.M., Trinh, P., Kirsh, D.: A corpus analysis of strategy video game play in starcraft: Brood war. In: Proceedings of the 33rd Annual Conference of the Cognitive Science Society (2011)
2. Laird, J.E., van Lent, M.: Human-level ai's killer application: Interactive computer games. In: Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence. pp. 1171–1178, AAAI Press (2000)
3. Buro, M.: Real-time strategy gaines: a new ai research challenge. In: Proceedings of the 18th International Joint Conference on Artificial Intelligence, IJCAI 2003, pp. 1534–1535. Morgan Kaufmann Publishers Inc., San Francisco (2003)
4. Yildirim, S., Stene, S.B.: A survey on the need and use of ai in game agents. In: Proceedings of the 2008 Spring Simulation Multiconference, SpringSim 2008, pp. 124–131. Society for Computer Simulation International, San Diego (2008)
5. Aha, D.W., Molineaux, M., Ponsen, M.: Learning to win: Case-based plan selection in a real-time strategy game. In: Muñoz-Ávila, H., Ricci, F. (eds.) ICCBR 2005. LNCS (LNAI), vol. 3620, pp. 5–20. Springer, Heidelberg (2005)
6. Langley, P., Choi, D.: A unified cognitive architecture for physical agents. In: Proceedings of the 21st National Conference on Artificial Intelligence, AAAI 2006, vol. 2, pp. 1469–1474. AAAI Press (2006)
7. Muñoz-Avila, H., Jaidee, U., Aha, D.W., Carter, E.: Goal-driven autonomy with case-based reasoning. In: Bichindaritz, I., Montani, S. (eds.) ICCBR 2010. LNCS, vol. 6176, pp. 228–241. Springer, Heidelberg (2010)
8. Josyula, D.P.: A unified theory of acting and agency for a universal interfacing agent. PhD thesis, College Park, MD, USA, AAI3202442 (2005)
9. Szczepanski, T., Aamodt, A.: Case-based reasoning for improved micromanagement in real-time strategy games. In: Proceedings of the Workshop on Case-Based Reasoning for Computer Games, 8th International Conference on Case-Based Reasoning, ICCBR 2009, pp. 139–148 (July 2009)
10. McGee, K., Abraham, A.T.: Real-time team-mate ai in games: a definition, survey, & critique. In: Proceedings of the Fifth International Conference on the Foundations of Digital Games, FDG 2010, pp. 124–131. ACM, New York (2010)
11. Choi, D.: Reactive goal management in a cognitive architecture. Cogn. Syst. Res. 12(3-4), 293–308 (2011)
12. Hartley, T., Mehdi, Q.: Online action adaptation in interactive computer games, vol. 7, pp. 28:1–28:31. ACM, New York (2009)
13. Grey, J., Bryson, J.J.: Procedural quests: A focus for agent interaction in role-playing-games. In: Proceedings of the AISB 2011 Symposium: AI & Games (2011)
14. Claypool, M., Claypool, K.: Latency and player actions in online games. Commun. ACM 49(11), 40–45 (2006)

15. Ballinger, C.A., Turner, D.A., Concepcion, A.I.: Artificial intelligence design in a multiplayer online role playing game. In: Proceedings of the 2011 Eighth International Conference on Information Technology: New Generations, ITNG 2011, pp. 816–821. IEEE Computer Society, Washington, DC (2011)
16. Synnaeve, G., Bessière, P.: Bayesian modeling of a human mmorpg player. CoRR abs/1011.5480 (2010)
17. Aranda, G., Botti, V., Carrascosa, C.: Mmog based on mas: the mmog layer. In: Proceedings of the 8th International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2009, Richland, SC, pp. 1149–1150. International Foundation for Autonomous Agents and Multiagent Systems (2009)
18. Synnaeve, G., Bessière, P.: Special tactics: A bayesian approach to tactical decision-making. In: CIG, pp. 409–416 (2012)