iCO_2 – Promoting Eco-driving Practice through Multiuser Challenge Optimization (Demonstration)

Marconi Madruga
National Institute of Informatics
2-1-2 Hitotsubashi, Chiyoda-ku
Tokyo 101-8430, Japan
+810342122650
marconi@nii.ac.jp

Helmut Prendinger
National Institute of Informatics
2-1-2 Hitotsubashi, Chiyoda-ku
Tokyo 101-8430, Japan
+810342122650
helmut@nii.ac.jp

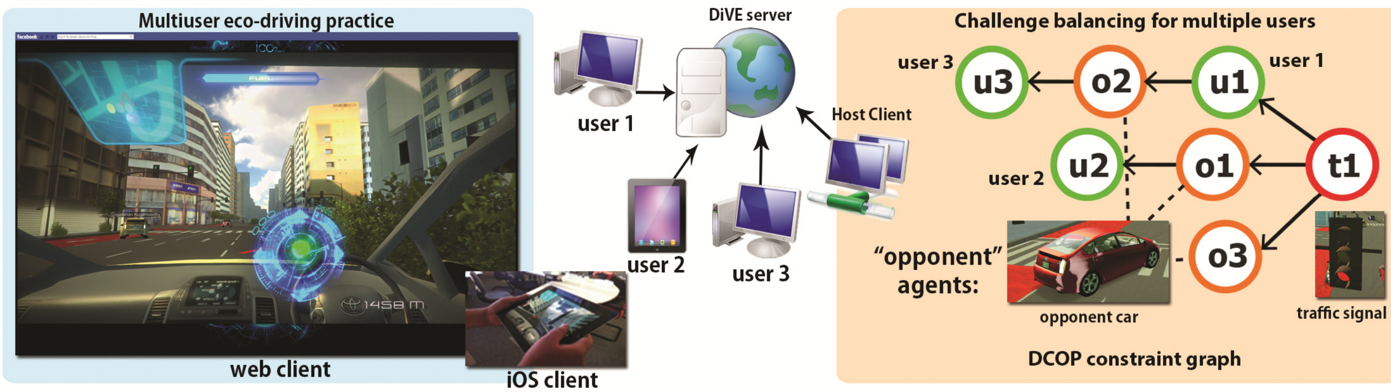

Figure 1. Overview of the iCO_2 application. Multiple users practice eco-driving in a shared 3D virtual environment that is also populated by computer-controlled agents acting as "opponents" to users' eco-driving. Distributed constraint optimization is used to optimize the difficulty level for *all* user drivers considering each user's skill level.

ABSTRACT
Eco-driving is a driving style that can significantly reduce fuel consumption and CO_2 emission. Current methods for eco-driving practice are inefficient or not easily accessible. Therefore, we introduce iCO_2, an online multi-user three-dimensional (3D) eco-driving training space, which was developed in Unity3D and made available as a Facebook application since September 2012. In iCO_2, agents are trained to act as "opponents" that create eco-challenges for users, i.e. situations that make eco-driving difficult. The (eco-)challenge is optimized for *all* users using distributed constraint optimization. iCO_2 is the first application to address the problem of multiuser real-time challenge balancing. Visitors of our demo will be able to join the simulation via Facebook (web client) or iPad (iOS client), and compete for the best eco-score in a shared 3D virtual environment depicting a part of Tokyo.

Categories and Subject Descriptors
I.2.11 [**Distributed Artificial Intelligence**]: Multi-agent systems; I.6.3 [**Simulation and Modeling**]: Applications

General Terms
Algorithms, Design, Human Factors

Appears in: *Proceedings of the 12th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2013)*, Ito, Jonker, Gini, and Shehory (eds.), May, 6–10, 2013, Saint Paul, Minnesota, USA.

Keywords
Distributed constraint optimization, coordination, real-time challenge balancing, social interaction

1. INTRODUCTION
Eco-driving is a driving style that offers many benefits, including the reduction of greenhouse gases, fuel cost savings, and greater safety and comfort for drivers. The positive effect of eco-driving is now clearly understood. So the training of the next generation of drivers becomes an important issue. We promote online eco-driving as a highly accessible, cost effective method to practice eco-friendly driving. However, current online eco-driving practice systems have several limitations: (1) single-driver experience; (2) static scenarios with simple circuits that are easy to memorize; (3) limited driving device (e.g. accelerator pedal).

Therefore, we have developed iCO_2, an eco-driving practice space that supports: (1) multi-user driving experience in shared simulation space; (2) dynamic and unpredictable scenarios through the inclusion of "opponent" agents that create increased levels of eco-challenges for the users; (3) several input options, including driving with game wheel, mouse, and iPad.

iCO_2 is a fully implemented "live" system. A simplified version was launched as a Facebook application in September, 2012.[1] For technical details on the multi-agent aspects of the system, please refer to our full paper in the AAMAS'13 proceedings.

[1] https://www.facebook.com/icotwo/

2. APPLICATION OVERVIEW

iCO_2 is an online tool for practicing eco-safe driving with multiple networked users. The scenario is a realistic 3D version of an area in Tokyo of about 1 km^2. Facebook users can control the direction and speed of the car operating a mouse-based interface (see Fig. 1, left side). The interface will change its color to indicate eco-friendliness in a continuous color range from green (eco-friendly) to red (eco-unfriendly). As a first approximation, eco-friendliness is defined by smooth acceleration and deceleration only. iCO_2 is designed as a computer game, where the user (or player) tries to drive as far as possible with a given amount of fuel. To receive more fuel, the user has to pass "checkpoints" which are spread around the city. The distance between checkpoints increases and the amount of fuel each checkpoint yields decreases. Colliding with other cars or running a red light will result in a fuel penalty.

iCO_2 was developed using the multiplatform game development software Unity3D (unity3d.com). Unity3D enables the client of the game to be ported to many platforms, such as standalone Windows/Mac, web (Facebook) and iOS (iPad). Regardless of the used client, all user drivers share the same virtual environment in the simulation. To enable multiuser functionality, we have developed our original massively multiuser networked 3D virtual environment technology, called DiVE (Distributed Virtual Environments). DiVE supports the management of user information, sharing data among clients, and data persistence. DiVE can currently accept up to 500 simultaneous users.

Another special type of client, the "host" client, hosts the opponents. From the perspective of the opponents, all users are remote entities. Hence, all user information that the opponents need in their decision-making is shared across the network. For instance, the system evaluates the challenge that a user is facing through a heuristic challenge function. These calculations are performed locally, at each client, and the result is synchronized across all clients. This separation allows us to build a lightweight client with high quality graphics on the user side, while the special host can assign all computational power on the execution of MAS, discussed next.

3. MULTI-AGENT TECHNIQUES

We apply real-time challenge balancing (RCB) [2] to avoid situations where the behavior of opponents (traffic signals and opponent cars) makes eco-driving for the user too easy or too difficult. In the offline learning phase (Q-learning), the opponents (traffic signal and car) learn an optimal policy to make eco-driving difficult for the user. The learning phase generates two Q-tables, one for the traffic signal and one for the opponent car. Note that there is no learning after the tables are generated.

During the simulation, each opponent car drives on a designated lane following a standard driving behavior model, while the traffic lights follow a standard fixed-time control policy. In this stage, whenever an opponent perceives a remote entity representing a user, it will decide what to do using the previously calculated Q-table. The light may choose between changing color or keeping its current state, whereas the car may choose to accelerate, brake or keep the current speed. To achieve challenge-balancing, an opponent can choose to decrease or increase the optimality of its chosen action to match the user's skill level. For instance, an opponent can choose to execute the 2^{nd} optimal action in the Q-table rather than the optimal one, in order not to frustrate the user with a too difficult challenge [1].

Distributed constraint optimization (DCOP) [3] is used to solve conflicts between the selected opponent actions in the presence of multiple users in the same area. The constraint graph representing the DCOP is a tree which matches the physical distribution of the entities in the virtual environment. The traffic light is the root node, and each branch represents the cars in a lane ahead of it. When a user changes lane, a graph reconstruction occurs and the calculation is reset.

To the best of our knowledge, our system is the first to apply a real-time challenge balancing mechanism to the case of multiple users in a shared space. A conflict between actions may happen, for instance, when a traffic signal attempts to create a high-difficulty eco-circumstance for one user driver who is in front of it (e.g. user u1 in Fig.1), which may accidentally and indirectly create a high-level challenge for another user driver following on the same lane behind the first user), who should instead experience a low-difficulty eco-circumstance (e.g. user u3 in Fig. 1). By solving the DCOP, we aim to find a set of actions that is executed by the opponents and optimal for all players involved. Details on the translation of our problem scenario to a DCOP are described in our AAMAS'13 full paper.

4. DEMONSTRATION

In the live demonstration of iCO_2, multiple visitors will be able to drive together and compete for the best eco-score using different types of clients (iPad, web). As a social incentive, a live ranking of the best eco-drivers will be recorded and shown in the demo session, and the best driver at AAMAS'13 will be announced later at our Facebook page.

A video of the iCO_2 application is available at: http://youtu.be/SVjgoypvKms

5. ACKNOWLEDGMENTS

This research was partly supported by a ''Grand Challenge'' Grant from National Institute of Informatics (NII), Tokyo and by the MEXT (Ministry of Education, Culture, Sports, Science, and Technology - Japan) scholarship applied for graduate studies at NII.

6. REFERENCES

[1] Andrade, G., Ramalho, G., Santana, H., and Corruble, V. 2005. Extending reinforcement learning to provide dynamic game balancing. In Proc. of the Workshop on Reasoning, Representation, and Learning in Computer Games, at 19th International Joint Conference on Artificial Intelligence (IJCAI'05), 7–12.

[2] Yannakakis, G. N., and Hallam, J. 2009. Real-time game adaptation for optimizing player satisfaction. IEEE Trans. on Computational Intelligence and AI in Games, Vol. 1, No. 2, 121-133.

[3] Kumar, A., Faltings, B., and Petcu., A. 2009. Distributed constraint optimization with structured resource constraints. In Proc. of the 8th Int. Conference on Autonomous Agents and Multiagent Systems (AAMAS '09), 923-930.