A Web-Based Multiplayer RTS using GCA*

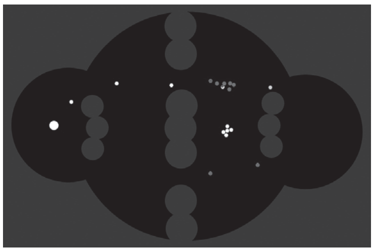
Computer Games, Networking, Artificial Intelligence / Advisor: Prof. Dr. Jürgen Eckerle
Expert: Han van der Kleij

Markus Roth and Dr. Jürgen Eckerle of the Bernese University of Applied Sciences developed a novel approach for collaborative pathfinding, named Generalized Collaborative A*. Using this method, a multiplayer Real-Time Strategy game was developed in Microsoft Silverlight to run inside a web browser. Besides advanced pathfinding, it offers modern RTS components such as unit types, bases, fog of war, advanced movement commands and more.

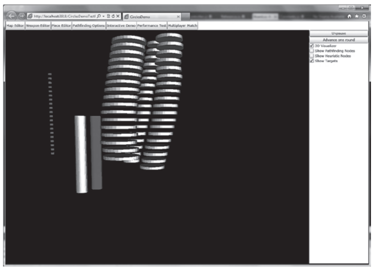
A multiplayer real-time strategy game (RTS) involves at least two players competing against each other by directing their armies, using strategy and tactics like generals would. In these games, the individual units use artificial intelligence techniques to act independently towards their strategically assigned targets. Thus, while a player might direct their troops to attack a certain group of opposing forces, the troops maneuver and exchange fire autonomously. This enables the player focus their attention on tactics and strategy, rather than forcing them to control each of their troops individually.

Given that RTS games often use terrain and other features blocking the movement of troops to make for a more complex game experience, the troops need advanced pathfinding algorithms to achieve their goals while not moving in the wrong direction, blocking each other, or getting stuck on obstacles. Ideally, troops would react exactly as the player expects them to, while simultaneously striving for optimal path sums as groups while reaching their potentially individually assigned targets. Since the user can issue new commands at any instant, these calculations have to be done in close to real-time. Since the publishing of the first modern RTS game in 1992, efforts have been ongoing to develop and improve such pathfinding techniques.

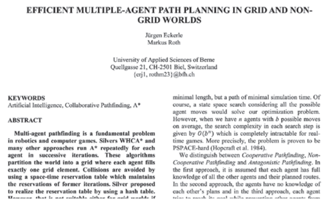
In this thesis, I continue these efforts by designing and implementing an RTS game using the collaborative pathfinding algorithm CGA*, developed by Dr. Jürgen Eckerle and myself in 2010. Being implemented in Microsoft Silverlight, it is the only completely web-based, full-fleshed multiplayer RTS with industry-standard pathfinding currently available.



A screenshot of the game, showing the abstract graphical style



A visualization of the three-dimensional reservation store of the pathfinding algorithm



EFFICIENT MULTIPLE-AGENT PATH PLANNING IN GRID AND NON-GRID WORLDS

Jürgen Eckerle
Markus Roth

University of Applied Sciences of Berne
Quellgasse 21, CH-2501 Biel, Switzerland
{erj1, rothm23}@bfh.ch

KEYWORDS
Artificial Intelligence, Collaborative Pathfinding, A*

ABSTRACT

Multi-agent pathfinding is a fundamental problem in robotics and computer games. Silvers WHCA* and many other approaches run A* repeatedly for each agent in successive iterations. These algorithms partition the world into a grid where each agent fills exactly one grid element. Collisions are avoided by using a space-time reservation table which maintains the reservations of former iterations. Silver proposed to realize the reservation table by using a hash table. However, that is not suitable either for grid worlds if

minimal length, but a path of minimal simulation time. Of course, a state space search considering all the possible agent moves would solve our optimization problem. However, when we have n agents with b possible moves on average, the search complexity in each search step is given by $O(b^n)$ which is completely intractable for real-time games. More precisely, the problem is proven to be PSPACE-hard (Hopcraft et al. 1984).

We distinguish between *Cooperative Pathfinding*, *Non-Cooperative Pathfinding* and *Antagonistic Pathfinding*. In the first approach, it is assumed that each agent has full knowledge of all the other agents and their planned routes. In the second approach, the agents have no knowledge of each other's plans and in the third approach, each agent tries to reach its goal while preventing other agents from

The conference paper describing the developed algorithm in detail

Markus Roth

maroth@gmail.com