Introduction

David Churchill, Abdallah Saffidine, Michael Buro

Okecávání:
Automated planning, i.e. finding a sequence of actions leading from a start to a goal state, is a central problem in artificial intelligence research with many applications such as robot navigation and theorem proving.
Úvod do problematiky – co je známo:
While search-based planning approaches have had a long tradition in the construction of strong AI systems for abstract games like Chess and Go, only in recent years have they been applied to modern video games, such as first-person shooter (FPS) and real-time strategy (RTS) games [1].
Co je známo detailněji + příklady:
Generating move sequences automatically has considerable advantages over scripted behavior, as anybody who tried to write a good rulebased Chess program can attest:

- Search naturally adapts to the current situation. By looking ahead it will often find winning variations, where scripted solutions fail due to the enormous decision complexity. For example, consider detecting mate-in-3 situations statically, i.e. without enumerating move sequences.
- Creating search-based AI systems usually requires less expert knowledge and can therefore be implemented faster. Testament to this insight is Monte Carlo tree search, a recently developed sample based search technique that revolutionized computer Go [2].

Vysvětlení problémů známých řešení:
These advantages come at a cost, namely increased runtime and/or memory requirements. Therefore, it can be challenging to adapt established search algorithms to large scale realtime decision problems, e.g. video games and robotics.
Náš přínos (vylepšení) – co jsme udělali + struktura článku:
In what follows, we will see how the Alpha-Beta search algorithm can be used to solve adversarial real-time planning problems with durative moves. We begin by motivating the application area — small scale combat in RTS games — and discussing relevant literature. We then define our search problem formally, describe several solution concepts, and then present our ABCD algorithm (Alpha-Beta Considering Durations) which we will evaluate experimentally. We conclude by discussing future research directions in this area.