**Chapter 1: introduction**

CHRISTEN FORD, Baldwin Wallace University

Path planning is a branch of artificial intelligence that is concerned determining routes for an artificial agent in both simulated and real environments. There have been many goals of path planning as well as many application domains. The most well-known involve a subset called shortest path algorithms. This type of path planning, as the name implies, is all about determining the shortest path from a source to a destination. Optimally, the best solution to this problem is to just follow a straight vector from the source to the destination. However, in practice, this is never achievable due to physical constraints such as obstacles and terrain or limits placed on the agent itself.

Another factor to consider when path planning is the number of dimensions to plan in. The obvious case and most easily approached is path planning in two dimensions. That is, path planning performed in a system that utilizes an (x, z) coordinate system. Many problems presented in two dimensions are easily solved by the various algorithms devised to do so, notably Dijkstra class algorithms. However, path planning becomes immeasurably more complex in terms of computation when the same problems are elevated to three dimensions. Unfortunately, this means that without advanced hardware, it becomes impractical to compute paths of any kind in three dimensions. This does not mean it cannot be achieved. Indeed, in constrained, regulated environments where inputs are limited, three dimensional paths can be determined. However, to have applicability to the real world, we must be able to at least model a spatially restricted sample of the world for pathing.

The final restriction on pathing in three dimensions is created by the enormous size of the sample space. With two dimensions of space, the sample space can be analyzed and a path can be computed in real time, regardless of complexity. If the sample space can be solved, a path can be determined. There currently exists only a small handful of path planning algorithms for computing paths in real time that are suitable for use in three dimensions. The most recent of these algorithms expands on the concept introduced by rapidly exploring random trees. As explained by Naderi et al. it extends the algorithm thorough constant resampling and rewiring to enable real time path planning (2015). This algorithm is known as rapidly exploring random trees or RT-RRT\*.

Given the recent advancements in power consumption for both central and graphical processing units. It is now possible to create a pathing algorithm that leverages advanced hardware to be able to create paths in three dimensions on the fly. While mobile computing has been a standby for decades, it has only recently reached a point where it matches the performance of traditional computing. This is further augmented by advancements in artificial intelligence as seen in platforms such as autonomous vehicles and the internet of things as well as robotics as showcased by Boston Dynamics.

Theoretically the improvements gained through distribution can be many. Most notable are improvements in the speed of calculations and the memory required to do so. Important is the fact that many modern graphical units involve computations across thousands of individual cores. Unlike a central processing unit which is better suited for throughput as opposed to speed, a graphics processing unit is capable by design of calculating at rates unachievable by a traditional processor. In addition, while central processing units contain multiple layers of localize memory cache, graphics processing units contain onboard with space thousands of times greater than the cache found in a traditional processor. Therefore, computation via the graphics pipeline weighs much more heavily on the development of artificial intelligence then does central processing. Cost is another factor to consider in this approach, however the cos of modern graphics hardware is now roughly equivalent to top tier computational units and is widely accessible to the public.

What this all leads to is the need for a new class of algorithms for a new world. Future algorithms will always build on those introduced in the past. This thesis serves to lay the framework for one such algorithm. This algorithm shall extend real-time rapidly exploring random trees to plan paths in real time, three-dimensional environments. It shall utilize advanced graphical hardware and processing techniques introduced in previous works. Finally, this algorithm shall be easily portable. Unlike several algorithms that utilize global data structures to compute paths, this algorithm shall be parallelized. This will allow us to compute paths across graphical cores, and possibly in the future across graphical units.

The goal of this thesis is to create an algorithm that under a set of specified criterion (specified in chapter three), can compute three dimensional paths in real time. However, even if the algorithm fails to meet these criterion, it will still classify as a success. Much of the work done in three-dimensional path planning is experimental. It is not as thoroughly researched as two-dimensional path planning. This leads to many failures in this field that further result in advancements at later times. As stated earlier, the need for three-dimensional path planning is rapidly increasing. A generalized approach to the planning problem can be utilized to solve problems in realms outside of computer science.