Evolving Motion Machines

Nathan Douglas
10160679

nathan.douglas@ucalgary.ca

1   Introduction

Emergent computing has been previously used in many areas before. In particular, the iterative approach taken by both evolutionary and genetic systems has repeatedly proven its' ability in several domains. The gradual approximation technique for a given goal function can create very interesting, efficient, and effective solutions. Further, some implementations have been remarkably effective in modelling real world behaviour. With each implementation and domain being so unique, there exists plenty of room for continued application of these techniques.

This research aims to expand on the existing pool of implementations using emergent approaches. General approaches are designed to either recreate natural behaviours or search for new optimal policies. This is done first by creating a proposed solution, or genotype, with some initial value selected for each variable part of the solution. The genotype or genotypes are then implemented and become their resulting phenotypes, which are then evaluated. Finally, the genotypes which created the most effective phenotypes are mutated slightly, and repeat the process of instantiation and evaluation. This creates the iterative process of searching for better and better solutions to a given goal function.

However, most research surrounding emergent computing is more focused on behaviours. The structure of the phenotype is generally static, while the actions and reasoning are given more careful attention. This system looks to use evolutionary strategies in a different way, focussing more on the resulting structure of a solution. To this end, a rather simple goal function is used: simply move as much as possible. The more complex aspect is in the genotype's large influence over the structure of the agent it generates.

Agents in this system are created from small, atomic pieces that join together to create larger collective structures. Atoms have different simple geometric shapes, different types of motion, and a variable direction and intensity of the motion. While atoms can mutate this direction and intensity as part of their evolutionary loop, they can also mutate additional adjoining atoms. These new pieces attach to their parent using a physical link, creating a large branching tree structure. With each atom still moving independently, this creates large agents with many independently moving subtrees.

This high degree on independence between subtrees creates a very dynamic structure with many different potential solutions. Ideally this variability allows for interesting results. As with evolution in nature, this system intends to mimic

the variation in structure of life on earth. As is shown below, this goal is somewhat met but the system itself is still too premature to be a comparable analog to natural selection.

2   Implementation

With the goal of moving as far as possible in each generation, an appropriate environment must first exist to evaluate within. Further, within the environment there are a list of features that must be included to support the evaluation. First and foremost is the ability to deserialize a given genotype into a phenotype, or agent, to be evaluated. Also needed is the inverse serialization ability. Then, agents need some physics simulator to operate within and source the evaluation of exactly how much distance has been covered.

To this end, Unity Engine is used

3   Results

4   Discussion

5   Future Work

6   Conclusion