Intelligent Agents CS 533: Homework2 Report

Vivswan Shitole and Rogen George

Oregon State University

1 Plots

1.1 Runtime versus Mapsize

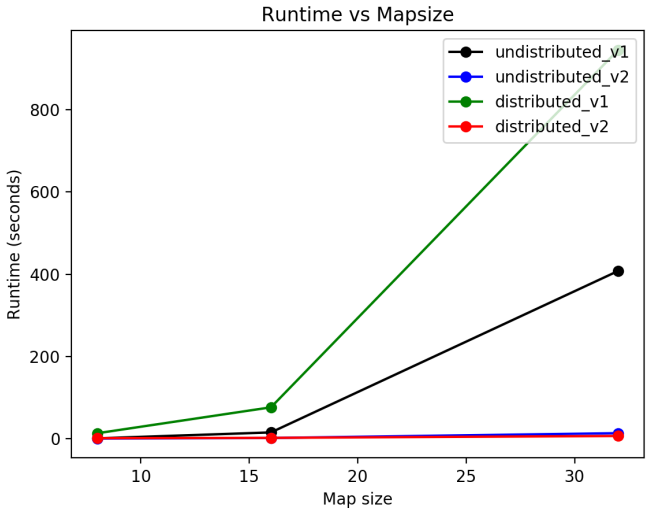

Fig. 1: Variation of runtime (in seconds) with the map sizes.

1.2 Runtime versus Number of workers

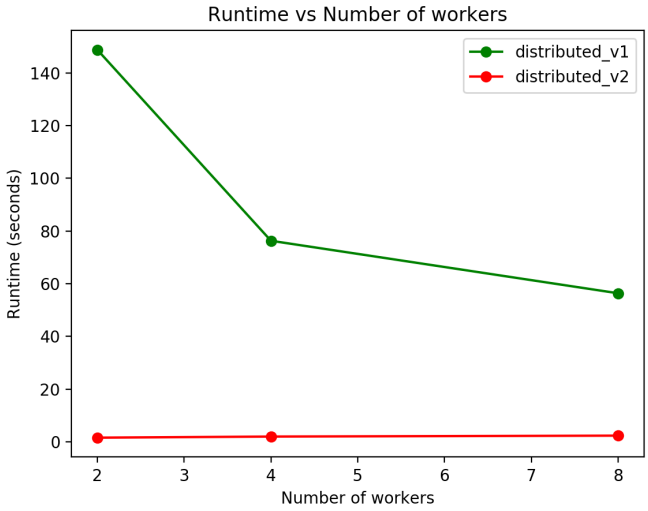

Fig. 2: Variation of runtime (in seconds) with the number of workers.

2 Why is the second distributed method is faster than the first one?

In the first distributed method, each of the 4 workers is responsible for updating the value of a single state. As soon as one of these workers is finished updating the value of the state assigned to it, the worker is re-initialized and assigned a new single state.

In the second distributed method, each of the 4 workers is working on a batch of states. Hence the workers need not be re-initialized after every state update, but only when they are done updating all the states in their assigned batch. This makes the update process more efficient with the workers updating all the states in their batch simultaneously without any interruptions. Less time is wasted on re-initializing and re-assigning the workers compared to the first distributed method.

3 Best distributed method versus Best non-distributed method

The best distributed method is the one which uses 4 workers updating 4 batch partitions of the state space values. The best non-distributed method is the one

using GetSuccessors() function to narrow down the number of next states (with non-zero transition probabilities). Following is a table of the runtimes of the two approaches with the corresponding plot.

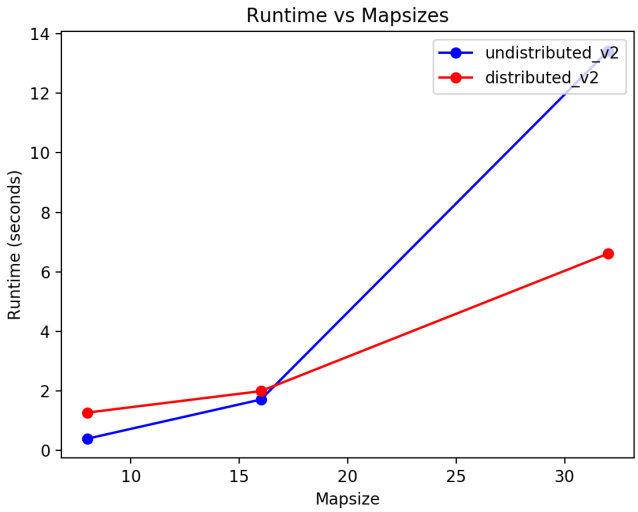

Fig. 3: Variation of runtime (in seconds) with map sizes for the best distributed and the best non-distributed methods.

We observe that the runtimes are almost similar, except for the map size of 32x32 where the efficiency of the distributed method obtains an edge over the non-distributed method. This is expected since the distributed method is updating values of multiple states simultaneously, compared to the non-distributed version which is updating the value of one state at a time. For smaller map sizes, this advantage of the distributed method is not apparent, overshadowed by the worker switching costs. For larger map sizes, the efficiency obtained by parallel calculations obtain an edge over the worker switching costs.