What we can do for Jeep in a Desert: Which is cleverer — human or learning-machine?

Akira Imada

Brest State Technical University
Moskowskaja 267 Brest 224017 Belarus
akira@bstu.by

Abstract – This article is not a report of success but a proposal of a benchmark that we believe not solvable by any of so-far-proposed machine learning techniques or computational evolutions. The benchmark is for robot navigation in two dimensional grid-world. A robot starting from somewhere in the grid should look for the exit of which the robot has no information about where. Furthermore, the robot needs to consume fuels to move, and the exit is far away from the starting point such that the robot must refill fuels to reach the exit. Assuming the robot can make it somehow in not so efficient way, the question is, "Can a robot elaborate its behavior by learning through a number of trials?" It might sound easy, but what we want is adaptive behavior not a deterministic action. This is extremely difficult, if not impossible.

Keywords – Robot navigation in a grid-world, A needle in a haystack, Jeep problem, Computational evolution, Machine-learning techniques.

I. Introduction

So, I lived all alone, without anyone I could really talk to, until I had to make a crash landing in the Sahara Desert six years ago. Something in my plane's engine has broken, and since I had neither a mechanic nor passengers in the plane with me. I was preparing to undertake the difficult repair job by myself. For me it was a matter of life or death: I had only enough drinking water for eight days. – "The Little Prince" by Antoine de Saint Exupery (translated by R. Howard.)

The research field of computational evolution or machine-learning techniques is now quite mature, and as such, we have had tremendously lots of success reports on how they solve a difficult problem, claiming like, *"The result solves a problem of indisputable difficulty in its field."* Or, *"Techniques of genetic and evolutionary computation are being increasingly applied to difficult real-world problems – often yielding results that are not merely interesting, but competitive with the work of creative and inventive humans."*[1]

Those claims might be true. However, the opposit also holds. Some problems would be extremely easy to be solved by a human, but remain unsolvable by any so-called artificial-intelligent technique. In this report, we propose one such problem as a benchmark, expecting it challenges our neural network, machine-learning, evolutionary-computation community, whatsoever.

The task is as follows. A robot navigates in a desert with a jeep which can carry a limited amount of fuel, starting from its base where the jeep can return later to refill the fuel. The jeep has containers to put some of its fuels somewhere in the desert to use it in future. The task is to find the exit of the desert far away from the base, by repeating the procedure: (i) start the base; (ii) navigate the desert; (iii) put fuels somewhere or find the fuels to get; and (iv) return to the base. Robot is allowed to return to the base pre-specified limited times.

This task is an extension of a mathematical puzzle called a *jeep-problem* in which a jeep explores one-dimensional desert under a constraint. A robot here navigates a two-dimensional grid-world instead of one-dimensional desert. The problem is studied here in a simulation, that is, a navigation robot explores a fictitious $N \times N$ grid-world. Hence, the motion of the jeep is from one cell to one cell to the north, south, east, or west; the distance of penetration is by *Manhattan distance*.

The question is, "Can a navigation robot elaborate its behavior by learning through a number of trials?" This should be done not in a deterministic way, but in an adaptive way.

II. A Preliminary Experiment

From a mountain as high as this one, he said to himself, I'll get a view of the whole planet and all the people on it... But he saw nothing but rocky peaks as sharp as needles. – "The Little Prince" by Antoine de Saint Exupery (translated by R. Howard.)

As a preliminary experiment, let's try a kind of two-dimensional version of *a-needle-in-a-haystack problem.* The task is to look for a needle, or equivalently, uniquely pre-determined location in the grid-world. The robot has no information of where the needle is. The navigation is by random walk. The robot is expected to eventually find the location, unless N is very large. Question is whether

[1] These two expressions are appeared in "Call for entries for \$10,000 in awards – The 2007 'Humies' – for human competitive results." This competition was held in the *Genetic and Evolutionary Computation Conference* in London in 2007.

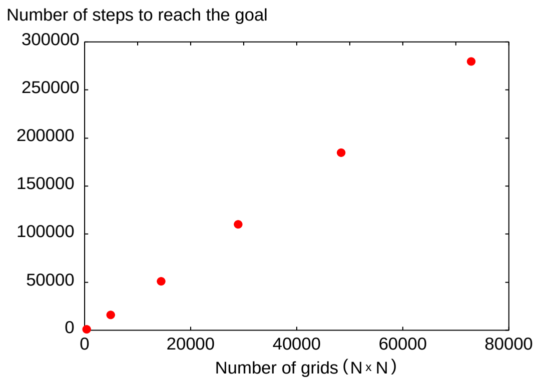

Fig. 1. Average number of steps during 100 runs until a robot who explores the $N \times N$ grid starting from $(1, 1)$ by a random walk eventually reaches (N, N) as a function of number of cells in the grid N^2.

the robot can minimize the path length by a learning algorithm later?

The expectation of number of steps of a robot to reach the exit is $O(N^2)$. The result of our experiment to confirm this is shown in Fig. 1 by the average number of steps in 100 runs plotted as a function of N^2. This might be called *a two-dimensional version of a-needle-in-a-hay-stack* problem.[2]

We tried this experiment by increasing the grid size. As size grows the task becomes difficult. The experiment is still on going, but according to our so-far observation of 1000 different runs, when grid size is 17,000 × 17,000, the minimum steps required was 25,987,691, and the robot reached the needle only in 119 out of 1,000 runs. The number of steps is starting to explode.

The question then will be, "A learning can enhance the efficiency?" In other words, "If a robot try it multiple times under a learning scheme, then the number of steps of the robot to reach the exit becomes shorter than the previous trials, or hopefully minimized?"

In Fig. 2 we show the experiment when the grid size is 96 × 96. First, by a random walk. Starting from (24,24) a robot walks aiming the goal at (72,72) of which the robot had no *a-priori* information.[3] We observed a 100 such runs and in the left of the Fig. 2 we show the minimum path out of those 100 different walks.

Then we tried a evolutionary learning, just as an example among many others, in which a possible trace of robot is expressed by a chromosomes whose gene is either 0, 1, 2, or 3, meaning to move one cell to the north, south, east or west, respectively. The length of one chromosome is N^2. Following this chromosome from one gene to the next, the robot moves from one cell to the next. After each movement it is checked if the current location is the goal or not. If it is the goal the walk is completed, otherwise to the next cell. The longest possible path length would be thus N^2 but most likely much less than that. The fitness is the number of steps to the goal. If the robot did not reach the goal after following all the N^2 genes, the walk failed. The selection is by *fitness proportionate*, reproduction is by *one-point crossover* and mutation is by replacement with other gene chosen at random with the probability of $1/N$. A result of trace after the evolution is shown in the right of the Fig. 2, and the fitness evolution is shown in Fig. 3.

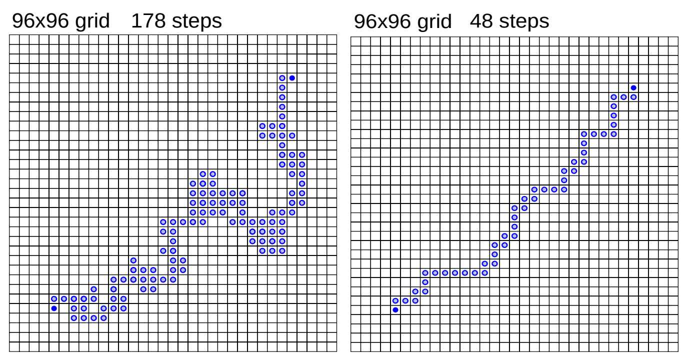

Fig. 2. In the grid-world of 96 starting from (24,24) a robot walks aiming the goal at (72,72) of which the robot had no *a-priori* information. Left: The path of minimum length among 100 trials by random walk. Right: Minimal path the robot found after an evolutionary learning as shown in Fig. 3. (Marginal area is omitted.)

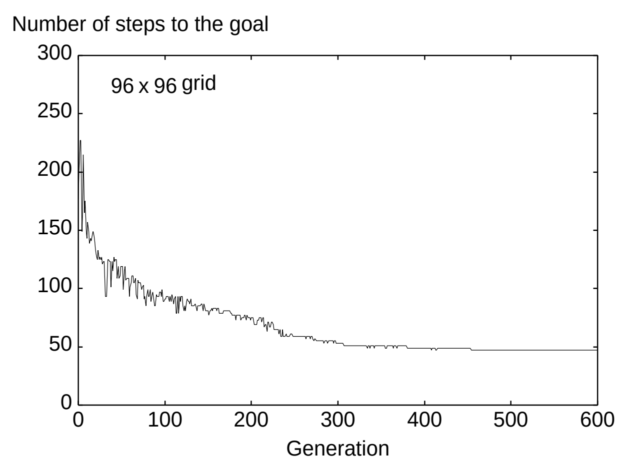

Fig. 3. An evolution of the number of steps to the goal starting with a population of random walks. We can see the convergence to the global minimum of 48 Manhattan distance.

III. Challenge

He answered, "Oh come on! Your know!" as if we were talking about something quite obvious. And I was forced to make a great mental effort to understand this problem

[2] The problem in general from a computational context was firstly described in 1987 by Hinton & Nowlan [1] as *a needle being a unique configuration of 20-bit binary string while all other configurations being a haystack.*

[3] The reason for the start and goal are far inside the grid is, otherwise robot found a more clever warp utilizing our toroidal character of the grid. Namely, robot could reach from (1,1) to (N, N) just with 1 step at the minimum.

all by myself. – "The Little Prince" by Antoine de Saint Exupery (translated by R. Howard.)

Can a learning robot survive in a desert with a jeep?

We now assume that the jeep are in the base located at the center of a desert. Again our world is a two-dimensional grid. This time, for some reasons which later become clear, size is 77×77, the coordinate of the bottom-left corner is $(-38, -38)$, and the top-right corner is $(38, 38)$ which is the only exit of the desert. The base is located at the origin $(0, 0)$. The grid is toroidal, that is, if the coordinate becomes $(N+1)$ and $-(N+1)$ then it is replaced with $-N$ and N, respectively.

A robot leaves the base with a jeep. The jeep moves the desert of grid from one cell to the next, each time by consuming one unit of fuel. The jeep has a tank for fuel whose capacity is 30 units. The jeep also has a container with which the robot can store some amount of fuel in the tank to put at any location in the desert for the next time usage. Since the exit is 76 *Manhattan-distance* apart from the base, the tank full of 30 units are not enough to reach the exit. The robot is allowed to go back to the base twice to refill the tank.

This is an extension of so-called a *jeep problem* where a jeep should maximize its penetration to one-dimensional desert under a constraint. See, for example, the WWW page of *Wolfram MathWorld*.[4] It reads:

> *"Maximize the distance a Jeep can penetrate into the (one-dimensional) desert using a given quantity of fuel. The Jeep is allowed to go forward, unload some fuel, and then return to its base using the fuel remaining in its tank. At its base, it may refuel and set out again. When it reaches fuel it has previously stored, it may then use it to partially fill its tank. This problem is also called the exploration problem (Ball and Coxeter 1987)."*

As far as we know, this has never extended to a two-dimensional world. We now summarize the problem.

Challenge (Jeep's survival in a desert)
Assume 77×77 toroidal lattice each of whose cells is expressed by (i,j) where $i, j = -38, -37 \cdots, 0, 1, 2, \cdots, 37, 38$. We call this grid a desert. The desert has only one exit at (38,38). Starting from (0,0) a robot navigates a jeep from one cell to the next. In order for the jeep to move one cell, it needs to spend one unit of fuel, and the jeep has the tank whose capacity is 30 units. The jeep also has a container with which the robot put some amount of fuel to any location of the desert for the next time usage. Allowing to go back to the base twice, can the robot learn how to reach the exit through a multiple times of experiences of failure?

[4] http://mathworld.wolfram.com/JeepProblem.html.

Or, we might be modified it like the original one, as follows. *The size is more generally $N \times N$ for a large enough N. Starting from also the base at (0,0) and being allowed to go back to the base R times to refill the fuels, the robot should penetrate the maximum distance from the base instead of aiming the exit of the desert.*

IV. Possible Approaches

A geographer is too important to go wandering about. He never leaves his study. But he receives the explorers there. He questions them, and he writes down what they remember. – "The Little Prince" by Antoine de Saint Exupery (translated by R. Howard.)

As such agents who could solve our problem, let's try here two candidates, among others. One is Neural Network and the other is Finite State machine (FSM). And as learning, we experiment with a Computational Evolution and Reinforcement Learning.

A. Neural Network under Evolutionary Computation

With architecture being Feed Forward or Recurrent, whichever it might be, input is what the jeep sees in each of the cells which jeep locates. So, it might be three-fold: (i) see nothing, (ii) see container previously put, and (iii) find the cell is base. Out put might be six-fold: (i) put a unit of fuel, (ii) get a unit of fuel, (iii) move north, (iv) move south, (v) move east or (vi) move west. We may use a population-learning, such as Genetic Algorithm, starting from a population of agents with all random weight connections, and fitness evaluation being by how close each individual agent approaches to the exit.

B. Finite State Machine under Reinforcement learning

FSM could also explore the desert. With two inputs – nothing to see or container there, and six actions — the same as those in the neural network above. The number of states should be arbitrary large. The behavior of the FSM is determined by the transition table which determine the next action and the next state according to the current state and input. Though these FSMs could learn the behavior also by a Computational Evolution as in Neural Network above, but here for a change, we might use Reinforcement Learning.

I will describe these two implementations more in detail in case our kindly reviewer generously accept this submission.

V. We know a solution, but...

I made an exasperated gesture. It is absurd looking for a well, at random, in the vastness of the desert. But even so, we started walking. – "The Little Prince" by Antoine de Saint Exupery (translated by R. Howard.)

As already mentioned, the task is an extension of the Jeep problem. This is quite an old problem, and we now have to confess that we know the analytical solution. Let's see the already known solution of the original version in the one-dimensional desert here.

1) *Start with 30 units of fuel.*
2) *Go forward 10 distances, put 10 units, and then go back to the base with the remaining 10 units of fuels, and refill 30 units again.*
3) *Go forward 10 with 30 units refilled, spending 10 units, and get 10 units there.*
4) *Go forward 6 further, spending 6 units and put there 8 units.*
5) *Go back to the base spending remaining 16 units*
6) *With 30 units again, go forward 16 , spending 16 units, and get 8.*
7) *Go forward further until spending all the remaining fuel, and eventually reach the point which is 38 apart from the base.*

This is how the jeep can penetrate to the desert with the maximum distance when allowed to go back to the base twice. You now notice the reason why those parameters in our two-dimensional version described here are highly artificially devised. It is to fit the problem.

As you now know, the solution can be simply applied to the two-dimensional grid-world. However, we must notice, we have infinite number of such optimal paths to the exit while only unique in the case of one-dimensional desert.

VI. Concluding Remarks

To put it frankly we have had no good results for this problem as of today.

Back in 1992, in their seminal paper, Jefferson et al. reported a similar experiment of trail-following problem. The world is also the two-dimensional toroidal grid in which a trail called the *John Muir trail* is specified. The trail is not continuous but includes empty gap(s) whose length is from one to three. The task of the agent is to move from square to square, traversing as much of the trail as possible. As a candidate of the agent, a FSM under Genetic Algorithm was proposed, while commenting, "Neural network will perform better than FSM." But we are wondering, how many input nodes are necessary in its minimal case. If we think it of one input, what, on earth, might be its structure?

Anyway, their results looked excellent using FSM which succeeded in following the 92 squares of the trail after the FSM's evolutionary learning.

Notice, however, the action of the successful FSM is totally deterministic. The result is natural since the solution is only one unique way.

Then what about in our problem? As we already mentioned, we have plenty of paths of the minimum length in the case of two-dimensional grid. What if the jeep which learned successfully follows always the same path as it found for the first time, totally neglecting the other possibilities? How can we call this behavior an *intelligent* one? All we want is an adaptive behavior of the successfully learned agents, like an intelligent human. As far as we know, we have never had such artificial intelligent behaviors after learning in any of those successful reports. Is it possible or just a fantasy?

References

[1] Hinton, G. E., and S. J. Nowlan (1987) "How Learning can guide evolution?" Complex Systems, Vol. 1, pp. 495–502.

[2] Remazeilles, A., and F. Chaumette (2007) "Image-based robot navigation from an image memory." Robotics and Autonomous Systems archive, Vol. 55(4), pp. 345-356.