AI - How does it cope in an arbitrary world

What is the world of the artificial intellect like

February 2001
this paper is a part from AI- Project

Dimiter Dobrev
d@dobrev.com
illustrations - Konstantin Lakov

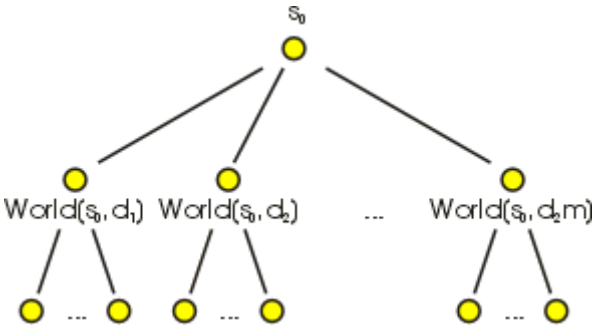
In this paper we are going to find out what is an arbitrary world and when two worlds are indistinguishable. Also, we are going to think whether the world is determined or not. We are going to ask ourselves how would AI cope, without knowing the world it got into. We are going to see that the way for understanding the world is the building of a correct model.

This paper is the sequel of "AI - What is this" (ed. 11/2000). We said there that we want to create a device which in an arbitrary world will cope not worse than a human being. Let us think what is an arbitrary world and how our device would cope in such world. We are going to ask ourselves how many are the possible world's states and when two worlds are indistinguishable?

We said that an arbitrary world is an arbitrary function **World(s, d)**, the arguments of which are the world's state and the influence that our device has on the world. The result of this function is the world's new state. We said that the device works out finite information (**m** bits), i.e. the possible influence actions towards the world are finite (2^m).

Let us have an arbitrary world. We are going to build the tree of the attainable world's states. An attainable world's state will be the one that can be reached by our device (or the one that the device can bring the world to). In the tree's root we are going to place the state s_0. This world's state will be reached at the moment of birth. Inheritors of s_0 will be the states World(s_0, d_i) where d_i runs through all possible actions (as we have already said, they are a finite number). These states can be reached in a moment one (if the action in moment zero was the respective one).

By analogy, we define the inheritors of the inheritors and so on. We get an infinite tree with countably many knots. We will call this tree the tree of the attainable states.

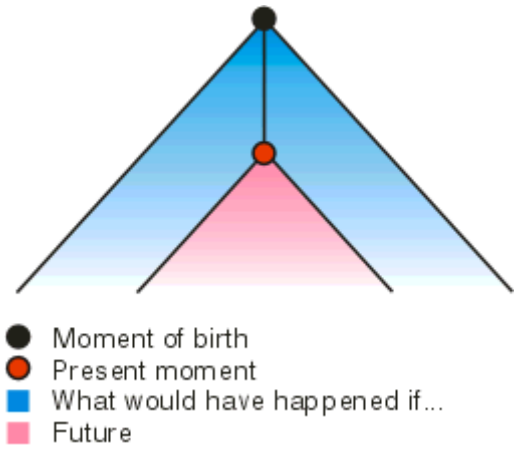


From the tree of the attainable states we can easily get another tree, which we will call the tree of the world. This will be the same tree but at each knot instead some state (s_i) we will juxtapose **View(s_i)**, i.e. instead the respective world's state we will juxtapose the information the device gets as an entrance when it is in that state (what it sees). Why did we call this tree with the pretentious name tree of the world? It is because that if two worlds have the same tree of the world then they are absolutely indistinguishable from the point of view of the

device. No matter what experiment it would carry out, it would get the same result in both worlds because with the same sequence of actions it would see the same things.

Now we are ready to answer the question how many the world's states are. They can be arbitrary many, but because only the attainable ones matter we can safely consider that the others do not exist. It means that if we throw out the unattainable states from a world, we will get a world indistinguishable from the given one. The newly obtained world will have no more than countably many states.

What is the life of our device? This is a path in the tree of the world, starting from the root **View(s_0)**. This path is potentially infinite but at a given moment **t** of its life the path has length **t**. What does our device know at the moment **t**? Up to this moment it knows only its own life path, the other knots are - what would have happened if it would have done another thing or what will happen in future. All these are things that our device does not know and only can suppose about them.

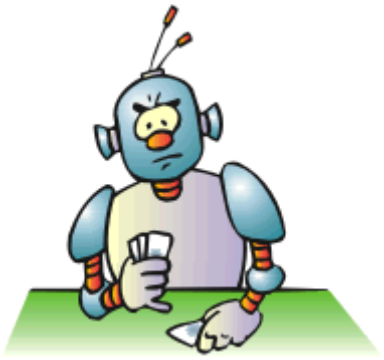


That is to say, our device is at the moment **t** of its life and its task is to understand the world, to understand the principles that make it move. This understanding of the world is necessary for it to be able in future to choose its way better and to get a better mark for its actions. Well, the understanding of the world is important, but how it can be done, how from a finite path in a infinite tree a whole tree can be built.

The latter (to be built a whole tree) is a too ambitious task. This means the world to be fully understood. If you understand fully the world you live in, you will not need to carry out experiments, because you will know the result you will get. For example, you will not have to look for your things, because you will already know where they are, even when you throw dice you will know what will the result be, also you will know the lottery numbers. As you can see, you do not know fully the world you live in and it is not sensible for us to hope our device to achieve such a result. We can have full knowledge of the world only in very simple worlds, but even there, even if we have a model, giving us a full description of the world, we can never be sure that this is a correct model. This model could have proved itself million times, but we do not have guarantees that in the million and first one it will not mislead us. For example, you have a model according to which the sun rises each morning, but you cannot be sure that it will rise tomorrow, because it is possible that in the world where you live the sun rises a finite number of times and today was the last sunrise.

That is to say, the task of AI is to find a model of the world and to act according to this model. It is not necessary for the model to be full and to tell everything about the world (we will want to tell more, of course). The model should be reliable, that is to say, it should not contradict the life experience of AI. As there are infinite many models that do not contradict our life experience we will aim to find the simplest one. For example, we will suppose that the sun will rise tomorrow again, because the model in which the sun rises every day is simpler than the model in which the sun rises ten thousand times and then stops.

Another question is whether we consider the function **World** as determined. Are the numbers from the lottery previously determined in the real world and is it the same about the weather. If they are determined and if we can find the dependency determining them, we could earn well or at least we will know when to take an umbrella and when - not.



According to some people there is fate that determines everything, that is why they accept the fate and get wet in the rain. Other people think that everything can be calculated, that is why they listen to the weather forecast. There is a third group of people who think that these are fortuitous events and for this reason they always have an umbrella. According to us, it does not matter whether this is determined or not. The only thing that interests us is whether we can foresee it. If something is determined by a very complicated dependency that we cannot understand it will be the same for us if this is fortuitous or not. The conclusion is that we can safely think the function **World** determined, but to assume that in the world there are phenomena which dependencies are difficultly understood or cannot be understood at all. In the model that we are going to build of the world these phenomena will be modelled as fortuitous.

There is one thing left. It is to tell what will the model of the world look like, how AI will find this model and how by possessing it will plan its actions so, to get a maximum result. Of course, the results will be good if the model is correct or, at least, it is adequate to the world to some extent. For now we will say that the model will consist of finite automata. Among the automata we will also use undetermined ones, in order to describe the fortuitous events.

The question what a finite automata is and how it could be the model of the world goes beyond the limits of this paper but the readers can take interest themselves and to find literature on this topic. A better studied question is that how the computer can plan its actions when it already has a model of the world. The most popular algorithm is the so-called Min-Max algorithm that is at the base of the play- chess programs. If the reader is interested how the computer thinks through Min-Max we would advise him to take a look at the program Strawberry Prolog that is on the CD enclosed to this magazine. There in the examples are a few games (Tick-Tack-Toe, Checkers and others) that use the algorithm Min-Max.