Designing an Electronic Mind Capable of Feeling, Thinking, Predicting, and Awareness

Milad Keramati

keramati900@gmail.com

ABSTRACT

In a problem facing agent, a situation can be categorized as different patterns and action can be taken based on the available information (known as method) as oppose to a simple value. Doing so will decrease the variety of situations and actions and as a result simplify the problem. Simple patterns and methods are generated at first but by detecting important patterns and methods and creating similar patterns and methods, the agent will be able to better recognize the situation it's in and find better solutions for the patterns respectively and as a result systematically broaden its knowledge over time.

By memorizing feelings (or rewards) and action result (situation) in a pattern, it's possible to make a tree of possible outcomes of an action related to a pattern and choose an action of the pattern that profit us the most by predicting future feelings and calculating the value and we know accuracy of our prediction based on similarity (or consistency) and number of results (or confidence).

I've also given my opinion and defined some standards regarding artificial intelligence, reinforcement learning, and designing agent in this paper.

1 INTRODUCTION

In reinforcement learning artificial intelligence problems, the agent might receive multiple input values, each with many possibilities. Same is true about output or action. And because the environment can be volatile, we need multiple results for each scenario to become more confident about our action and take actions with more accuracy. In complex problems, experiencing each input and trying every possible action multiple times is a very inefficient task. To solve this, we need to find similarities in inputs and label similar ones as a pattern and work on them instead. We can use a similar approach for taking action. By analyzing the results of actions related to a pattern, we can find out if they follow any rules depend on any data.

We need more understanding of the way human mind works (which is related to psychology) and use it to create and craft a mind for each specific problem. We should be able to explain and use some human mind related concepts in our mind such as thinking, feeling, predicting, awareness, hope, skill, experience, learning, confidence, deciding. A mind is not complete without these concepts; which have been addressed in this paper.

Using these principles, we can take advantage of this mind to solve a variety of problems; such as games, object recognition, and understanding human language. The usage is limitless.

2 ELEMENTS

Agent The entity that can take action, feel and interact with an environment.

Environment The entity that agent interacts with, gets its input from and sends its output to.

God The entity that has created the agent and maybe the environment and has the full control over the entire process.

World Consists of agent and environment; in other word everything.

Input Refers to the value(s) agent gets from its environment. The type (such as range and being continuous or discrete) of the value(s) depend on the environment.

Output/action Refers to the value(s) agent sends to its environment as a response. The type of the value(s) depend on the environment.

Method A method is a way to take action based on the available information.

Step A time point in life of the agent.

Episode A life of the agent. Consists of steps.

State One of several properties that describes current step.

Situation Consists of all states of current step.

Mind An agent can gain profit by acting in a *specific way* in the world. Finding the best *way* is the job of mind.

Feeling & feeling system Feeling is a numeric value that is calculated by the feeling system of agent (not environment) based on current step and all the agent is interested in is to act in a way to increase this value and prevent it to be low.

Concept Something that is defined and known by the agent which is hardwired and exists when the agent is born. This can be things like: numbers, first, last, addition, sum, greater than, equal, even, odd, etc. An agent may be familiar with multiple concepts.

Event A step in the past and all the data related to it which includes situation, feeling, action, and some more.

History Stored data from the past. Consists of events.

Pattern A way to describe a situation.

Experience A pattern and all the data related to it which is based on past events.

Termination condition God, agent, and environment may have the ability to end the life of agent (end the episode) and they do it by evaluating a defined condition known as termination condition.

2.1 SITUATION

All in the below can be considered part of situation; but to simplify the problem we may not consider all.

- Inputs (current and previous ones)
- Action/output (current and previous ones)
- Step/episode number

It can also include things that are in agent's mind. Such as:

- Number of times a method has been tried
- Consistency of results
- Matched patterns (current and previous ones)

Situation is used in the feeling system, pattern, and method.

2.2 FEELING

Feeling value is a continuous value and is in a specific range from negative to positive; with negative indicating bad feeling; zero indicating neutral; and positive indicating good.

3 ENVIRONMENT

Anything about environments is possible. They may or may not: be completely random, have memory, get effected by agent's actions, be smart, be irrational, be complex, be crazy.

In the beginning the agent *may or may not* know anything about the environment. Every environment works by an algorithm and agent should try to learn only about the part that effects its feeling.

Agents should be designed based on their environments and the available information we have about their environments to earn as much benefit as possible. Your agent does not have to be "complete" and work in every environment–general; it's not even possible to do. But should make specific agents that work well in their own field–special.

3.1 ASSUMPTIONS ABOUT ENVIRONMENT

In each world, the agent may be based on some assumptions about the behavior of environment. These assumptions can be based on guesses or knowledge.

Some examples of these assumptions include:

1. Likely that it understands and uses the concepts that we're familiar with.
2. Likely that it behaves *similarly* in the situations that are *similar*.
3. Likely that it's not 100% random and we can understand it and get good feelings by it–hope.

3.2 RANDOMNESS

An environment is random if we can't understand it and it seems unexpected. If we can, it's not. The more we understand it, the less random it is. Even if we get some numbers that are all the same, until we're not familiar with equality, it's random. *The goal* is to decrease randomness to understand and better control the problem to get better feelings. By *understanding environment*, we mean correctly predicting the future.

4 PATTERN

A pattern is for describing a situation, and a situation might be described in multiple ways; using multiple patterns. It's easy to understand pattern with some examples.

Consider a situation with two states:

st1 2.5
st2 5

Now consider these patterns:

pat1 $(st1 = 2.5)$
pat2 $(st2 = 5), (st2 > st1)$
pat3 $(st2 = st1), (st2 = 5)$

Each pattern is consisting of one or multiple basic patterns or BP. A BP is a condition based on a concept which can either be true or false. We say a pattern matches a situation (or is a match) if all its BPs are true.

1. *Pat1* has one true BP which uses the concept of equality. Thus *pat1* is a match.
2. *Pat2* has two BPs and they use the concepts of equality and greater than. And they are both true. Thus *pat2* is a match.
3. *Pat3* is not a match because the first BP is false.

We call the number of BPs of a pattern its *complexity*. For example, pat3 has complexity of 2.

An agent usually has many patterns. With each new step and new situation, all its patterns will be evaluated and matched ones will be identified.

4.1 CREATION OF NEW PATTERNS

With each new situation the agent tries to create new BPs that match the situation using its concepts. And each new BP creates a simple–complexity of 1–pattern. But creation of patterns with higher complexity is different. By detecting

an important pattern we try to create new patterns similar to the pattern but with one new BP added; resulting in one or multiple new patterns with higher complexity. These new patterns will make the agent more sensitive to some specific situations. The process of creating new and more complex pattern(s) from another pattern is known as *pattern extending*.

4.2 DETECTING IMPORTANT PATTERNS

Extending every pattern will result in excessive patterns with no use that just make the problem more complicated. Thus it's crucial to extend only important patterns. The methods for extending patterns include:

By feel If we get a feeling from a situation that is very different–too good or too bad–from average feeling, it's an indication of important situation and we should extend all the patterns that match the situation. Because previous actions might have had effect to reach this situation, it's a good idea to extend not only patterns of that situation but patterns of a few situations before it too.

By average feel If average feeling of the agent when a specific pattern matches is very different–high or low compare to other patterns–, extend the pattern.

By frequently If a pattern matches too much, extend the pattern.

5 METHOD

A method is a way to take action based on the available information. It's like a function in computer programming with the return value as action. Methods are agent's way to send data (output) to environment. By effecting one or multiple concepts on situation a method can take complex actions. The number of concepts used in a method is called its *complexity*. Taking return value of a method as action is called *executing a method*.

Consider this situation:

st1 0.5
st2 1

And these methods:

met1 0.7
met2 $0.5 * st1$
met3 $(0.2 * st2) - st1$

In this case:

1. *met1* is a simple method which uses no concept–complexity of 0–and returns a value. In our situation the action value would be: 0.7 (although it doesn't depend on situation)

2. *met2* uses the multiply concept and has complexity of 1. In this situation, action value is: $0.5 * 0.5 = 0.25$

3. *met3* uses two concepts, multiply, and subtraction–complexity of 2. Action value of *met3* in this situation is: $(0.2 * 1) - 0.5 = -0.3$

5.1 METHOD EXTENDING

The process of creating new methods by effecting new concepts on a method is called *method extending*. Extending a method will create one or multiple new methods similar to the original one but with one more concept and complexity. For most problems, zero complexity methods are enough. Some can't even be extended; for example, when our output is limited and includes a few discrete values.

Some basic and simple–low complexity–methods are defined for the agent at first which might do alright in some situations. These methods are known as basic methods. But it's important to detect important methods to create more methods similar to them by extending them. Doing so, the agent will get closer to the perfect solution.

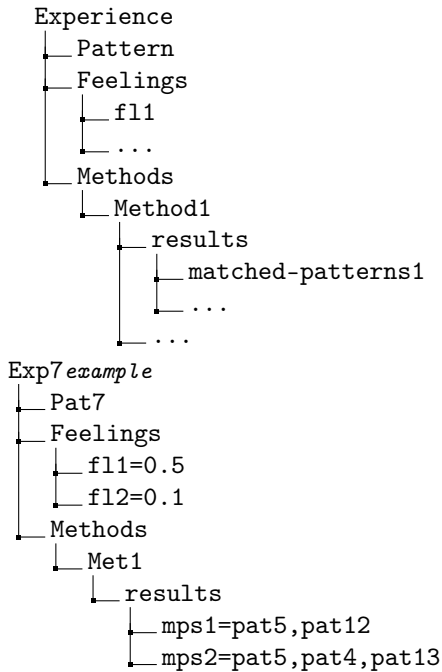### 5.2 DETECTING IMPORTANT METHODS *(theory)*

Detecting important methods and extending them can be done by these techniques:

By feel If average feeling of the agent after executing a method is very different–high or low compare to other methods –, extend the method.

By frequently If a method is used frequently, extend it.

6 EXPERIENCE

An experience is related to a pattern and all the data related to it which is based on past events. The data includes the feeling of agent in the pattern and all the taken methods when it was a match and the result (patterns that matched) of the taken methods which is related to the step after executing the method.

```
Experience
├──Pattern
├──Feelings
│   ├──fl1
│   └── ...
└──Methods
    └──Method1
        ├──results
        │   ├──matched-patterns1
        │   └── ...
        └── ...
Exp7 example
├──Pat7
├──Feelings
│   ├──fl1=0.5
│   └──fl2=0.1
└──Methods
    └──Met1
        └──results
            ├──mps1=pat5,pat12
            └──mps2=pat5,pat4,pat13
```

When a new pattern is created, a new experience will be created for it with some simple and basic methods that had been defined before. New–extended–methods from a method in an experience will be added to the experience.

When a pattern matches a situation, the agent should execute one of the methods related to the experience of the pattern. When a pattern matched too much and its methods have been taken too many times, we have a better understanding or the experience and its methods and can judge it based on its results. What we're interested to know about are:

Matched patterns Where would this method lead us to (what would the matched patterns be) if we execute this method.

Feeling What would we feel in the pattern.

These two should be predicted from the results we have which is based on past events. The way we use results to predict them depends on the behavior of the environment. But one simple way would be to average (mean) the results.

One important factor to consider is accuracy of our prediction which is directly affected by:

Confidence The more results we have, the more confident we are.

Consistency The more similar our data, the more consistent the data. In case of matched patterns, the more the same pattern matches, the more consistent the pattern.

Now we can predict feeling and patterns that will match by executing a method and we know accuracy of our prediction.

6.1 USELESS EXPERIENCE

Although we tried to created only useful and important patterns, the true value of a pattern is discovered in its experience after getting multiple results. If a method of an experience doesn't provide good prediction, it's better to be ignored and disabled and if all methods of an experience is disabled, disable the experience.

7 THINKING

In each step and situation the agent should take an action by executing a method. Among possible methods the agent should take the best method; that is, the method that benefits it the most–gives the best feeling.

Using its experiences, it is possible to predict the outcome of each method. It should consider the outcome not just in the next step–short-term–but in the next n steps–long-term. The process of evaluating possible actions or methods, considering their future effects and choosing one is called *thinking*.

For each situation one or multiple patterns might be matched. Using experience of these patterns, if we have any, we can predict the result and accuracy of each method of these experiences. And the same is true about experiences of each of the matched patterns of these results. Following this approach, we can create a tree that each path represents a possible future with a chain of feeling, matched pattern and accuracy. The bigger the depth of our tree, the further we can predict and the wiser our decision. Figure1 is an example of part of a thinking tree.
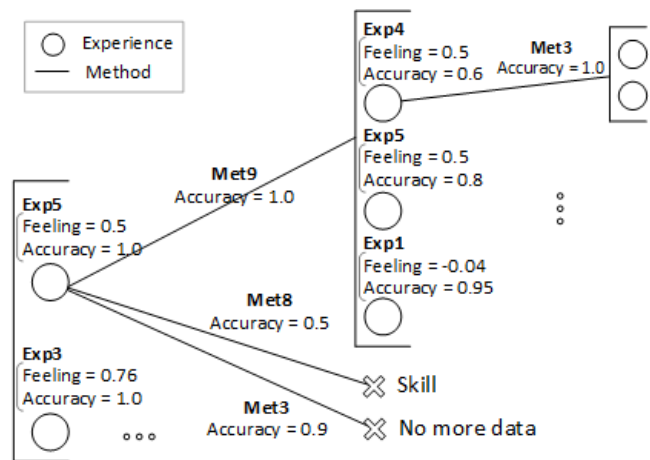

Figure 1: *Part of a thinking tree*

A path might end when one of these conditions is true:

1. It has reached the maximum depth limit.
2. Accuracy of our prediction about the next pattern (and experience) is low.
3. A method has a skill (will be explained later).

4

7.1 Path Value

We know that we should always choose the best method. This means the method that leads to the biggest value. To do it we should calculate the value of each path. Which is calculated by cumulative value of each method or branch in the path. And value of a method is directly affected by our prediction about *feeling* and by *accuracy* of our prediction.

The method of depth zero of the path is the important method in path. It's the method we predicted that by taking it we can reach the value. We consider the maximum path value associated with a depth zero method, the value of the method. It is possible to have the same method in different experiences of depth zero which should be considered considering their accuracies.

Following the approach gives no chance to *explore* new methods. To solve this, we can give low accurate methods some positive value known as *hope* value. The lower the accuracy, the more this value effects the value of the method and the lower its actual method value. Notice that if a method of an experience has not been taken before and has zero results, its confidence and accuracy will be zero.

7.2 Skill

Calculating values of methods of experiences is expensive and the computation increase exponentially as depth increases. But calculating value of a method every single time is not necessary. We can calculate it once and store it to use it later, instead of calculating value. A stored method value is called a *skill*. Skills are like unconsciousness; the agent just knows what to do without thinking about it. Skill is a crucial feature which without it the agent might practically become useless.

8 Designing Agent

Every agent should be designed and crafted according to its environment. And it's done by its creator–god–which is a human. Difficulty of designing an agent depends on complexity of its environment.

We should use the information we have about the behavior of environment–if we have any–in our advantage and consider it when designing the agent.

The important parts in designing agent are described below.

8.1 Situation

For most worlds inputs are enough for situation. Although it all depends on our knowledge about the environment. In the worst case we don't know anything about environment and should consider more than just the inputs. Depends on our understanding we might know that only a few inputs are important and should be considered as states.

8.2 Feeling System

Designing this depends on what we want from the environment. Just make sure to give good situations or the ones we seek, positive feeling and bad situations or the ones we want to avoid, negative feeling.

8.3 Pattern

Which conditions should be checked in basic patterns? Which state(s) with which concept?

The smaller the number of basic pattern, the simpler and faster our agent. Doing so requires knowledge about the environment. Basic concepts like equality, less than and bigger than are enough for patterns in most environments.

Not all situations might be needed in patterns and to decrease possible BPs it might be good to round or simplify some states for some concepts such as equality.

8.4 Method

Which states and concepts are needed for basic methods and for extending them?

The simpler our possible methods, the simpler and faster our agent. But doing so requires knowledge about the environment. For most environments the simple–zero complexity–methods are enough.

9 Weak Points

This is an ongoing project and I'm still trying to make it more similar to human mind. It has shown promising results in most worlds; but it seems that solving more complex worlds such as worlds requiring understanding human language need a more complex design which seems to include: connected and hierarchical patterns, conditional method, method capable of thinking, and maybe more.

Another issue is in complex worlds with *too* many inputs, states, or outputs which creates many possibilities and result in big tree when thinking which demands high computation power and as a result makes the agent slow. Using smart techniques to reduce these computations is an undeniable challenge and an important part of the design. I've not mentioned all these techniques in this paper as they are considered implementation details.

Another, not issue, but difficulty, is to design the agent according to the environment–situation, concepts, feeling system, pattern & method extending, etc.–which needs a deep understanding of the problem.

As a mind, it might not be good for things that human mind is weak at. It might be slow and cause error. Basically a mind is good for problems that include many states and possibilities, with the solution unknown.