Artificial General Intelligence: Prototype

Mykola Rabchevskiy

April 20, 2017

This paper describes a prototype of an Artificial General Intelligence (AGI) system. The prototype intended for testing of the AGI architecture introduced in previous papers [1],[2]. Distinctive features of the tested architecture include:

- Permanent autonomous learning, including the ability to learn ”from zero”
- Generalized decision making that is independent of a particular system mission
- Controllable behavior of the system that is dependent on a system mission
- Introspection of the system knowledge using human-readable knowledge representation
- Ability to extract AGI system experience
- Injection of experience (but not skills) extracted from another compatible system or provided by a human as alternative form of learning
- Ability to explain the decision made by the system in a human-readable form

1 DEFINITIONS

The artificial intelligence (AI) terminology used in the articles is as follows:

Abstract symbol, symbol, logical entity, entity : used as synonyms.

Behavior : What the system tries to achieve and what tries to avoid. Usable system must have a stable and predictable behavior (note differences between behavior and skills).

Effector : an actuator or a sensor (embodied or virtual) that accepts a request in a symbolic form and produces a response in a symbolic form. A finite predefined set of abstract symbols (an alphabet) is used for data exchange between the effector and AGI core.

Event : a symbol used for data exchange between the effector and AGI core.

Event sequence, history sequence : a time sequence of events where each event is an atomic entity or a concept that represents some sequence of events that occurred twice or more. The event sequence represents temporal relations between entities. Non-temporal relations are stored in the directed graph (see [1] and [2] for more details).

Experience : what was done and what was happened, a content of the event sequence.

Generality : an attribute of the AI system that implies the ability to use the same core system for a different system missions. A high intelligence level is optional (in a contrast to a ***strong*** AI).

Skills : known ways to achieve the desired outcome. Skills can be improved as the system learns (note differences between skills and behavior).

State : a situation represented by a tail of an event sequence. The state can be either an atomic entity or a concept that denotes a discovered sequence of events that occurred at least twice. The state is similar to the concept of context.

Modus Operandi : one of a few ways to calculate criteria for comparison of consequences of actions at the decision making moment. For example, different criteria can be used for safe and dangerous environments.

2 ARCHITECTURE

The AGI system consists of the ***core unit*** that is independent of the system mission, the ***behavior unit*** that is dependent on the mission and actually defines it, and a set of ***effectors*** (physical or virtual) that are also dependent on the mission. Effectors fall into two categories: ***actuators*** that are intended either to change the system state or affect the external world and ***sensors*** that are intended to obtain some information about either the system state or the external world.

Effectors are mediators between the system core and external world. Any information about the environment (including the embodiment if any) obtained by the core or any action performed by the system is a result of the effectors request. Upon receiving the request, the effector generates a response that contains either requested data (in case of a sensor) or details about the performed action (in case of an actuator). Both the request and response are sequences of abstract symbols.

The AGI prototype uses the ‘active sensing‘ approach, in which any acquired information depends on the decision made by the system. The system uses a standard symbolic interface that simplifies integration of various sensors and actuators into the system.

The core unit includes knowledge storage that employs a directed graph and a set of hash tables to maintain all types of knowledge (concepts, relation, sequences of logical entities, routines, and rules). All knowledge is represented in a symbolic form, and any piece of knowledge can be modified independently from other pieces using the introspection interface. Dictionaries are used for translation between internal and external representations of logical entities. The external representation is human-readable. The symbolic representation provides the ability to introspect a system using the interface that presents the knowledge in a human-readable form. The introspection channel can be also used for modification of the knowledge set.

Collected knowledge includes the historic sequence that describes what happened in the past (mainly, what requests were sent and what responses were received). The approach uses the logical compression of the historic sequence. If some subsequence occurs twice or more, then a new concept is created to denote it. Next, the new concept replaces all occurrences of the subsequence in the experience. Such ‘logical compression‘ is used recursively and produces hierarchies of concepts that describe all known subsequences in the whole collected experience. The tail of the historic sequence represents the current ***state***. Since logical symbol may represent sequence of atomic events, such state is a sort of context for the decision making.

3 DECISION MAKING

The traditional AI‘ approach to decision making is to formulate a goal and then perform a TASK defined by the goal, that is, to seek a way to reach the goal and once it is found, act accordingly. This approach works well for narrow, specialized AI if all possible goals are members of a predefined set, and the activity ends when the goal is reached (or the goal is acknowledged as unreachable) until the next task is requested (the most obvious example being any game).

Since the general AI should act autonomously, similar to a human or an animal, its activity should be continuous, in which case the goal assigning by a human becomes optional. The presented system ***mission*** is a generalization of the goal concept. Mission is a source for decision making during continuous flow of activity. In this sense, the AGI similar to a classic control system (such as a temperature regulator, a cruise control, etc.) that has a mission but no tasks, only repetitive decision making about what to do at each moment. The mission can have parameters that depend on the current situation and human instructions. The mission of the AGI system should be defined in some way and requires an appropriate set of effectors. Directives issued by a human or other external sources fall into a few categories:

- The ***command*** that is just directly mapped to equivalent request to some effector (such as 'turn light off') and is part of the introspection mechanism. The set of commands includes the red button command that turns the system off.
- The ***task*** that is similar to the goal of a narrow AI and represented by a sequence of commands and sub-tasks (for example 'go home'). In case of a previously unknown task, the solution can be done by using both the traditional approaches and help from a trusted source ('master') that can suggest how to reduce the unknown task to a sequence of actions and known subtasks.
- The ***activity*** mode issued by the trusted source that sets system mission parameters (for example 'move North-West').

Note that the lack of external directives does not result in deactivation of the autonomous system. In such case, the system acts on its own.

The main challenge of AI system development is achieving expected system activity that matches the system mission and avoid an unwanted action. Obviously, regardless of details, the system must have the module that is able to evaluate how well the situation matches the system mission and detect unwanted or prohibited situations. Safety and reliability of the system dictates immutability of such an unit. The behavioral unit is actually an artificial version of the natural feeling system that indirectly defines system behavior, that is, the system "feels good" when the event flow matches the system mission and "feels terrible" in case of an unwanted situation.

The evaluation of possible future situations at decision making moment is a way to compare variants of action. The traditional approach to the decision making is to construct a plan that leads to the desired state. In case of a complex task, intermediate goals are used as a way to reduce a complex problem to a set of simpler problems. However, when the system is mission-driven, there is no terminal point. Using an easy-to-reach state that provides good feeling as the surrogate goal is obviously a poor approach because such decision may easily lead to the unavoidable bad state in the future. This circumstance suggests using the forecasting of the action consequences as a base for making decisions instead of creating a surrogate goal.

The one-step forecast is based on the accumulated experience (including a possible preloaded experience). The forecast maps each available action to a set of possible consequences using the collected experience. Applying the forecasting procedure recursively to each possible new future state produces a ***forecast tree*** of reachable states. Each possible future state can be evaluated by the behavioral unit. A current forecast tree provides information for decision making.

The depth of the forecast tree is obviously restricted by the computational power of the system and acceptable time for decision making. The more complex the system and environment are the more computational power is required. An accumulated past knowledge is stored using a compressed representation. Each element of the past history included in the forecast tree may represent a sequence of atomic events. The more familiar the AGI system becomes with the environment, the longer forecast horizon is, despite the restricted depth of the forecast three due to increasing the compression rate.

In general, the action leads to a few known possible consequences. Many AI approaches use the most probable consequence as a base for the decision, and therefore, less probable variants are simply ignored. This methodology has two drawbacks:

- The consequence with a low probability of happening may be more important than others with higher probabilities (such as the consequence that leads to the prohibited state is not acceptable despite a low probability)
- The only source for the probability estimation is the collected experience. However, such experience depends on decisions made in the past. Therefore the probability estimation based on the past experience reflects not only the external world, but also the system itself. These two factors are hard to separate, and resultant probability may be unstable (even asymptotically).

To avoid these drawbacks, the current implementation does not use the probability estimation. All possible consequences are treated as having unknown probabilities, so that the only assumption used is that "if something has happened once it may happen again".

Decision making using a specific forecast is choosing a specific action where each available action is mapped to a set of possible consequences. Each consequence is evaluated by the behavioral unit. Such evaluation produces a sequence of feelings that can be either empty or contains a single item.

The action selection requires comparing the ***sequences*** of the expected feelings. This means that despite the well-ranged ***scalar*** feelings (from worst to best), a few different criteria can be used to define which feeling ***sequence*** is the most preferable. Below are examples of possible criteria for selecting the optimal (most preferable) action:

- Action that maximizes the best possible future state
- Action that minimizes the worst possible future state
- Action with an empty forecast that guaranties the obtaining of a new experience

Consequently, depending on the current situation (mood) and the system life cycle, a few ***modi operandi*** with different selection criteria can be used.

For example, early stages of the system life may be dedicated to collecting an experience in a special safe environment using exploratory modus operandi, and conservative modus operandi is intended for a less safe environment.

Selection of the modus operandi is a function of the behavioral unit. The modus operandi also can be set using the introspection interface.

The main control loop consists of two phases: collecting information about the current state using sensors and performing an appropriate action using actuators. Extending collected experience using obtained information is a shadow process that coexists with both stages.

4 TESTING AGI MISSION

The test mission based on the virtual embodiment and environment. In the discussed test, the virtual world and a set of virtual effectors were simple enough for analysis yet sufficiently complex for testing the most significant aspects of the proposed approach. After analyzing a few variants, the following one was selected:

- The virtual world is discrete and is represented by a two-dimensional 32 x 32 array of squared cells that is wrapped in both directions (toroidal topology without borders).
- Each cell may be vacant or contain obstacles, such as rocks (inedible obstacles) or shrubs (edible obstacles).
- The mission of the AGI-equipped creature is to eat shrubs and avoid collisions with obstacles when moving within a specified area.
- The attempt to eat a rock, leads to system damage; whereas the attempt to eat a vacant cell leads to system distraction.
- When the creature swallows a shrub, the cell occupied by the shrub becomes vacant. Then a new shrub begins to grow in a randomly selected point. The world is mutable, but the number of shrubs remains constant.
- At the start of simulation, rocks and shrubs are distributed randomly over the terrain.

- System embodiment is a virtual creature equipped with four sensors that can detect what is located in four adjacent cells only. The system cannot obtain the information about its location using sensors.
- Actuators can be used to move into one of the four adjacent cells or swallow content of one of adjacent cells.
- The preloaded knowledge includes information about available actions (detecting and moving), but no information about the world.
- The behavioral unit produces feeling that evaluates a consequence of a performed action. The possible feeling values have 6 levels, ordered from worst to best: ***hurt, upset, wavering, calm, content, and happy***, with ***calm*** being a neutral feeling.
- Decision making uses four ***modi operandi***:
  - ***curious***: prefers action that expands the experience
  - ***dynamic***: maximizes the wanted consequences
  - ***prudent***: minimizes the unwanted consequences
  - ***venture***: is a hybrid between curious and dynamic.

A virtual creature starts with zero knowledge about the world and autonomously learns to live in a mutable environment. Monitoring the process was implemented using a web interface that displays the current map of the world, the information about the current decision and expected consequences, and statistical data about the process, such as the total number of generated concepts, the histogram of the length of discovered patterns, the percentage of each feeling for the last N steps, and so on (Fig.1).

The experimentation has confirmed the ability to achieve autonomous learning from zero using the described approach. The most effective autonomous learning was achieved by using the curious modus operandi on the initial stage and then switching to venture modus operandi after the system accumulated a sufficient amount of knowledge about the world. C++ was used for AGI system coding. The core system was implemented as a template class. The average time of decision making is around 1 millisecond by using single 64-bit core at 3.0GHz, so most of simulation time is consumed by web interface.

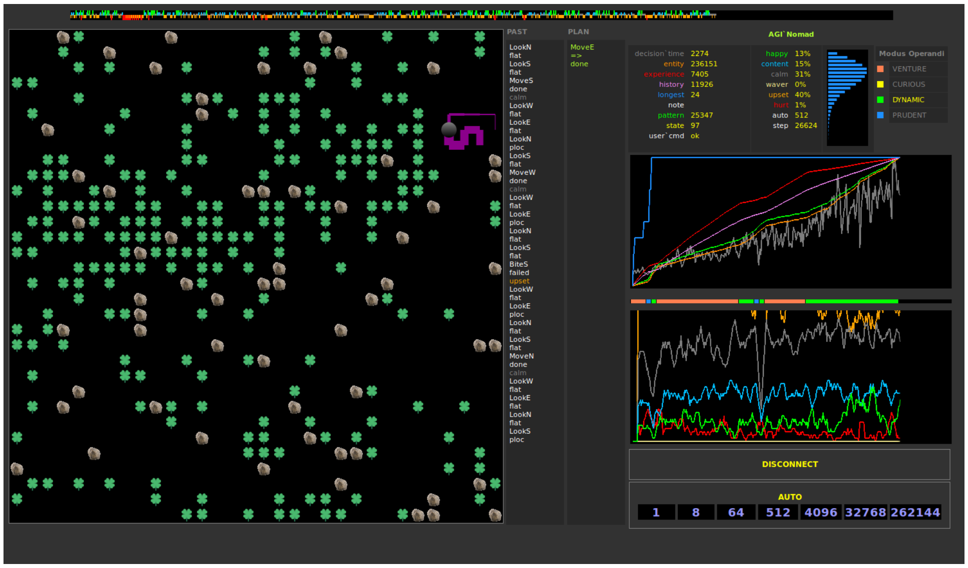

Figure 1: Introspection interface

5 FUTURE RESEARCH AND DEVELOPMENT

The results of the testing of the proposed AGI approach confirmed the claimed capabilities for a case of an AGI system with a simple mission that operates in a simple environment. Future tests should use more complex missions in order to evaluate the scalability of the approach and test features that have not yet been covered by previous tests, including usage of human-defined rules that restrict or improve the undesirable system behavior and extend skills.

REFERENCES

1. Mykola Rabchevskiy. Artificial General Intelligence: Engineering approach. 2010.

2. Mykola Rabchevskiy. Artificial General Intelligence: Blueprint. 2013.

