Where to go? What to do?
An Abstraction of Autonomous System Behavior

Compsim LLC

Abstract:

The questions: "Where to go? What to do?" can be used to encapsulate the abstract behavior of autonomous systems. This may be of interest if one is attempting to architect the capabilities of unmanned autonomous systems. "Where to go?" might even be included in the "What to do?" activities, but since "Where to go?" might generally be a task that is performed in pursuit of "What to do?", then it has been separated in this paper.

The "Autonomous Entities":

For this paper we are defining "Autonomous Entities" as objects that are expected to perform tasks without direct human control, yet are created by humans to perform those tasks. This definition includes virtual (software) agents, but excludes your pets, etc. This paper attempts to abstract how these Autonomous Entities pursue the goals that they have been assigned by humans, or subservient goals they derive themselves.

"Where to go":

For mobile entities (Unmanned Systems or USs), such as Unmanned Aerial Vehicles (UAVs), Unmanned Ground Vehicles (UGVs), or Unmanned Underwater Vehicles (UUVs), etc.), the "Where to go" problem is usually the preliminary part of any action. Subservient to "Where to go" is the "How to go" problem.

"What to do":

Now the key issues: "What to do" and then "How to do it". "What to do" is pretty simple for your (child's) electric train that runs around the circular track in your living room. "What to do" is almost a binary decision for your electric train. If it is provided with power it follows the track. If it is provided with too much power, it will fall off the track: The End. No power - No movement. "Where to go", if one wanted to define the reasoning for the electric train, would be to follow the track. This concept is pretty simple... "How to do" is only slightly more complex: given so much power; go so fast.

"What to do" for More Complex Systems:

This simple abstraction defines the constant problem that must be addressed by all autonomous systems. The problems are seldom as simple as the electric toy train. In this "digital age" the conventional approach has been to translate the problems into IF | THEN | ELSE logic. But maybe this is not the best approach.

If we allow our toy electric train the ability to exercise reason and judgment (and we forget about certain physical laws of nature), then we might define a policy for our little train with two goals:

1. Given power, move forward
2. If you go too fast, die (by jumping off the track)
    a. Determine if it is going too fast by balancing speed and center of gravity of the load (so you can go faster if you are carrying more weight, as long as it has a low center of gravity, or not as fast if the center of gravity is higher…)

Now your little train is balancing alternatives. It is balancing risk and reward.

Autonomous Entity Decisions:

The Unmanned Systems will be "manufactured" to pursue some kind of goals (their reasons for being manufactured). They will be expected to pursue those goals while observing some kind of "rules of engagement" or operational guidelines that give them some direction in applying their resources to pursue their goals. The Unmanned Systems will be able to observe their situation by absorbing sensor values and other information they are provided. They will have to interpret this information and combine it with their "rules of engagement". Internally they will be architected to support a hierarchy of decision-making actions to pursue the over-reaching goals.

As the industry attempts to integrate human-like behavior one might want to consider how humans process information in the pursuit of their changing goals. The human is constantly applying judgment and reasoning in order to interpret information and balance alternatives. Deciding to work, rest, eat, play is a continuous balancing act. How to work, how to play, etc. in order to balance resources, tools, funds… is also a decision to allocate resources in different ways.

The conventional practice of hard-coding (IF | THEN | ELSE logic) will not be practical to address these complex behaviors. The USs will have to balance the pros and cons of many optional approaches. Similarly they will have to balance risk and reward as they apply their own resources to address the "How to" issues: how fast, how hard, how much… Many, if not all, decisions and actions will require the application of "judgment" to most effectively respond to the complex balancing act that will be required.

Humans perform these types of problem solving actions in the right hemisphere of the brain. This parallel processing function allows the brain to balance short term and long term goals (each with their own optional decisions and actions) with their own risks and rewards. It would be difficult (or impossible) to script solutions to these problems with conventional IF | THEN | ELSE logic, even with the use of higher level mathematics.

Information Objects:

To solve judgmental types of problems, humans objectify information. These information objects are then quantified (valued for their importance) and then integrated with other information objects that have been quantified. Different options are selected from balanced collections of information objects.

Similarly, information objects are balanced in the process of allocating resources to define “how to perform”. Often the “how” (how much) decision is part of a feedback loop. When a human is driving a car and it starts to slide around a corner, the human evaluates the amount of slide and makes a correction to the heading, the throttle, and/or the brakes. The feedback takes place as the human observes the results and corrects the heading, throttle, and/or brakes. While this may appear to be continuous, it is just a repetitive process like any other feedback loop. For a human the amount of correction is based on experience. For a machine the correction is based on the “policy”. For legacy machines this might mean a PID loop.

So, to implement judgmental / adaptive capabilities into Unmanned Systems requires that the information will be objectified. In some cases this simply means that sensor data can be treated as an object. A speed sensor could provide one piece of information. For visual systems the problem is more complex. An image may contain many “objects”. For a human, before judgment is applied to the “big picture”, objects are extracted from the image. For industrial vision systems, a process called “feature extraction” is used to isolate “features” from an image. These features could be objects, or they could be combined to define objects. Once they are translated into objects they can then be used in the judgmental process. Size, distance, color, count, etc can all be used to qualify objects. In some cases the absence of an object can be treated as an object. For example a doorway can be identified as part of a path, or a gap in a shoreline can be treated as an object when trying to find a way into a harbor.

Two characteristics define theses information objects: they can be tagged with a name, and they can be valued. So a human would be able to describe these objects by giving them a name (a common name or a symbol). Then, within some context, the human would be able to describe an instantaneous importance level of that object. In the “real-world”, however, the instantaneous importance of information items is constantly changing. And the English language (or any other spoken or written language), is not effective in explicitly defining this rapidly changing information. This is an analog process that is carried out in the right hemisphere of the brain without requiring translation to a “language”.

In order for autonomous systems to exercise the adaptive behavior necessary to address the “where to go” / “what to do” problems, they will need to objectify information similar to how humans utilize the right hemisphere of the brain.

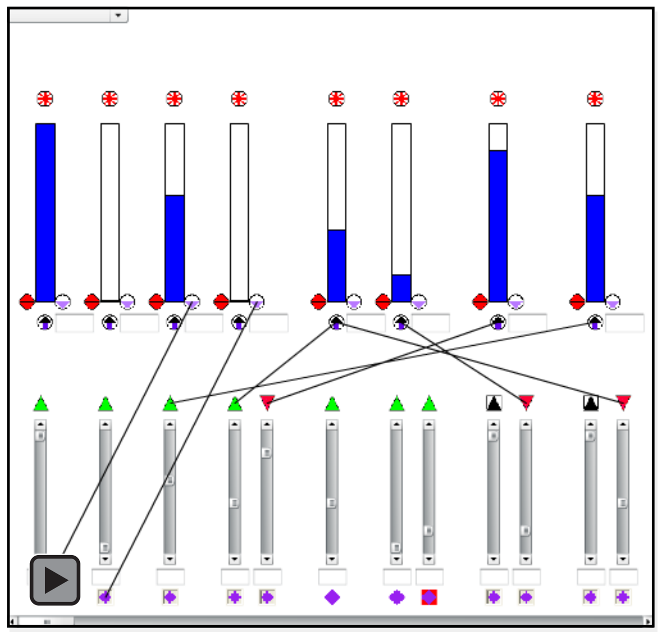
KEEL Technology Fuses Relative Information:

Compsim's Knowledge Enhanced Electronic Logic (KEEL) Technology provides the means to process dynamic, non-linear, inter-related, multi-dimensional problem sets. It provides a way to interpret the risk and reward alternatives by balancing their impacts in the decision-making process. Complex inter-relationships can easily be modeled without resorting to complex mathematical models.

The KEEL "Dynamic Graphical Language" (DGL) provides a way to describe policies that define how information is to be interpreted in a complex environment. The DGL is used to create KEEL Engines that can easily be integrated into almost any platform or architecture. The high performance, small memory footprint, KEEL Engines are suitable for the embedded, real-time systems that will be required for the USs.

The "Where to Go" and "What to Do" decisions are selection types of decisions that benefit from KEEL Technology when balancing options that have dynamically changing, weighted values. The "How to Go" and "How to Do" decisions are relative decisions that greatly benefit from the way that KEEL Engines balance information associated with risk and reward.

As embedded software functions, the KEEL Engines can be integrated into software agents for intelligent software systems



Interactive Content
Adjust Inputs
Input 0 for Train Speed
Input 1 for Center of Gravity for Load
Movie introducing the KEEL Dynamic Graphical Language

KEEL Technology (the processing model and the dynamic graphical language) is covered by granted US patents. KEEL Technology is only available from Compsim.

KEEL® (Knowledge Enhanced Electronic Logic) is a registered trademark of Compsim LLC. An introduction to KEEL technology can be found on the Compsim website (http://www.compsim.com) in the paper titled "Knowledge Enhanced Electronic Logic for Embedded Devices" in the Papers section. A more comprehensive discussion is provided in the paper titled "Decisions and Actions in KEEL" in the same section.

Compsim LLC is a technology company providing next generation cognitive technology for application in military, medical, transportation, industrial automation, governmental / business, and electronic gaming markets. Compsim licenses its KEEL® technology for use in embedded devices, software applications and for the Internet. The website is: http://www.compsim.com.

Compsim LLC
PO Box 532
Brookfield, Wisconsin 53008
(262) 797-0418