Features of Intelligence Required by Unmanned Ground Vehicles

James S. Albus
National Institute of Standards and Technology
Gaithersburg, MD 20899

ABSTRACT

A definition of intelligence is given in terms of performance that can be quantitatively measured. Behaviors required of unmanned ground vehicles are described and computational requirements for intelligent control at seven hierarchical levels in a military scout platoon are outlined. Metrics and measurements are suggested for evaluating the performance of unmanned ground vehicles. Calibrated data and test facilities are suggested to facilitate the development of intelligent systems.

KEYWORDS: intelligence, intelligent systems, unmanned ground vehicles, scout platoon, autonomous vehicles, metrics, measures

1. DEFINITIONS

The definition of intelligence is a controversial subject. Hardly any two persons define intelligence the same. Some even question whether intelligence can be defined at all. Yet, if we are to perform serious research on intelligent systems, we must not only be able to define intelligence, we must be able to quantitatively measure it. Thus, for the purpose of discussion of the issues addressed in this paper, we will define intelligence as follows [1]:

Df: **intelligence**

the ability to act appropriately in an uncertain environment

Df: **appropriate action**

that which maximizes the probability of success

Df: **success**

the achievement or maintenance of behavioral goals

Df: **behavioral goal**

a desired state of the environment that a behavior is designed to achieve or maintain

This definition of intelligence addresses both biological and machine embodiments. It admits a broad spectrum of behaviors, from the simple to the complex. We deliberately do not define intelligence in binary terms (i.e., this machine is intelligent and this one is not, or this species is intelligent and this one is not) and we do not limit our definition of intelligence to behavior that is beyond our understanding. Our definition includes the entire spectrum of intellectual capabilities from that of a paramecium to that of an Einstein, from that of a thermostat to that of the most sophisticated computer system. We include the ability of a robot to spot-weld an automobile body, the ability of a bee to navigate in a field of wild flowers, a squirrel to jump from limb to limb, a duck to land in a high wind, and a swallow to catch insects in flight above a field of wild flowers. We include the ability of blue jays to battle in the bushes for a nesting site, a pride of lions to conduct a coordinated attack on a wildebeest, and a flock of geese to migrate south for the winter. We include a human's ability to bake a cake, play the violin, read a book, write a poem, fight a war, or invent a computer.

Our definition of intelligence recognizes degrees, or levels, of intelligence. These are determined by the following parameters: 1) the computational power and memory capacity of the system's brain (or computer), 2) the sophistication of the processes the system employs for sensory processing, world modeling, behavior generation, value judgment, and communication, and 3) the quality and quantity of information and values the system has stored in its memory. The measure of intelligence is success in solving problems, anticipating the future, and acting so as to maximize the likelihood of achieving goals. Success can be measured by various criteria of performance (including life or death, pain or pleasure, reliability in goal achievement, cost in time and resources, and others.) Different levels of intelligence produce different probabilities of success.

Our definition of intelligence also has many dimensions. For example, the ability to understand what is visually perceived is qualitatively different from the ability to comprehend what is spoken. The ability to reason about mathematics and logic lies along a different dimension from the ability to compose music and verse. The ability to choose wisely involves both the ability to predict the future and the ability to accurately assess the cost or benefit of predicted future states. Along each of these dimensions, there exists a continuum. Thus, the space of intelligent systems is a multidimensional continuum wherein non-intelligent systems occupy a point at the origin.

At a minimum, intelligence requires the ability to sense the environment, to make decisions, and to control action. Higher levels of intelligence may include the ability to recognize objects and events, to represent knowledge in a world model, and to reason about and plan for the future. In advanced forms, intelligence provides the capacity to predict the future, to perceive and understand what is going on in the world, to choose wisely, and to act successfully under a large variety of circumstances so as to survive, prosper, and replicate in a complex, competitive, and often hostile environment.

From the viewpoint of control theory, intelligence might be defined as a knowledgeable "helmsman of behavior." Intelligence is a phenomenon which emerges as a result of the integration of knowledge and feedback into a sensory-interactive, goal-directed control system that can make plans and generate effective purposeful action to achieve goals.

From the viewpoint of psychology or biology, intelligence might be defined as a behavioral strategy that gives each individual a means for maximizing the likelihood of success in achieving its goals in an uncertain and often hostile environment. Intelligence results from the integration of perception, reason, emotion, and behavior in a sensing, perceiving, knowing, feeling, caring, planning, and acting system that can formulate and achieve goals.

2. REQUIREMENTS FOR UNMANNED GROUND VEHICLES

The features of intelligence required by an Unmanned Ground Vehicle (UGV) depends on many factors, such as:

What does the UGV have to do?

Does it simply wander through a lab looking for soft drink cans?

Does it have to operate outside? Travel long distances? Perform difficult tasks?

How complex and uncertain is the environment?

Where is it expected to operate? On well marked roads? On unmarked roads? Gravel or dirt roads? Roads grown up with weeds and brush? Off roads? In tall grass and weeds? In woods? Does it have to cross streams? Are there bridges or fords available? What kind of maps are available? How accurate are they? How recent?

How dynamic and hostile is the environment?

Are there moving obstacles? What are the lighting conditions? Are obstacles located above or below ground level? Are there other agents competing for the goal? Are there enemy agents with deadly weapons?

What are costs, risks, and benefits?

What are the stakes? Life or death? Win or lose?

What are goals?

Attack? Defend? Escape? Detect and track enemy targets? Remain undetected?

What are tasks?

Pick up an object? Use a tool? Dig a ditch? Cross a stream? Establish an observation post? Discover an enemy vehicle? Analyze enemy behavior? Identify a face in a crowd?

What sensors are available?

CCD cameras? FLIRs? LADARs? Radars? Sonars? Inertial? GPS? Beacons? Reflectors? Tactile? Force? Encoders?

What actuators are to be controlled?

Manipulators? Grippers? Power train? Legs or Wheels? Steering? Brakes? Switches?

How much is known apriori?

Maps? Lists of objects and their attributes? State of objects? Behavior of objects? Rules?

What skills and abilities are required?

Locomotion? Manipulation? Perception? Communication? Reasoning? Speech understanding? Written text understanding? In what languages?

The above questions are so open ended that it is futile to try to address all these issues simultaneously. To focus our efforts, we select an example of a problem that is difficult enough to be challenging, well defined enough to quantitatively measure performance, easy enough that it probably can be achieved using available technology, and useful enough that it is worth spending time and resources to solve it. The problem that we have selected it that of an unmanned ground vehicle for military scout operations.

3. A SCOUT PLATOON EXAMPLE

To illustrate the types of issues that will be addressed, an example is given below of a seven level hierarchy for a scout platoon attached to a battalion. The specific numbers and functions described in this example are illustrative only. They are meant only to illustrate how the generic structure and function of an intelligent system might be instantiated in the 4D/RCS architecture [2] designed for the Army's Demo III experimental unmanned ground vehicle program. [3]

Exact numbers for the actual system are still under development.

Level 7 -- Battalion

An armored battalion is a unit that consists of a group of M1 or Bradley companies and a scout platoon. A computational node at level 7 of the 4D/RCS architecture corresponds to a battalion headquarters unit, consisting of a battalion commander, several company commanders, a scout platoon leader, and support staff. (In principle, any or all of these could be humans or intelligent agent software processes. In practice, they are all humans.)

The battalion headquarters unit plans activities and allocates resources for the armored companies and the scout platoon attached to the battalion. Incoming orders to the battalion are decomposed by the battalion commander into assignments for the companies and the scout platoon. Resources and assets are allocated to each subordinate unit, and a schedule is generated for each unit to maneuver and carry out assigned operations. Together, these assignments, allocations, and schedules comprise a plan. The plan may be devised by the battalion commander alone, or in consultation with his subordinate unit leaders. The battalion level planning process may consider the exposure of each unit's movements to enemy observation, and the traversability of roads and cross-country routes. The battalion commander typically defines the rules of engagement for the units under his command and works with his unit leaders to develop a schedule that meets the objectives of the mission orders given to the battalion. In the 4-D/RCS battalion node, plans are computed for a period of about 24 hours(h) and recomputed at least once every 2 h, or more often if necessary. Desired positions for each of the subordinate units at about 2 h intervals are computed.

The 4D/RCS architecture provides a surrogate battalion node in each individual vehicle to perform the functions of the battalion headquarters unit when the vehicle is not in direct communication with its chain of command. The surrogate node plans activities for the vehicle on a battalion level time scale and estimates what platoon and section level operations should be executed to follow that plan. The surrogate battalion node considers the exposure of scout platoon operations to enemy observations, and the traversability of roads and cross-country routes.

In the surrogate battalion node in each vehicle, the 4-D/RCS world model maintains a knowledge database containing a copy of the battalion level knowledge database that is relevant to that vehicle. It contains names and attributes of friendly and enemy forces and of the force levels required to engage them. Maps have a range of 1000 km (i.e. more than the distance that a vehicle is likely to travel in a 24 h day at a Demo III speed of 36 km per hour (10 m/s)) with a resolution of about 400 m. Maps describe the terrain and location of friendly and enemy forces (to the extent that they are known), and roads, bridges, towns, and obstacles such as mountains, rivers, and woods. Battalion level maps may be updated from intelligence reports.

4-D/RCS sensory processing in the surrogate battalion node integrates information about the movement of forces, the level of supplies, and the operational status of all the units in the battalion, plus intelligence about enemy units in the area of concern to the company. This information is used to update maps and lists in the knowledge database so as to keep it accurate and current.

The surrogate battalion node also contains value judgment functions (e.g., calculating the risk of casualties) that enable the battalion commander to evaluate the cost and benefit of various tactical options. To the extent that the knowledge, skills, and abilities in the surrogate battalion node is identical with that in the real battalion node, the surrogate battalion node will make the same decisions as the real battalion headquarters node.

An operator interface allows human operators (either on-site or remotely) to visualize information such as the deployment and movement of forces, the availability of ammunition, and the overall situation within the scope of attention of the battalion commander. The operator can intervene to change priorities, alter tactics, or redirect the allocation of resources.

Output from the battalion level through the company commanders and scout platoon leader comprise input commands to the company/platoon level. Armor company commanders and the scout platoon leader are expected to issue commands to their respective units, monitor how well their units are following the battalion plan, and make adjustments as necessary to keep on plan. New output commands may be issued at any time, and typically consist of tasks expected to require about 2 h to complete.

Level 6—Platoon

A scout platoon is a unit that typically consists of ten HMMWVs or Bradley vehicles organized into one or more sections. For the Demo III project, a scout platoon will consist of six manned HMMWVs and four UGVs. A 4-D/RCS node at the Platoon level corresponds to a scout platoon headquarters unit. It consists of a platoon commander plus his/her section leaders. (Any of these could be humans or intelligent agent software processes, in any combination.) The platoon commander and section leaders plan activities and allocate resources for the sections in the platoon. Platoon orders are decomposed into job assignments for each section. Resources are allocated, and a schedule of activities is generated for each section. Movements are planned relative to major terrain features and other sections within the platoon. Inter-section formations are selected on the basis of tactical goals, stealth requirements, and other priorities. At the platoon level, plans are computed for a period of about 2 h into the future, and replanning is done about every 10 min, or more often if necessary. Section waypoints about 10 min apart are computed.

The surrogate platoon node in each vehicle performs the functions of the platoon headquarters unit when the vehicle is not in direct communication with the chain of command. It plans activities for the vehicle on a platoon level time scale and estimates what vehicle level maneuvers should be executed in order to follow that plan. Movements are planned relative to major terrain features and other vehicles within the platoon.

At the platoon level, the 4-D/RCS world model symbolic database contains names and attributes of targets, and the weapons and ammunition necessary to attack them. Maps with a range of about 100 km (i.e. more than the distance a platoon is likely to travel in 2 h) and resolution of about 40 m describe the location of objectives, and routing between them. Sensory processing integrates intelligence about the location and status of friendly and enemy forces. Value judgment evaluates tactical options for achieving section objectives. An operator interface allows human operators to visualize the status of operations and the movement of vehicles within the section formation. Operators can intervene to change priorities and reorder the plan of operations. Section leaders are expected to sequence commands to their respective sections, monitor how well their sections are following the platoon plan, and make adjustments as necessary to keep on plan. The output from the platoon level through the section leaders are commands issued to sections to perform maneuvers and engage enemy units in particular sectors of the battlefield. Output commands may be issued at any time, but typically are planned to change only about once every 5 min.

Level 5—Section

A scout section is a unit that consists of a group of individual scout vehicles such as HMMWVs and UGVs. A 4-D/RCS node at the section level corresponds to a section leader and vehicle commanders (humans or intelligent software agents). The section leader assigns duties to the vehicles in his section and coordinates the vehicle commanders in scheduling cooperative activities of the vehicles within a section. Orders are decomposed into assignments for each vehicle, and a schedule is developed for each vehicle to maneuver in formation within assigned corridors taking advantage of local terrain features and avoiding obstacles. Plans are developed to conduct coordinated maneuvers and to perform reconnaissance, surveillance, or target acquisition functions. At the section level, plans are computed for about 10 min into the future, and replanning is done about every 1 min, or more often if necessary. Vehicle waypoints about 1 min apart are computed.

The surrogate section node in each UGV performs the functions of the section command unit when the UGV is not in direct communication with the section commander. The surrogate node plans activities for the UGV on a section level time scale and estimates what vehicle level maneuvers should be executed in order to follow that plan.

At the section level, the 4-D/RCS world model symbolic database contains names, coordinates, and other attributes of other vehicles within the section, other sections, and potential enemy targets. Maps with a range of about 10 km and a resolution of about 30 m are typical. Maps at the section level describe the location of vehicles, targets, landmarks, and local terrain features such as buildings, roads, woods, fields, streams, fences, ponds, etc. Sensory processing determines the position of landmarks and terrain features, and tracks the motion of groups of vehicles and targets. Value judgment evaluates plans and computes cost, risk, and payoff of various alternatives. An operator interface allows human operators to visualize the status of the battlefield within the scope of the section, or to intervene to change priorities and reorder the sequence of operations or selection of targets. Vehicle commanders issue commands to their respective vehicles, monitor how well plans are being followed, and make adjustments as necessary to keep on plan. Output commands to individual vehicles to engage targets or maneuver relative to landmarks or other vehicles may be issued at any time, but on average are planned for tasks that last about 1 min.

Level 4—Individual vehicle

The vehicle is a unit that consists of a group of subsystems, such as locomotion, attention, communication, and mission package. A manned scout vehicle may have a driver, vehicle commander, and a lookout. Thus, a 4-D/RCS node at the vehicle level corresponds to a vehicle commander plus subsystem planners and executors. The vehicle commander assigns jobs to subsystems and schedules the activities of all the subsystems within the vehicle. A schedule of waypoints is developed by the locomotion subsystem to avoid obstacles, maintain position relative to nearby vehicles, and achieve desired vehicle heading and speed along the desired path on roads or cross-country. A schedule of tracking activities is generated for the attention subsystem to track obstacles, other vehicles, and targets. A schedule of activities is generated for the mission package and the communication subsystems. Waypoints and task activities about 5 s apart out to a planning horizon of 1 min are replanned every 5 s, or more often if necessary.

At the vehicle level, the world model symbolic database contains names (identifiers) and attributes of objects -- for example, the size, shape, and surface characteristics of roads, ground cover, or objects such as rocks, trees, bushes, mud, and water. Maps are generated from on-board sensors with a range of about 500 m and resolution of 4 meters. These maps are registered and overlaid with 40 meter resolution data from Section level maps. Maps represent object positions (relative to the vehicle) and dimensions of road surfaces, buildings, trees, craters, and ditches. Sensory processing measures object dimensions and distances, and computes relative motion. Value judgment evaluates trajectory planning and sensor dwell time sequences. An operator interface allows a human operator to visualize the status of operations of the vehicle, and to intervene to change priorities or steer the vehicle through difficult situations. Subsystem controller executors sequence commands to subsystems, monitor how well plans are being followed and modify parameters as necessary to keep on plan. Output commands to subsystems may be issued at any time, but typically are planned to change only about once every 5 s.

Level 3—Subsystem level

Each subsystem node is a unit consisting of a controller for a group of related Primitive level systems such as Primitive mobility, Gaze control, Communication, and Mission package sub-subsystems. A 4-D/RCS node at the Subsystem Level assigns jobs to each of its Primitive sub-subsystems and coordinates the activities among them. A schedule of Primitive mobility waypoints and Primitive mobility actions is developed to avoid obstacles. A schedule of pointing commands is generated for aiming cameras and sensors. A schedule of messages is generated for communications, and a schedule of actions is developed for operating the mission package sub-subsystems. The Primitive mobility way points are about 500 ms apart out to a planning horizon of about 5 s in the future. A new plan is generated about every 500 ms.

At the Subsystem level, the world model symbolic database contains names and attributes of environmental features such as road edges, holes, obstacles, ditches, and targets. Vehicle centered maps with a range of 50 meters and resolution of 40 cm are generated using data from range sensors. These maps represent the shape and location of terrain features and obstacle boundaries. The Demo III LADAR and stereo cameras measure position and range (out to about 50 m) of surfaces in the environment. Sensory processing computes surface properties such as dimensions, area, orientation, texture, and motion. Value judgment supports planning of steering and aiming computations, and evaluates sensor data quality. An operator interface allows a human operator to visualize the state of the vehicle, or to intervene to change mode or interrupt the sequence of operations. Subsystem executors compute at a 5 Hz clock rate. They sequence commands to primitive systems, monitor how well plans are being followed, and modify parameters as necessary to keep on plan. Output commands to Primitive sub-subsystems may be issued at any 200 ms interval, but typically are planned to change on average about once every 500 ms.

Level 2— Primitive level

Each node at the primitive level is a unit consisting of a group of controllers that plan and execute velocities and accelerations to optimize dynamic performance of components such as steering, braking, acceleration, gear shift, camera pointing, and weapon loading and pointing, taking into consideration dynamical interaction between mass, stiffness, force, and time. Communication messages are encoded into words and strings of symbols. Velocity and acceleration set points are planned every 50 ms out to a planning horizon of 500 ms.

The world model symbolic database contains names and attributes of state variables and features such as target trajectories and edges of objects. Maps are generated from camera data. Five meter maps have a resolution of about 4 cm. Driving plans can be represented by predicted tire tracks on the map, and visual attention plans by predicted fixation points in the visual field.

Sensory processing computes linear image features such as occluding edges, boundaries, and vertices and detects strings of events. Value judgment cost functions support dynamic trajectory optimization. An operator interface allows a human operator to visualize the state of each controller, and to intervene to change mode or override velocities. Primitive level executors keep track of how well plans are being followed, and modify parameters as necessary to keep within tolerance. Primitive executors compute at a 20 Hz clock rate. Output commands are issued to the Servo level to adjust set points for vehicle steering, velocity, and acceleration or for pointing sensors or weapons platforms. Output commands are issued every 50 ms.

Level 1—Servo level

Each node at the servo level is a unit consisting of a group of controllers that plan and execute actuator motions and forces, and generate discrete outputs. Communication message bit streams are produced. The servo level transforms commands from component to actuator coordinates and computes motion or torque commands for each actuator. Desired forces, velocities, and discrete outputs are planned for 20 ms intervals out to a planning horizon of 50 ms.

The world model symbolic database contains values of state variables such as actuator positions, velocities, and forces, pressure sensor readings, position of switches, and gear shift settings. Sensory processing detects events, and scales and filters data from individual sensors that measure position, velocity, force, torque, and pressure. Sensory processing also computes pixel attributes in images such as spatial and temporal gradients, stereo disparity, range, color, and image flow. An operator interface allows a human operator to visualize the state of the machine, or to intervene to change mode, set switches, or jog individual actuators.

Executors servo actuators and motors to follow planned trajectories. Position, velocity, or force servoing may be implemented, and in various combinations. Servo executors compute at a 200 Hz clock rate. Motion output commands to power amplifiers specify desired actuator torque or power every 5 ms. Discrete output commands produce switch closures and activate relays and solenoids.

The above example illustrates how the 4-D/RCS multilevel hierarchical architecture assigns different responsibilities and duties to various levels of the hierarchy with different range and resolution in time and space at each level. At each level, sensory data is processed, entities are recognized, world model representations are maintained, and tasks are decomposed into parallel and sequential subtasks, to be performed by cooperating sets of agents. At each level, feedback from sensors reactively closes a control loop allowing each agent to respond and react to unexpected events.

At each level, there is a characteristic range and resolution in space and time, a characteristic bandwidth and response time, and a characteristic planning horizon and level of detail in plans. The 4-D/RCS architecture thus organizes the planning of behavior, the control of action, and the focusing of computational resources such that functional processes at each level have a limited amount of responsibility and a manageable level of complexity.

4. DEMO III CONTROL HIERARCHY

Figure 1 is a high-level block diagram of the first five levels in the 4-D/RCS architecture for Demo III. On the right, Behavior Generation modules decompose high level mission commands into low level actions. The text beside the Planner and Executor at each level indicates the planning horizon, replanning rate, and reaction latency, and the rate at which new commands are typically generated at each level. Each planner has a world model simulator that is appropriate for the problems encountered at its level.

In the center of Figure 1, each map as a range and resolution that is appropriate for path planning at its level. At each level, there are symbolic data structures and segmented images with labeled regions that describe entities, events, and situations that are relevant to decisions that must be made at that level. On the left is a sensory processing hierarchy that extracts information from the sensory data stream that is needed to keep the world model knowledge database current and accurate.

At the bottom of Figure 1 are actuators that act on the world and sensors that measure phenomena in the world. The Demo III vehicles will have a variety of sensors including a laser range imager (LADAR), stereo CCD (charge coupled device) cameras, stereo forward looking infra red (FLIR) devices, a color CCD, a vegetation penetrating radar, GPS (Global Positioning System), an inertial navigation package, actuator feedback sensors, and a variety of internal sensors for measuring parameters such as engine temperature, speed, vibration, oil pressure, and fuel level. The vehicle also will carry a Reconnaissance, Surveillance, and Target Acquisition (RSTA) mission package that will include long-range cameras and FLIRs, a laser range finder, and an acoustic package.

In Figure 1, the bottom (Servo) level has no map representation. The Servo level deals with actuator dynamics and reacts to sensory feedback from actuator sensors. The Primitive level map has range of 5 m with resolution of 4 cm. This enables the vehicle to make small path corrections to avoid bumps and ruts during the 500 ms planning horizon of the Primitive level. The Primitive level also uses accelerometer data to control vehicle dynamics and prevent roll-over during high speed driving.

The Subsystem level map has range of 50 m with resolution of 40 cm. This map is used to plan about 5 s into the future to find a path that avoids obstacles and provides a smooth and efficient ride. The Vehicle level map has a range of 500 m with resolution of 4 m. This map is used to plan paths about 1 min into the future taking into account terrain features such as roads, bushes, gullies, or tree lines. The Section level map has a range of 5000 m with resolution of about 40 m. This map is used to plan about 10 m into the future to accomplish tactical behaviors. Higher level maps (not shown in Figure 1) are used to plan section and platoon missions lasting about 2 and 24 h respectively. These are derived from military maps and intelligence provided by the digital battlefield database.

4D/RCS planners are designed to generate new plans well before current plans become obsolete. Thus, action can always take place in the context of a recent plan, and feedback through the executors can close reactive control loops using recently selected control parameters. To meet the demands of Demo III, the 4D/RCS architecture specifies that replanning should occur within about one-tenth of the planning horizon at each level (e.g., replanning at the Vehicle level will occur about every 5 s.)

Executors can react to sensory feedback even faster (e.g., reaction at the Vehicle level will occur within 500 ms). If the Executor senses an error between its output CommandGoal and the predicted state (status from the subordinate BG Planner) at the GoalTime, it may react by modifying the commanded action so as to cope with that error. This closes a feedback loop through the Executor at that level within the specified reaction latency.

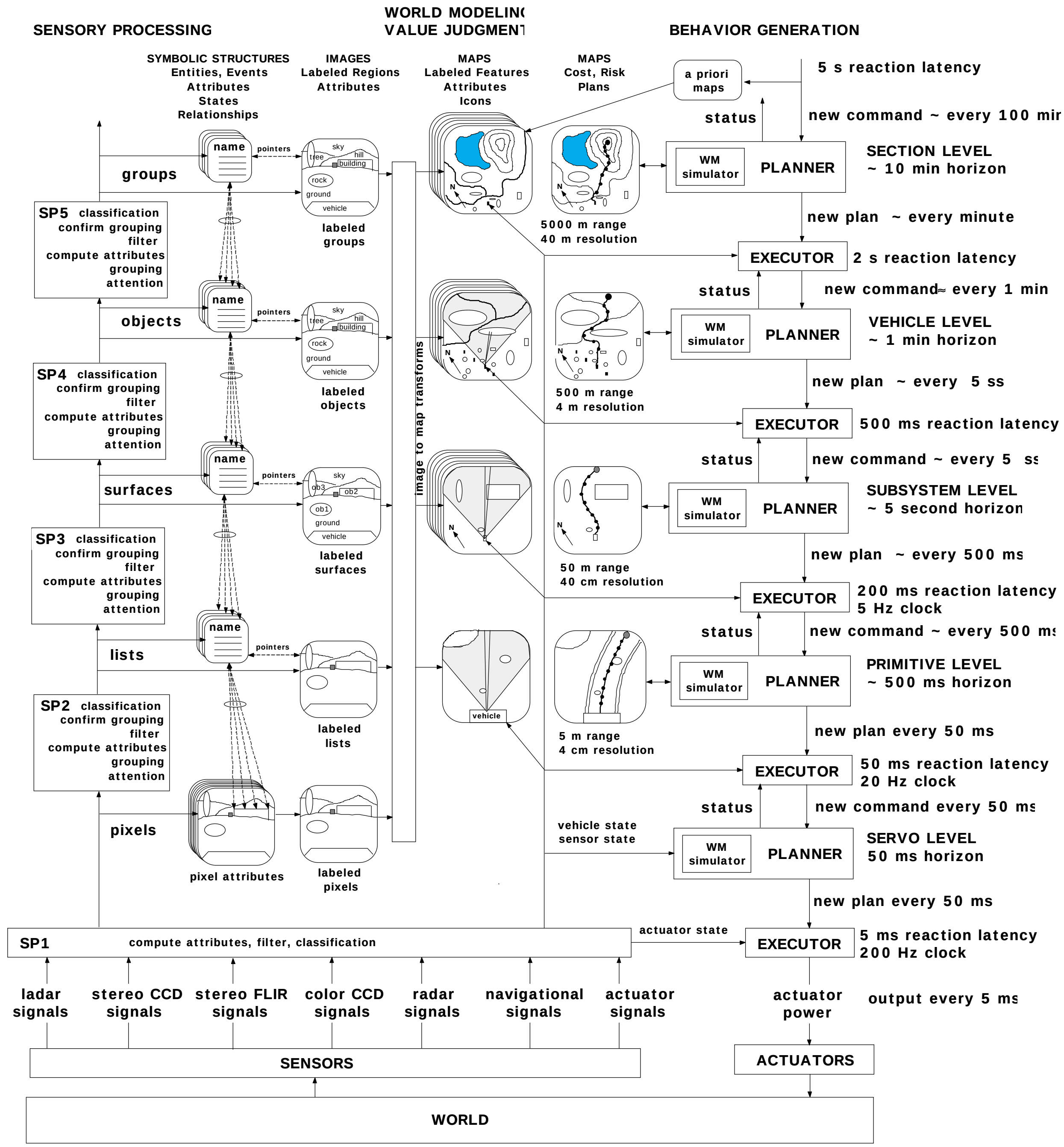

Figure 1. Five levels of the 4-D/RCS architecture. On the right are Planner and Executor modules. In the center are maps for representing terrain features, road, bridges, vehicles, friendly/enemy positions, and the cost and risk of traversing various regions. On the left are Sensory Processing functions, symbolic representations of entities and events, and segmented images with labeled regions.

The type of Executor reaction depends on the size and nature of the detected error. If the error is small, the Executor may simply modify its CommandedAction in a manner designed to reduce the error. For example, if the status reported from the subordinate planner indicates that the vehicle is going to arrive at the goal point late, the Executor might modify its CommandedAction to speed up or delete some low priority activities. However, if the error is out of range, the Executor may select a stored emergency plan from an exception handler, substitute it for the current plan, and modify its CommandedAction and CommandGoal to its subordinate planner appropriately. For example, an event such as the discovery of an unexpected obstacle in the AM planned path (generated by the Vehicle Planner) may cause the AM planner to make a plan that deviates significantly from its commanded goal. In this case, the Vehicle level Executor may modify its CommandedAction in a manner designed to buy time for the Vehicle level Planner to generate a new AM plan. For example, it may command the AM level to reduce speed or stop and direct AM driving cameras or RSTA sensors to collect information about the obstacle while a new AM plan is being generated by the Vehicle level planner. All of this Executor response should take place within the 500 ms reaction latency of the Vehicle level Executor.

Typically, evoking an emergency plan will cause the Executor to request its Planner to immediately begin a new replanning cycle. As shown in Figure 1, the period required for replanning at the Vehicle level is 5 s. The replanning period at the AM level is 0.5 s. Thus, the emergency plan evoked by the Vehicle level Executor can handle the problem of what the AM level should plan to do over the next 5 s while the Vehicle level planner generates a new AM plan out to its 1 min planning horizon.

5. GENERIC BEHAVIORS OF SCOUT VEHICLES

Navigate from A to B

Point A may be several km from point B. What kind of roads are available? How much traffic will be present? A scout vehicle may be required to stay off of roads, to maneuver through hilly fields and woods, and cope with fences, washes, and streams.

Avoid obstacles

The simplest obstacles are those that stick up from flat ground and are not obscured by foliage. The most difficult are ditches that are obscured by foliage. It is important to be able to distinguish grass and weeds that the vehicle can drive through from grass and weeds that conceal obstacles. In some cases, the only way to tell the difference is to drive slowly and stop when the vehicle encounters stiff resistance, or when the front wheels drop over the edge of a ditch, or sink into the mud.

Compute terrain attributes and classify terrain features

The first requirement is to map the terrain geometry and topology. The second is compute attributes such as color, texture, slope, size, and shape of regions of terrain. The third is to compare attributes of terrain regions with class attributes so as to classify terrain regions as road, dirt, grass, rocks, brush, trees, and bogs.

Drive autonomously

Driving autonomously covers a wide range of situations. Driving on an empty freeway is quite different from driving in downtown Istanbul. Driving with traffic on a freeway requires the ability to recognize lane markings, detect and track other vehicles, detect and avoid obstacles in the roadway, and obey road signs.

Driving at normal human speeds on narrow roads and cross country is more difficult. Road edges may be poorly defined and lane markings often do not exist. There may be bumps or ditches that will damage the vehicle if struck at high speeds.

Autonomous driving in suburban or downtown streets requires the ability to detect and predict the behavior of pedestrians, other vehicles, to read road signs, and respond to traffic signals, including hand signals from humans.

In driving cross country, there is no guarantee that a chosen path is even feasible. There may be hidden obstacles such as ditches, streams, fences, hills, brush, or woods that are impassable. The vehicle must be able to back up, and try alternate routes when the planned path is blocked.

Classify landmarks, objects, places, and situations

It is easy to get lost. GPS is not always available. Critical path waypoints may not appear on a map, or may be incorrectly represented. The unexpected appearance of an enemy may require immediate action. The ability to recognize a likely spot for an enemy sniper in time to take evasive action may be critical to survival.

Recognize and track other vehicles, avoid collisions

On-coming traffic on narrow roads is a major problem. One must drive very close to oncoming vehicles to stay on the road. One must estimate whether the oncoming vehicle is in its own lane on its own side of the road, and whether there is room on the road for two vehicles to safely pass. To do that one must detect the road edges at a great distance and measure the relative position of the on-coming vehicle between the road edges. There is very little margin for error in space or time.

Predict behavior of pedestrians and other vehicles in traffic

Driving in traffic requires the self vehicle to not only detect, but to predict where pedestrians and other vehicles will be in the future. For example, on a two lane road, on-coming traffic may consist of one vehicle passing another. The self vehicle must predict whether the on-coming vehicle in the self vehicle lane will return to its own side before a head-on collision occurs. On a one lane road, it may be necessary for the self vehicle to pull over and let an on-coming vehicle pass, or wait for the on-coming vehicle to pull over so that the self vehicle can pass. On a narrow mountain road, it may be necessary to back up to a place where it is wide enough for two vehicles to pass each other.

Learn from experience and from human instructors

Adjust behavior to situation and priorities. Use reward and punishment from human instructors to learn skills and behaviors. Use experience from multiple simulated scenarios to learn from experience.

6. METRICS AND MEASURES

A metric is a unit of measure. Examples include the meter, the second, the kilogram, the volt, Plank's constant, and Avogadro's number.

Measurements are made by comparing something against the unit of measure. A measurement can be made of the length of the coastline of the British Isles, the height of the Eiffel Tower, the mass of the Queen Mary, the length of a day, or the charge on an electron. There are many parameters related to measurement including accuracy, precision, resolution, observability, and uncertainty.

What is it about intelligent systems that can be measured? If an intelligent system is defined as a system with the ability to act appropriately in an uncertain environment, then we can measure the appropriateness of its behavior. And, if appropriate behavior is defined as that which increases the likelihood of achieving a goal, then the ability of a system to achieve goals in an uncertain environment is a measure of intelligence.

At least three things are required to measure the ability of a system to achieve goals. First, we need to define the goals and set criteria for achieving them. Second, we need to provide an environment in which to make the measurements. Third, we need to define a procedure for scoring performance that takes into account the difficulty of the goals, and the complexity and uncertainty of the environment

What kinds of measurements can be used to measure performance? One possibility is to develop one or more benchmark tests, and measure speed, accuracy, efficiency, level of difficulty, and cost. These measurements can then be weighted for importance and summed to provide an overall score.

Another approach is to devise competitions wherein different intelligent systems can compete against each other for a score. Competitions can involve direct physical interactions such as in football or tennis, measurements of time as in skiing or bobsleding, or competitions that consider both style and difficulty as in ice skating, diving, and gymnastics. Again, performance measurements can be weighted for importance and summed to provide a score.

What kind of metric can be used to measure the performance of an intelligent systems? One possible metric is the performance of a human being. Another possible metric is the performance of a standard baseline system. In either case, the performance of the intelligent system under test can be compared with the performance of a human being (or baseline system) under similar conditions. The difference in performance, the level of difficulty of the test, and the weighting for importance of the test all combine to give a score.

Measures of performance can be devised for subsystem performance, individual system performance, or group or team performance. For example, for subsystems, benchmark tests can be devised to measure the performance of sensory processing algorithms, world model predictors, or behavior generation planners. One might measure the difference between predictions and observations, or the difference between plans and actions. Benchmark tests can also be devised to measure the accuracy of knowledge about the world. For example, one can measure the difference between perceived terrain geometry derived from sensors and ground truth from calibrated test courses. One can measure the latency between requesting and receiving information about the world. Individual system performance can be measured and scored against standard tasks that are typically required of human scout vehicles. Similarly, team performance can be measured and scored in war games wherein opposing forces are tested in battle fighting scenarios.

What is needed?

Calibrated test facilities are needed to test the performance of sensors and systems in the field under realistic conditions. High fidelity simulation facilities are needed to generate repeatable test data for software debugging and testing. Data from calibrated sensors, mixed with a known noise, and accompanied by ground truth are needed to test sensory processing and world modeling algorithms. World model data with values assigned to entities and events is needed to test behavior generation planning and control algorithms.

Large scale test and training facilities are needed to test performance of systems in large scale operations and to develop tactics and training for integration of autonomous systems with manned forces. A wide variety of benchmark tests and competitions are needed to test intelligent system performance under a wide variety of environmental conditions. A rigorous regimen of testing, debugging, and reliability engineering will be needed before intelligent systems become robust enough to operate reliably under a wide variety of operational conditions.

6. REFERENCES

[1] Albus, J.S., "Outline for a Theory of Intelligence," IEEE Transactions on Systems, Man and Cybernetics, Vol. 21, No. 3, pgs. 473-509, May/June, 1991

[2] Albus, J. *4D/RCS: A Reference Model Architecture for Demo III, Version 1.0, NISTIR 5994,* Gaithersburg, MD, 1998

[3] Shoemaker, C., Bornstein, J., Myers, S., and Brendle, B. "Demo III: Department of Defense testbed for unmanned ground mobility," *SPIE Conference on Unmanned Ground Vehicle Technology, SPIE Vol. 3693*, Orlando, FA, April, 1999