A Hierarchy of Qualitative Representations for Space *

Benjamin Kuipers
Computer Science Department
University of Texas at Austin
Austin, Texas 78712 USA

Abstract

Research in Qualitative Reasoning builds and uses discrete symbolic models of the continuous world. Inference methods such as qualitative simulation are grounded in the theory of ordinary differential equations. We argue here that cognitive mapping — building and using symbolic models of the large-scale spatial environment — is a highly appropriate domain for qualitative reasoning research.

We describe the *Spatial Semantic Hierarchy* (SSH), a set of distinct representations for space, each with its own ontology, each with its own mathematical foundation, and each abstracted from the levels below it. At the control level, the robot and its environment are modeled as a continuous dynamical system, whose stable equilibrium points are abstracted to a discrete set of "distinctive states." Trajectories linking these states can be abstracted to actions, giving a discrete causal graph level of representation for the state space. Depending on the properties of the actions, the causal graph can be deterministic or stochastic. The causal graph of states and actions can in turn be abstracted to a topological network of places and paths. Local metrical models, such as occupancy grids, of neighborhoods of places and paths can then be built on the framework of the topological network while avoiding their usual problems of global consistency.

This paper gives an overview of the SSH, describes the kinds of guarantees that the representation can support, and gives examples from two different robot implementations. We conclude with a brief discussion of the relation between the concepts of "distinctive state" and "landmark value."

The Spatial Semantic Hierarchy

Building on recent progress in robot exploration and map-building, we propose an *ontological hierarchy* of representations for knowledge of large-scale space.

An ontological hierarchy shows how multiple representations for the same kind of knowledge can coexists. Each level of the hierarchy has its own *ontology* (the set of objects and relations it uses for describing the world) and its own set of inference and problem-solving methods. The objects, relations, and assumptions required by each level are provided by those below it.

The dependencies among levels in the hierarchy help clarify which combinations of representations are coherent, and which states of incomplete knowledge are meaningful.

In this paper, we formalize the computational model of the cognitive map as developed by Kuipers and his students (Kuipers 1978; Kuipers & Byun 1988; 1991). That theory was motivated by two insights from observations of human spatial reasoning skills and the characteristic stages of child development (Lynch 1960; Piaget & Inhelder 1967; Hart & Moore 1973). First, a *topological* description of the environment is central to the cognitive map, and is logically prior to the metrical description. Second, the spatial representation is grounded in the sensorimotor interaction between the agent and the environment.

The *Spatial Semantic Hierarchy* (SSH) (Kuipers & Levitt 1988) abstracts the structure of an agent's spatial knowledge in a way that is relatively independent of its sensorimotor apparatus and the environment within which it moves. The following informally describes the knowledge at the different SSH levels, which will be described more formally below.

- The *sensorimotor system* of the robot provides continuous sensors and effectors, but no direct access to the global structure of the environment, or the robot's position or orientation within it.
- At the *control level* of the hierarchy, the ontology is an egocentric sensorimotor one, without knowledge of fixed objects or places in an external environment. A *distinctive state* is defined as the local maximum found by a hill-climbing control strategy, climbing the gradient of a selected sensory feature, or *distinctiveness measure*. Trajectory-following control laws take the robot from one distinctive state to the neighborhood of the next, where hill-climbing can

*This work has taken place in the Qualitative Reasoning Group at the Artificial Intelligence Laboratory, The University of Texas at Austin. Research of the Qualitative Reasoning Group is supported in part by NSF grants IRI-9216584 and IRI-9504138, by NASA contract NCC 2-760, and by the Texas Advanced Research Program under grant no. 003658-242.

find a local maximum, reducing position error and preventing its accumulation.

- At the *causal level* of the hierarchy, the ontology consists of *views*, which describe the sensory images at distinctive states, and *actions*, which represent trajectories of control laws by which the robot moves from one view to another. A causal graph of associations $\langle V, A, V' \rangle$ among views, actions, and resulting views represents both declarative and imperative knowledge of routes or action procedures.
- At the *topological level* of the hierarchy, the ontology consists of *places*, *paths*, and *regions*, with connectivity and containment relations. Relations among the distinctive states and trajectories defined by the control level, and among their summaries as views and actions at the causal level, are effectively described by the topological network. This network can be used to guide exploration of new environments and to solve new route-finding problems. Using the network representation, navigation among distinctive states is not dependent on the accuracy, or even the existence, of metrical knowledge of the environment.
- At the *metrical level* of the hierarchy, the ontology for places, paths, and sensory features is extended to include metrical properties such as distance, direction, shape, etc. Geometrical features are extracted from sensory input, and represented as annotations on the places and paths of the topological network.

Two fundamental ontological distinctions are embedded in the SSH. First, the continuous world of the sensorimotor and control levels is abstracted to the discrete symbolic representation at the causal and topological levels, to which the metrical level adds continuous properties. Second, the egocentric world of the sensorimotor, control, and causal levels is abstracted to the world-centered ontologies of the topological and metrical levels.

Formalizing the levels of the hierarchy draws on different bodies of relevant theory: the sensorimotor and control levels on control theory and dynamical systems; the causal level on logic and stochastic transition models; the topological level on logic and simple topology; the geometrical level on estimation theory and differential geometry.

The Spatial Semantic Hierarchy approach contrasts with more traditional methods, which place geometrical sensor intepretation (the most expensive and error-prone step) on the critical path prior to creation of the topological map (Chatila & Laumond 1985; Moravec & Elfes 1985). The SSH is consistent with, but more specific than, Brooks' (1986) subsumption architecture, particularly levels 2 and 3.

The SSH representational framework has been implemented on several different simulated and physical robots. Figure 1 (modified from (Kuipers & Byun 1991)) shows how the control level definition of states and trajectories grounds the topological description of places and paths, which in turn supports exploration and planning while more expensive sensor fusion methods accumulate metrical information. When metrical information is available, it can be used to optimize travel plans or to disambiguate apparently identical places, but when it is absent navigation and exploration remain possible. Figure 2 demonstrates a fragment of behavior of an RWI B12 robot, using a ring of 12 sonar sensors, as it follows control laws and identifies a distinctive place in the indoor office environment.

The Sensorimotor Level

The robot has an objective location in the environment, but it does not have direct access to a representation of that location in an absolute frame of reference. Assume that the environment is two-dimensional, so that the *state* of the robot has three dimensions: position (x, y) and orientation θ. The vector of state variables is $\mathbf{x} = [x, y, \theta]^T$. The robot also has a memory M including symbolic descriptions of goals, beliefs, etc., which can influence the choice of control law, hence behavior.

The robot has a vector of sensors providing input $\mathbf{s} = [s_0, \ldots s_{n-1}]^T$ and a vector of motor outputs $\mathbf{u} = [u_0, \ldots u_{k-1}]^T$ by which it can change its position in the environment.

The sensor values are a function of the robot's state,

$$[s_0, \ldots, s_{n-1}]^T = \mathbf{s} = \Psi(\mathbf{x}) = \Psi(x, y, \theta). \quad (1)$$

All variables are piece-wise continuous functions of time. This model treats the environment as static, with the only changes being to the robot's position and orientation.

The "physics of the environment" (or dynamics of the robot),

$$[\dot{x}, \dot{y}, \dot{\theta}]^T = \dot{\mathbf{x}} = \Phi(\mathbf{x}, \mathbf{u}) = \Phi(x, y, \theta, u_0, \ldots u_{k-1}) \quad (2)$$

specifies how the state, and hence the sensory values, change with time as a function of the current state and the motor outputs. The robot does not have direct access to its state variables, but only to the sensory information $\mathbf{s}(t)$ provided to it as it moves through the environment.

The Control Level

The purpose of the SSH control level is to select and execute control laws for travel through the environment. During exploration, locally well-behaved features of the sensory input are identified and used to construct suitable control laws. During travel through a known environment, control laws are retrieved from the causal level of the cognitive map.

During a particular segment i of reactive behavior, the robot moves through the environment by setting its motor vector in response to its sensory inputs, according to a control law χ_i.

$$[u_0, \ldots, u_{k-1}] = \mathbf{u} = \chi_i(\mathbf{s}) = \chi_i(s_0, \ldots s_{n-1}) \quad (3)$$

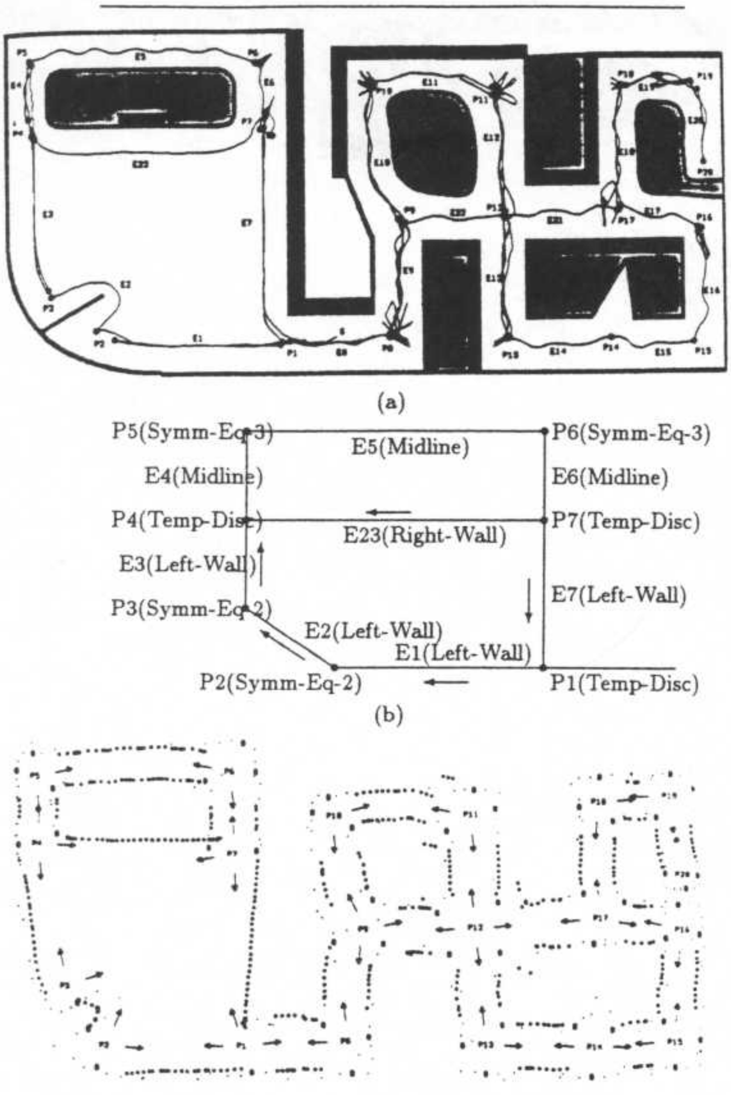

Figure 1: Simulated NX robot applies SSH exploration and mapping strategy.

(a) The simulated NX robot uses range-sensors to explore and map an environment (Kuipers & Byun 1991). The exploration and control strategies identify random and systematic sensor errors, and thus provide robustness. (b) The topological map (fragment) identifies places and paths with the distinctiveness measures that define them (e.g., equidistance from nearby obstacles, discontinuous sensor changes), and represents their connectivity relations. (c) The metrical map consists of annotations on each place and path, and can be relaxed into a global 2D frame of reference.

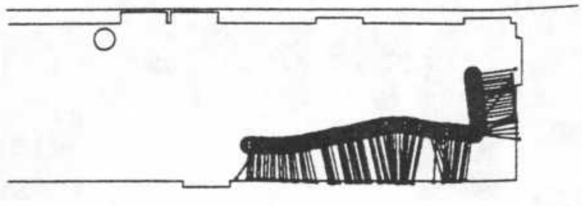

Figure 2: Spot, a physical robot, applying SSH control strategies.

Spot moves along a right wall, identifies a distinctive place, turns left, and begins to follow the next wall. This figure plots the position of the robot and the single most relevant sonar reading, on a map of the corridor it is exploring.

Although χ_i is purely reactive (i.e., determined by $\mathbf{s}$), its selection depends both on the currently perceived environment, and on goals and other aspects of the robot's state not described at the control level. For example, sitting at an intersection, the robot's goals determine whether to invoke a control law for a left or right turn, or to continue straight. For a given choice of control law χ_i, equations (1), (2), and (3) define a dynamical system that describes the behavior of the robot interacting with its environment.

Distinctiveness Measures

A critical step in our approach is the identification of a discrete set of *locally distinctive states* within a continuous state-space. A locally distinctive state can be defined in terms of the behavior of a control law if we can identify a continuous *distinctiveness measure* with an isolated local maximum in the current neighborhood.

A *distinctiveness measure* (or "d-measure") d is a continuous function $d(\mathbf{s}) \rightarrow \Re$. The set $D = \{d_0, \dots d_{m-1}\}$ of distinctiveness measures depends on the environment and sensorimotor system of the particular robot. A d-measure can be used to define a point-like distinctive state, such as the state equidistant from three obstacles and oriented midway between two of them; or a path-like trajectory, such as the midline of a corridor. Pierce and Kuipers (1994) show how d-measures and control laws can be learned from unguided experience.

Each d-measure d has an *appropriateness measure* $a_d(\mathbf{s}) \rightarrow [0, 1]$ that specifies the degree to which d is useful for control. a_d need not be continuous, and it may depend on goals or other aspects of the robot's state, as well as the sensory input stream $\mathbf{s}(t)$ to the robot. It is sometimes useful to think of a d-measure d as having a prerequisite $\pi_d(\mathbf{s})$ which must be true for d to be defined. This can be subsumed by the appropriateness measure: $\pi_d(\mathbf{s}) \equiv a_d(\mathbf{s}) > \epsilon$, for some user-specified $\epsilon \geq 0$.

A *neighborhood* $nbd(d)$ of the distinctiveness measure d is a connected subset of the set of states where $\pi_d(\mathbf{s})$

is true. That is, a given distinctiveness measure may have several disconnected neighborhoods in different parts of the environment.

Local Control Laws

Navigation at the control level is an alternation between two different types of control laws: *hill-climbing* control laws to reach a nearby local maximum of a d-measure, and *trajectory-following* control laws to move from one part of the state space to another. One way to express these is through a simple but general local control law associated with a given d-measure d, specifying a direction of change in the state space of the robot:

$$\dot{\mathbf{x}} = [\dot{x}, \dot{y}, \dot{\theta}]^T = k_1 \nabla d + k_2 N_d \tag{4}$$

where ∇d is the gradient of d in the state space, and N_d is a unit vector orthogonal to ∇d.

Since more than one d-measure $d \in D$ may be appropriate during a given trajectory, we take the weighted average in the spirit of heterogeneous control (Kuipers & Åström 1994):

$$\dot{\mathbf{x}} = \frac{\sum_{d \in D} a_d(t)\,[k_1 \nabla d + k_2 N_d]}{\sum_{d \in D} a_d(t)} \tag{5}$$

where $a_d(t)$ is an *appropriateness measure* for d. When d is not meaningful, $a_d(t) = 0$. Note that as the robot moves, the effective number of participating local control laws may change.

Other compositional approaches to control include potential field methods (Arkin 1989; Slack 1993) and fuzzy control (Mamdani 1974; Kosko 1992). Appropriateness measures and other parameters of the control laws χ_i may be acquired and optimized by function-learning methods including neural nets (e.g. (Pomerleau 1993)) and memory-based learning (Atkeson, Moore, & Schaal 1996; Moore, Atkeson, & Schaal 1996).

Hill-Climbing. Starting in the state where a trajectory-following control law terminates, identify the applicable hill-climbing d-measure(s). For a hill-climbing control law of the form (4), the $k_1 \nabla d$ term points the robot toward the local maximum, and $k_2 = 0$. A *distinctive state* $\langle x, y, \theta \rangle$ is the state of the robot when a hill-climbing control law terminates; i.e., when $\dot{\mathbf{x}} = \nabla d = 0$.

Trajectory-Following. Starting at a locally distinctive state, select and obey a trajectory-following control law (or a sequence of local control laws) until it terminates. For a trajectory-following control law of the form (4), the term $k_1 \nabla d$ keeps the robot on the desired trajectory, and $k_2 N_d$ moves it along the trajectory in one direction or the other depending on the sign of k_2. A trajectory-following control law terminates when $\dot{\mathbf{x}}$ changes discontinuously (or very quickly). In equation (5) this would happen when some $a_d(t)$ suddenly becomes zero while the corresponding control action $[k_1 \nabla d + k_2 N_d]$ is non-zero.

For example, a robot starts at one end of a corridor, facing "open space." It takes a trajectory consisting of open-loop motion into the corridor it faces, then following the midline to the end of the corridor. Upon reaching the end, the robot does hill-climbing to position itself equidistant from nearby obstacles.

Putting Control into Action

The local control law (5) provides a desired direction of motion $\dot{\mathbf{x}}$ in state space, which must be translated into values for the robot's motor output variables $\mathbf{u}$.

In simple cases, the dynamics of the robot (equation (2)) will have a pseudo-inverse Φ^{-1} so that, given $\mathbf{x}$ and a desired $\dot{\mathbf{x}}$, we can directly compute

$$\mathbf{u} = \Phi^{-1}(\mathbf{x}, \dot{\mathbf{x}}) \text{ such that } \dot{\mathbf{x}} = \Phi(\mathbf{x}, \mathbf{u}). \tag{6}$$

In general (i.e., for a robot with non-holonomic motion constraints), there may be no way to achieve a desired $\dot{\mathbf{x}}$ for a given state $\mathbf{x}$ (cf. (Latombe 1991)). In such a case, we specify the control goal as a net change $\Delta\mathbf{x}$ to be obtained over some period of time. Then we assume the ability to plan a sequence of continuous actions (e.g., (Penberthy & Weld 1994)), or to retrieve a previously developed control plan:

$$p = plan(\mathbf{x}, \Delta\mathbf{x}), \text{ such that } \mathbf{u} = p(\mathbf{x}, t) \tag{7}$$

has the desired effect of reaching the state $\mathbf{x} + \Delta\mathbf{x}$. Note that, as with parallel parking, the intermediate states of the plan p may be farther from the goal than the initial or final states. Further extensions will be required to cope with pedestrians and other unexpected obstacles.

Equation (5), along with either (6) or (7), provides an instance of the control law χ_i required by equation (3). Thus, the robot's behavior *during a single hill-climbing or trajectory-following segment* consists of the state-evolution of a particular dynamical system. Higher-level symbolic reasoning intervenes at the joints between these segments to determine which dynamical system controls the behavior.

The Causal Level

When a sequence of control laws — trajectory-following then hill-climbing — reliably takes the robot from one distinctive state to another, we abstract the sequence of control laws to an action A, and the two distinctive states to the sensory images, or views, V and V', obtained there. Their association is represented by the schema $\langle V, A, V' \rangle$.

When this abstraction can be applied across the environment, the continuous state space in which the robot is described as following the trajectories of a dynamical system is abstracted to a discrete state space in which the robot is described as performing a sequence of discrete actions.

Views, Actions, and Schemas

A *view* is a description of the sensory input vector $\mathbf{s}(t) = [s_1(t), \ldots s_n(t)]$ obtained at a locally distinctive state, $\langle x, y, \theta \rangle$. A view could be a complete snapshot of $\mathbf{s}(t)$, or it could be a partial description, consistent with more than one value of $\mathbf{s}$.

An *action* denotes a sequence of one or more control laws which can be initiated at a locally distinctive state, and terminates after a hill-climbing control law with the robot at another distinctive state. A typical action might consist of an open-loop trajectory-following control law to escape from the current neighborhood, then a closed-loop trajectory-following control law to reach a new neighborhood, and finally a hill-climbing control law to reach a new distinctive state.

A *schema* is a tuple $\langle V, A, V' \rangle$, representing the (temporally extended) event in which the robot takes a particular action A, starting with view V and terminating with view V'.

In the following, $holds(V, s_0)$ means that the view V is observed in situation s_0; $do(A, s_0)$ means that action A is initiated in situation s_0; and $result(A, s_0)$ denotes the situation resulting after action A is initiated in situation s_0 and terminates in a new distinctive state. The schema $\langle V, A, V' \rangle$ has two meanings:

$$\begin{array}{ll} \text{declarative:} & holds(V, s_0) \rightarrow holds(V', result(A, s_0)) \\ \text{imperative:} & holds(V, now) \Rightarrow do(A, now). \end{array}$$

The declarative meaning is standard situation calculus (McCarthy & Hayes 1969). The imperative meaning is intuitively clear, but not formalized.

Procedurally, in order for a complete schema $\langle V, A, V' \rangle$ to be created from observations during behavior, the partially filled schema $\langle V, A, nil \rangle$ must be preserved in working memory during the time required to carry out the action A to termination. In case of interruption, it may be that only the partial schema is stored in long-term memory. The partially filled schema $\langle V, A, nil \rangle$ lacks the declarative meaning of the complete schema, but has a restricted version of the imperative meaning:

$$\text{imperative:} \quad holds(V, now) \Rightarrow do(A, now).$$

Routines

A *routine* is a set of schemas, indexed by initial view. It represents the sequence of actions and intermediate views in a behavior that moves the robot from an initial to a final distinctive state. A routine can be used either as a description of the behavior, or as a procedure for reproducing it.

Consider the alternating sequence of views and actions $V_0, A_0, V_1, A_1, V_2, \ldots, V_{n-1}, A_{n-1}, V_n$ leading from V_0 to V_n.

- A routine R is *complete* from view V_0 to V_n if R contains the schema $\langle V_i, A_i, V_{i+1} \rangle$ for each i from 0 to $n-1$.
- A routine R is *adequate* from V_0 to V_n if R contains either $\langle V_i, A_i, V_{i+1} \rangle$ or $\langle V_i, A_i, nil \rangle$ for each i from 0 to $n-1$.

An adequate routine supports "situated action": physical travel from state V_0 to V_n within the environment (Agre & Chapman 1987). It also generalizes naturally to causal graphs such as *universal plans* (Schoppers 1987), which are sets of rules specifying the actions to take at *each* state in a state-space to move toward a given goal. In addition to situated action, a complete routine supports cognitive operations such as mental review or verbal description of the route in the absence of the environment.

The Topological Level

The topological map describes the environment as a collection of places, paths, and regions, linked by topological relations such as connectivity, order, containment, boundary, and abstraction. Places, paths, and boundary regions are created from experience represented as a sequence of views and actions. They are created by *abduction*, positing the minimal additional set of places, paths, and regions required to explain the sequence of observed views and actions.

- A *place* describes part of the robot's environment as a zero-dimensional point. A place may lie on zero or more paths. A place may also be defined as the abstraction of a region.
- A *path* describes part of the robot's environment, for example a street in a city, as a one-dimensional subspace. It may describe an order relation on the places it contains, and it may serve as a boundary for one or more regions. The two directions along a path are $dir = +1$ and $dir = -1$.
- A *region* represents a two-dimensional subset of the robot's environment. The set of places in a region share a common property. A region may be defined by one or more boundaries, by a common frame of reference, or by its use in an abstraction relation.

Co-occurrence Implies Topological Connections

The "current context" or "You-Are-Here pointer" describes the current state of the explorer. The topological level adds the current place, path, and 1-D direction to the current context. Simultaneous presence of several descriptions in the current context implies a topological connection.

$$\begin{array}{l} current\_place(p) \wedge current\_view(v) \rightarrow at(v, p) \\ current\_path(p) \wedge current\_direction(d) \wedge \\ \quad current\_view(v) \rightarrow along(v, p, d) \\ current\_place(p) \wedge current\_path(path) \rightarrow \\ \quad on\_path(p, path). \end{array}$$

The 1D topological order of places along the current path is inferred from a *Travel* action and the current direction. We can also infer, or abduce, topological boundary and containment relations among regions, paths, and places.

Abduction to Places and Paths from Views and Actions

The definition of topological places is coupled with a categorization of actions into those that change the current place, called *Travel* actions, and those that leave the current place the same, called *Turn* actions. An action description includes a term representing the observed magnitude of the corresponding control laws, from internal effort sensors such as odometry. Since an action must begin and end at a locally distinctive state, not every magnitude of *Turn* or *Travel* is a meaningful action.

- $\langle V, (\texttt{Turn}\ a), V'\rangle$ means that $place(V) = place(V')$ and a is monotonically related to the magnitude of the turn from $\theta(V)$ to $\theta(V')$. ($place(V)$ denotes the place where the robot observed V. Since places can have identical views, it is not necessarily a function only of V.)
- $\langle V, (\texttt{Travel}\ d), V'\rangle$ means that $place(V) \neq place(V')$ if $d \neq 0$, and d is monotonically related to the distance traveled from $place(V)$ to $place(V')$.

We use the following observations as the basis for abduction of the connectivity properties of places and paths, given sequences of views and actions. Since different places could provide the same sensory image, sophisticated inference and even physical travel may occasionally be required to identify the current place from the current view (Kuipers & Byun 1988; Dudek *et al.* 1991).

- Every view is observed at a place.
  $$\forall view\ \exists place\ at(view, place)$$
- A *Turn* leaves the traveller at the same place.
  $$\langle V, (\texttt{Turn}\ a), V'\rangle \rightarrow \exists place\ [at(V, place) \wedge at(V', place)] \quad (8)$$
- A *Travel* leaves the traveller on the same path, facing the same direction. If the distance traveled is nonzero, the starting and ending places are different.
  $$\langle V, (\texttt{Travel}\ d), V'\rangle \wedge d \neq 0 \rightarrow$$
  $$\exists p_1, p_2\ [p_1 \neq p_2 \wedge at(V, p_1) \wedge at(V', p_2)]$$
  $$\langle V, (\texttt{Travel}\ d), V'\rangle \rightarrow$$
  $$\exists path, dir\ [along(V, path, dir) \wedge along(V', path, dir)]$$

The topological level supports an array of problem-solving methods, augmenting graph search with heuristics based on the boundary and containment relations (not described here) that regions add to the topological map.

We have implemented a system that takes alternating sequences of views and actions from tours of a simulated urban environment and builds causal, topological, and local 1-D metrical descriptions of the environment.

The Metrical Level

Local 1-D Geometry

Observations of the magnitudes of actions provide information about the local geometry of places and paths. $\langle V, (\texttt{Travel}\ d), V'\rangle$ provides evidence about the distance between two places on the current path. $\langle V, (\texttt{Turn}\ a), V'\rangle$ provides evidence about the angle between obstacles and/or paths at the current place. This information can be represented as 1-D (linear or circular) metrical properties of the individual places and paths in the topological map. These properties are accumulated incrementally by the same abductive process that builds the topological map.

Local 2-D Geometry

If we take into account the fact that the topological map is embedded in a 2-D space, we can incrementally accumulate local descriptions of place neighborhoods and path segments as 2-D manifolds. Occupancy grids (Moravec 1988; Konolige 1995), sonar target maps (Leonard & Durrant-Whyte 1992; MacKenzie & Dudek 1994), and generalized cylinders (Nevatia & Binford 1977; Brooks 1981) are three representations for 2-D manifold descriptions of local place neighborhoods and path segments.

Global 2-D Geometry

Once the topological and local metrical descriptions are sufficiently rich and reliable, these descriptions can be relaxed into a global 2-D frame of reference (figure 1(c)). However, this representational transformation is never on the critical path for exploration, map-learning, route-planning, or navigation.

Guarantees

A state s is *localizable* if, starting from s, there is a reliable method for traveling to a distinctive state, and thus being localized within the topological map. The localizable states are defined in terms of (a) the selection criteria for control laws, as embodied in the appropriateness measures $a_d(\mathbf{s})$, and (b) the basins of attraction defined by those control laws, considered as dynamical systems. A state s is *reachable* if, starting at a localizable state, there is a reliable method for the robot to travel to s. Using the framework defined above, we can analyze which states in the physical environment are localizable and/or reachable, giving various levels of knowledge in the cognitive map.

Discussion

The concept of "distinctive state" as used here for cognitive mapping appears to generalize certain aspects of the concept of "landmark value" as used for qualitative simulation.

Landmark values in QSIM corresponding to sign changes, operating region transitions, or extreme points in the behavior all represent individual real numbers. Ordinal relations with these landmarks support straight-forward qualitative hill-climbing, so they are distinctive within their quantity spaces. The only landmark values with a different character are those corresponding to initial values, which represent universally quantified variables ranging over a set defined by pre-existing landmark values.

Both cognitive mapping and qualitative simulation rely on abstracting a continuous underlying space to a discrete set of objects with symbolic names and symbolic relationships. In spite of the differences between the domains, I believe that the common structure will prove to be important.

References

Agre, P. E., and Chapman, D. 1987. Pengi: An implementation of a theory of activity. In *Proc. 6th National Conf. on Artificial Intelligence (AAAI-87)*. Morgan Kaufmann.

Arkin, R. C. 1989. Motor schema-base mobile robot navigation. *International Journal of Robotics Research* 8(4):92–112.

Atkeson, C. G.; Moore, A. W.; and Schaal, S. 1996. Locally weighted learning. *Artificial Intelligence Review*. in press.

Basye, K.; Dean, T.; and Vitter, J. S. 1989. Coping with uncertainty in map learning. In *Proc. 11th Int. Joint Conf. on Artificial Intelligence (IJCAI-89)*.

Brooks, R. A. 1981. Symbolic reasoning among 3D models and 2D images. *Artificial Intelligence* 17:285–348.

Brooks, R. A. 1986. A robust layered control system for a mobile robot. *IEEE Trans. on Robotics and Automation* RA-2(1):14–23.

Brooks, R. A. 1990. The behavior language; user's guide. Technical Report Memo 1227, MIT Artificial Intelligence Laboratory, Cambridge, MA.

Cassandra, A. R.; Kaelbling, L. P.; and Littman, M. L. 1994. Acting optimally in partially observable stochastic domains. In *Proc. 12th National Conf. on Artificial Intelligence (AAAI-94)*, 1023–1028. Cambridge, MA: AAAI Press/The MIT Press.

Chatila, R., and Laumond, J.-P. 1985. Position referencing and consistent world modeling for mobile robots. In *IEEE International Conference on Robotics and Automation*, 138–170.

Courtney, J. D., and Jain, A. K. 1994. Mobile robot localization via classification of multisensor maps. In *IEEE International Conference on Robotics and Automation*, volume 2, 1672–1678. Los Alamitos, CA: IEEE Computer Society Press.

Dean, T. L., and Wellman, M. P. 1991. *Planning and Control*. San Mateo, CA: Morgan Kaufmann.

Dean, T.; Basye, K.; Chekaluk, R.; Hyun, S.; Lejter, M.; and Randazza, M. 1990. Coping with uncertainty in a control system for navigation and exploration. In *Proc. 8th National Conf. on Artificial Intelligence (AAAI-90)*. Cambridge, MA: AAAI Press.

Dean, T.; Angluin, D.; Basye, K.; Engelson, S.; Kaelbling, L.; Kokkevis, E.; and Maron, O. 1992. Inferring finite automata with stochastic output functions and an application to map learning. In *Proc. 10th National Conf. on Artificial Intelligence (AAAI-92)*. Cambridge, MA: AAAI Press.

Dudek, G.; Jenkin, M.; Milios, E.; and Wilkes, D. 1991. Robotic exploration as graph construction. *IEEE Trans. on Robotics and Automation* 7(6):859–865.

Engelson, S. P., and McDermott, D. V. 1992. Maps considered as adaptive planning resources. In *AAAI Fall Symposium on Applications of AI to Real-World Autonomous Mobile Robots, Working Notes*.

Froom, R. 1991. Acquiring effective knowledge of environmental geometry for minimum-time control of a mobile robot. In *Proceedings of the 1991 IEEE International Symposium on Intelligent Control*.

Froom, R. 1992. Approximate maps for high-speed control of a mobile robot. In *Mobile Robots VII: Proceedings of SPIE – the International Society for Optical Engineering v. 1831*.

Gat, E. 1991. Alfa: a language for programming reactive robotic control systems. In *IEEE International Conference on Robotics and Automation*.

Gladwin, T. 1970. *East is a Big Bird: Navigation and Logic on Puluwat Atoll*. Cambridge, MA: Harvard University Press.

Hart, R. A., and Moore, G. T. 1973. The development of spatial cognition: A review. In Downs, R. M., and Stea, D., eds., *Image and Environment*. Chicago: Aldine Publishing Company.

Hutchins, E. L. 1995. *Cognition in the Wild*. Cambridge, MA: MIT Press.

Konolige, K. 1995. A refined method for occupancy grid interpretation. unpublished draft manuscript.

Kortenkamp, D., and Weymouth, T. 1994. Topological mapping for mobile robots using a combination of sonar and vision sensing. In *Proc. 12th National Conf. on Artificial Intelligence (AAAI-94)*, 979–984. AAAI Press/The MIT Press.

Kosko, B. 1992. *Neural Networks and Fuzzy Systems*. Englewood Cliffs, NJ: Prentice-Hall.

Kuipers, B. J., and Åström, K. 1994. The composition and validation of heterogeneous control laws. *Automatica* 30(2):233–249.

Kuipers, B., and Byun, Y. T. 1988. A robust qualitative method for spatial learning in unknown environments. In *Proc. 7th National Conf. on Artificial Intelligence (AAAI-88)*. Los Altos, CA: Morgan Kaufmann.

Kuipers, B., and Byun, Y.-T. 1991. A robot exploration and mapping strategy based on a semantic hierarchy of spatial representations. *Journal of Robotics and Autonomous Systems* 8:47–63.

Kuipers, B. J., and Levitt, T. 1988. Navigation and mapping in large scale space. *AI Magazine* 9(2):25–43. Reprinted in Advances in Spatial Reasoning, Volume 2, Su-shing Chen (Ed.), Norwood NJ: Ablex Publishing, 1990.

Kuipers, B. J. 1978. Modeling spatial knowledge. *Cognitive Science* 2:129–153. Reprinted in Advances in Spatial Reasoning, Volume 2, Su-shing Chen (Ed.), Norwood NJ: Ablex Publishing, 1990.

Latombe, J.-C. 1991. *Robot Motion Planning*. Boston: Kluwer Academic Publishers.

Lazanas, A., and Latombe, J.-C. 1992. Landmark-based robot navigation. In *Proc. 10th National Conf. on Artificial Intelligence (AAAI-92)*, 816–822. Cambridge, MA: AAAI Press/The MIT Press.

Leonard, J. J., and Durrant-Whyte, H. F. 1992. *Directed Sonar Sensing for Mobile Robot Navigation*. Boston: Kluwer Academic Publishers.

Lynch, K. 1960. *The Image of the City*. Cambridge, MA: MIT Press.

MacKenzie, P., and Dudek, G. 1994. Precise positioning using model-based maps. In *IEEE International Conference on Robotics and Automation*, volume 2, 1615–1621. Los Alamitos, CA: IEEE Computer Society Press.

Mamdani, E. H. 1974. Applications of fuzzy algorithms for control of a simple dynamic plant. *Proc. IEE* 121:1585–1588.

Mataric, M. J. 1992. Integration of representation into goal-driven behavior-based robots. *IEEE Trans. on Robotics and Automation* 8(3):304–312.

McCarthy, J., and Hayes, P. J. 1969. Some philosophical problems from the standpoint of artificial intelligence. In Meltzer, B., and Michie, D., eds., *Machine Intelligence 4*. Edinburgh: Edinburgh University Press. 463–502.

Moore, A. W., and Atkeson, C. G. 1993. Memory-based reinforcement learning: converging with less data and less real time. In Connell, J. H., and Mahadevan, S., eds., *Robot Learning*. Kluwer Academic Publishers.

Moore, A. W.; Atkeson, C. G.; and Schaal, S. 1996. Memory-based learning for control. *Artificial Intelligence Review*. to appear.

Moravec, H., and Elfes, A. 1985. High resolution maps from wide angle sonar. In *IEEE International Conference on Robotics and Automation*, 116–121.

Moravec, H. P. 1988. Sensor fusion in certainty grids for mobile robots. *AI Magazine*.

Nevatia, R., and Binford, T. O. 1977. Description and recognition of curved objects. *Artificial Intelligence* 8(1):77–98.

Ohya, A.; Nagashima, Y.; and Yuta, S. 1994. Exploring unknown environment and map construction using ultrasonic sensing of normal direction of walls. In *IEEE International Conference on Robotics and Automation*, volume 1, 485–492. Los Alamitos, CA: IEEE Computer Society Press.

Penberthy, J. S., and Weld, D. S. 1994. Temporal planning with continuous change. In *Proc. 12th National Conf. on Artificial Intelligence (AAAI-94)*, 1010–1015. Cambridge, MA: AAAI Press/The MIT Press.

Piaget, J., and Inhelder, B. 1967. *The Child's Conception of Space*. New York: Norton. First published in French, 1948.

Pierce, D., and Kuipers, B. 1994. Learning to explore and build maps. In *Proc. 12th National Conf. on Artificial Intelligence (AAAI-94)*. AAAI/MIT Press.

Pomerleau, D. A. 1993. Knowledge-based training of artificial neural networks for autonomous robot driving. In Connell, J. H., and Mahadevan, S., eds., *Robot Learning*. Kluwer Academic Publishers.

Schiele, B., and Crowley, J. L. 1994. A comparison of position estimation techniques using occupancy grids. In *IEEE International Conference on Robotics and Automation*, volume 2, 1628–1634. Los Alamitos, CA: IEEE Computer Society Press.

Schoppers, M. J. 1987. Universal plans for reactive robots in unpredictable environments. In *Proc. 10th Int. Joint Conf. on Artificial Intelligence (IJCAI-87)*. San Mateo, CA: Morgan Kaufmann.

Simmons, R., and Koenig, S. 1995. Probabilistic robot navigation in partially observable environments. In *Proc. 14th Int. Joint Conf. on Artificial Intelligence (IJCAI-95)*, 1080–1087. San Mateo, CA: Morgan Kaufmann.

Slack, M. G. 1993. Navigation templates: mediating qualitative guidance and quantitative control in mobile robots. *IEEE Trans. on Systems, Man and Cybernetics* 23(2):452–466.

Watkins, C. J. C. H. 1989. *Learning from delayed rewards*. Ph.D. Dissertation, King's College, University of Cambridge.