To appear, Seventeenth International Joint Conference on Artificial Intelligence (IJCAI-01), Seattle, Washington, USA, August 4-10, 2001.

A Logical Account of Causal and Topological Maps

Emilio Remolina and **Benjamin Kuipers***
Computer Science Department
The University of Texas at Austin
Austin, TX 78712, USA
e-mail: {eremolin,kuipers}@cs.utexas.edu

Abstract

We consider the problem of how an agent creates a discrete spatial representation from its continuous interactions with the environment. Such representation will be the *minimal* one that explains the experiences of the agent in the environment. In this paper we take the Spatial Semantic Hierarchy as the agent's target spatial representation, and use a circumscriptive theory to specify the minimal models associated with this representation. We provide a logic program to calculate the models of the proposed theory. We also illustrate how the different levels of the representation assume different spatial properties about both the environment and the actions performed by the agent. These spatial properties play the role of "filters" the agent applies in order to distinguish the different environment states it has visited.

1 Introduction

The problem of map building –how an agent creates a discrete spatial representation from its continuous interactions with the environment– can be stated formally as an abduction task where the actions and observations of the agent are explained by connectivity relations among places in the environment [Shanahan, 1996, Shanahan, 1997, Remolina and Kuipers, 1998]. In this paper we consider the Spatial Semantic Hierarchy (SSH) [Kuipers, 2000, Kuipers and Byun, 1988, Kuipers and Byun, 1991] as the agent's target spatial representation. The SSH is a set of distinct representations for large scale space, each with its own ontology and each abstracted from the levels below it. The SSH describes the different states of knowledge that an agent uses in order to organize its sensorimotor experiences and create a spatial representation (i.e. a map). Using the SSH representation, navigation among places is not dependent on the accuracy, or even the existence, of metrical knowledge of the environment.

*This work has taken place in the Intelligent Robotics Lab at the Artificial Intelligence Laboratory, The University of Texas at Austin. Research of the Intelligent Robotics lab is supported in part by NSF grants IRI-9504138 and CDA 9617327, and by funding from Tivoli Corporation.

In order to define the *preferred models* associated with the experiences of the agent, we use a circumscriptive theory to specify the SSH's (minimal) models. Different models can exist that explain the same set of experiences. This occurs because the agent could associate the same sensory description to different environment states, or because the agent has not completely explored the environment. The different SSH levels assume different spatial properties about the environment and the actions performed by the agent. These spatial properties play the role of "filters" the agent applies in order to distinguish the different environment states it has visited. For instance, at the SSH causal level two environment states are considered the same if any sequence of actions started at these states renders the same sequence of observations. At the SSH topological level, two environment states are considered the same if they are at the same place along the same paths. Finally, at the SSH metrical level, two environment states are the same, if it is possible to assign to them the same coordinates in any frame of reference available to the agent. In sections 3 and 4 we make precise the claims above.

Finally, we use the SSH circumscriptive theory as the specification for a logic program used to implement the abduction task. In the paper we provide the logic program for the SSH causal level theory and illustrate how to encode the minimality condition associated with this theory. We have implemented the program using Smodels [Niemelä and Simons, 1997] and confirm that the theory yields the intended models.

2 Related Work

The SSH grew out of the TOUR model proposed in [Kuipers, 1977, Kuipers, 1978]. Other computational theories of the cognitive map have been proposed: [Kortenkamp *et al.*, 1995, McDermott and Davis, 1984, Leiser and Zilbershatz, 1989, Yeap, 1988]. These theories share the same basic principles: the use of multiple frames of reference, qualitative representation of metrical information, and connectivity relations among landmarks. They differ in how they define what a landmark is, or the description (view, local 2D geometry) associated with a landmark. Except for McDermott and Davis, none of the theories above has a formal account like the one presented in this paper for the SSH.

Considering map building as a formal abduction task has been proposed by Shanahan [Shanahan, 1996, Shanahan, 1997]. He proposes a logic-based framework (based on the circumscriptive event calculus) in which a robot constructs a model of the world through an abductive process whereby sensor data is explained by hypothesizing the existence, locations, and shapes of objects. In Shanahan's work, space is considered a real-valued coordinate system. As pointed out in [Shanahan, 1997], a problem of Shanahan's approach is the existence of many minimal models (maps) that explain the agent's experiences. We have alleviated this problem by considering the SSH topological map instead of an Euclidean space as the agent's target spatial representation.

The problem of distinguishing environment states by outputs (views) and inputs (actions) has been studied in the framework of automata theory [Basye *et al.*, 1995]. In this framework, the problem we address here is the one of finding the smallest automaton (w.r.t. the number of states) consistent with a given set of input/output pairs. Without any particular assumptions about the environment or the agent's perceptual abilities, the problem of finding this smallest automaton is NP-complete [Basye *et al.*, 1995].

The SSH [Kuipers, 2000, Kuipers and Byun, 1988, Kuipers and Byun, 1991] 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. At the *SSH control level*, the agent and its environment are modeled as continuous dynamical systems whose equilibrium points are abstracted to a discrete set of *distinctive states*. A distinctive state has associated a *view* describing the sensory input obtained at that distinctive state. The control laws, whose executions define trajectories linking these distinctive states, are abstracted to *actions*, giving a discrete causal graph representation for the state space. The causal graph of states and actions can in turn be abstracted to a topological network of *places*, *paths* and *regions* (i.e. the *topological map*). 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 global metrical consistency problems. In the next sections we formally describe the SSH causal and topological levels.

3 SSH Causal level

We use a first order sorted language in order to describe the SSH causal level. The sorts of this language include *distinctive states*, *views*, *actions* and *schemas*. The sort of distinctive states corresponds to the names given by the agent to the fixpoints of hill-climbing control strategies. It is possible for the agent to associate different distinctive state names with the same environment state. This is the case since the agent might not know at which of several environment states it is currently located. A distinctive state has an associated view. We use the predicate $View(ds, v)$ to represent the fact that *v* is a *view* associated with *distinctive state* ds. We assume that a distinctive state has a unique view. However, we do **not** assume that views uniquely determine distinctive states (i.e. $View(ds, v) \wedge View(ds', v) \not\rightarrow ds = ds'$). This is the case since the sensory capabilities of an agent may not be sufficient to distinguish distinctive states.

An action has a unique type, either *travel* or *turn*, associated with it. We use the predicate $Action\_type(a, type)$ to represent the fact that the type of action *a* is *type*. Turn actions have associated a unique turn description, either *turnLeft*, *turnRight* or *turnAround*. We use the predicate $Turn\_desc(a, desc)$ to indicate that *desc* is the turn description associated with the turn action *a*.

A schema represents an action execution performed by the agent in the environment. An action execution is characterized in terms of the distinctive states the agent was at before and after the action was performed.[1] We use the predicate $CS(s, ds, a, ds')$ to denote the fact that according to schema s, action a was performed starting at distinctive state ds and ending at distinctive state ds'. While schemas are explicit objects of our theory, most of the time it is convenient to leave them implicit. We introduce the following convenient notation:

$$\langle ds, a, ds' \rangle \equiv_{def} \exists s \; CS(s, ds, a, ds')$$
$$\langle ds, type, ds' \rangle \equiv_{def} \exists a \; \left\{ \langle ds, a, ds' \rangle \wedge Action\_type(a, type) \right\}$$
$$\langle ds, desc, ds' \rangle \equiv_{def} \exists a \; \left\{ \langle ds, a, ds' \rangle \wedge Turn\_desc(a, desc) \right\}$$

Example 1

Consider a robot moving in the environment depicted in figure 1. The robot moves from distinctive state *a* to distinctive state *b* by performing a follow-midline action, *ml*. Then the robot performs the same action to move to distinctive state *c*. We assume that all corridor intersections look alike ($v+$). This set of experiences can be described by the formulae:

$$Action\_type(ml, travel) \; , CS(s1, a, ml, b) \; , \; CS(s2, b, ml, c) \; ,$$
$$View(a, v+) \; , \; View(b, v+) \; , \; View(c, v+) \; .$$

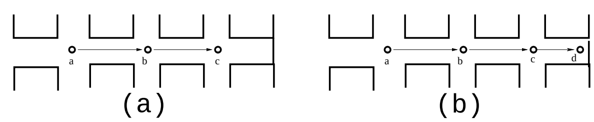


Figure 1: (a) Distinctive states *a*, *b* and *c* are not distinguishable at the causal level. Topological information is needed in order to distinguish them. (b) All distinctive states are distinguished at the causal level given the new information $\langle c, travel, d \rangle$.

Given this set of experiences, at the SSH causal level distinctive states *a*, *b* and *c* are not distinguishable. Any known sequence of actions renders the same set of views. However, at the SSH topological level all these distinctive states are distinguishable since the robot has traveled from *a* to *b* and then to *c* following the same *path* (see example 3). Should the robot continue the exploration and visit distinctive state d, with view $\sqsupset$, then by relying just on known actions and views the agent can distinguish all distinctive states it has visited (see example 2). {*end of example*}

The agent's experiences in the environment are described in terms of *CS*, *View*, *Action_type* and *Turn_desc* atomic formulae. Hereafter we use **E** to denote a particular agent's experience formulae. By $\mathbf{HS}(\mathbf{E})$ we denote the formulae

[1] An action execution also has metrical information associated with it. This metrical information represents an estimate of, for example, the distance or the angle between the distinctive states associated with the action execution.

stating that the sorts of schemas, distinctive states, views and actions are completely defined by the sets of *schema*, *distinctive states*, *view* and *action* constant symbols occurring in *E* respectively.[2] By $\mathbf{DT}$ we denote our domain theory, the formulae stating that: (-) the sets {turn, travel}, {turnLeft,turnRight,turnAround}, completely define the sorts of *action_types* and *turn_descriptions*; (-) an action has associated a unique action type ; (-) distinctive states have associated a unique view; (-) the description associated with an action is unique; (-) turn actions have associated a turn description; (-) the type of actions as well as the qualitative description of turn actions is the one specified in *E*. The SSH causal theory **CT(E)** defines when two distinctive states are indistinguishable at the SSH causal level. We use the predicate $ceq(ds, ds')$ to denote this fact. We will assume that actions are *deterministic*: [3]

$$\langle ds, a, ds' \rangle \wedge \langle ds, a, ds'' \rangle \rightarrow ds' = ds'' \ . \tag{1}$$

CT(E) is the following nested abnormality theory [Lifschitz, 1995]:

$$CT(E) \ = \ E,\ HS(E),\ DT,\ Axiom\,1,\ CEQ\_block$$

where **CEQ_block** is defined as

$$\begin{aligned}
&\{\ max\ \ ceq:\\
&\ ceq(ds,ds') \rightarrow ceq(ds',ds),\\
&\ ceq(ds,ds') \wedge ceq(ds',ds'') \rightarrow ceq(ds,ds''),\\
&\ ceq(ds,ds') \rightarrow View(ds,v) \equiv View(ds',v), \quad (2)\\
&\ ceq(ds_1,ds_2) \wedge \langle ds_1,a,ds_1' \rangle \wedge \langle ds_2,a,ds_2' \rangle \rightarrow ceq(ds_1',ds_2') \quad (3)\\
&\}
\end{aligned}$$

It can be proved that the predicate *ceq* defines an equivalence relation on the sort of distinctive states. Axiom 2 states that indistinguishable distinctive states have the same view. Axiom 3 states that if distinctive states ds and ds' are indistinguishable and action *a* has been performed for both ds and ds', then the action links these states with indistinguishable states. By maximizing *ceq* we identify distinctive states that cannot be distinguished by actions and/or views, and thereby minimize the set of states represented by the model.

Axioms 2 and 3 allow us to prove the following useful lemma:

Lemma 1 *Let A denote a sequence of action symbols. Let $A(ds)$ denote the distinctive state symbol resulting of starting the sequence A at distinctive state ds or $\perp$ if A is not defined for ds.*[4] *Then,*

$$\begin{aligned}
&ceq(ds,ds') \wedge A(ds) \neq \perp \wedge A(ds') \neq \perp\\
&\quad \rightarrow View(A(ds),v) \equiv View(A(ds'),v) \ \ .
\end{aligned}$$

Example 2

Consider the situation depicted in Figure 1b, with the corresponding schemas and views as in example 1. Using lemma 1 one can conclude that all distinctive states *a*, *b* and *c* are distinguishable by actions and views alone. For instance, $\{ml, ml\}(a) = c$, $\{ml, ml\}(b) = d$, $View(\{ml, ml\}(a), v+)$, $View(\{ml, ml\}(b), \sqsupset)$, and consequently, $\neg ceq(a, b)$. {*end of example*}

The Herbrand models of $CT(E)$ are in a one to one correspondence with the answer sets [Gelfond and Lifschitz, 1991] of the logic program in Figure 2.[5] In this program, the X and Y variables range over distinctive states and the variable V ranges over views in E. The sets of rules 4 and 5 are the facts corresponding to the agent's experiences. Rules 6-8 require ceq to be an equivalence class. Rules 8 and 9 are the counterpart of axiom 2. Rule 11 is the counterpart of axiom 3. In order to define the maximality condition of ceq, the auxiliar predicate $p(X, Y, X1, Y1)$ is introduced. This predicate reads as *"If X and Y were the same, then $X1$ and $Y1$ would be the same"*. The predicate $dist(X, Y)$ defines when distinctive states X and Y are distinguishable. Constraint 12 establishes the maximality condition on *ceq*: $ceq(X, Y)$ should be the case unless X and Y are distinguishable.[6]

$$\{cs(ds,a,ds') \leftarrow .\quad : cs(ds,a,ds') \in E\} \tag{4}$$
$$\{view(ds,v) \leftarrow .\quad : view(ds,v) \in E\} \tag{5}$$
$$\begin{aligned}
&ceq(X,Y), \neg ceq(X,Y) \leftarrow \ .\\
&p(X,Y,X,Y) \leftarrow .\\
&p(X,Y,X2,Y1) \leftarrow p(X,Y,X1,Y1), ceq(X1,X2).\\
&p(X,Y,X1,Y2) \leftarrow p(X,Y,X1,Y1), ceq(Y1,Y2).\\
&p(X,Y,X2,Y2) \leftarrow p(X,Y,X1,Y1), cs(X1,A,X2), cs(Y1,A,Y2).\\
&p(X,Y,Y1,X1) \leftarrow p(X,Y,X1,Y1).\\
&p(X,Y,X1,Y2) \leftarrow p(X,Y,X1,Y1), p(X,Y,Y1,Y2).\\
&\\
&dist(X,Y) \leftarrow p(X,Y,X1,Y1), view(X1,V), not\ view(Y1,V).\\
&dist(X,Y) \leftarrow p(X,Y,X1,Y1), not\ view(X1,V), view(Y1,V).
\end{aligned}$$
$$\leftarrow not\ ceq(X,X). \tag{6}$$
$$\leftarrow ceq(X,Y), not\ ceq(Y,X). \tag{7}$$
$$\leftarrow ceq(X,Y), ceq(Y,Z), not\ ceq(X,Z). \tag{8}$$
$$\leftarrow ceq(X,Y), view(X,V), not\ view(Y,V). \tag{9}$$
$$\leftarrow ceq(X,Y), not\ view(X,V), view(Y,V). \tag{10}$$
$$\leftarrow not\ ceq(X1,Y1), ceq(X,Y), cs(X,A,X1), cs(Y,A,Y1). \tag{11}$$
$$\leftarrow not\ ceq(X,Y), not\ dist(X,Y). \tag{12}$$

Figure 2: Logic program associated with CT(E).

[2]That $sort$ is completely defined by the constant symbols $s_1, \ldots, s_n$ means that an interpretation for $sort$ is the *Herbrand* interpretation defined by the set $\{s_1, \ldots, s_n\}$.

[3]Throughout this paper we assume that formulas are universally quantified.

[4]Given an action symbol A and distinctive state ds, $A(ds) = ds'$ if the schema $\langle ds, A, ds' \rangle$ has been observed, otherwise, $A(ds) = \perp$. Moreover, $A(\perp) = \perp$. The definition is then extended to action sequences in the standard way. Notice that $A(ds)$ being well-defined relies on our assumption that actions are deterministic (Axiom 1).

[5]See extended version of this paper [Remolina and Kuipers, 2001] for a proof.

[6]We have implemented this logic program in Smodels [Niemelä and Simons, 1997]. In the implementation, one has to add variable domain restrictions to the different rules. For example, rule

$$ceq(X,Y), \neg ceq(X,Y) \leftarrow \ .$$

becomes

$$ceq(X,Y), \neg ceq(X,Y) \leftarrow \ dstate(X), dstate(Y)$$

where $dstate$ is our predicate to identify the sort of distinctive states.

4 SSH Topological Level

We are to define the SSH topological theory, $\mathbf{TT}(\mathbf{E})$, associated with a set of experiences E. The language of this theory is a sorted language with sorts for *places*, *paths* and *path directions*.[7] The main purpose of $TT(E)$ is to minimize the set of paths and places consistent with the given experiences E. A place can be a *topological place* (hereafter place) or a *region*. A place is a set of distinctive states linked by turn actions. A region is a set of places. We use the predicates *tplace* and *is_region* to identify these subsorts. A path defines an order relation among places connected by travel with no turn actions. They play the role of streets in a city layout. We use the predicate *tpath* to identify the sort of paths. By minimizing the extent of $tplace$, is_region and $tpath$ we minimize the sort of places and paths respectively.[8] The language of the SSH topological level includes the following other predicates: *teq(ds,ds')* – distinctive states ds and ds' are *topologically* indistinguishable; *at(ds,p)* – distinctive state ds is at place p; *along(ds,pa,dir)* –distinctive state ds is along path pa in direction dir; *OnPath(pa,p)* –place p is on path pa; *PO(pa,dir,p,q)* –place p is before place q when facing direction dir on path pa (PO stands for Path Order).

TT(E), is the following nested abnormality theory:

$$\begin{aligned}&\forall p,\, tplace(p) \equiv \neg is\_region(p)\,, \forall pa,\ tpath(pa)\ , \quad (13)\\ &\{min\ is\_region\ :\\ &\quad CT(E)\ , T\_block\ , AT\_block\quad \}\end{aligned}$$

The first line in Axioms 13 says that topological places and regions are the two subsorts of places, and that the predicate $tpath$ represents the sort of paths. The block **CT(E)** is the one defined in the previous section. The block **T_block** defines the predicates $\widehat{turn}$, $\widehat{travel}$, and $\vec{travel}$ such that $\widehat{turn}$ is the equivalence closure of the schemas $\langle \cdot, turn, \cdot\rangle$; $\widehat{travel}$ and $\vec{travel}$ are the equivalence and transitive closure of the schemas $\langle \cdot, travel, \cdot\rangle$.

The block **AT_block** (Figure 3) is the heart of our theory.[9] The purpose of this block is to define the extent of the predicates *tpath*, *tplace*, *at*, *along*, *PO* and *teq*, while identifying a minimum set of places and paths that explain E. The block has associated the circumscription policy[10]

$$\textbf{circ}\ tpath \succ along \succ PO \succ OnPath \succ tplace\ \textbf{var}\ \vec{SSHpred}$$

where $\vec{SSHpred}$ stands for the tuple of predicates *at*, *teq*, *travel_eq*, and *turn_eq*.[11] This circumscription policy states (among others) that a minimum set of paths is preferred over a minimum set of places. Next we discuss the axioms in AT_block.

$$\begin{aligned}
&\{\ :\\
&teq(ds,ds') \equiv \exists p\ \left\{ceq(ds,ds') \wedge at(ds,p) \wedge at(ds',p)\right\}, && (14)\\
&at(ds,p) \rightarrow tplace(p), && (15)\\
&\exists ! p\, at(ds,p), && (16)\\
&\langle ds, turn, ds'\rangle \wedge at(ds,p) \rightarrow at(ds',p), && (17)\\
&at(ds,p) \wedge at(ds',p) \rightarrow turn_eq(ds,ds'), && (18)\\
&\{min\ \ turn_eq : && (19)\\
&\quad teq(ds,ds') \wedge teq(dr,dr') \wedge \widehat{turn}(ds',dr') \rightarrow turn_eq(ds,dr),\\
&\quad turn_eq(ds,ds') \wedge turn_eq(ds',ds'') \rightarrow turn_eq(ds,ds'')\quad \}\\
&along(ds,pa,dir) \rightarrow tpath(pa), && (20)\\
&at(ds,p) \wedge at(ds',q) \wedge \vec{travel}(ds,ds') \rightarrow && (21)\\
&\quad \exists pa,\ dir\ \left\{PO(pa,dir,p,q) \wedge along(ds,pa,dir) \wedge along(ds',pa,dir)\right\},\\
&along(ds,pa,dir) \wedge along(ds,pa1,dir1) \rightarrow pa = pa1, && (22)\\
&at(ds,p) \wedge at(ds',p) \wedge along(ds,pa,dir) \wedge && (23)\\
&\quad along(ds',pa,dir) \rightarrow teq(ds,ds'),\\
&\Big\{\langle ds, turn_desc, ds'\rangle\ \wedge\ turn_desc \neq turnAround \wedge && (24)\\
&\qquad along(ds,pa,dir) \wedge along(ds',pa1,dir1)\Big\} \rightarrow pa \neq pa1,\\
&\langle ds, turnAround, ds'\rangle \rightarrow along(ds,pa,dir) \equiv along(ds',pa,-dir), && (25)\\
\\
&PO(pa,pos,p,q) \equiv PO(pa,neg,q,p), && (26)\\
&\neg PO(pa,dir,p,p), && (27)\\
&PO(pa,dir,p,q) \wedge PO(pa,dir,q,r) \rightarrow PO(pa,dir,p,r), && (28)\\
&PO(pa,dir,p,q) \rightarrow OnPath(pa,p) && (29)\\
&OnPath(pa,p) \wedge OnPath(pa,q) \wedge tpath(pa) \rightarrow && (30)\\
&\quad \exists ds, ds'\ \{at(ds,p) \wedge at(ds',q) \wedge travel_eq(ds,ds')\},\\
&\{min\ \ travel_eq : && (31)\\
&\quad teq(ds,ds') \wedge teq(dr,dr') \wedge \widehat{travel}(ds',dr') \rightarrow travel_eq(ds,dr),\\
&\quad travel_eq(ds,ds') \wedge travel_eq(ds',ds'') \rightarrow travel_eq(ds,ds'')\ \ \}\\
&\textbf{circ}\ \ tpath \succ along \succ PO \succ OnPath \succ tplace\ \ \textbf{var}\ \ \vec{SSHpred}\\
&\}
\end{aligned}$$

Figure 3: AT_block.

Predicate teq is the equivalence relation defined by axiom 14. $teq(ds, ds')$ is the case whenever ds and ds' cannot be distinguished by views and actions (i.e. $ceq(ds, ds')$) and it is consistent to group ds and ds' into the same place. If we assume that views uniquely identify distinctive states (e.g. $View(ds, V) \wedge View(ds', V) \rightarrow ds = ds'$), then predicates ceq and teq will reduce to equality. This is expected since all that is required to identify a distinctive state is its view.

Every distinctive state is at a unique place (Axiom 16). Whenever the agent $turns$, it stays at the same place (Axiom 17). Distinctive states grouped into a topological place should be $turn$ connected (modulo teq) (Axiom 18). *Travel* actions among distinctive states are abstracted to topological paths connecting the places associated with those distinctive states (Axiom 21). A distinctive state is along at most one path (Axiom 22). At each place there is at most one distinctive state along a given path direction (Axiom 23). Turn actions other

[7] The sort of directions is completely defined by the symbols *pos* and *neg*.

[8] Notice that our logic has sorts for *places* and *paths* but in order to minimize these sorts we have to explicitly have predicates representing them.

[9] Notice that the predicate is_region is not mentioned in the theory of figure 3. In the next section we will add to this theory axioms dealing with regions. For the purpose of this section, the minimization of is_region in conjunction with $\forall p,\ tplace(p) \equiv \neg is\_region(p)$ implies (the default) $\forall p\ tplace(p)$.

[10] The symbol $\succ$ indicates prioritized circumscription (see [Lifschitz, 1994] section 7.2).

[11] Block 19 in Figure 3 states that the predicate $turn\_eq$ corresponds to the relation $\widehat{turn}$ modulo teq. Block 31 defines $travel\_eq$ to be the relation $\widehat{travel}$ modulo teq.

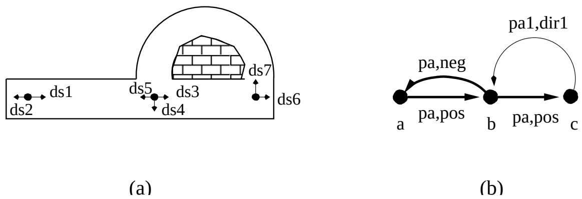


Figure 4: The environment in (a) illustrates a case where different paths intersect at more than one place. (b) depicts the topological map associated with this environment.

than *turnAround* change the path the initial and final distinctive states are at (Axiom 24). *TurnAround* actions relate distinctive states being in the same path but opposite directions (Axiom 25). The order of places in a given path direction is the inverse of the order of places in the other path direction (Axiom 26). Axioms 27 and 28 require $PO(pa, dir, \cdot, \cdot)$ to be a non-reflexive transitive order for the places on pa. Places ordered by a path should belong to that path (Axiom 29). Axiom 30 requires the agent to have traveled among the places on a same path.

Our theory does not assume a "rectilinear" environment where paths intersect at most in one place. It is possible for different paths to have the same order of places (see Figure 4). Topological information can distinguish distinctive states not distinguishable by view and actions.

Example 3

Consider the scenario of example 1. Since the same view is experienced at *a*, *b* and *c*, the extent of ceq is maximized by declaring $ceq = true$. Using the topological theory, from axiom 16 we conclude that there exist places P and Q, such that $at(a, P)$ and $at(c, Q)$. Since it is the case that $\vec{travel}(a, c)$, from axioms 21 and 27 we conclude, for instance, that $P \neq Q$. Distinctive states a and c are topologically distinguishable though they are "causally indistinguishable" (i.e. $ceq(a, c) \wedge \neg teq(a, c)$). $\{$*end of example*$\}$

Given a minimal model M of $TT(E)$, the SSH topological map is defined by the extent in M of *tpath, tplace, along, PO* and *at*. Since the positive and negative direction of a path are chosen arbitrarily (Axiom 21), there is not a unique minimal model for $TT(E)$. We will consider these "up to path direction isomorphic" models to be the same. However, it is still the case that the theory $TT(E)$ has minimal models that are not isomorphic up to path direction (see Figure 5).

5 SSH Boundary Regions

In addition to connectivity and order among places and paths, the topological map includes topological boundary relations: assertions that a place lies to the right of, to the left of, or on a path. In order to determine boundary relations we formally state the following default heuristic. Suppose the agent is at an intersection on a given path, and it then turns right. If the agent now travels, any place it finds while traveling with no turns will be on the right of the starting path. When conflicting information exists about whether a place is to the right or

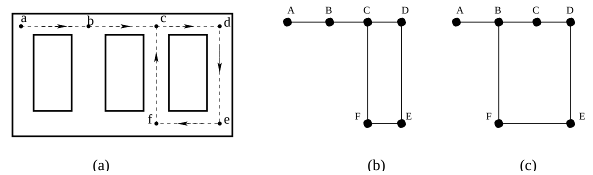


Figure 5: (a) The robot goes around the block visiting places A,...,F,C in the order suggested in the figure. Intersections B and C look alike to the agent. Two minimal models can be associated with the set of experiences in (a) (see (b) and (c)). Topological information is not enough to decide whether the agent is back to B or C. Notice that if the agent accumulates more information, by turning at c and traveling to d, then it can deduce that the topology of the environment is the one in (b). In addition, when available, metrical information can be used to refute the incorrect topology.

left of a path, we deduce no boundary relation (see Figure 6).

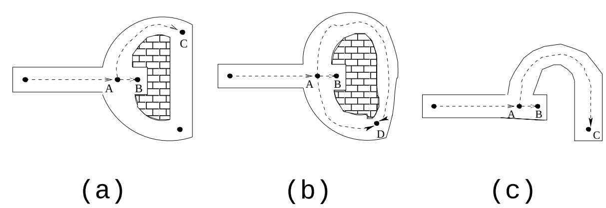


Figure 6: Different environments illustrating how our default to determine boundary relations works. In (a), we conclude by default that place *C* is to the left of the path from *A* to *B*. In (b) we conclude nothing about the location of place *D* with respect to the path from *A* to *B*. In (c), we conclude that place *D* is to the left of the path from *A* to *B*. This is the case since there is no information to conclude otherwise.

We use the predicates $TotheRightOf/TotheLeftOf(p1, pa, dir, pa1, dir1)$ to represent the facts that (i) *p1* is a *place* on both paths, *pa* and *pa1*, and (ii) when the agent is at *place p1* facing in the direction *dir* of *pa*, after executing a turn right (left) action, the agent will be facing on the direction *dir1* of *pa1* (see Figure 7). The predicates *TotheLeftOf* and *TotheRightOf* are derived from the actions performed by the agent at a place:

$$\langle ds, turnRight, ds1\rangle \wedge at(ds, p) \wedge along(ds, pa, dir) \wedge along(ds1, pa1, dir1) \rightarrow TotheRightOf(p, pa, dir, pa1, dir1) \quad (32)$$

We use the predicates $LeftOf(pa, dir, lr)$ and $RightOf(pa, dir, rr)$ to denote that *region* lr (rr) is the left (right) region of path *pa* with respect to the path's direction *dir*. The left/right regions of a path are unique, disjoint, and related when changing the path direction (i.e $LeftOf(pa, dir, r) \equiv RightOf(pa, -dir, r)$). From the relative orientation between paths at a place, we deduce the relative location of places with respect to a path (see Figure 7): [12]

$$TotheRightOf(p1, pa, dir, pa1, dir1) \wedge PO(pa1, dir1, p1, p) \wedge RightOf(pa, dir, rr) \wedge \neg Ab(pa, p) \rightarrow in\_region(p, rr) \quad (33)$$

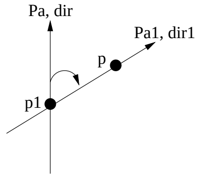


Figure 7: Path *Pa1* is to the right of path *Pa* at place *p1*. Place *p* is after place *p1* on path *pa1*. By default, we conclude that place *p* is to the right of path *pa*.

The predicate **Ab** is the standard "abnormality" predicate used to represent defaults in circumscriptive theories [Lifschitz, 1994]. Axiom 33 states that *"normally"*, if at place *p1* path *pa1* is to the right of path *pa*, and place *p* is after *p1* on path *pa1*, then it should be the case that *p* is on the right of *pa* (Figure 7). In order to capture this default, boundary regions domain theory axioms[13] are added to the block **AT_block** (see Figure 3). Since we are interested in the extent of the new predicates *in_region, LeftOf, RightOf, TotheLeftOf* and *TotheRightOf*, we allow them to vary in the circumscription policy. The new circumscription policy becomes

$$\textbf{circ}\ tpath \succ along \succ PO \succ Onpath \succ \textbf{Ab} \succ \textbf{is\_region} \succ$$
$$\textbf{in\_region} \succ tplace\ \ \textbf{var}\ \ new\vec{SSH}pred$$

where $new\vec{SSH}pred$ stands for the tuple of predicates *at*, *along*, *teq* , *travel_eq*, *turn_eq*, **LeftOf**, **RightOf**, **TotheLeftOf**, and **TotheRightOf**. The circumscription policy states that boundary relations should be established even at the expense of having more places on the map. In addition, by minimizing the predicates *is_region* and *in_region*, we require the models of our theory to have only the regions that are explicitly created by the agent, and not arbitrary ones.

Example 4

Boundary relations determine distinctions among environment states that could not be derived from the connectivity of places alone. Consider an agent visiting the different corners of a square room in the order suggested by Figure 8a. In addition, suppose the agent defines *views* by characterizing the direction of walls and open space. Accordingly, the agent experiences *four* different views, *v1-v4*, in this environment.

The set of experiences *E* in the environment are:

$View(ds1, v1)$	$View(ds2, v2)$	$View(ds3, v1)$
$View(ds4, v2)$	$View(ds5, v1)$	$\langle ds1, turnRight, ds2\rangle$
$\langle ds2, travel, ds3\rangle$	$\langle ds4, travel, ds5\rangle$	$\langle ds3, turnRight, ds4\rangle$

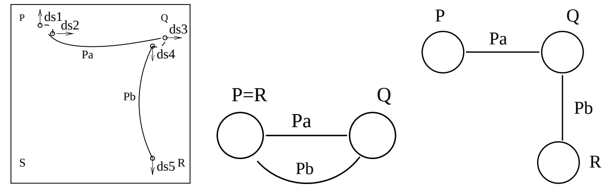
Suppose that the agent does not use boundary regions when building the topological map. Then the minimal topological model associated with *E* has two paths[14] and two places. In this model, $teq(ds1, ds5)$ is the case. The environment looks perfectly symmetric to the agent (Figure 8b).!!

Suppose now that the agent relies on boundary regions. Let *P*, *Q*, *R*, be the topological places associated with

[12]The predicate *in_region(p,r)* states that *place* p is in *region* r.

[13]In the spirit of axioms 32-33.

[14]Notice that from $\langle ds3, turnRight, ds4\rangle$ and Axiom 24 we can deduce that $Pa \neq Pb$ in Figure 8b.



(a) (b) (c)

Figure 8: (a) The figure shows the sequence of actions followed by an agent while navigating a square room. Starting at distinctive state ds1, distinctive states are visited in the order suggested by their number. Dashed lines indicate Turn actions. Solid lines indicate Travel actions. (b) and (c) depict the topological map associated with the environment in (a) without and using boundary regions, respectively.

ds1, *ds3* and *ds5* respectively. From Axiom 21, let *Pa*, *Pb*, dir_a and dir_b be such that $PO(Pa, dir_a, P, Q)$, $along(ds2, Pa, dir_a)$, $along(ds3, Pa, dir_a)$, $PO(Pb, dir_b, Q, R)$, $along(ds4, Pb, dir_b)$, and $along(ds5, Pb, dir_b)$ hold. From Axiom 32 we can conclude then $TotheRightOf(Q, Pa, dir_a, Pb, dir_b)$. In the proposed model, the extent of *Ab* is minimized by declaring $Ab = false$ and consequently from Axiom 33 we conclude $in\_region(R, right(Pa, dir_a))$ where $right(Pa, dir_a)$ denotes the right region of Pa when facing dir_a. Finally, since a path and its regions are disjoint, and $OnPath(Pa, P)$ is the case, we conclude $P \neq R$ and so $\neq teq(ds1, ds5)$. The resulting topological map is depicted in Figure 8c. {*end of example*}

If the agent's sensory capabilities are so impoverished that many distinctive states are perceived to be similar, then metrical information could be used to distinguish different environment states. Figure 9 summarizes different representations an agent could build depending on the spatial properties it relies on.

6 Conclusions

Starting with an informal description of the SSH we have formally specified its intended models. These models correspond to the models of the circumscriptive theory *TT(E)*. The formal account of the theory allows us to illustrate the deductive power of the different SSH ontologies. For instance, example 4 shows how the use of boundary relations allows the agent to determine distinctions among environment states that could not be derived from the connectivity of places and paths alone.

The theory *TT(E)* is rather complex so it may be difficult to determine the effect of the different defaults in combination. However, it is possible to translate this theory into a logic program whose answer sets determine the models of *TT(E)*. We have illustrated the case for the SSH causal theory *CT(E)*, but the same techniques apply for *TT(E)*. The major subtleties in the translation are the minimality and maximality conditions associated with the theory. We have used Smodels to calculate the models of *TT(E)* and confirm that the theory yields

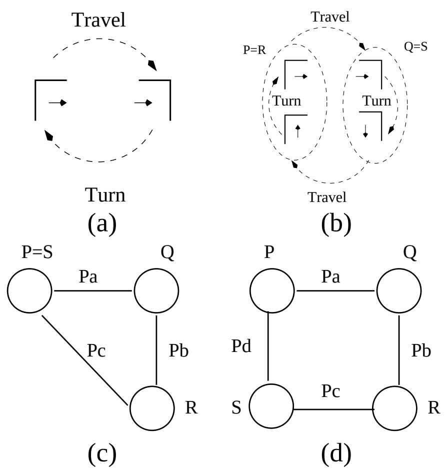


Figure 9: Consider the same environment and agent as in figure 8. Assumes the agent keeps turning right and following the left wall until it is back to distinctive state $ds1$, at place P. Only two kind of views $|\rightrightarrows$ and $\rightrightarrows|$ are observed by the agent. Next we summarizes different maps the agent could build depending on the spatial properties it relies on. (a) If the agent only relies on causal information, the map consists of two states. (b) When topological information is used, but without boundary relations, the map consists of four states and two places. (c) When boundary relations are used, the map consists of six states and three places. There is no fixed correspondence between the three places in the map and the four indistinguishable places in the real world. (d) If metrical information is accurate enough to refute the hypothesis $P = S$, the map will consist of eight states and four places.

the intended models. However, when the number of distinctive states is big, Smodels may not be able to ground the theory as the number of rules associated with the program grows exponentially. We are still working on solving this problem.

Acknowledgments

We are grateful to Vladimir Lifschitz for his valuable feedback during this work and for suggesting the use of Smodels to implement the ideas proposed here. We also thank the anonymous referees for their valuable comments on this paper.

References

[Basye *et al.*, 1995] K. Basye, T. Dean, and L. P. Kaelbling. Learning dynamics: System identification for perceptually challenged agents. *Artificial Intelligence*, 72(1):139–171, 1995.

[Gelfond and Lifschitz, 1991] M Gelfond and V. Lifschitz. Classical negation in logic programs and disjunctive databases. *New Generation Computing*, 9:365–385, 1991.

[Kortenkamp *et al.*, 1995] D. Kortenkamp, E. Chown, and S. Kaplan. Prototypes, locations, and associative networks (PLAN): towards a unified theory of cognitive mapping. *Cognitive Science*, 19:1–51, 1995.

[Kuipers and Byun, 1988] B. Kuipers and Y. T. Byun. A robust qualitative method for spatial learning in unknown environments. In Morgan Kaufmann, editor, *AAAI-88*, 1988.

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

[Kuipers, 1977] B. Kuipers. *Representing Knowledge of Large-Scale Space*. PhD thesis, Artificial Intelligence Laboratory, MIT, 1977.

[Kuipers, 1978] B. J. Kuipers. Modeling spatial knowledge. *Cognitive Science*, 2:129–153, 1978.

[Kuipers, 2000] B. Kuipers. The spatial semantic hierarchy. *Artificial Intelligence*, 119:191–233, 2000.

[Leiser and Zilbershatz, 1989] D. Leiser and A. Zilbershatz. THE TRAVELLER: a computational model of spatial network learning. *Environment and Behavior*, 21(4):435–463, 1989.

[Lifschitz, 1994] V. Lifschitz. Circumscription. In *Handbook of Logic in Artificial Intelligence and Logic Programming*, volume 3, pages 297–352. Oxford University Press, 1994.

[Lifschitz, 1995] V. Lifschitz. Nested abnormality theories. *Artificial Intelligence*, (74):351–365, 1995.

[McDermott and Davis, 1984] D. V. McDermott and E. Davis. Planning routes through uncertain territory. *Artificial Intelligence*, 22:107–156, 1984.

[Niemelä and Simons, 1997] I. Niemelä and P. Simons. Smodels - an implementation of the stable model and well-founded semantics for normal logic programs. In *4th International Conference on Logic Programming and Nonmonotonic Reasoning*, number 1265 in LNCS, pages 420–429. Springer-Verlag, 1997.

[Remolina and Kuipers, 1998] E. Remolina and B. Kuipers. Towards a formalization of the Spatial Semantic Hierarchy. In *Fourth Symposium on Logical Formalizations of Commonsense Reasoning, London*, January 1998.

[Remolina and Kuipers, 2001] E. Remolina and B. Kuipers. A logical account of causal and topological maps. Technical Report AI01-288, The University of Texas at Austin, http://www.cs.utexas.edu/users/qr, April 2001.

[Shanahan, 1996] M. Shanahan. Noise and the common sense informatic situation for a mobile robot. In *AAAI-96*, pages 1098–1103, 1996.

[Shanahan, 1997] M. Shanahan. Noise, non-determinism and spatial uncertainty. In *AAAI-97*, pages 153–158, 1997.

[Yeap, 1988] W. K. Yeap. Towards a computational theory of cognitive maps. *Artificial Intelligence*, 34:297–360, 1988.