

How Should Robots Think about Space?

Ernest Davis
Dept. of Computer Science
New York University
davis@cs.nyu.edu

November 18, 2015

Abstract

A robot's actions are carried out over time through space. The computational mechanisms that the robot uses to choose and guide actions must therefore at some level reflect the structure of time and space. This essay discusses what ontology for space and time should be built into a robot. I argue that the designers of a robot generally need to have in mind explicit ontologies of time and space, and that these will generally be the real line and three-dimensional Euclidean space, though it is often advantageous to project down to a lower-dimensional space, when possible. I then discuss which spatial or spatio-temporal regions should be considered "well-behaved" and which should be excluded as "monsters". I propose some principles that the class of well-behaved regions should observe, and discuss some examples.

1 Artificial Intelligence and Ontology

Artificial intelligence (AI), like computer science generally, is a branch of engineering, not science or philosophy or scholarship. The goal of the field, in ambitious moods, is to construct programs and robots that can behave as intelligently as people across a wide range of activities; in more modest moods, to build programs that can do fairly well at some activity that requires intelligence; but in any case to build something. Therefore, from the standpoint of AI research the question of whether X *actually* exists is not particularly important; the question is, will it be helpful to a robot¹ to think about X . The choice or design of an ontology is part of the software engineering process; and if an ontology is helpful in building a robot, that is all we ask of it. Of course, since robots work in the real world, there are often advantages to using ontologies that bear some relation to reality; however, verisimilitude is definitely a secondary consideration as compared to usefulness. Likewise, since human beings are incomparably more intelligent than any AI program that exists, the AI programmer may well want to pay attention to what psychologists, linguists, and anthropologists tell her about how the human mind works or how natural language works; however, she is not bound by those findings if she decides that they don't fit into her program design.

What ontologies a robot needs depends, of course, on what tasks it is supposed to be doing; and since human intelligence is open-ended and there is no end to the things that people can think about, there is likewise no end to AI ontologies. If the robot you are trying to build is supposed to answer questions about cell biology, then it will need an ontology that includes ribosomes and endoplasmic

¹For purposes of both brevity and concreteness, I will largely frame the discussion in this essay in terms of robots. However, most AI systems are not connected to physical robots; and much of the discussion in this paper applies equally to AI systems of many other kinds that deal with spatial information; e.g. to natural language processing systems that have to interpret spatial terms in text.

reticulums; if it is supposed to prove theorems in category theory, it will need an ontology that includes categories and morphisms; if it is supposed to read and understand the *Harry Potter* books, it will need to suspend disbelief and use an ontology that includes horcruxes, magic wands, and potions. Ultimately, a robot with full-fledged intelligence should be develop these ontologies by itself on the fly as it learns biology or category theory or reads *Harry Potter*, the way people do. How that is to be done is a deep problem for AI research, about which not very much is known; I am not going to address that in this essay.

Rather, this essay focuses on one particular type of core competency: how a robot could reason about spatial relations and about motion for a variety of everyday settings, and what ontology of space it should use for that. I will begin by making this concrete by giving a few examples of the kinds of everyday settings that involve spatial reasoning that I have in mind. I will then discuss the role of an ontology in building a program; the need for ontologies; the need for a distinctive spatial ontology; the argument for using multiple spatial ontologies; and the advantages and difficulties of using standard Euclidean geometry (\mathbb{R}^3) as a spatial ontology.

2 Examples

Let me begin by discussing a few examples of the kinds of reasoning I have in mind. These have been chosen to be natural and simple examples that illustrate a variety of geometric features, geometric entities, and forms of intelligent spatial reasoning.

2.1 Exhibit A: A cheese grater

Figure 1 shows a common make of cheese grater. (Readers who own this object and happen to be near their kitchen might find it useful to take it out and contemplate it directly.)

What do we want a household robot to understand about a cheese grater? Most obviously, the robot should know how to use the grater: Place the grater vertically with the base flush on a table top; hold the handle in one hand; hold the cheese flush against the appropriate side of the grater at the top; slide the cheese downward against the face of the grater, pushing inward; repeat until enough cheese has been grated; collect the grated cheese from the table inside the grater.

If all we want the robot to do is to carry out this procedure by rote, and we are willing to preprogram it for this particular procedure, then we may well be able to get away without any kind of general reasoning ability or ontology; all we would need is a set of rules mapping percepts to the corresponding action. We will discuss this further in section 4.

But a truly intelligent robot should be able to go beyond just rote execution. A truly intelligent robot would be able to adapt this plan to varying circumstances: If the table surface is slanted, then the base should be placed flush against it, leaving the grater off the vertical; in this case, the grater should generally be placed so that the surface being used is uppermost. If the grater would scratch the table and so cannot be placed directly on it, then either another object should be placed between the grater and the table, or the grater can be held the air with the other hand; this will require a tighter grip than grating in the usual way. If the holes are blocked on the inside, then they must be cleared before the grater can be used. The robot should be able to predict what will happen if the grater is used in some unusual way; for instance, if the food is pulled up the side of the grater rather than down, that will work for the small holes, but not for the circular “cheddar cheese” holes (in the foreground in figure 1). The robot should be able to understand verbal instructions for using the grater or to give verbal instructions. The robot should be able to apply its knowledge of this particular grater to other graters of related but different design. These kinds of tasks presumably



Downloaded from http://commons.wikimedia.org/wiki/File:Cheese_Grater.jpg 12/28/13 and reproduced under a Creative Commons licence

Figure 1: Cheese Grater

require the robot to understand the geometrical features of the grater and their relation to the physical interactions of grater, manipulators, and food being grated.

The geometry of the grater is rather complex: a complete description needs to specify the overall frustum shape plus handle, the shapes of the individual holes, and the patterns in which the holes are laid out. Ontologically, however, the spatial entity associated with the grater is fairly simple. The grater is, for virtually all intents and purposes, a rigid object. Therefore the regions that any particular grater occupies at two different times are congruent to one another, and the grater moves continuously as a function of time. Two graters of the same make occupy nearly congruent spatial regions; two different graters of this general category occupy regions that are not congruent but have a number of geometric features in common.

The geometry of the cheese as it is being grated is harder to characterize. For concreteness, let us consider a cheddar cheese being grated on the circular holes. There is the block of cheese held in the hand, the pile of strips that have been grated lying in a pile, and transient states when a strip of cheese has been partially cut off from the main block. It would be laborious, and certainly beyond the scope of this paper, to give a detailed characterization of the geometric properties of the cheese. However, it is clear that a truly intelligent robot of the kind we are contemplating, or a human using the grater, has some understanding of the geometric features of the cheese. For instance, the robot should know that the strips come off the part of the cheese that has been in contact with the grater, and that they leave semi-cylindrical indentations in that end of the cheese; it should be astonished if the block of cheese started wearing away at the opposite end or in the middle or if, after grating, the block of cheese had deep conical pits going inward into the block.

2.2 Exhibit B: A string bag

Figure 2 shows a string bag with vegetables. For our purposes, the issues are not very different in kind from those involved in the cheese graters; but we bring the example to illustrate what a broad



Downloaded from http://commons.wikimedia.org/wiki/File:String_bag.jpg 12/28/13 and reproduced under a Creative Commons licence

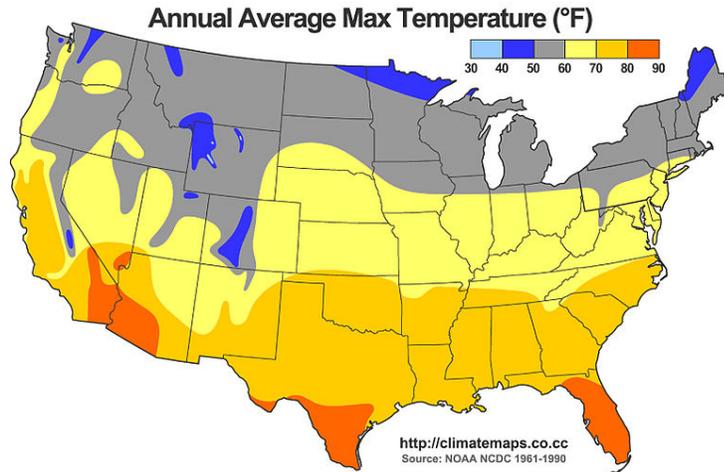
Figure 2: String Bag with Vegetables

range of spatial features are involved in even simple tools of this kind. Again, we would expect a truly intelligent robot, not merely to be able to use a particular bag in a standard way but to be able to reason about how the bag would work in unusual circumstances or could be used for unusual purposes.

As with the grater, the spatial ontology of any particular string bag can be characterized in terms of the space of regions that the bag could be made to occupy. If it is acceptable to approximate the individual strings as one dimensional curves, then a feasible region for the bag is a mapping of the strings into space that preserves the arc length of strings and the connections between them. If it is necessary to deal with the three-dimensionality of the strings, then the definition becomes more complex. Still more complex is the geometric characterization of string bags in general (i.e. networks of strings that will successfully function as a string bag)

2.3 Exhibit C: A Temperature Map

Our third example is the ground air temperature distribution over a large region, illustrated in figure 3. This illustrates that spatial knowledge is not always knowledge about a particular thing at a particular place. The air temperature is a continuous field over the cross-product of the surface of the earth times the time line. Unless the robot is working as a meteorologist, it probably does not need a detailed physical model of the weather, but even the robot on the street should have some basic spatial information available, much of it in the form of general rules rather than absolute rules: At any given time, temperatures are lower further north and in the mountains. At any place, date of the year, and time of day the temperature is predictable with reasonably high confidence in a range of about plus or minus 20 degrees Fahrenheit, though outliers are not very uncommon. The temperature difference between places ten miles apart is generally not very large, though again outliers can occur, particularly when there are strong differences in topography or a topographical boundary. The body of knowledge for the average person, other than purely chronological (“When I went to Chicago in January, it was -10° with a 30 mph wind”), and the range of types of reasoning are both probably much smaller than with a grater; but some significant reasoning must be done,



Downloaded from
http://commons.wikimedia.org/wiki/File:Average_Annual_High_Temperature_of_the_United_States.jpg
 12/28/13 and reproduced under a Creative Commons licence

Figure 3: Annual Average Max Temperature

and it involves knowledge of a form that is quite different from a grater.

3 What is the role of an ontology in a program?

The word “ontology” has become a buzz-word in AI, with a number of different meanings, some quite banal (e.g. what kinds of buttons are on a web-based order form). In this essay, what I mean by the word is the underlying conceptualization of the world that the programmer has in mind in building the program. The meanings of some of the data structures — those that refer to the outside world rather than the internal state of the program — can be described in terms of entities or relations in the ontology, and the operations on these data structures can mostly be justified in terms of the ontology. Importantly, the underlying ontology can be different from the class of entities that can be individually denoted by the program’s data structures.

Let me give a couple of examples. Almost all numerical calculations on computers that require non-integer values are done using floating point numbers. There are fairly complex rules for doing various kinds of mathematical computations over these, and still more complex rules for estimating the accuracy of your answer when you’re done. However, the underlying ontology, as I am using the term, is the traditional real line \mathbb{R} , in the sense that the real line is the “gold standard” to which floating point operations are essentially always referred. When one speaks of the accuracy of a floating point calculation, one means the accuracy as compared to the true value given in the theory of the reals; and a floating point algorithm is acceptable to the degree that it gives answers that correspond to the true real value. For example, if I give MATLAB the problem

$$(\sqrt{153} + \sqrt{147}) \cdot (\sqrt{153} - \sqrt{147})$$

I get the answer $6 - 1.243 \cdot 10^{-14}$. The reason that this is a good answer is that (a) it is very close to 6, which is the true answer in the theory of the reals; (b) the people who know about this kind of thing assure me that it would be difficult or impossible to do systematically better than MATLAB if one is committed to computing intermediate results in floating point form, rather than manipulating

the symbolic expression.

There are alternative methods for representing real numbers. Standard double-precision floating point notation represents the number “two and two-thirds” as (essentially) “ $0.101\dots$ (52 bits) $10 \cdot 2^2$ ”. One could use decimal notation and represent it as “ $0.26666\dots 67 \cdot 10^1$ ”. One could use an exact fraction: “ $8/3$ ”. One can go further and represent an arbitrary algebraic number x exactly² as a pair of a polynomial p for which x is a zero and an interval range that contains x and no other zero of p ; for instance “ $\langle x^5 - 7x + 1, [0, 1] \rangle$ ” represents the root of $x^5 - 7x + 1$ that lies between 0 and 1 (≈ 0.14287). The set of numbers that are exactly representable in each case is different, and so are the algorithms that carry out a given calculation. But the underlying ontology is the same: namely the real line.

Another, similar example: If a programmer is writing a “physics engine” to simulate interactions among rigid solid objects, she will probably begin by choose some specific idealized model of solid object physics: e.g. perfect rigidity, uniform density, Newtonian mechanics, Coulomb friction, inelastic collisions. This, then, is the ontology. The program will almost certainly not calculate the results exactly — for example, calculating the exact time of collisions between two rotating objects is in general very difficult — rather, the calculations are an approximation of this model, and the algorithms are tuned to achieve the best possible approximation of the model. If the programmer later needs to incorporate some phenomenon that is outside the scope of this idealization, she will probably first consider how to modify the model, or create a separate model for the phenomenon, rather than simply diving directly into the code and tweaking it to get the phenomenon. Certainly, in most cases, she is well advised to do so.

This definition of ontology gives a partial answer to what is known as the “symbol grounding” problem: What determines the meaning of a symbol in a computer program? On this view of ontology, a symbol means what the programmer intends that it should mean. That does not solve the general symbol grounding problem because it does not give any account of symbols that the program generates on the fly but, as far as it goes, it seems reasonable.

4 Is it necessary to have an ontology of space?

Is it useful for a robot to have an ontology of space? That is, is it a good idea for someone programming a robot to think through what kinds of spatial entities and spatial relations she intends to use, and how the data structures express those?

Perhaps not. Recently, remarkable successes in AI tasks have been achieved by systems in which the AI engineer presents a powerful, general-purpose machine learning program with a large collection of sample inputs and outputs, and leaves it to the program to work out all the intermediate complexities. For example, Cho et al. [1] have built a machine translation system, of state-of-the-art quality, which works purely by being shown parallel texts in English and French, with no built-in ideas of syntax, semantics, or meaning in either language.

In some cases, a machine learning algorithm of this kind will spontaneously develop internal structures that a human observer can see correspond to “natural” domain features or concepts. However, even when that occurs, that is only an interesting side effect; it is not an inherent or an intended part of the working of the algorithm. In many cases, when the learning algorithm is done, human observers really have no idea why it works or what its internal structures might “mean”; all they can say is that, experimentally, it carries out the task successfully.

²This data structure *by definition* denotes this exact algebraic number, because that is what the programmer means by the data structure. The programmer presumably provides functions that carry out exact calculations over these data structures, e.g. an addition function that takes two addends represented in this form and returns the sum, also represented in this form.

As of the date of writing, there have been few applications of this technology to robotic control and manipulation, but no doubt those are coming as well. Many enthusiasts for this kind of approach believe that it will suffice to achieve all of AI. If so, then the answer to the question in the chapter title would be “There is no need for a robot to think about space,” or at least, “There is no need for us to think about how a robot should think about space,” or indeed any other abstract concept.

This approach has a number of points in its favor. First, it passes the buck; rather than worry about what to do with space, we can simply rely on a machine learning facility which we will need anyway for other purposes. Second, it will presumably by design tend to home in on spatial properties that are important to the robot and ignore those that are not, which can be a difficult problem for hand-designed ontologies.

Third, there is a strong argument to be made that this must in principle be possible. After all, any concept that a person knows was acquired in one of three ways: he learned it himself from experience, or he was, explicitly or implicitly, taught it from a store of cultural capital, or it was innate. In each of these cases, the concept was learned from experience at some point at some level; either the individual learned it, or the culture learned it, or the species or one of its ancestors learned it through evolution. In any case, the concept must be learnable.

However, I personally suspect that there are limits to how far this kind of approach can be taken, with anything like the kinds of learning techniques and data sources currently under consideration. These techniques shine at narrowly defined tasks and achieve a certain level of success at broadly defined tasks; but achieving human levels of performance on complex tasks would seem to require sophisticated representations of knowledge and reasoning abilities that are substantially beyond the machine-learning technology we currently have. For example, this kind of technology might well be able to learn to grate cheese, but it is harder to see how it could achieve the kind of flexible planning and other kinds of reasoning described in section 2 without being separately trained on each instance.

Another alternative approach uses a programmer, rather than a machine learning algorithm, to deal with the spatial issues, but releases her from the responsibility of thinking through a coherent theory of space. Rather she programs catch-as-catch-can; recording the spatial information that the robot will need in unsystematic and probably subtly inconsistent data structures, and writing subroutines to deal with specific tasks.

There is certainly a lot of this kind of software out there, and some of it works well enough. But it is not what we who teach computer science recommend as good software engineering; it tends to lead to code that is brittle, inextensible, and hard to maintain. Particularly in AI applications, where one wants to attain the maximum degree of flexibility, and to be able to use the same knowledge for a multitude of different purposes, it is important to have data structures with well-defined meanings and functions that carry out well-defined computations or actions. For these kinds of definitions, a clear ontology is an invaluable asset.

5 Should a robot have a separate ontology of space?

Should a robot have a clearly delineated ontology of space as a separate conceptual entity or should space be just an aspect of other theories? The second alternative might be something along the following lines. The robot knows about concrete real-world entities – e.g. the cheese, the grater, itself — and it knows about relations among these. Some of these relations are strongly spatial e.g. “The block of cheese is touching the face of the grater”; some are non-spatial e.g. “This is mild, low-fat cheddar cheese”; and some combine spatial and non-spatial elements e.g. “I am pushing the cheese downward against the face of the grater.” But there are no abstract spatial entities — there

are triangular objects, but no triangles — and reasoning that is based purely on spatial relations does not form a separate or particularly distinct part of the overall reasoning apparatus. This is a somewhat holistic view or a Machian view.

I do not know of any attempts to develop an AI system or robot along these lines. From the standpoint of software engineering, there are good reasons for having a separate spatial reasoning component with abstract spatial objects; these are, in fact, much the same reasons that geometry, abstracted from its specific applications, became a separate discipline 2500 years ago.

- There is generally a clear distinction between purely geometric reasoning and reasoning that involves other issues. (As a point of contrast, it can be very difficult to draw a line between folk psychology and folk sociology.)
- The validity of a geometric inference is essentially independent of the application, though the importance of one or another type of geometric inference depends strongly on the application.
- There is a largely self-contained body of techniques used for geometric calculation and geometric inference which are not applicable to other subject matters.

Therefore placing geometric reasoning in a separate component with abstract geometric entities serves the software engineering goal of modularization; this significantly reduces the risk of unnecessary repetition of code, of inconsistencies, and of errors.

6 Should a robot have a single ontology of space or multiple ontologies?

Perhaps a robot should be able to think about space in a number of different ways corresponding to different tasks or circumstances. For instance, it could model space as three-dimensional when grating cheese; as the two-dimensional surface of a sphere when thinking about geography; and as a network of points and edges when planning a route through the subway. There is, in fact considerable psychological evidence that suggest that people use multiple spatial models in this way. In particular, people are often extremely poor at geometric reasoning in three-dimensions — for instance, at visualizing the three dimensional rotation of an object of any complexity — and prefer to project down to two dimensions, if that is in any way an option.

From an engineering standpoint the chief advantages of using multiple models is that a specialized model can be tuned to the task and can support more efficient algorithms. There are two drawbacks. The first drawback of multiple models are that when two tasks that use different models interact, information has to be translated from one model to the other. The second drawback is that if the wrong model is chosen for a task, incorrect or suboptimal results may be obtained.

In some cases these drawbacks hardly apply. For instance, the low-levels of vision proceed with little or no input from higher-level reasoning; the representation used here can therefore be entirely optimized to this particular process. The only constraint is that, at the end of the vision process, the output has to be presented in a form that other processes (e.g. manipulation) can use. However, aside from a few autonomous, specialized processes like low-level vision, intelligent spatial reasoning would seem to require that information be quite freely exchangeable between all kinds of cognitive tasks: high-level vision, physical reasoning, navigation, cognitive mapping, planning, natural language understanding and so on.

The general opinion here, with which I concur, is that the simplifications offered by working in specialized ontologies outweigh the costs of moving back and forth between them. In particular, the

ability to reduce problems from three to one or two dimensions seems to be extremely powerful, when it is possible. However, we do not actually know what difficulties may lie here; the technical issues of transferring information between spatial ontologies have not been extensively studied, and may be problematic.

7 What is the model of space?

What ontology of space is appropriate for tasks that substantively involve fully three dimensional space, like grating cheese? The traditional answer, of course, is \mathbb{R}^3 , three-dimensional Euclidean space. Indeed this is so universally accepted that many mathematicians would no doubt be surprised to learn that it is at all controversial. But it is. Many people are quite uncomfortable with \mathbb{R}^3 . And it certainly has features that give one pause: it is full of monsters such as Zeno's paradoxes, the space-filling curve and the Banach-Tarski paradox;³ it raises all kinds of unreal distinctions such as the difference between a topologically open region and its closure; and most of its elements cannot be finitely named.

It seems to me, however, that the conventional wisdom has settled on \mathbb{R}^3 for a number of very compelling reasons that more than make up for these drawbacks.

Correspondence to reality: The physicists tell us that Euclidean geometry certainly breaks down on the cosmic scale and may break down on the smallest scales (10^{-43} meters or so).⁴ However within the eleven orders of magnitude from about 0.2 mm to about 20,000 km that span the range of normal human interaction, it is spectacularly accurate to within the accuracy of measurement. Discrepancies can almost always be convincingly attributed to problems with the measurement technique rather than discrepancies between the world's geometry and \mathbb{R}^3 .

Continuity. If an object goes from one side of a line to the other side it must cross the line. If an object goes from outside a closed curve to inside, it must cross the curve.

Invariance under rotation: Rigid objects can rotate while preserving their shape, and the theory of space must have a way of describing this. This is not so easy to achieve.⁵ An interesting point of comparison is rotating a pixel image. In computer graphics, if you have an image of a long narrow bar, and you use a naive algorithm to rotate it, the result may well have a visibly jagged edge. If you carry out a long sequence of rotations, computing each new shape from the previous one, without reference to the original shape, the geometric information can be entirely lost; depending on the algorithm, the bar may split or may fatten into a circle. The graphicists have tricks for getting around this; these are based, partially on the \mathbb{R}^2 theory of rotation, partially on an understanding of how the eye reads images, partly on empirical findings of what visually works. But the point, for our purposes, is that the geometric aspects of these corrections is based on a gold standard which is the \mathbb{R}^2 theory of rotation; so in the terms discussed in section 3, the ontology used here is in fact \mathbb{R}^2 .

Moreover, a physical theory of solid objects should presumably allow them to rotate at a uniform rate, and to be in a well-defined spatial state at exact fractions of one complete rotation. In a real-based theory, this involves terms of the form $\cos(\pi/n)$ and $\sin(\pi/n)$ for integer n ; an alternative ontology must provide something analogous. A physical theory that gives rise to non-uniform

³The space-filling curve, the Banach-Tarski paradox, and some other technical terms are defined in a glossary at the end of the chapter.

⁴It also breaks down in the neighborhood of a black hole; but that is the least of the problems facing a robot in the neighborhood of a black hole.

⁵The Pythagorean proof that the ratio of the diagonal of an isosceles right triangle to its leg is irrational fundamentally depends on rotation. Comparing the lengths of two non-parallel line segments depends on rotating one of the lines so that the two are collinear. In affine geometry, where rotation is undefined, the ratio of the lengths of two non-parallel lines is also undefined.

rotation, like a solid pendulum, similarly entails the existence of various kinds of non-elementary functions. What one can effectively compute or what one needs to compute about these in any given situation is a secondary question; the critical point is that the theory of space and motion had better allow these to exist. If an alternative theory of space is to be used, then one has to find an alternative formulation of the laws of motion that similarly guarantees that future states are geometrically meaningful.⁶

Other measures give other kinds of real values; for instance, the ratio of the volume of a sphere to the volume of the circumscribing cube is $\pi/6$. However, here it seems to me there is more ontological wiggle room to escape these commitments; one could possibly have an ontology in which there are no exact spheres or no exact cubes or no exact measurements of volumes. The invariance of rigid objects under rotation is less forgiving; you want to be sure that, if you pick up the cheese grater and carry it around, its geometric properties remain the same. Of course, solid objects are not perfectly rigid, but it should be the business of the physical theory to tell us about the slight deformations and wear, not the geometric theory. If your spatial ontology makes it *geometrically impossible* for the grater to maintain its shape, it seems to me that you are off to a bad start.

It is sometimes argued that, since programs cannot actually manipulate real numbers, they cannot actually be working in \mathbb{R}^3 . However, on the view of ontology that I have taken in section 3, that argument does not hold water. The programs are using one representation or another, but what they are doing makes sense because the representations are approximations to \mathbb{R}^3 .

I am not, of course, arguing that the man on the street explicitly knows the theory of the reals as formulated by Cauchy, Dedekind, etc. I am not even arguing that he knows it implicitly in the same sense that a native speaker of a language has implicit knowledge of the grammatical rules. What I am saying is that the man on the street has an understanding of spatial relations, much of it non-verbalized, that he uses for physical reasoning and other cognitive tasks, and that, in general, \mathbb{R}^3 is an extremely good model for that understanding, much better than any other model we know of. Likewise, if a programmer wants to build a robot to do this kind of reasoning, she will do better to ground her program in \mathbb{R}^3 than any other alternative.

8 Welcoming friends while keeping out monsters

If we accept Euclidean geometry as our ontology of space, and we view the spatial extent of an object or other physical entity as a set of points in \mathbb{R}^3 , then we are faced with the problem of distinguishing which sets of points can be reasonably viewed as physically meaningful and which are purely mathematical constructs of no physical significance. Our general approach here follows the tradition of many generations of mathematicians;⁷ whatever we are doing, we build a castle with a moat and a drawbridge. Mathematical entities that will be well-behaved for what we are doing are welcomed in; monsters that will make trouble are kept out. Which mathematical entities are considered well-behaved and which are monsters depends on what we are doing inside.⁸

⁶For instance, an ontology of space in which all points have rational coordinates would be consistent with a theory in which all rotations are rational rotations i.e. (in 2D) rotations by angle θ such that $\cos(\theta) = a/c$ and $\sin(\theta) = b/c$ where a, b, c are a Pythagorean triple (see Glossary). But then the position at which a rotating object collides with a stationary object would in general be undefined; it would either have to overshoot (in which the two objects interpenetrate) or undershoot (in which the collision occurs while there is still a finite gap between the objects).

⁷Physicists and engineers rarely if ever worry about this; they use their physical intuition to avoid worrying about physically meaningless problems. But AI programs only have the physical intuitions that we program into them. The purpose of our analysis here is precisely to determine what *are* the physical intuitions that will enable the AI program to avoid worrying about meaningless problems.

⁸One semester as an undergraduate, I took both Analysis II, which was mostly the theory of distributions, and Functions of a Complex Variable. In Analysis, the *ne plus ultra* of well-behaved functions were C^∞ functions with compact support; distributions could be defined as linear functionals over that space. In Complex Variables, these

However, deciding on a proper admissions policy is not always easy. A successful policy must exclude the monsters, but it must admit all the mathematical entities that the work inside the castle requires. Moreover, one has to be sure that the work inside the castle will not *breed* monsters inside the keep; it is often very difficult to be sure that that cannot happen. I do not have a final answer to propose on this question for the problem of the reasoning about space and motion by a household robot. After some preliminary groundwork discussing physical idealization, we will first consider the policies for inviting friends and then consider a few examples of monsters and the issues they raise.

8.1 A preliminary comment on physical idealization

All of the examples that I will discuss below arise from the use of one or another idealized models of physical reality. The specific models in my discussion are, first, the general model of matter as continuous rather than atomic, and, second, the specific model of rigid solid objects as perfectly rigid; additional examples would involve other kinds of idealizations (e.g. liquids as perfectly incompressible). All of the particular mathematical anomalies that I discuss go away if a more sophisticated model is used. Thus each of our castles and moats is designed for the purpose of properly formulating a theory that in any case is a false idealization. One can certainly reasonably ask whether this is worth the trouble; might we not simply be better off working once and for all with the true physical theory?

I don't think so. First, quantum theory is, of course, by no means clear sailing logically, and even the reduction of thermodynamics to statistical mechanics has its slippery patches. More importantly for our purposes, the atomic/quantum model of matter is extremely difficult to use for large scale problems of mechanics, and it is therefore rarely used for these. There is a huge scientific and engineering literature that uses only a continuous model of matter, and then there is another huge literature that integrates continuous and atomic models. Therefore it is important to work out how continuous models can be reasoned about. Idealized models of physics such as continuum mechanics are necessarily somewhat wrong when one measures sufficiently accurately, and completely wrong when applied far outside their limits (e.g. on the very small scale), but they should at least be consistent, and, within their limits, they should not give answers that are nonsensical. We will return to this point in section 9.

There is also a historical/epistemological issue. The atomic theory was fiercely rejected by many physicists throughout the nineteenth century, and there were holdouts as late as the 1910's; the issue was not absolutely settled until experiments in the 1910's by Perrin and others that reliably determined the size of the atom.⁹ To the best of my knowledge, no one ever formulated a precise formal axiomatization of the continuous theory of matter, but presumably the theory is sufficiently logically coherent that this could have been done; and a precise statement of this kind would have had to address the issues that we raise below. If no such axiomatization could have been formulated, then one is left with the strange conclusion that the continuous theory of matter could have been excluded as impossible purely on the basis of observations at the medium scale with no experiments at the small scale. That seems counterintuitive.

8.2 What spatial regions do we need?

We turn now to the following question. If we accept that the spatial extent of a solid object is a set of points in \mathbb{R}^3 , what kinds of sets can meaningfully be considered as the extent of a single solid object? For instance: A sphere seems reasonable, a single point seems doubtful, a disconnected

were all monsters because they are not analytic.

⁹Both Dalton's chemical theory and the Maxwell/Boltzmann kinetic theory of heat have the unsettling feature that, though they are based on particles, they give no indication of how large the particles should be.

set seems very doubtful, as does the set of all points with rational coordinates. For convenience of reference we use the phrase “object-region” to mean a region that could plausibly be the extent of a solid object.

It turns out, in fact, that what needs to be done is not so much to identify a particular hard-and-fast rule for distinguishing object-regions, but rather to choose a collection of object-regions that will work together well.

Ultimately, a full specification of the ontology should precisely define the collection of object-regions to be used; but some aspects of this definition are rather arbitrary and we will not try to resolve them here. For instance: do we want to assume that all regions are polyhedra, or that all regions are smooth, or do we want an ontology that includes both smooth regions and polyhedra? The advantage of including polyhedra is that rectangular boxes and so on are useful entities with nice physical properties; and artifacts that are actually polyhedral to a very high precision are quite common. The drawbacks of including polyhedral regions are that some aspects of physics are defined in terms of the normal to the surface, and for non-smooth objects, since there is no unique normal, one has to work around this (which can be done, but it takes work); and that most though not all ostensibly polyhedral objects actually have visibly curved edges if you look closely enough. The advantages of restricting the class of regions to polyhedral objects are, first, that polyhedra support comparatively efficient algorithms when there are exact specifications; and, second, that the class of polyhedra has nice closure properties e.g. it is closed under union. The advantage of allowing non-polyhedral objects is that sphere, ovoids, helical screws and so on are also useful physical entities; they can be manufactured or arise in nature to very high precision; and it is a substantial nuisance to have a theory that posits that only high-precision polyhedral approximations for these exist. We will not here attempt to decide this.

Rather, we will enumerate some desirable features of the class of regions that correspond to solid objects.

1. The ontology includes the shape of any solid object that can be manufactured out of plastic, stainless steel, paper, string, and so on. Of course, sufficiently complex shapes can be hard to apprehend; but it would be strange to have an ontology that ruled them out as geometrically meaningless. Geometrically, we can interpret this as the constraint that for any interior-connected geometric complex in \mathbb{R}^3 there is an object-region that is an arbitrarily good approximation, or at least an approximation¹⁰ that is accurate to within the limits of perception (a fraction of a millimeter). Any such shape can be constructed by gluing together little tetrahedra.
2. Solid objects can be moved around rigidly in space. Hence, the ontology should be closed under rotation and translation. It should be closed under reflection as well; otherwise, when you look at the reflection of an object in a mirror, you could see an inconceivable shape.
3. Solid objects deform continuously under physical processes such as thermal expansion and bending. It will certainly be convenient to view these as truly continuous changes, rather than as discrete.¹¹ Therefore the ontology should be rich enough to support continuous change of shape. For instance the class of polyhedra with rational coefficients would not satisfy this constraint.
4. Solid objects that meet in a face can be glued together. Geometrically, this means that if A

¹⁰There is more than one measure of approximation for regions in Euclidean space; Hausdorff distance (see Glossary) will do fine here.

¹¹The amount of change that physics predicts can certainly be very small, even for reasonable sized objects undergoing perceptible changes. For instance if a 1 cm cubed object is moving at a speed of 1 cm per second, the relativistic contraction is $5 \cdot 10^{-23}$ m.

and B are regions in the ontology that meet along an extended face, then the union $A \cup B$ is either in the ontology or very closely approximable within the ontology.

5. Objects can be cut or broken. It is not clear how these processes are best categorized geometrically. However, however you characterize these processes, they must generate shapes that are acceptable object-regions. This is a joint constraint on the theories of breaking and cutting and the definition of object-regions.
6. Invariance under scale transformations. This constraint is more doubtful; it is more a geometric convenience than the requirement of a physical theory. It is not possible to apply scale transformations to physical objects; the laws of physics are not scale invariant and the particle model of matter is not scale invariant. In a continuous model of matter, scale invariance would be natural, but could be avoided if that were desirable. For instance, one could posit that any solid object has to contain a sphere of radius 0.01 mm in order to be physically coherent.¹²

8.3 Other physically meaningful regions

There may be point sets that we wish to describe as physically meaningful that cannot be the extent of a solid object; these may well satisfy different constraints. For instance, as we will discuss in section 8.7, we may well wish to posit that the extent of a solid object cannot be a lower-dimensional region such as a geometric point, curve, or surface. However, if we wish to describe the geometry of the contact between two solid objects, then that contact can be a single point, a curve, or a two-dimensional surface, or the union of a collection of each of these. One might wish to posit that all solid objects are bounded, but, for reasoning in a terrestrial context, to idealize the atmosphere as occupying an unbounded region. One might wish to reason about the swath swept out by an object over time; what class of regions that would include depends on what kinds of motions are possible. For each of these, if they are to be included, one has to reason separately what kinds of regions are needed and what kinds are monsters to be barred; the criteria are likely to depend to some extent on the choice of solid object regions, but may also involve further considerations.

8.4 Out and out monsters: Set of points with rational coordinates/Banach-Tarski paradox

This is, naturally, often enough what children mean when they ask, “Is it true?” They mean, “I like this, but is it contemporary? Am I safe in my bed?” The answer, “There is certainly no dragon in England today,” is all they want to hear.

– J.R.R. Tolkien, “On Fairy-Stories”.

We now turn to the monsters. We begin with two well-known monsters that presumably have no physical significance. Relative to some fixed coordinate system, let Q be the set of points in $[0, 1]^3$ with rational coordinates. Let B be one of the non-measurable components of the Banach-Tarski paradox. We can admit Q as a set — excluding it would inconveniently complicate the formulation of set theory. If we need the axiom of choice (which is debatable) we can admit B as a set.¹³ However, it seems safe to say that neither Q nor B components have any physical significance. Therefore, for example, our robot does not have to be able to solve problems that start “Suppose that there is an object of shape Q on the table.”

¹²More precisely, one would want to posit that any bump or indentation contains a sphere of 0.01 mm; in the technical jargon, the region is r -regular with $r = 0.01$ mm (see Glossary).

¹³John Stillwell’s chapter in this collection discusses in great depth the intricate relations between set theory and the theory of the real numbers and Euclidean space.

However, the price of excluding Q and B is, as we have said, that we have to be sure that we do not breed them inside the castle. Specifically, we have to be sure that our theory of fabricating objects does not include any means of fabricating an object of shape Q ; and that our theory of slicing an orange does not allow us to slice out a piece of shape B . Those constraints certainly seem plausible; proving them, however, may not be easy.

8.5 A monster we may breed but may be able to live with: The infinitely bouncing ball

Suppose that a ball is bouncing on the ground with a partially elastic collision. Let us idealize the ball and the ground as perfectly rigid. Suppose that the coefficient of restitution is $1/2$. Then each bounce is half as high as the previous one, and therefore takes time $1/\sqrt{2}$ as long as the previous bounce. The ball therefore bounces infinitely often times before coming to rest at time $\sqrt{2}/(\sqrt{2}-1)$. Thus we have here a rather Zenonian behavior (in sense of Zeno's paradoxes) emerging from a perfectly natural physical idealization; we have bred the monster inside the castle.

Of course, in reality, this analysis breaks down at the point when the time between bounces is comparable to the period of the fundamental vibrational mode of the ball. At that point you have something like a spring bouncing on the ground and the behavior is dominated by the damped harmonic oscillation of the ball itself. (It is nonetheless challenging to establish that no Zenonian behaviors are possible in this improved model.)

The question is, though, is this Zenonian behavior so monstrous that we are forced to abandon the convenient idealization of perfectly rigid objects in order to avoid it? There is a case to be made that it actually has no intolerable consequences; the velocity is well-defined and finite (except at the moments of collision, but that's true of any theory of rigid objects collision) as is the momentum and the energy, and there is no actual difficulty predicting the behavior either at the instant when the bounces converge or past it. Having infinitely many events in finite time does rule out a reasoning strategy in which one always reasons from one distinguished event to the next; but that is a poor strategy whenever there are large numbers of unimportant events, even if that large number is finite. (For instance, if you shake a can half full of sand, you do not want to reason about every collision between every pair of grains of sand.)

Note finally that, if the ball has additionally some horizontal motion, then the swathe that it sweeps out will have some geometrically anomalous features. Whether this is OK depends, first on whether we are concerned about swathes at all, and second on what are the well-behavedness conditions for swathes.

8.6 Each beast is tame, but the herd is monstrous: Zeno's box of blocks

Consider the hypothetical situation shown in figure 4. An open box of depth 1 is filled with a collection of cubical blocks. The blocks in the top row have side $1/2$, those in second row have side $1/4$, those in the third row have side $1/8$, and so on.

Is this a monster to be barred? Probably, but that is not entirely certain. Certainly a force-based formulation of Newton's theory does not work here; since there are no bricks that touch the bottom of the box, there is nothing for the bottom of the box to be exerting an upward force on. On the other hand, an energy-based formulation works fine; the total potential energy is well-defined and finite, and the current state is obviously a local minimum. Note also that, in a continuous model of solid material, each block has a perfectly reasonable shape; it is just the overall collection that is problematic.

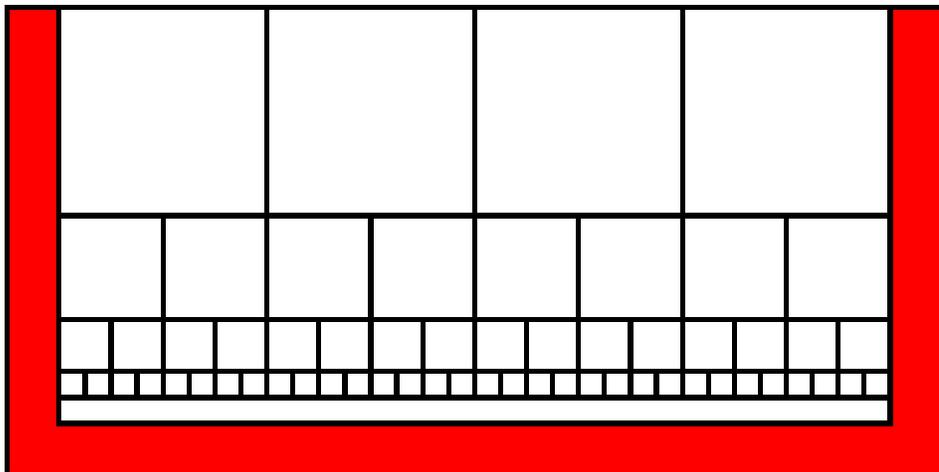


Figure 4: Zeno's box of blocks

The more serious difficulty is making sure that this kind of monster is not bred within the castle. For instance, one's theory of breaking objects or cutting objects might in some cases generate an infinite collection of objects of this kind. Someone might, for example, propose a fractal model of his beard in which there are k hairs of length 2^{-k} or some such. Then if he shaves . . . Or a continuous theory of liquids might generate a fractal distribution of droplets; if these then freeze, there are infinitely many pieces of ice.

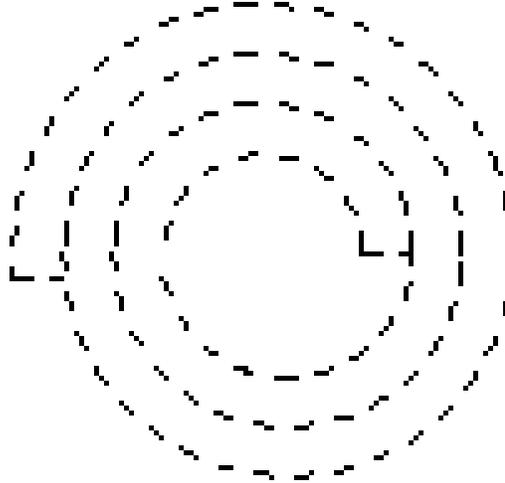
All these problems vanish, of course, once one switches from a continuous theory of matter to an atomic theory; but, as discussed above, the costs of avoiding a continuous theory are large.

8.7 Ghosts: Solid objects of lower-dimensional shape

Point objects, cords that are one-dimensional curves, and plates that are two-dimensional surfaces are common in the exercises for physics textbooks. Should a robot be prepared to find them in the kitchen?

I would argue that not only are these obviously impossible in an atomic theory, they are more trouble than they are worth in a continuous matter theory. For instance, suppose you drop a grain of sand on to a paper plate and it sits there for a while. You can now take it off the top of the plate, but not from the bottom. If sand and plate are modelled as objects of some thickness, this follows directly from the constraints that the interiors of object cannot overlap and that objects move continuously. However, if the sand is a point object and the plate is a two-dimensional, then the natural geometric interpretation of the sand on the plate is simply that the point is in the surface. Extra state must be added to keep track of which side the sand is on; keeping track of this state becomes more difficult if the plate's shape changes (e.g. by being folded). Characterizing the possible interactions of two cords characterized as one-dimensional curves is even more challenging. The various kinds of infinite densities associated with these objects are also a nuisance to deal with.

Therefore, it seems to be that a robot does best to consider lower-dimensional objects as purely calculational conveniences approximating a reality that is in fact solidly three-dimensional. Specifically, the qualitative behavior (the sand stays on one side of the plate) is worked out based on a three-dimensional theory. If numerical values are needed, it may be convenient to do calculations based on a lower-dimensional model.



The boundary is dashed to indicate that it is excluded from the spatial extent of the rope.

Figure 5: Coil of Rope

8.8 Yoking a monster to the plow: non-regular open regions for solid objects

Perhaps the most obviously physically meaningless distinction that comes with the geometry of \mathbb{R}^3 is the distinction between a topologically open region, which does not contain its boundary points, and a topologically closed region, which does contain its boundary points. What could possibly be the physical difference, for any purpose, between the closed unit sphere $\bar{B}(\mathbf{o}, 1) = \{\mathbf{x} \mid \text{dist}(\mathbf{x}, \mathbf{o}) \leq 1\}$ which includes its boundary and the open unit sphere $B(\mathbf{o}, 1) = \{\mathbf{x} \mid \text{dist}(\mathbf{x}, \mathbf{o}) < 1\}$ which does not include its boundary? By the same token it seems at first glance obvious, it is assumed in much of the AI spatial reasoning literature, that a physically meaningful region has no lower-dimensional pieces deleted or sticking out (topologically regular). What could be the physical difference between the open unit ball, and the same ball missing a point, or a line or a two-dimensional slice?

Nonetheless, there is an argument to be made that solid objects should be idealized specifically as open sets rather than closed sets, and moreover as open sets that are not necessarily topologically regular. Consider a situation where a flexible solid object is bent around back on itself; e.g. a coil of rope. One can create an idealized model of this situation by viewing the region occupied by the object to be the open set that it fills, excluding the boundaries where one part touches another. One then formulates the associated physical theory as stating that two parts $P1$ and $P2$ of object O cannot be easily separated if the boundary surface where $P1$ meets $P2$ is part of the interior of O , but may be easily separable if the boundary surface is part of the boundary of O (figure 5). For example, the rope in Figure 5 can easily be uncoiled, but not easily split into two pieces of rope.

It may seem that this is too clever by half, and that it will sooner or later get you into trouble, like having a carriage pulled by a hippopotamus. But I think it is actually OK. Essentially, we're using a trick from the old days of programming, in which programmers would find a few unused bits and stuff some extra information into them. The theory of the real geometry gives us, in the distinction between including or excluding boundary points, an extra degree of freedom that we do not otherwise need for physical theories, and we're using it to encode, so to speak, the important distinction between surfaces across which an object does and does not cohere to itself.

8.9 How could one verify a choice of well-behaved regions?

Our discussion above is quite unsystematic; shapes of category A are monsters that will get you into trouble, whereas as far as I can tell it should be safe to consider shapes of category B as well-behaved. How could one achieve greater confidence that a choice of well-behaved conditions will work out, ideally in terms of a theorem?

Ideally, I think, the kind of theorem that would answer this question would be along the following lines. Suppose that the programmer has worked out the following:

- a. A proposed physical theory e.g. “Newtonian physics of rigid solid objects with Coulomb friction and collisions with varying degrees of elasticity, in a uniform gravitational field.” The theory does not have to be a deterministic theory; a partially specified theory is also an option. The theory may well allow exogenous influences; for instance, it may allow the possibility that an agent can move blocks around, without otherwise providing a model of the agent. If so, then the theory presumably also specifies constraints on these influences
- b. A proposed collection of object-regions; e.g. “Interior-connected, bounded, topologically closed regular polyhedra.” This would presumably satisfy the conditions discussed in section 8.2, or at least those that derive from physics covered in (a).
- c. A collection of problems that are considered well-posed. The standard form of a well-posed problem is an initial-value problem: Given an initial situation and exogenous influences satisfying the physical constraints in (a) where objects have the shapes in (b), how does the system evolve over time? But there may also be other forms of problems to be solved, depending on the theory and its role in reasoning.

The theorem to be proved, then, would have the form of showing that all well-posed problems have a solution; in the case of initial-value problems, that there is a trajectory consistent with the problem specification and the physical theory. Such results have of course been obtained and studied in great depth for systems of time-dependent ODE’s and PDE’s. I do not know of any such results for the rather different kinds of physical theories that tend to arise for the household robot; few such theories have been developed in sufficient detail to support such a theorem. In particular, the characterization of exogenous influences in a theory with a robot seems to be difficult.

This is, in any case, meta-analysis; that is, it is not reasoning that we need the robot to do, it is meta-reasoning to prove that the robot’s reasoning will not arrive at a contradiction. Accomplishing it will in no ways help the robot, it will merely give us more confidence in the robot.

9 Two Objections

Some readers will find the above discussion not merely unsatisfying but well-nigh preposterous. There are, I think, two major objections.

The first objection returns to the issue we addressed in section 8.1. “These problems have got to be all pseudo-problems; it is simply not credible that building a robot that can manipulate cheese graters and string bags requires thinking about topologically closed and open sets. The problems arise from pushing on the physical idealizations too hard and in the wrong direction. The fact that when you do that, the idealization will break in a strange way is neither surprising nor important; and attempting to fix the broken theory with bubble gum and rubber bands is neither computer science nor anything else. The correct approach to the problem is to find out how an automated reasoner would use multiple idealizations, how it would choose between them and integrate them.”

I would say that there is some truth in this; integrating multiple idealizations is certainly a very important problem and a very poorly understood one, and it is likely that once we have attained a better understanding of multiple models, our viewpoint on these problems will shift to some degree; major conceptual advances are apt to have that effect. But it seems to me that the seeming plausibility of the above arguments rests on a false extrapolation from people. For people, it is obvious when a physical prediction is nonsensical, and it is much easier to detect nonsense than to determine the consequences of a physical theory. Nothing is inherently obvious for a robot; a “sanity check” on a prediction is just another kind of physical reasoning. In an automated reasoner, a sanity check is just another physical theory, and in many ways a harder one to implement than a physics engine. It may have some privileged place in terms of how multiple theories are integrated, but as far as we know it is not different in kind from the theories that it is critiquing.

For example, in the bouncing ball example, the conclusion that the ball bounces infinitely many times may intuitively seem very strange, and therefore it would seem that the theory of rigid solid object should be rejected here. Perhaps it should; but that may not be at all easy for an automated reasoner to see. After all, in many ways this case is particularly well-suited to the rigid body approximation, since the collisions are all very low energy; generally a rigid-body approximation does *better* with low energy collisions than with high energy collisions. Moreover, on standard measures, such as the mean squared distance between the prediction and behavior, the rigid body model does fine; it predicts accurately that the ball will bounce lower and lower and then come to rest. Where it does badly is in predicting the number of collisions (infinite vs. finite); but the number of collisions is not a standard measure for evaluating the quality of predictions, nor necessarily an important one. The sanity check module cannot automatically send up a red flag each time it sees an infinity in a prediction; as Zeno’s paradox of Achilles and the tortoise shows, perfectly simple physical behavior can be full of infinities, depending on how it is construed.¹⁴ In short even once we have a metatheory of multiple models, the kinds of problems that we have looked at above are likely to persist, though they may well take a different form.

I am not claiming that a programmer of a robot *has* to think through these ontological issues; only that it will be helpful to do so. As an imperfect analogy, a native speaker of English does not have to be consciously aware of the difference between regular and irregular past tense forms of verbs or the difference between a gerund and a participle; but these distinctions are useful, both in characterizing the linguistic behavior of speakers and in teaching English as a second language.

The second objection is that the above analysis is nothing more than a defense of the status quo in conventional mathematics and demonstrates only that Ernest Davis, having spent much effort over half a century becoming indoctrinated in that status quo, is now too hidebound and unimaginative to accept any alternative.

There may be some truth in that as well, but there it seems to me that the burden of proof is on the challenger. Over the course of three and a half centuries \mathbb{R}^3 has proved itself an extremely powerful spatial ontology for doing all kinds of physical reasoning at all scales except the extremely small and the extremely large. If an alternative can be proposed which is adequate for the purposes of a robot’s physical reasoning, then the AI community, including myself, will do our best to judge it fairly. Until then, we are stuck with \mathbb{R}^3 , which means that we have to do something about the monsters. If you want to do carpentry, then you are better off with a power saw that is powerful enough to cut wood but has inconvenient safety features than with one which avoids the potential dangers from the start but can’t cut wood.

¹⁴Slight variants on standard AI reasoning techniques actually do generate a version of the Achilles paradox. Suppose that you posit there are important times, important places, and important events; when an object arrives at an important place, that is an important event; the time when an important event occurs is an important time; and the place of an object at an important time is an important place. Then, as Achilles chases the tortoise, infinitely many important events occur, by the standard argument.

10 Beyond physical reasoning: Formal models, friends, and monsters in other aspects of automated reasoning

HORATIO: O day and night, but this is wondrous strange!

HAMLET: And therefore as a stranger give it welcome.

– *Hamlet*, Act I, scene 5

Among the many subject matters that a general artificial intelligence has to deal with, reasoning about physical objects and their spatial relations is one where the relation between the real world and the mathematical model is clearest and works best. In many other areas, there either is no model at all or the gap between practice and theory is much larger.

In particular, the issue of deciding which problems an AI system needs to address and which it can reject as ill-formed or meaningless arises in many different context and forms. In natural language understanding: Does a text understanding system need to understand newly coined words, expressions, or metaphors? Does it need to deal with the Liar sentence? Does it need to be able to get out of *Jabberwocky*, *Finnegan's Wake*, or the first chapter of *The Sound and the Fury* what a human reader gets out of them? In computer vision: Does a computer vision system need to be able to read a political cartoon, to characterize an animal it has not seen before, to recognize the Easter Island statues as people, to interpret a Magritte, a Monet, a Picasso, a Jackson Pollack? In planning: Is “Buy low, sell high” an executable plan? How about “Do something your adversary doesn’t expect,” or “Carry out an experiment whose result will surprise you”? In reasoning about people: Does a theory of “naïve psychology” need to subsume daydreams, nocturnal dreams, self-destructive behavior, or delusional behavior?

The ability of people to dramatically expand their horizons, to embrace what is radically new, grounding it in previous experience without reducing it to previous experience, is among the most magical characteristics of human cognition, and one of the abilities that we are furthest from attaining in AI. The ability to create mathematics and to understand mathematical entities and mathematical arguments is an extraordinary manifestation of this gift.

Glossary of Terms

Banach-Tarski paradox: It is possible to divide the sphere of radius 1 into finitely many pieces that can be rearranged to form two spheres of radius 1. The Banach-Tarski theorem can be proven given the axiom of choice. There are models of set theory that exclude the axiom of choice in which the statement is false.

Hausdorff distance: Let \mathbf{A} and \mathbf{B} be subsets of \mathbb{R}^n (or for that matter subsets of any metric space). The Hausdorff distance between \mathbf{A} and \mathbf{B} is a measure of how different these two sets are. Specifically it is the maximum of two quantities: (a) the maximum (or supremum) distance from any point $\mathbf{a} \in \mathbf{A}$ to the closest point in \mathbf{B} ; (b) the maximum distance from any point $\mathbf{b} \in \mathbf{B}$ to the closest point in \mathbf{A} . The Hausdorff distance is a metric over closed bounded subsets of \mathbb{R}^n .

Pythagorean triple: Three integers a, b, c such that $a^2 + b^2 = c^2$; e.g. 3,4,5 or 12,5,13.

r -regular: Let $r > 0$ be a real number. A region P in \mathbb{R}^n is r -regular if, for every point \mathbf{p} , there is a sphere of radius r containing \mathbf{p} whose interior is entirely inside P or there is a sphere of radius r containing \mathbf{p} whose interior is entirely outside P [2].

Space-filling curve: A one-dimensional curve that includes every point in an extended two-dimensional region. More precisely, a continuous function from the unit interval $[0,1]$ to the unit

square $[0, 1] \times [0, 1]$.

Topologically regular: A topologically regular open region is equal to the interior of its closure; intuitively, it does not have any lower-dimensional pieces — points, curves, or surfaces — missing. A topologically regular closed region is equal to the closure of its interior; intuitively; it does not have any lower-dimensional pieces sticking out.

Additional Reading

R. Casati and A. Varzi, *Holes and Other Superficialities*, MIT Press, 1994.

E. Davis, “The Naive Physics Perplex”, *AI Magazine*, vol. 19, no. 4, Winter 1998, 51-79.

A. Galton, *Qualitative Spatial Change*, Oxford U. Press, 2000.

References

[1] K. Cho et al. “Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation,” *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014.

[2] P. Steldinger *Image Digitization and its Influence on Shape Properties in Finite Dimensions*, IOS Press, 2008.