

Solving Everyday Physical Reasoning Problems by Analogy using Sketches

Matthew Klenk, Ken Forbus, Emmett Tomai, Hyeonkyeong Kim, and
Brian Kyckelhahn

Qualitative Reasoning Group, Northwestern University
1890 Maple Avenue, Evanston, IL, USA
Contact: m-klenk@northwestern.edu

Abstract

Understanding common sense reasoning about the physical world is one of the goals of qualitative reasoning research. This paper describes how we combine qualitative mechanics and analogy to solve everyday physical reasoning problems posed as sketches. The problems are drawn from the Bennett Mechanical Comprehension Test, which is used to evaluate technician candidates. We discuss *sketch annotations*, which define conceptual quantities in terms of visual measurements, how modeling decisions are made by analogy, and how analogy can be used to frame comparative analysis problems. Experimental results support the plausibility of this approach.

Introduction

Understanding common sense reasoning about the physical world is one of the goals that motivated qualitative reasoning (QR) research from the beginning. Despite its success at capturing many important aspects of reasoning about technical domains, little progress has been made on applying QR ideas to common sense reasoning per se. A key difference between common sense problems and reasoning in technical domains is breadth. In domains such as electronics or thermodynamics, a small library of components and relationships between them suffice to describe the systems of interest. This is not true for everyday reasoning, where the number of types of entities that can potentially be involved is at least in the tens of thousands. A second important feature of common sense reasoning is robustness, by which we mean the ability to draw conclusions even with partial knowledge. QR already provides one piece of the puzzle, by enabling natural conclusions to be drawn without detailed numerical information. However, existing QR techniques tend to assume complete and correct domain theories, which are applied to construct situation-specific models as needed to solve a given problem. By contrast, (1) mental models research [14] suggests that people's models are often incomplete and incorrect, and (2) psychological evidence suggests that people often miss opportunities to apply relevant principles in everyday life. How then can we explain the robustness of common sense reasoning?

Forbus & Gentner [7] suggest that the use of analogy provides a missing piece of the puzzle. Here we do not refer to cross-domain analogies (e.g., seeing heat as a liquid), which are rare, can be risky, and are prized precisely because good ones require considerable insight. Instead, we focus on within-domain analogies, where a new situation is understood in terms of a prior example (e.g., seeing a person pushing a wheelbarrow as like another person pushing a different wheelbarrow, or a shopping cart). The prior example might have been understood in terms of the person's domain theory, but it might have also been understood in terms of an explanation that is completely specific to that example (e.g., the stability of this building decreases as its height increases). One is reminded of similar experiences, and the explanation of those experiences is applied to the current problem. Common sense is learned via experience by accumulating examples. Examples can be used directly as analogs and generalized to form more abstract knowledge.

Using sketches in common sense reasoning is a particularly good venue for exploring these ideas because sketches are concrete. A sketch depicts a particular system, and general principles are articulated in terms of how they apply to this specific situation. Sketches and diagrams are used heavily in teaching and learning about physical domains. For example, their importance in physical thinking is indicated by the structure of the Bennett Mechanical Comprehension Test (BMCT), an examination used to evaluate applicants for technical positions. BMCT problems consist of diagrams depicting physical scenarios, with multiple-choice questions about their qualitative properties. The BMCT is extremely broad, including questions about statics, dynamics, heat, and electricity, all stated in terms of everyday situations. The BMCT is also used by cognitive psychologists as an independent measure of mechanical aptitude and spatial ability. In QR terms, BMCT problems can be divided into two aspects: model formulation and computing the answer from the model. As indicated below, computing the answer can typically be done by existing QR techniques, with one or two extensions. The most serious difficulty is in formulating the model. The compositional modeling methodology [3, 21] assumes complete and correct domain theories, and says little about the mapping from structural descriptions to structural abstractions. We claim that the problem of mapping from the broad vocabulary of entities and relationships used in the everyday world to a more refined set that can be used to describe conceptual models is central to understanding common sense reasoning.

This paper describes a system we have constructed which solves problems from the BMCT, using the similarity-based qualitative reasoning model outlined above. It uses a new cognitive architecture, *Companion Cognitive Systems* [10], which is applying these ideas more broadly. Here we focus on three novel qualitative modeling ideas that were needed to build this system: (1) *sketch annotations* define conceptual properties in terms of visual quantities, (2) *using analogy to apply abstractions and models to structural descriptions*, and (3) *using analogy to frame comparative analyses*. We start with a brief review to ground the discussion, and then describe each idea in turn. The overall architecture of the system is described next, followed by some experimental results. We end with related work not mentioned elsewhere and a discussion of future work.

Background

Sketching is a powerful way to work out and communicate ideas. The *nuSketch* model [11] takes sketching to be a combination of interactive drawing and conceptual labeling. While most sketch understanding systems focus on recognition, *nuSketch* systems are based on the insight that recognition is not necessary in human-to-human sketching. The *sketching Knowledge Entry Associate* (sKEA) [12] is the first *open-domain* sketch understanding system. Anything that can be described in terms of sKEA's knowledge base can be used in a sketch. sKEA's knowledge base consists of a 1.2 million fact subset of Cycorp's Cyc KB¹, which includes over 38,000 concepts, over 8,000 relations, and over 5,000 logical functions. We have added to that our own representations of qualitative physics, visual properties and relationships, spatial knowledge, and representations to support analogical reasoning, but the vast majority of the content that we deal with was independently developed. The breadth of this KB makes it an excellent platform for exploring common sense reasoning.

Glyphs are the basic constituent of sketches. A glyph consists of its *ink*, which indicates its visual properties, and its *entity*, which is the thing depicted by the glyph. Entities can be instances of any of the concepts in the KB. Sketches are further structured into *bundles* and *layers*. In this paper all of the sketches involve only a single bundle, so we ignore bundles. Layers decompose different aspects of a subsketch, e.g., two systems being compared side by side would be drawn in the same bundle, but each system on a different layer. sKEA computes a variety of visual relationships between glyphs based on ink [23]. For example, RCC8 qualitative topology relationships [1] are computed for every pair of glyphs in a layer.

We use Gentner's structure-mapping theory of analogy and similarity [13]. In structure-mapping, analogy and similarity are defined in terms of structural alignment processes operating over structured representations. The

output of this comparison process is one or more *mappings*, constituting a construal of how the two entities, situations, or concepts (called *base* and *target*) can be aligned. A mapping consists of a set of *correspondences*, a set of *candidate inferences*, and a *structural evaluation score*. A *correspondence* maps an item (entity or expression) from the base to an item in the target. A candidate inference is the surmise that a statement in the base might hold in the target, based on the correspondences. The structural evaluation score indicates overall match quality.

We use two cognitive simulations based on structure-mapping theory here. The *Structure-Mapping Engine* (SME) does analogical mapping [2]. SME uses a greedy algorithm to compute approximately optimal mappings in polynomial time [8]. The base and target descriptions can be pre-stored cases, or dynamically computed based on queries to a large knowledge base [19]. MAC/FAC [9] models similarity-based retrieval. The first stage uses a special kind of feature vector, automatically computed from structural descriptions, to rapidly select a few (typically three) candidates from a large case library. The second stage uses SME to compare these candidates to the probe description, returning one (or more, if they are very close) of them as what the probe reminded it of. As performance systems, both SME and MAC/FAC have been used successfully in a variety of different domains, and as cognitive models, both have been used to account for a variety of psychological results [6].

Sketch Annotations

In everyday sketching, people annotate sketches of physical entities with conceptual information that would not appear in the actual situation. In architectural drawings, annotations indicate distances between walls and the widths of windows. In sketches explaining principles, annotations indicate important properties, such as physical quantities (e.g., width of the base of a ladder) and where forces are applied. We introduced *annotation glyphs* to provide this capability in sKEA. Like other glyphs, an annotation glyph consists of its ink and the entity it is representing. However, annotation glyphs also refer to one or more other glyphs in the sketch, depicting the entity (or entities) that they are providing information about. We call these the *references* for the annotation glyph.

We have found three kinds of annotation glyphs useful to date. *Linear annotations* indicate linear distances, either along a single reference or between two references. Two special subclasses of linear annotations are *X-coordinate* and *Y-coordinate annotations*, which refer to the projection of the measurement onto the appropriate axis. *Force annotations* indicate where, and in what direction, a force is applied to the reference. *Rotational annotations* indicate the references direction of rotation. Figures 1a and 1b illustrate each type of annotation.

¹ The conventions used in this KB are documented at www.opencyc.org, although our KB includes a larger subset of Cyc than OpenCyc does.

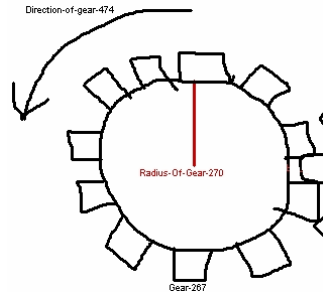


Figure 1a: Gear with length and counter clockwise rotation annotations

Annotation glyphs express relationships about their reference(s) in three ways. Force annotations have their direction and application surface computed from their ink and their referenced glyphs. Rotational direction annotations assume qualitative rotational motion information about the reference. Finally, linear annotations compute a distance measurement using *anchor points* on the reference. Anchor points are used to specify which parts of the reference(s) that the annotation is tied to. Each glyph has nine anchor points: Its centroid, the rightmost top, leftmost top, top rightmost, and so on clockwise around the glyph. Anchor points provide symbolic descriptions that can be projected as candidate inferences from an example to a new situation (e.g., the distance from the left bottommost point to the right bottommost point of a reference).

Annotation glyphs create visual quantities that can be linked to conceptual quantities in a concept map interface attached to sKEA. For example, in describing stability of a building, one wants to say that the stability of the building (a continuous conceptual quantity) decreases as its height (a continuous visual quantity) increases. Annotation glyphs allow us to do this by defining visual quantities in terms of measurable properties of glyphs. These visual quantities can then be linked in situation-specific causal models to the conceptual quantities. These statements are made using a concept map interface. Every entity and quantity in the sketch automatically becomes available in the concept map, so that other statements can be made about them. For causal models, we use qualitative process theory [5], specifically the notion of influences. In this case, the stability of the building is \propto_Q the height of the building. These example-specific statements can be applied by analogy to other situations, thus providing a means of formulating models without a complete and correct domain theory.

Qualitative Mechanics

Qualitative mechanics (QM) concerns the same material as traditional mechanics, e.g., the effects of energy and forces on bodies, but from a qualitative perspective. We assume that people learn many aspects of qualitative mechanics in infancy, so we treat the structural abstractions and model fragments of qualitative mechanics as part of the starting endowment of the system, rather than as something to be learned. Our QM domain theory is drawn from [16, 20].

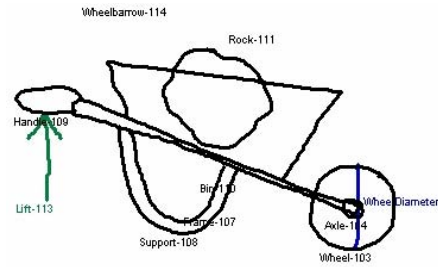


Figure 1b: Wheelbarrow with assumed force and length annotations

Specifically, we use their qualitative representations of surfaces, force transfer, torque transfer, and center of rotation.

How these structural abstractions can and should be applied to everyday situations is learned by our system, via sketched examples. When a person is entering an example, they need to provide both the everyday concepts that are used to describe the entities in the system and the appropriate structural abstractions. This is done in two ways. First, the interface used for conceptual labeling includes both types of concepts. For example, the chassis of a wheelbarrow might be indicated to be an instance of *VehicleChassis* (an everyday concept) and as an instance of *RigidObject* (a QM abstraction). Second, sKEA automatically computes visual/conceptual relationships between pairs of glyphs if they are touching or one is inside another (as indicated by the computed RCC8 relationships). sKEA offers the user the opportunity to specialize these relationships to provide QM relationships, e.g., that the beam of a seesaw can pivot around the seesaw's base.

Sometimes systems must be viewed from multiple perspectives. In understanding how a wheelbarrow works, for example, it makes sense to draw the individual parts, since each contributes differently to how it functions. But if we are considering how hard it will be to lift a wheelbarrow, we need to consider these entities as a single rigid object. Consequently, we extended sKEA to include *group glyphs*, which introduce a new entity to represent a selected set of entities, to handle such situations.

By noting what QM concepts and relationships are applicable to each aspect of an example, analogy can be used to apply this information to new situations. The everyday properties and relationships provide the commonalities needed to be reminded of relevant examples and the information needed to align the new situation with the example. The technical vocabulary of QM can then be projected, via candidate inferences, to the new situation. This enables the first-principles QM domain theory to be applied to the new situation.

Problem Solving

There are two kinds of questions in the BMCT. *Outcome questions* ask the examinee to select which qualitative

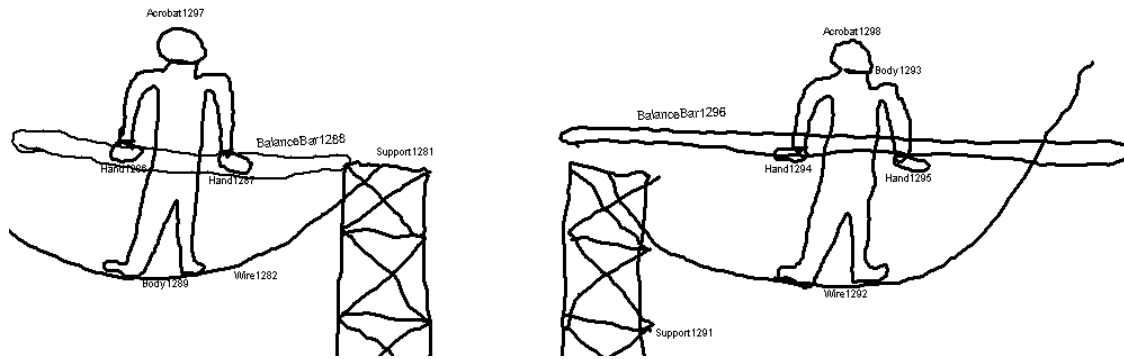


Figure 2: Problem Sketch Drawn in sKEA

behavior will occur in a situation, e.g., "which way will the indicated gear turn: clockwise, counterclockwise, or not at all?" *Differential qualitative analysis (DQA) questions* concern relative values of the same (or similar) properties between two (or sometimes three) distinct scenarios. Both kinds of questions rely on the same process of model formulation via analogy, so we begin there.

To retrieve a relevant case, the system starts by using MAC/FAC on the problem sketch with low level visual properties removed, i.e. glyph orientations and relative sizes. For outcome problems the first retrieval is used. For DQA questions, the retrieval must also contain candidate inferences that causally constrain the goal quantity. Should the first retrieval prove unsatisfactory, a second retrieval is performed. If that, too, fails, the low level visual properties are added back into the probe and up to two more retrieval attempts are made. After MAC/FAC has selected a suitable retrieval, SME is used to generate a mapping between the example, base, and the problem situation, target. All candidate inferences produced by this mapping pertaining to structural abstractions and causal models (but not specific parameter values) are assumed to be true for the problem.

Once the model is formulated, solving an outcome problem involves standard qualitative reasoning. The QM domain theory is applied to the problem to answer the question. Note that this critically relies on the structural abstraction information imported by analogy from the example: Without it, there would be no model fragments.

For the DQA questions, the problem now contains causal information about the quantity in question. We compare the situations in the problem, using SME. This mapping provides the correspondences between the two systems needed for DQA. This is an important departure from [24]: We discovered that analogy provides a general mechanism for providing the frame of reference needed for comparative analyses. This allows a wider class of systems to be analyzed, since the correspondences between systems are computed dynamically. For example, some problems require DQA between different parts of the same system, e.g., which wheel of a railcar presses harder on the rail, when the load is nearer to the back wheel? This does not fit the standard "perturbed system" format of DQA. But SME creates an appropriate reference frame by mapping the

railcar to itself, constraining the rear wheel to correspond to the front wheel.

Once the reference frame is set up, differential qualitative analysis proceeds by chaining backward from the goal quantity through the causal model. Conceptual quantities that are not causally constrained by other parameters or are not known to be different are assumed to be the same, i.e., a value of UnchangedDQ. DQ values for visual quantities are measured from the sketch, using the annotation information inferred via analogy from examples.

Example

Figure 2 shows a problem from the BMCT sketched in our system. The question, "Which acrobat will find it easier to keep her balance? (If equal, mark C)", was translated into the following query:

```
(solveDQProblem BMCT-S-17-MEK Object-1297
Object-1298 ((QPQuantityFn Stability) Object-
1297) ?value))
```

BMCT-S-17-MEK indicates the context in which the problem will be solved, Object-1297 and Object-1298 represent the acrobats A and B respectively, and the DQA problem will be to determine the DQ value of the Stability of Object-1297, Acrobat A.

Since our system relies on analogy for problem solving, finding a good example in the system's memory is crucial. Good analogs are similar situations containing causal models constraining relevant quantities. One example, drawn to depict the sentence "A person using a pole to maintain their balance", is shown in Figure 3. The stability of the person is causally constrained by the length of the pole, a visual quantity defined by a linear sketch annotation. This fact is asserted in this sketch via a concept map interface. The anchor points stored for this annotation are:

```
(startPointOf AnnotationGlyph-11
(LeftmostBottomPointFn Glyph-3))
(endPointOf AnnotationGlyph-11
(RightmostTopPointFn Glyph-3))
```

Retrieval is done for each situation depicted in the problem sketch (Figure 2), in this case twice. During retrieval, the commonalities of each situation containing a person, a rope, and a pole are enough for MAC/FAC to retrieve this example. The causal model and the annotation

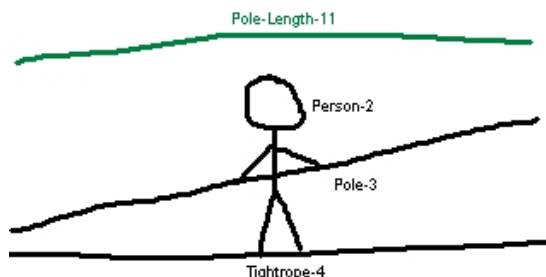


Figure 3: Example Sketch with Pole-Length Linear Annotation

glyph's anchor points are imported as candidate inferences into each situation in the problem sketch. As a result, the stability of Acrobat A is causally constrained by the visual quantity of Acrobat A's Pole-Length. The same causal model exists for Acrobat B as well.

Because this is a DQA problem, the system makes an analogy between the Acrobat A's layer (the base) and Acrobat B's layer (the target) to frame the comparison. The causal model of Acrobat A's stability is traversed until a visual quantity is found. The system measures and compares Acrobat A's Pole-Length and its corresponding quantity, Acrobat B's Pole-Length, given by the framing analogy. The measurement is determined by the previously imported anchor points. Because the value of the measurement increases from base to target, the DQ value of the Acrobat A's Pole-Length is increased. This DQ value is propagated through the positive causal link and the system answers that the stability has increased from Acrobat A to Acrobat B. The system responds to the query by returning the correct answer:

```
(solveDQProblem BMCT-S-17-MEK Object-1297
Object-1298 ((QPQuantityFn Stability) Object-
1297) IncreasedDQ)
```

Experiment

We conducted an experiment to see how well this model performs. We selected 13 of the 68 problems on the BMCT, focusing on problems involving net force, revolution rate, stability, and smoothness of ride. We developed a list of 18 example situations, 15 of which were intended to be good analogs for specific test questions. These examples were described via a single sentence of text. We recruited three graduate students, with varying degrees of familiarity with sKEA, to serve as knowledge enterers (KEs), drawing each of the situations on the list. Our goal was to determine how well this model performs when given case libraries drawn by different individuals, and by combinations of individuals.

Method

The KEs were told to sketch each situation, breaking it down into glyphs just far enough to explain the principle(s) that they thought were operating in the situation. While

drawing, they used sKEA's visual/conceptual relationships interface to fill in their intended relationships, selecting from the candidates offered. KEs were also given a list of non-QM conceptual parameters that were relevant in this subset of the test (smoothness of ride, revolution rate, and stability), and asked to add these to the sketch when appropriate, along with a causal model describing how they are constrained by any visual quantities. The visual quantities were defined via annotations, and the other aspects of the causal model were entered via a concept map interface linked to sKEA. After they considered their sketch complete, they ran the QM system to derive model fragment instances. If they were not satisfied with the model fragment instances found, they were encouraged to modify their sketch (typically by adding structural abstraction information, through sKEA's standard conceptual labeling mechanism, or a visual/conceptual relationship) until they were. For example, in a sketch depicting two meshed gears and one of the gears is rotating (as indicated by an annotation), then there is something wrong if there is no mention of torque transfer in the active model fragment list. Once finished, each example sketch was stored in a case library for that particular KE.

The 13 problems (11 DQA and two outcome problems) were drawn by a fourth graduate student serving as the problem enterer (PE), an experienced sKEA user. These problems were presented to the system in a series of seven trials, one for each case library alone as well as each pair wise combination and finally all the examples in the system.

Results and Discussion

Library	# Correct Retrievals (of 13)	# Correct Answers (of 13)
KE1	7	5 (p < .45)
KE2	10	10 (p < .001)
KE3	6	2 (p < .96)
KE1+2	11	9 (p < .008)
KE1+3	9	5 (p < .45)
KE2+3	10	10 (p < .001)
KE1+2+3	12	10 (p < .001)

Table 1: Problem Solving Results

A summary of the results appears in Table 1. The correct retrieval column lists the number of times the system retrieved one of the good analogs for the problem sketch; as defined above. The correct answer column lists the number of times the system provided the correct answer. Of the 13 problems, only one was not solved in any of the trials, as discussed later. KE2 had the most experience with sKEA leading to similar representations to the PE, and KE3 had the least experience, providing some serious variability.

Recall that the system would not be able to solve any of these problems without the examples to work from. Four of the seven trials had statistically significant results ($P < 0.05$). The system's performance improves as better examples are accumulated. The results using combinations of case libraries support this model's claim to be general purpose and scalable. In each of the combination trials,

example sketches from at least two of the case libraries were used to formulate correct answers.

Question Type (number)	# Correct Retrieval	# Answers Produced	# Answers Correct
Outcome Questions (14)	10	14	10
DQA Questions (77)	55	57	41

Table 2: Results by Question Type

Table 2 breaks down the results by question type to analyze the failures. The outcome questions errors occurred only from failures to retrieve the correct case from the case library. The system would reason from whatever structural abstractions it found from the example case. The four failures occurred when the system confused a gear rotating inside another gear with two gears rotating side by side due to annotation glyph placement. Failure detection in qualitative mechanics is difficult because it is an implicit assumption that the mapping with the example case will provide the correct structural abstractions. People seem to handle this by recognizing contradictions in their reasoning.

The problem solving failures in the DQA problems were much more interesting, due to their complexity. During the model formulation phase, the retrieval algorithm determines that the mapping with the example case will yield a causal model for the quantity in question. For 20 of the 77 problems, a retrieval causally constraining the sought quantity could not be found. Frequently this was because the KEs drew the situation at a different level of abstraction than the PE. This was especially true for the one problem that was never solved correctly where the KEs' sketches contained between 7 and 8 glyphs and the PE's contained only 5 glyphs. The second failure type is when the causal model contains elements that do not exist in the problem sketch due to a bad mapping. These are the remaining 16 DQA failures. An example of this appears in Figure 4.

The difference in numbers of glyphs and entities

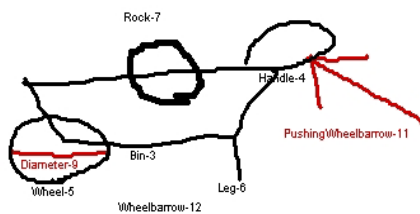


Figure 4a. Example Wheelbarrow

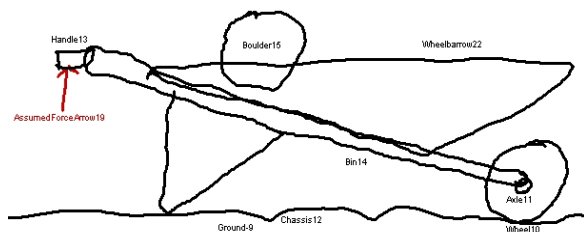


Figure 4b. Problem Wheelbarrow

depicted in these sketches is difficult for our system to handle. The problem sketch contains glyphs for the ground and the axle, but the example does not. The chassis in the problem sketch is conceptually labeled as a leg in the example. Fundamental differences like these lead to incorrect mappings, e.g., the bin in the example to the chassis in the problem. This leads to incorrect candidate inferences, which in turn forces the system to attempt to reason about surfaces or glyphs that do not exist.

People seem to have several methods for dealing with such problems. First, people try other examples, going back to memory to find an example that is more productive. A simple version of this is already implemented in the system. Also, people use *rerepresentation* [26] to bring the base and target into closer alignment. Knowledge about depiction seems crucial: If two sketches are misaligned, simplifying the more complex one, or postulating new glyphs in the simplified one, seems to be a promising strategy.

These results indicate the strength and breadth of our common sense reasoning system. The major components have been designed in a domain independent manner from the model formulation via analogy, to the use of sketch annotations to define visual quantities, and the use of analogy to frame comparative analyses.

Related Work

Previous efforts in qualitative mechanics have created systems that can reason about clocks [20], mechanisms [15, 22], and internal combustion engines [16]. Some researchers use high-resolution inputs or CAD systems to produce structural descriptions containing metric information which can also be exploited by the system. Aside from annotations and low-level visual properties included in similarity computations, we do not exploit metric information at all. SketchIt [22] used sketched input, but allowed only a handful of abstract types to be drawn.

The use of experience in model formulation was proposed by Falkenhainer [4], for improving compositional modeling by choosing appropriate perspectives and levels of detail. Unlike this effort, he assumed a complete and correct domain theory as a starting point.

Similarity-based qualitative simulation [25] also uses analogy for qualitative reasoning, but focuses on multi-state reasoning, unlike the single-state problems addressed here.

Discussion

This work provides evidence that qualitative reasoning combined with analogical reasoning is a promising explanation for how common sense reasoning works. The mapping from structural descriptions to structural abstractions, we hypothesize, is learned incrementally from examples. As shown here, examples can be used by analogy to aid in formulating models for new situations. Similarly, sketch annotations provide a means for defining visual quantities in examples which can be used by analogy to define visual quantities for new situations. Analogy also plays an important role in qualitative analysis once the

model has been formulated, e.g. our use of mappings as frames of reference for differential qualitative analysis.

As noted earlier, there are several paths to explore in future work. Variations on the baseline model of analogical processing used here constitute one line of investigation. This includes more aggressive retrieval strategies (e.g., using multiple analogs), rerepresentation to improve mappings, and using the SEQL model [17] to learn generalizations from examples. We are also designing an interactive interface so that we can expand the system's capabilities by teaching it. For more than half of the incorrect answers, the system knew that it could not provide a good answer. This is an opportunity to explore interactive learning while problem solving. For instance, given an incorrect mapping or retrieval, providing feedback about what a better mapping or retrieval would be gives the system evidence on how to reorganize its methods for encoding situations, which should lead to improved performance. Our goal is to expand its capabilities via instruction to include the full range of phenomena covered in the Bennett Mechanical Comprehension test. Being able to perform at an expert level on such an exam would be a landmark in qualitative and common sense reasoning.

Acknowledgements

This work was supported by DARPA IPTO.

References

1. Cohn, A. Calculi for Qualitative Spatial Reasoning. *Artificial Intelligence and Symbolic Mathematical Computation*, LNCS 1138, eds: J Calmet, J A Campbell, J Pfalzgraf, Springer Verlag, 124-143, 1996.
2. Falkenhainer, B., Forbus, K. and Gentner, D. The Structure-Mapping Engine. *Proceedings of the Fifth National Conference on Artificial Intelligence*. 1986.
3. Falkenhainer, B. and Forbus, K. Compositional modeling: finding the right model for the job. *Artificial Intelligence*. 51:95-143. 1991.
4. Falkenhainer, B. Modeling without amnesia: Making experienced-sanctioned approximation. *The Sixth International Workshop on Qualitative Reasoning in Physical Systems*. 44-55, Edinburgh, Scotland, Heriot-Watt University. 1992.
5. Forbus, K. 1984. Qualitative Process Theory. *Artificial Intelligence*, (24), 1984, pp 85-168
6. Forbus, K. Exploring analogy in the large. In Gentner, D., Holyoak, K., and Kokinov, B. (Eds.) *Analogy: Perspectives from Cognitive Science*. MIT Press. 2001.
7. Forbus, K., Gentner, D. 1997. Qualitative mental models: Simulations or memories? *Proceedings of QR97*, Cortona, Italy.
8. Forbus, K., Ferguson, R. and Gentner, D. Incremental structure-mapping. *Proceedings of the Cognitive Science Society*, August. 1994.
9. Forbus, K., Gentner, D., and Law, K. MAC/FAC: A model of similarity-based retrieval. *Cognitive Science*, 19, 141-205. 1994.
10. Forbus, K. and Hinrichs, T. Companion Cognitive Systems: A step towards human-level AI. *AAAI Fall Symposium on Achieving Human-level Intelligence through Integrated Systems and Research*, October, 2004. Washington, DC.
11. Forbus, K., Lockwood, K., Klenk, M., Tomai, E. and Usher, J. 2004. Open-domain sketch understanding: The nuSketch approach. *Proceedings of the AAAI Fall Symposium on Making Pen-based Interaction Intelligent and Natural*, October, Washington, D.C.
12. Forbus, K., and Usher, J. 2002. Sketching for knowledge capture: A progress report. *IUI '02*, San Francisco, CA.
13. Gentner, D., Structure-mapping: A theoretical framework for analogy, *Cognitive Science* 7(2), 1983
14. Gentner, D. and Stevens, A. *Mental Models*. Lawrence Erlbaum Associates, Hillsdale, New Jersey, 1983.
15. Joskowicz, L., & Sacks, E. Configuration Space Computation for Mechanism Design. *IEEE-94*. 1993.
16. Kim, H. Qualitative reasoning about fluids and mechanics. Ph.D. dissertation and ILS Technical Report, Northwestern University. 1993.
17. Kuehne, S., Forbus, K., Gentner, D. and Quinn, B. SEQL: Category learning as progressive abstraction using structure mapping. *Proceedings of CogSci*. August, 2000.
18. Lenat, D. B., & Guha, R. V. (1989). Building large knowledge-based systems : representation and inference in the Cyc project. Reading, MA: Addison-Wesley.
19. Mostek, T., Forbus, K, Meverden, C. Dynamic case creation and expansion for analogical reasoning. *Proceedings of AAAI-2000*. Austin, TX. 2000.
20. Nielsen, P. A Qualitative Approach to Rigid Body Mechanics. Ph.D. dissertation, University of Illinois at Urbana-Champaign. 1988.
21. Rickel J. and Porter B. Automated Modeling for Answering Prediction Questions: Selecting the Time Scale and System Boundary, *AAAI-94*, pp. 1191-1198, Cambridge, MA: AAIT/MIT Press. 1994.
22. Stahovich, T. F., Davis, R., and Shrobe, H., "Qualitative Rigid Body Mechanics," *Artificial Intelligence*, Vol. 119, No. 1-2, pp 19 - 60, 2000.
23. Tomai, E., Forbus, K., and Usher, J. 2004. Qualitative spatial reasoning for geometric analogies. *Proceedings of the 18th International Qualitative Reasoning Workshop*, Evanston, Illinois, USA, August
24. D. Weld "Comparative Analysis," *Artificial Intelligence*, 36:333-374, October 1988.
25. Yan, J. and Forbus, K. Similarity-based qualitative simulation: A preliminary report. *Proceedings of the 18th International Qualitative Reasoning Workshop*, Evanston, Illinois, USA, August. 2004.
26. Yan, J., Forbus, K. and Gentner, D. 2003. A theory of rerepresentation in analogical matching. *Proceedings of CogSci03*.