4 Analysis of BotE Strategies

At the heart of BotE reasoning is a library of strategies that transform a problem into other, possibly easier problems. In this section we start with a formalization of the problems and strategies. We then present a knowledge level analysis [Newell, 1982] of the strategic knowledge in BotE reasoning. We present seven different strategies that are compositional and cover a large range of problems from various domains.

An abstract way to represent a BotE problem is **(Q O ?V)** where **Q** is the quantity, **O** the object and **?V** is the unknown value that is being sought. For example, in the question "How many K-8 teachers are there in US?," **Q** is cardinality, and **O** is the collection of K-8 teachers in US. Note that such decomposition of a question into a quantity-object pair is not necessarily unique, for example, "How much money is spent on newspapers in USA per year?" can be decomposed into:

- **Q**=Cost, **O**=All newspapers sold every year in US
- **Q**=Annual Sales, **O**=All US newspapers

Given a problem as **(Q O ?V)**, a strategy transforms it into a set of other problems $\{(Q_i\ O_i\ ?V_i)\}$ such that $?V_i$ are already known or easier to estimate. Besides transforming the problem, each strategy contains the answers to the following questions:

- When does it apply?
- How do we combine $?V_i$s to find **?V**?
- How sure are we of our estimate of **?V**?

With strategies represented as above, there are three syntactic possibilities for a strategy based what aspect of problem it transforms:

1. *Object-based:* $(Q\ O\ ?V) \rightarrow \{(Q\ O_i\ ?V_i)\}$
2. *Quantity-based:* $(Q\ O\ ?V) \rightarrow \{(Q_i\ O\ ?V_i)\}$
3. *System-based:* $(Q\ O\ ?V) \rightarrow \{(Q_i\ O_i\ ?V_i)\}$

Next we look at each of these in turn. We present the seven strategies which are numbered S1 through S7.

4.1 Object-based Strategies

An object-based strategy related an object, **O**, to a set of objects, $\{O_i\}$, such that the quantity values for those objects, $\{?V_i\}$,combine in a known way to estimate the original quantity, **?V**. Note that this since we are estimating the same quantity, this combination function can only be addition or subtraction since **?V** and $\{?V_i\}$ have to have the same dimensions. Below are three object based strategies:

S1. Mereology: The mereology strategy transforms an object into other objects that are its parts. If **Q** is an extensive parameter, then, $?V=\Sigma ?V_i$. For example, the weight of a basket of fruits is the sum of weights of all the fruits and the basket. If **O** is homogeneous, i.e., composed of the same kind of objects, then the above sum reduces to a product of number of parts and the value for each part. This strategy requires making a closed world assumption, namely, that we know all the parts of the original object. In order for this strategy to be applicable, the parts should be non-overlapping in the quantity. If **Q** is an intensive parameter, then, $?V=\Sigma w_i*?V_i$, where w_i is the weight of the ith part. For example, the density of a mixture is the weighted average of the densities of the constituents.

S2. Similarity: The similarity strategy transforms the object into other object(s) which are similar to it. For example, if asked for the population of Austria, a reasonable guess could be the population of Switzerland, based on the similarity of the two countries. The strategy can be implemented in two ways:

- **O** is similar to $O_1 \Rightarrow ?V=?V_1$
- **O** is similar to $\{O_i\} \Rightarrow ?V=Average(?V_i)$

If two objects are similar, it doesn't warrant the inference that values of all the quantities for two objects are similar. For example, another grad student in my department probably gets paid similar to me, but doesn't necessarily weigh the same. Or, consider the example that if two basketball players are similar, it is more likely that there heights are similar, than if the case was of professors. This notion of what features can be inferred from a similar example was called *projectability* by Goodman (1955/1983). There is increasing psychological evidence that projectability is based on *centrality* of the feature [Ahn, 1998; Hadjichristidis *et al*, 2004]. A feature is central to the extent that features depend on it. In our above example, height is central to basketball players, but not to professors. We have operationalized this notion of centrality as the structural support of the inference in computation of similarity using Structure Mapping Engine [Falkenhainer *et al*, 1989].

S3. Ontology: The ontology strategy tries to find other objects from the ontology hierarchy which might be used to guess the quantity in question. In the simplest form, if **O** is an instance of O_1, then I can use the knowledge about the class to guess the value for the instance. For example, if I know that Jason Kidd is a point guard[1], then I can use the knowledge that point guards are relatively shorter than other players on the team to guess his height. If we didn't have information about point guards, we could even use the fact that Jason Kidd is a basketball player to guess his height.

Clearly, the higher up we go, the less accurate will be the estimate. Conceptually, the category hierarchy can be considered as specific → subordinate→ basic-level→ superordinate [Rosch, 1975]. An example of the four levels of hierarchy will be a specific chair in my living room, armchair, chair, furniture. In this framework, we can see that

[1] The point guard is one of the standard positions in a regulation basketball game. Typically one of the smallest players on the team, the point guard's job is to pass the ball to other players who are responsible for making most of the points.

not only the accuracy decreases as we go higher, but also that we can go no higher than the basic level. Besides that, as in the similarity strategy, the centrality of the quantity is proportional the accuracy of the estimate.

4.2 Quantity-based Strategies

A quantity-based strategy relates a quantity, **`Q`**, to a set of quantities, **`{Q`$_i$`}`**, such that the values of these quantities (for the object **`O`**) can be combined in a known way to derive the original quantity. Note that the combination function has to satisfy dimensional constraints, i.e., **`?V`** and **`f({?V`$_i$`})`** have to have the same units, where **`f`** is the combination function. Below are the two quantity-based strategies:

S5. Density: This strategy converts a quantity into a density quantity and an extent quantity. Here, density is used in a general sense to mean average along any dimension: we talk of electric flux density, population density, etc. Rates, averages, and even quantities like teachers per student are considered densities. For example, number of K-8 teachers in US can be estimated by multiplying the number of teachers per student by number of students.

S4. Domain Laws: This strategy converts a quantity into other quantities that are related to it via laws of the domain. Domain laws include laws of physics as well as rules of thumb. For example, Newton's second law of motion, **`F=m*a`**, relates the force on an object to its mass and acceleration. The application of domain laws by the problem solver requires formalizing the assumptions and approximations implicit in the laws. This has been well explored in compositional modeling [Falkenhainer and Forbus, 1991; Nayak, 1994]. In BotE reasoning, since we are not interested in an exact answer, but an approximate estimate, aggressively applying approximations to simplify the problem solving becomes crucial. Some of the approximations are:

- Geometry: Assume simplest shape, e.g. Consider a spherical cow [Harte, 1988].
- Distribution: Assume either a uniform distribution, or a Dirac-delta (point mass).
- Calculus: Integrals can be simplified by sums or average multiplied by extent, and differentials by differences.
- Algebra: Use simplification heuristics to reduce the number of unknowns [Pisan, 1998].

4.3 System-based Strategies

System-based strategies transform both the quantity and the object into other quantities and objects. They are so called as they represent relationships between quantities of a system as whole. For example, for a system with no external force, the momentum remains conserved. This translates into a conservation equation that relates the masses and velocities of the parts of the system. It would seem like that this effect can be obtained by sequentially applying a quantity-based and an object-based strategy (or vice versa) since all the above strategies are compositional. There are two reasons for keeping this a separate type of strategy: 1) it represents a reasoning pattern that is different, 2) sometimes it is much more efficient to apply a system-based strategy, e.g., applying conservation of momentum leads to safely ignoring all the internal forces which need to be made explicit if one is not applying a system-based strategy. The two system-based strategies are:

S6. Balances: Many physical quantities remain conserved for a system, e.g., energy, mass, momentum, angular momentum, etc. As a result of this, often one can write a balance equation that relates the expressions that denote the value of the quantity in two different states of the system. To write a balance equation, appropriate assumptions about the system in consideration have to be made.

S7. Scale-up: This is often an empirical BotE strategy. A smaller model that works under the same physical laws can be used to estimate the quantity values for a full-scale prototype. To ensure that the scale-up is valid, all the dimensionless groups must be kept the same in the model and the prototype. For example, the Reynolds number is a dimensionless group that corresponds to the nature of flow (laminar, transient or turbulent), and for a flow model to be valid for scaling up, the Reynolds number must be the same in both situation.

4.4 Discussion

BotE reasoning is very powerful because of its applicability to many domains. Clearly a human expert, or a computer problem solver, can do better with more facts about quantities and more strategies. The above analysis is an attempt to identify the core strategies that have a broad coverage. We don't claim that these are all the strategies. We arrived at these strategies by an analysis of knowledge that our problem solver was using. To confirm that we have a set of strategies which have broad coverage, we manually went through Clifford Swartz's "Back-of-the-envelope Physics." Swartz's book is a collection of estimation problems (along with solutions) from various domains in physics. We looked at all the problems (n=44) from Force and Pressure, Rotation and Mechanics, Heat, and Astronomy. Even though the goal of our work is not tied to Physics, this provides an independent confirmation about our strategies. Table I shows the results of our empirical analysis. The table shows the number of instances of each strategy, and the corresponding percentage, for Swartz's book as well as for the set of problems that BotE-Solver can currently solve. We went through every solved problem and every time a problem was transformed into another, it was classified as one of the seven strategies or other. We counted an application of the ontology strategy if it was specifically mentioned, since trivially almost all problems apply that strategy, as most problems are not about specific instances but classes of things.

Table I: Distribution of strategies.

Source →		Swartz		BotE-Solver	
Strategy ↓		Number of times used.	% of times used	Number of times used.	% of times used
S1	Mereology	11	14	9	32
S2	Similarity	5	6	5	18
S3	Ontology	6	8	0	0
S4	Density	10	13	10	36
S5	Domain laws	29	37	3	11
S6	Balances	11	14	1	4
S7	Scale-up	2	3	0	0
	Other	5	6	0	0
	Total	79	100	28	100

Some observations from table 1:

1. Coverage: The seven strategies account for 94% of strategy use in the problems from Swartz's book. The 5 strategies that under the "other" category contain four instances of designing experiments to estimate a quantity, and one instance of a complex problem from statistical mechanics.
2. Domain-specificity: Strategies S1 through S4 are domain independent, and account for 40% of strategy use, while 60% of strategies are domain specific, with the largest component being domain laws. Interestingly, only 15% of the strategies in BotE-Solver are domain specific. This is expected because the problems that BotE-Solver currently solve are from common-sense domains, as compared to Swartz's book where most problems involve applying laws of physics.

As of now, we haven't implemented the Ontology and the Scale-up strategy in BotE-Solver, hence no instances of those are found. In the next section, we present a short description of the current implementation of BotE-Solver.