Summary of Class Discussion of January 29, February 3 and February 5, 2003

Scribe: Catherine Anderson

February 12, 2003

1 Constraint synthesis: a clarification

We discussed the following issue of the paper by Freuder [1]. The issue concerned the effect of globally propagating constraints. It was thought that there was no looping until quiescence. The situation where this was thought to occur would be in the following case:

- Node N_{J1} is added to the network and linked to all of the appropriate $N_{H's}$ (reminder N_{Ji} is of arity 1 higher than N_{Hi}).
- N_{H1} is locally propagated to N_{J1} with all tuples removed from N_{J1} that do not have support in N_{H1}.
- Likewise, all other $N_{H's}$ are locally propagated to N_{J1}.
- Then N_{J1} is propagated which means that it is globally propagated through all of its neighbors. If any of the neighbors are modified, then that neighbor is globally propagated through all of its neighbors except N_{J1}.
- Now node N_{J2} is added to the network and linked to all appropriate $N_{H's}$.
- All linked $N_{H's}$ are locally propagated to N_{J2} with all tuples removed from N_{J2} that do not have support in these $N_{H's}$.
- Then N_{J2} is propagated.
- If any tuple is removed from N_{H2} (a node that is also linked to N_{J1}) and this tuple was the only support for a tuple in N_{J1}, then N_{J1} would no longer be consistent. This is were it was thought that a mistake had been made.
- However, in re-examining the full definition of Propagate, it was agreed that when an N_H is modified during the propagation of N_J, that N_H is globally propagated through all of its neighbors except N_J, which, in this instance would include N_{J1}, thus removing the inconsistency.

2 Critiquing articles

Dr. Choueiry would like us to learn how to properly review and critique technical articles. The conferences in the field of A.I. are extremely strict about the quality of paper accepted for a conference. Each paper is reviewed by three and sometimes four people. Each person will offer their remarks, suggestions and questions, all of which must be resolved before the paper is deemed ready for the conference. The proceedings from the conferences are bound and archived with an ISBN number.

Dr. Choueiry stressed that any critique made to an article will most likely be taken as an insult, so they should be made as respectfully as possible. It should also be kept in mind that it is important to acknowledge corrections made to your paper.

One acceptable style for writing up the review of an article is as follows:

Type of problem, Page number, Paragraph number, line number, brief explanation. Examples:
Typo, page 144, paragraph 2, line 1, ”Hello” instead of ”Helo”.
or
Clarification needed, page 49, paragraph 4, equation (2), can X be less then Y for all values of z?

The types of problems could be typos, equation incorrectness, misuse of words, unsubstantiated claim, missing reference, point where clarification is needed, or a minor error.

As we read and discuss the book by Rina Dechter, all critiques of the manuscript should be sent to Cate Anderson at anderson@cse.unl.edu.

3 Chapter 3, Section 3

The following comments are in addition or elaboration of the slides presented by Dr. Choueiry.

- The definition of Path consistency, Definition 3.3.2, defines the property of path consistency and not the algorithm.
- The definition applies to networks containing non-binary constraints in addition to those containing binary.
- The definition, while defining the property of path consistency, does not guarantee arc-consistency. This is the result of the limit of $i \neq j$ in the relation R_{ij}.

4 Chapter 3, Section 4

- The algorithm for Revise-3 modifies the constraints of the network, not the domains.
- The algorithms PC-1 and PC-3 (which are analogous to AC-1 and AC-2) are not the same as those defined by Mackworth [2]. Dechter's algorithms modify the binary constraints as opposed to Mackworth's modifying *also* the domains.
- In the algorithm Revise-i, only relations of cardinality i are updated, not those of $(i-1)$. This can be seen in the complexity analysis. Because of this, i-consistency does not guarantee $(i-1)$ consistency. This is quite different from the synthesis of constraints as presented by Freuder, which does guarantee $(i-1)$ consistency.

5 Chapter 3, Section 5

- In the concept of generalized arc-consistency it should be noted that the ”arcs” apply to the hyper-arcs of a non-binary network.
- In Generalized Arc-Consistency, which applies to non-binary networks, once again the domains are updated.
- In Relational Arc-Consistency (not to be confused with relational consistency), the relations are updated. This is not a recommended procedure for implementation in search, because the cost in space of having to track the deletion at each stage of instantiation in a backtrack search could be much greater then simply keeping track of the domain members.
- Dechter's definition of *global constraints* is different that the traditional definition. Traditionally global constraint is defined as a constraint that has all of the variables in the network as its scope. Dechter defines a global constraint as a constraint of order larger then 2. The scope of the constraint should also be parameterized.
- With quite a few of the more common global constraints, relational description are given. These are generally used for constraints where the number of allowed tuples is too large to be written extensionally.

- Specialized algorithms have been developed for the different global relations, which ensure generalized arc consistency, while controlling the combinatorial blow-up. These algorithms take advantage of the topology of the constraint network (i.e., shape of the network) semantics of the constraints. Exact applications were not discussed.
- Several of the more common global constraints are `AllDifferent` (same as `Mutex`), sum, cardinality, cumulative, and cycle constraints.
- Bounds Consistency can be used in the case of a well totally domain in either a continuous or discrete problem. While it ensures that the interval endpoints are arc-consistent, it is a weaker notion of consistency, in that it doesn't check on the inner values and so cannot guarantee arc-consistency.
- In spite of not guaranteeing arc-consistency, bounds consistency is very cost effective, for example, $O(nlogn)$ instead of $O(n^3)$. The mechanism is to tighten the endpoints until the endpoints are arc-consistent. This is an application of using interval mathematics.

6 Chapter 3, Section 6

- Dechter uses the description of numeric constraints when what she describes is really linear and inequality Constraints. Numeric constraints could involve logarithmetic or trigonometric expression, as well as linear.

  Both numeric and boolean constraints have special syntactic forms that can be used to achieve consistency.
- Algebraic constraints (or more exact, linear constraints) can be composed through algebraic elimination. The is the solving of a system of equation, the same number of independent equations as variables An example of a binary linear constraint would be the constraint of bounded difference. Applying arc-consistency filters the domains, and composition (elimination) adds binary constraints.
- In the case of boolean constraints, each variable has one or two value; 1 (for true)and 0 (for false). A binary boolean constraint is expressed as a binary clause. Constraint composition is achieved through unit resolution (or, better generalized resolution), using the procedure Unit propagation. The input of this procedure is a CNF sentence and the output is the equivalent theory without unit clauses appearing as literals in any non-unit clause.
- There are several specialized case of constraint problem where arc and path consistency can guarantee to solve the problem. Examples of these classes are Tree structured graphs (which has width 1), 2-SAT and Horn Theories.
- A Horn theory is one in which each clause has at most one positive literal. Consistency can be achieved in $O(n^2)$.
- A causal graph is an example of one that does not need a high level consistencyq checking to be solved.

7 Chapter 3, Section 8

- A different method of indicating a constraint network was developed in the Database field. Instead of a square node connected to the variable nodes, to indicate a non-binary constraint, the nodes are enclosed by a blob (see Fig. 3.17). In this case, 3-consistency implies binary relations, and 4-consistency implies ternary relations.

8 Suggestions on manuscript

- Clarification, page 72, paragraph 3, line 5, should emphasize that $i \neq j$ in R_{ij}.
- Clarification, page 77, paragraph 2, line 3, should emphasize that for D_y, it should be emphasized that y is not in S.

- Clarification, page 77, paragraph 2, line 4, should emphasis that for i-consistent, variable i must be distinct from variable in $i - 1$.
- Typo, page 78, Figure 3.15, Title of algorithm should be I-consistency-1(R), note I-consistency(R), in agreement with the caption of the figure.
- Format style, page 79, Section title 3.4.1, why have a Section 3.4.1 if there is not other 3.4 section?
- Clarification needed, page 80, definition 3.5.1 and Proposition 3.5.2, the variable t is used to mean two different things. In Definition 3.5.1, it is used to mean a tuple of a given relation, while in Proposition 3.5.2 it is used to mean the tightness of a constraint.
- Writing style, page 80, top of page, there should be a brief description of the section between the section heading 3.5, and the first subsection, 3.5.1.
- Misuse of Word, page 82, paragraph 4, line three. ”implicitly” $\longrightarrow$ ”intentionally”.
- Misstatement, page 83, paragraph 2, line 2. When S is totally defined ordered set, should be when the domains of the elements of S are totally defined ordered sets.
- Formatting, page 82, paragraph 3, Italics should be ended at some point in the paragraph instead of letting it extend to the end of the chapter.
- Inconsistency in notations: throughout the chapter, constraints are sometimes called C, sometimes R. This is OK but makes the non-experienced reader uncomfortable...

References

[1] Eugene C. Freuder. Synthesizing Constraint Expressions. *Communications of the ACM*, 21 (11):958–966, 1978.

[2] Alan K. Mackworth. Consistency in Networks of Relations. *Artificial Intelligence*, 8:99–118, 1977.