Constraint-Based Inference: A Bridge between Constraint Processing and Probability Inference

Student: Le Chang
Supervisor: Alan K. Mackworth

Department of Computer Science, University of British Columbia
2366 Main Mall, Vancouver, B.C. Canada V6T 1Z4
{lechang, mack}@cs.ubc.ca

1 Introduction

Constraint-Based Inference (CBI) [1] is an umbrella term for various superficially different problems including probabilistic inference, decision-making under uncertainty, constraint satisfaction, propositional satisfiability, decoding problems, and possibility inference. These problems share a common target of discovering new constraints from a set of given constraints over individual entities. The new constraints reveal previously undiscovered properties of those entities.

Along with the development of inference approaches for concrete CBI problems, researchers are increasingly aware that these problems share common features in representation and essentially identical inference approaches. In the constraint processing community, Semiring CSP [2] and Valued CSP [3] are two of the most widely studied generalized frameworks that abstract most soft constraint proposals, including Max CSPs, Fuzzy CSPs, Weighted CSPs, and Probabilistic CSPs, as well as classic CSPs. Based on the two frameworks, arc consistency is also extended as soft arc consistency [4] to handle soft constraints [5, 6].

In the probability inference community, the generalized algorithmic and representation frameworks have been studied for the past ten years. The generalization of the bucket or variable elimination algorithm [7] can be used in probability assessment, most probable explanation, maximum a posteriori hypothesis, and maximum expected utility problems. Junction tree algorithms, another class of widely studied exact probability inference approaches, are also generalized into single abstract frameworks, such as the Generalized Distributive Law (GDL) [8], the generalized bucket-tree elimination [9], and the valuation algebra framework [10]. These generalized frameworks, though focus on probability inference problems, provide opportunities to solving general CBI problems, especially to solving classic and soft constraint processing problems.

We explicitly use the semiring concept to generalize various CBI problems into a single formal representation framework with a broader coverage of the problem space, based on the synthesis of existing generalized frameworks from both constraint processing and probability inference communities. Based on our generalized CBI framework, an extensive comparative study of exact and approximate inference approaches between constraint processing and probability inference is commenced. The goals of our work are: (1) study the most important common characteristics of various CBI problems; (2) analyze and compare different inference approaches; (3) borrow design ideas from other fields and improve the inference approaches' efficiency in their own domains; and (4) significantly reduce the amount of implementation work targetted previously at the individual problems.

2 A Semiring-Based Generalized Framework for CBI Problems

There are two essential operators in real world CBI problems: (1) combination, which corresponds to an aggregation of constraints, and (2) marginalization, which corresponds to focusing of a specified constraint to a narrow domain. We use the commutative semiring structure to represent these two operators.

Definition 1 (Commutative Semiring). *Let* $\mathbf{A}$ *be a set. Let* $\oplus$ *and* $\otimes$ *be two closed binary operators defined on* $\mathbf{A}$*. Here we define operator* $\otimes$ *as taking precedence over operator* $\oplus$*.* $\mathbf{S} = \langle \mathbf{A}, \oplus, \otimes \rangle$ *is a commutative semiring if (1)* $\oplus$ *and* $\otimes$ *are all associative and commutative; (2)* $\otimes$ *and* $\oplus$ *have identity elements* $\mathbf{1}$ *and* $\mathbf{0}$*, respectively; and (3) there is a distributivity of* $\otimes$ *over* $\oplus$*.*

A CBI problem is defined in terms of a set of variables with values in finite domains and a set of constraints on these variables. We use commutative semirings to unify the representation of constraint-based inference problems from various disciplines into a single formal framework [1]. Formally:

Definition 2 (Constraint-Based Inference (CBI) Problem). *A constraint-based inference (CBI) problem* $\mathbf{P}$ *is a tuple* $(\mathbf{X}, \mathbf{D}, \mathbf{S}, \mathbf{F})$ *where:*

- $\mathbf{X} = \{X_1, \cdots, X_n\}$ *is a set of variables;*
- $\mathbf{D} = \{\mathbf{D_1}, \cdots, \mathbf{D_n}\}$ *is a collection of finite domains, one for each variable;*
- $\mathbf{S} = \langle \mathbf{A}, \oplus, \otimes \rangle$ *is a commutative semiring;*
- $\mathbf{F} = \{f_1, \cdots, f_r\}$ *is a set of constraints. Each constraint is a function that maps value assignments of a subset of variables to values in* $\mathbf{A}$

Definition 3 (The Combination of Two Constraints). *The combination of two constraints* f_1 *and* f_2 *is a new constraint* $g = f_1 \otimes f_2$*, where* $Scope(g) = Scope(f_1) \cup Scope(f_2)$ *and* $g(\mathbf{w}) = f_1(\mathbf{w}_{\downarrow Scope(f_1)}) \otimes f_2(\mathbf{w}_{\downarrow Scope(f_2)})$ *for every value assignment* $\mathbf{w}$ *of variables in the scope of the constraint* g*.*

Definition 4 (The Marginalization of a Constraint). *The marginalization of* X *from a constraint* f*, where* $X \in Scope(f)$*, is a new constraint* $g = \bigoplus_X f$*, where* $Scope(g) = Scope(f)_{-X}$ *and* $g(\mathbf{w}) = \bigoplus_{x_i \in \mathbf{D}_X} f(x_i, \mathbf{w})$ *for every value assignment* $\mathbf{w}$ *of variables in the scope of the constraint* g*.*

According to the definitions of the CBI problem and the basic constraint operations, we define the abstract inference and allocation tasks for a CBI problem.

Definition 5 (The Inference Task for a CBI Problem). *Given a subset of interested variables* $\mathbf{Z} = \{Z_1, \cdots, Z_t\} \subseteq \mathbf{X}$*, let* $\mathbf{Y} = \mathbf{X} \setminus \mathbf{Z}$*, the inference task for a CBI problem* $\mathbf{P} = (\mathbf{X}, \mathbf{D}, \mathbf{S}, \mathbf{F})$ *is defined as computing:*

$$g_{CBI}(\mathbf{Z}) = \bigoplus_{\mathbf{Y}} \bigotimes_{f \in \mathbf{F}} f \tag{1}$$

Given a CBI problem $\mathbf{P} = (\mathbf{X}, \mathbf{D}, \mathbf{S}, \mathbf{F})$, if $\oplus$ is idempotent, in other words, $a \oplus a = a, \forall a \in \mathbf{A}$, we can define the allocation task for a CBI problem.

Definition 6 (The Allocation Task for a CBI Problem). *Given a subset of variables* $\mathbf{Z} = \{Z_1, \cdots, Z_t\} \subseteq \mathbf{X}$*, let* $\mathbf{Y} = \mathbf{X} \setminus \mathbf{Z}$*, the allocation task for a CBI problem* $\mathbf{P} = (\mathbf{X}, \mathbf{D}, \mathbf{S}, \mathbf{F})$ *is to find a value assignment for the marginalized variables* $\mathbf{Y}$*, which leads to the result of the corresponding inference task* $g_{CBI}(\mathbf{Z})$*. Formally, we compute:*

$$\mathbf{y} = \arg \bigoplus_{\mathbf{Y}} \bigotimes_{f \in \mathbf{F}} f \tag{2}$$

where arg *is a prefix of operator* $\oplus$*. In other words,* $\arg\oplus$ *is an operator that returns arguments of the* $\oplus$ *operator.*

Many CBI problems from different disciplines can be embedded into our semiring-based unified framework [1]. These problems include the decision task and allocation task of CSP and SAT, Max SAT and Max CSP, Fuzzy CSP, Weighted CSP, probability assessment, most probable explanation (MPE), dynamic Bayesian networks (DBN), possibility inference with various t-norms, and maximum likelihood decoding.

3 Comparative Studies of Inference Approaches

3.1 Generalized Arc Consistency for Probability Inference

Arc consistency [11] is one of the most important techniques for binary classic CSPs. It is straightforward to extend it as generalized arc consistency [12, 13] to handle non-binary classic CSPs. Arc consistency is also extended as soft arc consistency [4] based on the Semiring CSP [2] and Valued CSP [3] frameworks. We applied generalized arc consistency to probability inference problems [14]. Here the success of generalized arc consistency in probability inference depends only on the existence and property of the combination absorbing element. More specifically, given a CBI problem defined on a commutative semiring $\mathbf{S} = \langle \mathbf{A}, \oplus, \otimes \rangle$, if $\exists \alpha_{\otimes} \in \mathbf{A}$ s.t. $e \otimes \alpha_{\otimes} = \alpha_{\otimes} \otimes e = \alpha_{\otimes}$ for any element $e \in \mathbf{A}$ and $\alpha_{\otimes} = \mathbf{0}$, the combination absorbing element $\alpha_{\otimes}$ can be used to detect non-valid domain values and simplify the involved constraints. For example, $false$, 0, and ∞ are combination absorbing elements of classic CSPs, probability inferences, and Max

CSPs, respectively. All the existing arc consistency enforcing algorithms can be generalized and migrated to handle a concrete CBI problem that satisfies this condition. If a CBI problem is defined on a monotonic semiring, we can also use an element to approximate the combination absorbing element. This approach results in an approximate arc consistency enforcing algorithm [14].

3.2 Junction Tree Algorithms for Soft Constraints

The major motivation that prompts researchers to use junction tree (JT) algorithms to solve CBI problems is handling multiple queries by sharing intermediate computational results. In general, junction tree algorithms assign constraints to clusters and combine constraint in the same cluster. The combined constraint is marginalized and passed as a message between clusters. Following a specified message-passing scheme, the junction tree reaches consistency and any subset of variables can be queried through marginalizing out other variables in the cluster that contains these variables.

The basic version of message-passing scheme in JT algorithms is Shenoy-Shafer (SS) architecture [15]. SS architecture has no special requirement for the two operators, thus general CBI problems, including all soft constraint proposals, can be processed without any modification. If the two operators of the commutative semiring have special properties, the message-passing schemes can be modified to achieve better computational efficiency. For example, the idempotent combination operator $\otimes$ implies that repeatedly combining the same information will not produce new information. This property enables us proposing a JT-Idemp message-passing scheme in [1]. Classic CSPs and Fuzzy CSPs can be processed by JT algorithms with this scheme. For CBI problems such as Max CSPs and Weighted CSPs that are defined on semirings with an invertible combination operator $\otimes$, in other words, $\forall a \in \mathbf{A}, \exists a^{-1} \in \mathbf{A}, s.t. a \otimes a^{-1} = \mathbf{1}$, two popular message-passing schemes in probability inference, the Lauritzen-Spiegelhalter (LS) architecture [16] and the HUGIN architecture [17], can be generalized and applied. Details of these generalized JT algorithms for CBI problems can be found in [1].

3.3 Loopy Message Propagation for General CBI Problems

As already known, in JT algorithms both time and space complexities are bounded by the maximum cluster size. To maintain junction tree properties, the maximum cluster size is usually large in practical problems. Loopy message propagation [18] is an approximate probability inference approach that aims at relaxing the junction tree properties to make computation feasible. Using junction graphs, instead of junction trees, means that the message-passing may not terminate due to the introduction of loops. Also messages will be repeatedly counted. Both of these bring errors of inference for general CBI problems. However, we claimed in [1] that for CBI problems with a idempotent combination operator, such as Classic CSPs and Fuzzy CSPs, the loopy message propagation is an exact inference approach. Our experimental results also showed that the loopy message propagation yields high quality inference approximation for general CBI problems like Max CSPs and probability assessment. Like message-passing schemes in JT algorithms, the schemes in the loopy message propagation can be revised according to different semiring properties to achieve better computational efficiency.

3.4 Stochastic and Context-Specific Inference

Basically, all the inference algorithms generalized in our framework are systematic approaches to CBI problems. In concrete application domains, stochastic approaches, such as sampling techniques in probability inference, are very successful. Given the fact that underlying representations of these problems are highly analogous, an abstract representation of stochastic or hybrid (stochastic and systematic) inference approaches should be a reasonable goal for future research. Context-specific inference has drawn a lot of attentions in the probability inference community recently. The same idea appears in SAT as the DPLL algorithm [19] for a long time. After instantiating one or several variables (context), the original CBI problem will be probably simplified. Solving the simplified problem together with backtracking techniques provide another promising inference approach for general CBI problems.

4 Conclusion

Many real world inference problems, under the umbrella term of Constraint-Based Inference (CBI), share common features in the problem representation and essentially identical inference approaches. The observation prompts us proposing a generalized semiring-based algorithmic and representation framework for CBI problems, based on the synthesis of existing generalized frameworks from both constraint processing and probability inference communities. Our unified CBI framework provides opportunities for researchers from different fields to reinterpret many familiar approaches in their domains at a higher level.

More specifically, we are aiming at comparing various exact and approximate inference approaches in both constraint processing and probability inference communities. Based on the semiring-based unified framework for CBI problems, we show that arc consistency, one of the most important techniques in constraint processing, can be applied to general CBI problems like probability inference. We also show that the widely studied exact and approximate inference approaches in probability inference, such as junction tree algorithms and loopy message propagation, are suitable to be applied in constraint processing. In other words, based on the semiring-based unified CBI framework, we can borrow ideas from other fields to improve the inference algorithm design in constraint processing. Our knowledge in handling constraints is also applicable in other domains. The unified CBI framework acts as a bridge in exchanging ideas between researchers from different fields.

Acknowledgments

We thank the anonymous reviewers for their comments on this paper. This research was supported by the Natural Sciences and Engineering Research Council of Canada and the Institute for Robotics and Intelligent Systems. Alan K. Mackworth holds a Canada Research Chair.

References

1. Chang, L.: Generalized constraint-based inference. Master's thesis, Dept. of Computer Science, Univ. of British Columbia (2005)
2. Bistarelli, S., Montanari, U., Rossi, F.: Semiring-based constraint satisfaction and optimization. J. ACM **44** (1997) 201–236
3. Schiex, T., Fargier, H., Verfaillie, G.: Valued constraint satisfaction problems: Hard and easy problems. In: IJCAI95, Montreal (1995) 631–637
4. Schiex, T.: Arc consistency for soft constraints. In: Proceedings of CP2000. (2000) 411–424
5. Cooper, M., Schiex, T.: Arc consistency for soft constraints. Artificial Intelligence **154** (2004) 199–227
6. Bistarelli, S.: Semirings for Soft Constraint Solving and Programming. Springer-Verlag (2004)
7. Dechter, R.: Bucket elimination: A unifying framework for probabilistic inference. In: 12th Conf. on Uncertainty in Artificial Intelligence. (1996) 211–219
8. Aji, S.M., McEliece, R.J.: The generalized distributive law. IEEE Transactions on Information Theory **46** (2000) 325–343
9. Kask, K., Dechter, R., Larrosa, J.: Unifying cluster-tree decompositions for automated reasoning. Submitted to the AIJ (2003)
10. Kohlas, J., Shenoy, P.: Computation in valuation algebras. In: Handbook of Defeasible Reasoning and Uncertainty Management Systems, Volume 5: Algorithms for Uncertainty and Defeasible Reasoning. Kluwer, Dordrecht (2000) 5–40
11. Mackworth., A.K.: Consistency in networks of relations. Artificial Intelligence **8** (1977) 99–118
12. Mackworth, A.K.: On reading sketch maps. In: IJCAI77. (1977) 598–606
13. Mohr, R., Masini, G.: Good old discrete relaxation. In: European Conference on Artificial Intelligence. (1988) 651–656
14. Chang, L., Mackworth, A.K.: A generalization of generalized arc consistency: From constraint satisfaction to constraint-based inference. In: 5th Workshop on Modelling and Solving Problems with Constraints, Edinburgh (2005) (8pp.).
15. Shenoy, P.P., Shafer, G.: Axioms for probability and belief-function propagation. In: Uncertainty in Artificial Intelligence 4. North-Holland (1990) 169–198
16. Lauritzen, S.L., Spiegelhalter, D.J.: Local computations with probabilities on graphical structures and their application to expert systems. In: Readings in Uncertain Reasoning. Kaufmann (1990) 415–448
17. Jensen, F.V., Lauritzen, S.L., Olesen, K.G.: Bayesian updating in causal probabilistic networks by local computations. Computational Statistics Quarterly **4** (1990) 269–282
18. Murphy, K.P., Weiss, Y., Jordan, M.I.: Loopy belief propagation for approximate inference: An empirical study. In: Conf. on UAI. (1999) 467–475
19. Davis, M., Logemann, G., Loveland, D.: A machine program for theorem-proving. Commun. ACM **5** (1962) 394–397