TOWARDS A UNIFIED SOLUTION FOR CONSTRAINT-SATISFACTION PROBLEMS: A SURVEY-PROPAGATION APPROACH BASED ON NORMAL REALIZATIONS

Ronghui Tu, Yongyi Mao and Jiying Zhao

School of Information Technology and Engineering, University of Ottawa
800 King Edward Ave., Ottawa, Ontario, Canada K1N 6N5
Email: {rtu, yymao, jyzhao}@site.uottawa.ca

ABSTRACT

Motivated by the celebrated success of survey propagation (SP) in solving k-SAT problems and its recent applications in coding and data compression, this paper approaches general constraint satisfaction problems from a unified perspective, aiming at developing a general SP-style algorithmic framework for such problems. Although many aspects along our direction remain open, in this paper, we have arrived at a unified combinatorial framework, "lifting" the solution space to what we call the space of all "rectangles". We also present a Markov random field (MRF) formalism over this space using a normal realization. This MRF formalism then brings to surface a new SP-style algorithm family, which contains the existing SP algorithms as a special case.

1. INTRODUCTION

Survey propagation (SP) [1] is a recent revolutionary algorithmic discovery in solving a classical class of NP-complete problems, the $k-$SAT problems [2]. As a message-passing procedure on the *factor graph* representation [3] of the problem instance, SP has been shown to be effective even for very large and difficult instances of $k-$SAT problems (see, e.g., [1] and [4]). Recently, the philosophy of SP has been applied, on a case-by-case basis, to solving other constraint satisfaction problems (see, e.g., [5]). In the context of communications and data compression, very recently, Yu and Aleksic have applied SP to coding for a special class of broadcast channels [6], and Wainwright and Maneva have applied a generalized version of SP [4] to source-coding problems [7]; in both cases, great successes were demonstrated.

In the context of $k-$SAT problems, the key of SP is the introduction of a "joker" symbol (*) to every variable alphabet, where a variable taking the joker symbol means that it is free to take any values in the original alphabet, and a variable taking a non-joker symbol means that it is constrained to taking a given value in the original alphabet. A milestone in the understanding of SP in this context is the work of [4]. In [4], the authors introduced a combinatorial framework — a Markov Random Field (MRF) formalism over the extended configuration space (i.e., each variable alphabet includes the additional joker symbol). Based on this formalism, the authors of [4] generalized SP to a family of algorithms parameterized by a single real number between 0 and 1, as a solver for $k-$SAT problems. In addition, it is shown in [4] that this family of algorithms may be interpreted as an instance of the well-known sum-product (or belief-propagation) algorithm [3] on their MRF model. The MRF formalism of [4] and the sum-product interpretation therein are important in two respects. On one hand, they allow the well-developed analytic tools for the sum-product algorithm to be used in the understanding of the SP algorithms. On the other hand, since there naturally exists a partial ordering on the space of all extended configurations, a hierarchical (lattice) structure "connects" the original satisfying configurations, which may otherwise form a large number of "disconnected clusters"; that is, lifting the k-SAT problem from the original configuration space to the extended configuration space provides a combinatorial framework, potentially enabling a deeper understanding of the inherent structure of the problem.

Motivated by [4], we recently have introduced "generalized state variables" to $k-$SAT problems and constructed normal realizations [8] for these problems [9]. In that framework, SP can also be reduced from the sum-product algorithm and the reduction thereof appears simpler and more transparent.

In this paper, we consider general constraint satisfaction problems of arbitrary form, where the variables may take values from an arbitrary finite alphabet and the constraints may assume arbitrary forms. We do require however that the variables are sparsely constrained so as to allow for SP-style local message-passing algorithms. The contributions of this paper are outlined as follows.

We introduce a generic combinatorial framework for constraint satisfaction problems, generalizing that of [4]. Specifically, a key component of this framework is a transformation of the problems of finding a satisfying point in the original space to a problem of finding a "rectangle" (Cartesian product) containing satisfying points. Each "side" of a rectangle takes on values from the power set of original alphabet, and the notion of "joker" in this framework has the natural interpretation as a symbol in the power set, namely, the original alphabet itself. The counter-parts of the fundamental objects in [4] such as partial ordering, connectivity of solutions and the notion of cores are also made explicit in this generalized framework. We also present an MRF formalism for the set of all valid rectangles, based on normal realization, generalizing the work of [9] on $k-$SAT problems. Potentially useful for other purposes, this MRF formalism, primarily under sum-product message-passing rules, naturally leads to a rich SP-style algorithm family — which we call the marginalization-restriction (MR) algorithm — generalizing the existing SP algorithms as special cases.

The remainder of this paper is organized as follows. In Section 2, we formulate constraint satisfaction problems in a generic form. In Section 3, we present a combinatorial framework for such problems. In Section 4, we present the normal realization associated with this framework. In Section 5, we present the family of MR algorithms. The paper is briefly concluded in Section 6. Length constraints prohibit the inclusion of some details, and all proofs are omitted.

0-7803-9528-X/06/$20.00 ©2006 IEEE

2. A GENERIC FORMULATION OF CONSTRAINT-SATISFACTION PROBLEMS

Let V be a finite set indexing a set of variables $\{x_v : v \in V\}$, where each variable x_v takes on values from some set χ_v. For any subset $U \subseteq V$, we will use x_U to denote the variable set $\{x_v : v \in U\}$. We note that depending on the context, x_U may also be interpreted as a configuration in $\prod_{v \in U} \chi_v$.

Let C be another finite set indexing a set of constraints $\{\Gamma_c : c \in C\}$, the form of which will be specified subsequently. For each $c \in C$, let $V(c)$ be some subset of V, indexing the set of variables constrained by Γ_c. Symmetrically, for each $v \in V$, we will denote the set $\{c : v \in V(c)\}$ by $C(v)$, namely, $C(v)$ indexes the set of all constraints involving variable x_v. Since each constraint Γ_c applies only on variables $x_{V(c)}$, we will identify constraint Γ_c as a subset of the Cartesian product $\prod_{v \in V(c)} \chi_v$. Thus a constraint satisfaction problem is completely specified by $(V, C, \{\chi_v : v \in V\}, \{V(c) : c \in C\}, \{\Gamma_c : c \in C\})$. Then the objective of a constraint satisfaction problem may be phrased as finding a solution for equation

$$\prod_{c \in C} [x_{V(c)} \in \Gamma_c] = 1, \tag{1}$$

where the notation $[P]$, for any boolean proposition P, is the usual Iverson's convention [3], namely, evaluating to 1 if P, and to 0 otherwise.

Clearly, (1) can be represented by a factor graph, which is indeed a canonical representation in the context of $k-$SAT problems. The factor graph presentation of a toy problem is shown in Figure 1. We will use $E(G)$ to denote the set of edges of factor graph G.

For any U and S with $S \subset U \subseteq V$ and any subset $\Omega \subseteq \prod_{v \in U} \chi_v$, we denote

$$\Omega_{|S} := \{\alpha \in \prod_{v \in S} \chi_v : (\alpha, \beta) \in \Omega \text{ for some } \beta \in \prod_{v \in U \setminus S} \chi_v\},$$

that is, $\Omega_{|S}$ is the *projection* of Ω on S.

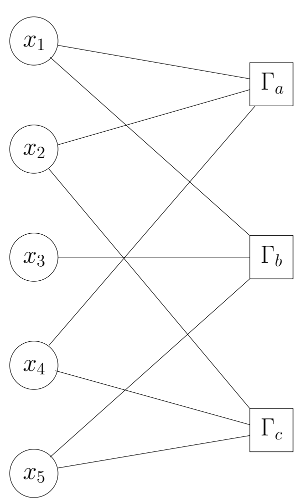

Fig. 1. A factor graph for a constraint satisfaction problem specified by $[(x_1, x_2, x_4) \in \Gamma_a][(x_1, x_3, x_5) \in \Gamma_b][(x_2, x_4, x_5) \in \Gamma_c] = 1$.

3. A GENERAL COMBINATORIAL FRAMEWORK

To establish a general combinatorial framework for arbitrary constraint satisfaction problem, for each $v \in V$, let χ_v^* be the power set of χ_v, i.e., $\chi_v^* = \{a : a \subseteq \chi_v\}$. For each variable x_v in the given problem, we introduce a new variable y_v, taking values from χ^*. A configuration y_U, for any subset $U \subseteq V$, is understood as the Cartesian product $\prod_{v \in U} y_v$, which we refer to as a *rectangle*. Given rectangle y_U and some $v \in U$, we refer to its $v-$component y_v, as the $v-$*side* of y_U. Clearly, rectangle y_U has $|U|$ sides and if any side of y_U is the empty set, then so is the rectangle.

Given a variable index $v \in V$ and a non-empty rectangle $y_{V(c)\setminus\{v\}}$, we define

$$\mathtt{F}_c(y_{V(c)\setminus\{v\}}) := \left((y_{V(c)\setminus\{v\}} \times \chi_v) \cap \Gamma_c \right)_{|\{v\}}.$$

That is, set $\mathtt{F}_c(y_{V(c)\setminus\{v\}})$ is the largest subset of χ_v in which every element, when paired with some sequence in $\prod_{u \in V(c)\setminus\{v\}} \chi_u$, makes constraint Γ_c satisfied.

The following lemma is immediate.

Lemma 1 *Let $y_{V(c)\setminus\{v\}}$ and $y'_{V(c)\setminus\{v\}}$ be two rectangles for some c and $v \in V(c)$. Then if $y_{V(c)\setminus\{v\}} \subset y'_{V(c)\setminus\{v\}}$, then $\mathtt{F}_c(y_{V(c)\setminus\{v\}}) \subseteq \mathtt{F}_c(y'_{V(c)\setminus\{v\}})$.*

We say that a rectangle y_V is *valid* for constraint Γ_c if y_V is non-empty and for every variable index $v \in V(c)$, $y_v \subseteq \mathtt{F}_c(y_{V(c)\setminus\{v\}})$; otherwise, we say it is *invalid* for Γ_c. A rectangle is said to be *globally valid* (or simply *valid*) if it is valid for all constraints. We will now denote the set of all valid rectangles by Ξ.

Lemma 2 *If x_V satisfies (1), then $x_V \in y_V$ for some $y_V \in \Xi$.*

That is, the union of all valid rectangles contains all solutions to (1). As we will show later, this result will underline the shift of solution paradigm for satisfaction problems from finding a "point" in the solution space to finding a rectangle containing the solutions.

To make a correspondence with the the framework of [4], we also generalize the notion of "core" in [4]: a valid rectangle $y_V \in \Xi$ is said to be a *core* if for every constraint Γ_c and every variable index $v \in V$, $y_v = \bigcap_{c \in C(v)} \mathtt{F}_{v,c}(y_{V(c)\setminus\{v\}})$.

We now define a partial order on Ξ via a directed graph. Let each node on the graph represent a rectangle in Ξ; there is a directed edge from node y_V to node y'_V if and only if $y_v = y'_v$ for every $v \in V$ except for some u, at which $y'_u \supset y_u$ and $|y'_u \setminus y_u| = 1$. Now on this directed graph, for any two nodes y_V and y'_V, we write $y'_V \prec y_V$ if there is a directed path from y'_V to y_V. Clearly, $\prec$ is a partial order on Ξ.

Theorem 1 *Rectangle $y_V \in \Xi$ is a maximal element if and only if y_V is a core.*

Similar to the lattice structure in [4], the combinatorial structure of Ξ, namely, the above-defined directed graph and the partial order thereof, may serve as a fundamental framework underlying a constraint satisfaction problem. For $k-$SAT problems, in which variable alphabet is $\{0, 1\}$, each side variable y_v takes values from $\chi_v^* = \{\emptyset, \{0\}, \{1\}, \{0, 1\}\}$. For valid rectangles, $y_v = \emptyset$ is not allowed, thereby reducing the alphabet of side variables to $\{\{0\}, \{1\}, \{0, 1\}\}$. Under the one-to-one correspondence between $\{\{0\}, \{1\}, \{0, 1\}\}$ and $\{0, 1, *\}$ defined by $\{0\} \leftrightarrow 0$, $\{1\} \leftrightarrow 1$, and $\{0, 1\} \leftrightarrow *$, it is easy to verify that the directed graph defined above precisely reduces to the lattice in [4]. In light of this correspondence, we remark that the view of the joker symbol as “no warning” in the existing literature may be less well-principled.

In [4], it was pointed out that the satisfying solutions of difficult instances of $k-$SAT problems form exponentially many “disconnected clusters” and that the lattice of [4] allows a connection between the “clusters”. As one naturally expects that a similar behaviour exists in general satisfaction problems, we now make precise the notions of “connectedness” and “clusters” intended by [4] and pondered in [5].

Let S be a subset of $\prod_{v\in V}\chi_v$. Two configurations $x_V, x'_V \in \prod_{v\in V}\chi_v$ are said to be *connected via* S, denoted by $x_v \overset{S}{\sim} x'_V$ if $x_V = x'_V$ or if there exists a sequence of pairs

$$(x_V, \alpha_1), (\alpha_1, \alpha_2), \ldots, (\alpha_{n-1}, \alpha_n), (\alpha_n, x'_V)$$

such that every $\alpha_i \in S$ and

$$D(x_V, \alpha_1) = D(\alpha_1, \alpha_2) = \ldots = D(\alpha_{n-1}, \alpha_n) = D(\alpha_n, x'_V) = 1,$$

where $D(\cdot)$ is the Hamming distance. Configurations x_V and x'_V are said to be *separated* by S if they are not connected via S. Two sets $B, B' \subseteq \prod_{v\in V}\chi_v$ are said to be *separated* by S if for every $x_V \in B$ and every $x'_V \in B'$, x_V and x'_V are separated by S. If B and B' are not separated by S, then they are said to be *connected* via S. Given a subset $S \subseteq \prod_{v\in V}\chi_v$, it can be verified that for any subset $B \subseteq S$, connectedness $\overset{S}{\sim}$ is an equivalence relation on B, and we refer to an equivalence class in B induced by equivalence relation $\overset{B}{\sim}$ as a *cluster* of B with respect to S.

Theorem 2 *For every three rectangles y_V, y'_V, and $y''_V \in \Xi$, if $y_V \prec y''_V$ and $y'_V \prec y''_V$, then y_V and y'_V are connected via y''_V.*

Therefore, whether or not two satisfying configurations are connected (via the set of all satisfying configurations), if there is a common rectangle via which they are connected, then they are connected via a core.

As the set of all satisfying configurations may form a large number of separate clusters, algorithms based on local search in $\prod_{v\in V}\chi_v$ often fail to work. The connectivity of Ξ however may enable efficient solvers, where SP is such an example. As we will see in Section 5, there in fact exists a rich family of algorithms within this framework, well beyond SP.

4. NORMAL REALIZATIONS

We now define an MRF over the set of all rectangles in $\prod_{v\in V}\chi_v$ representing a distribution over Ξ.

For each edge $(x_v, \Gamma_c) \in E(G)$ in the canonical factor graph representation of the problem, we introduce a generalized state variable (referred to as state variable hereafter) $s_{v,c}$ which takes on values in set $\chi_v^* \times \chi_v^*$. Each state variable $s_{v,c}$ will also be written as an ordered pair $(s_{v,c}^L, s_{v,c}^R)$, where the first component is referred to as the left state, and the second component is referred to as the right state. For any $v \in V$ and any subset $C' \subseteq C(v)$, we use $s_{v,C'}$ to denote the variable set $\{s_{v,c} : c \in C'\}$. Likewise, for any $c \in C$ and any subset $V' \subseteq V(c)$, we use $s_{V',c}$ to denote the variable set $\{s_{v,c} : v \in V'\}$. In addition, we will use $s_{V,C}$ to denote the set of all state variables. Similar notations apply to left and right states.

In the MRF, local potential functions consist of *left functions* — each for a $v \in V$ and denoted by g_v — and a set of *right functions* — each for a $c \in C$ and denoted by f_c.

The left function g_v for each $v \in V$ is defined as

$$g_v(y_v, s_{v,C(v)}) := W_L(y_v, s_{v,C(v)}) \cdot [y_v \subseteq \bigcap_{c\in C(v)} s_{v,c}^R] \times$$
$$[y_v \neq \emptyset] \cdot \prod_{c\in C(v)} [s_{v,c}^L = \left(\bigcap_{c\in C(v)\setminus\{c\}} s_{v,c}^R \right) \cap y_v]$$

where $W_L(y_v, s_{v,C(v)})$ is a non-negative weighting function.

The right function f_c for each $c \in C$ is defined as

$$f_c(s_{V(c),c}) := W_R(s_{V(c),c}) \cdot \prod_{v\in V(c)} [s_{v,c}^R = \mathtt{F}_c(s_{V(c)\setminus\{v\},c}^L)]$$

where $W_R(s_{V(c),c})$ is another non-negative function.

We then define a global function F via the above-defined local functions:

$$F(y_V, s_{V,C}) := \prod_{v\in V} g_v(y_v, s_{v,C(v)}) \cdot \prod_{c\in C} f_c(s_{V(c),c}). \quad (2)$$

Then function F is readily expressed as a factor graph. Notice that in this factor graph, all symbol variables have degree one and all state variables have degree two, giving rise to a normal realization [8]. Following [8], such a factor graph can be more compactly represented as a Forney graph, where each state variable is represented as an edge, each side variable represented as a terminal or “half-edge”, and local functions represented as nodes. The Forney graph representing the normal realization of the toy problem in Figure 1 is shown in Figure 2.

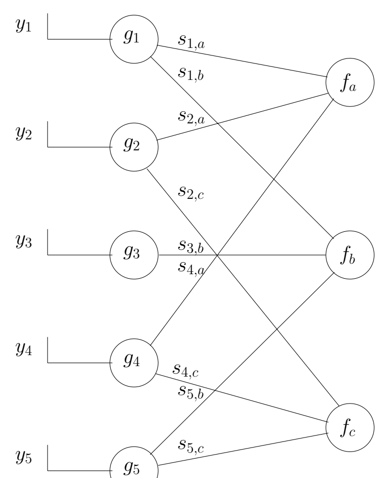

Fig. 2. The Forney graph representing the normal realization of the toy problem in Figure 1.

As is standard, when treating function F (upon normalization) as a probability mass function, the factorization structure of F defines an MRF.

Lemma 3 *If the weighting functions $W_L(\cdot)$ and $W_R(\cdot)$ are strictly positive, then for every sequence $(y_V, s_{V,C})$ in the support of F, it holds that*

$$s_{v,c}^L \subseteq y_v \subseteq s_{v,c}^R.$$

for every (v, c) such that $(x_v, \Gamma_c) \in E(G)$.

Theorem 3 *If the weighting functions $W_L(\cdot)$ and $W_R(\cdot)$ are strictly positive, then the support of function $F(\cdot)$ with state sequences punctured gives rise to precisely Ξ.*

At this end, we conclude that the normal realization presented here precisely realizes Ξ, and thus forms a natural representation of the combinatorial framework in Section 3. As the notion of states are fundamental in studying the structure of codes, we expect normal realizations and Forney graphs to provide additional insight in understanding the structure of a constraint satisfaction problem.

5. SOLVING THE PROBLEM

Now if we apply the sum-product or max-product message-passing rule to the MRF defined above, a generic algorithm for an arbitrary constraint satisfaction problem immediately arises. If the Forney graph is a tree, one can immediately conclude that the max-product algorithm returns a valid rectangle with the highest probability mass under the MRF model and that the sum-product algorithm returns a set of sides, each for a side variable y_v and equal to a subset of χ_v that is shared as the $v-$side most "heavily" by the valid rectangles, where "heavy" is in the sense of maximizing the sum of the probability mass of the sharing rectangles – the Cartesian product of these sides gives rise to a rectangle. Then one may use a local search algorithm to find a satisfying solution within the returned rectangle.

In light of the decimation procedure involved in SP, the above algorithm framework is modified as follows, and designated *marginalization and restriction (MR) algorithm*.

For a given function $H(y_V, s_{V,C})$, for each $v \in V$, we denote $h_v^H(y_v) := \bigoplus_{y_{V\setminus\{v\}}, s_{V,C}} H(y_V, s_{V,C})$, where $\bigoplus$ is a generalized notion of summation. We refer to h_v^H as a marginal function of $H(y_V, s_{V,C})$. It is well-known that the set of all marginal functions $\{h_v^H(y_v) : v \in V\}$ can be computed or approximated using the sum-product algorithm, provided that the function $H(\cdot)$ has a factor graph representation.

An MR algorithm is an iterative procedure where in iteration l, we set $H_l(y_V, s_{V,C}) := H_{l-1}(y_V, s_{V,C}) \prod_{v\in U_{l-1}} [y_v = \alpha_v]$. and H_0 is set to F. In each iteration, the set of marginals $\{h_v^{H_l} : v \in V\}$ are first computed using the sum-product — this step is referred to as the marginalization step. A subset of variable indices $U_l \subseteq V$ are then selected and for each $v \in U_l$, $\alpha_v := \arg\max_{y_v} h_v^{H_l}(y_v)$ is determined. The criterion for selecting U_l is such that for each $v \in U_l$, $h_v^{H_l}(\alpha_v)$ need to be sufficiently large, either in a sense relative to $h_v^{H_l}$ evaluated at other y_v, or in a sense relative to other $h_u^{H_l}(\alpha_u)$, $u \in V \setminus \{v\}$, or in a sense combining both aspects. This step is referred to as the restriction step, since the result of this step will be used to restrict the function H^{l+1} to a smaller space. The procedure iterates until finding a solution for equation $H_l(y_V, s_{V,C}) > 0$ is sufficiently simple with some local search algorithm.

It is worth noting that when computing the marginals, one may consider the option of initializing the messages to some particular forms, so as to simplify the updating equations or to drive the algorithm to some desired behaviour.

In addition, we remark that for large variable alphabet χ_v, χ_v^* becomes prohibitively large. In this case, some modification or approximation of MR is necessary for computation to be manageable; proper choice of the weighting functions may also be incorporated for this purpose.

For $k-$SAT problems, the normal realization formalism is given in [9]. One can easily verify that a version of MR algorithm family applied to the normal realization in [9] gives rise to the generalized SP and decimation algorithm presented in [4].

6. CONCLUSION

Generalizing the work of [4], in this paper, we introduce a generic combinatorial framework for arbitrary constraint satisfaction problems. We then introduce the notion of normal realization — generalizing from our results in [9] — to arbitrary such problems. The normal realizations naturally fit in the combinatorial framework, and are believed to be fundamental in understanding the combinatorial nature of the problem instance. Based on normal realizations, we then introduce a new class of algorithms, which we call the MR algorithms, for solving arbitrary constraint satisfaction problems. The existing survey propagation algorithm and its generalization appear to be a special case of the MR family.

This work appears to have opened up a rich direction of future research. On one hand, many problems in communications may seek solutions within this framework. On the other hand, many questions in this framework remain to be answered — for example, how to systematically construct the weighting functions in the normal realization? What is the dynamics of the MR algorithms? How messages should be initialized for a given problem? It is our hope that this work inspire more research activities along these directions.

7. REFERENCES

[1] M. Mézard, G. Parisi, and R. Zecchina, "Analytic and algorithmic solution of random satisfiability problems," *Science*, , no. 297, pp. 812–815, 2002.

[2] S. A. Cook, "The complexity of theorem-proving procedures," in *3rd Annual ACM Symposium on Theory of Computing*. Association for Computing Machinery, 1971, pp. 151–158.

[3] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 498–519, Feb 2001.

[4] E. Maneva, E. Mossel, and M. J. Wainwright, "A new look at survey propagation and its generalizations," in *SODA*, 2005, pp. 1089–1098.

[5] A. Braunstein, M. Mézard, M. Weight, and R. Zecchina, *Constraint satisfaction by survey propagation*, Sep. 2003, http://arxiv.org/abs/cond-mat/0212451.

[6] W. Yu and M. Aleksic, "Codig for the blackwell channel: a survey propagation approach," in *IEEE International Symposium on Information Theory (ISIT)*, Adelaide, Australia, 2005, pp. 1583–1587.

[7] M. J. Wainwright and E. Maneva, "Lossy source encoding via message-passing and decimation over generallizaed codewords of LDGM codes," in *IEEE International Symposium on Information Theory (ISIT)*, Adelaide, Australia, 2005, pp. 1493–1497.

[8] Jr. G. D. Forney, "Codes on graphs: normal realizations," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 520–548, Feb. 2001.

[9] R. Tu, Y. Mao, and J. Zhao, "On generalized survey propagation: normal realization and sum-product interpretation," submitted to IEEE International Symposium on Information Theory 2006.