AND/OR Search Spaces and the Semantic Width of Constraint Networks

Student: Robert Mateescu
Supervisor: Rina Dechter

School of Information and Computer Science
University of California, Irvine, CA 92697-3425
{dechter,mateescu}@ics.uci.edu

Abstract. In recent work we introduced the perspective of AND/OR search spaces for various types of graphical models [3, 4, 6]. In this paper we provide a broader exposition of the subject for the case of constraint networks and we introduce the new concept of *semantic width*, that captures, more accurately than the well known parameter *tree-width*, the size of the minimal AND/OR search graph. The semantic width can explain the often noted disparity between the actual size of a compiled OBDD and the upper bound exponential in the path-width.

1 Introduction

The primary contribution of this paper consists in using the *AND/OR search space* paradigm [3, 4, 6] to define the new concept of *semantic width* of a constraint network. The well known parameter tree-width is graph based, and therefore cannot capture context sensitive information. This often results in a very loose upper bound on the actual complexity of the problem, a typical example being the result of compilation scheSanDisk SDCFH-1024-901 1 GB Ultra II CompactFlash Card mes such as OBDDs. The semantic width is based on the notion of equivalent constraint networks. The idea is to capture the intrinsic hardness of a problem by the smallest width equivalent network.

2 AND/OR Search Spaces for Constraint Networks

A *constraint network* $\mathcal{R} = \langle X, D, C \rangle$ is defined by a set of variables $X = \{X_1, ..., X_n\}$, their respective finite domains $D = \{D_1, ..., D_n\}$ and a set of constraints $C = \{C_1, ..., C_t\}$. Each constraint is a pair $C_i = (S_i, R_i)$, where $S_i \subseteq X$ is the scope of the relation R_i, denoting the allowed combination of values: The *constraint graph* has variables as its nodes and edges between variables appearing in the same constraint.

Definition 1 (pseudo tree [5]). *Given an undirected graph $G = (V, E)$, a directed rooted tree $T = (V, E')$ defined on all its nodes is called* pseudo tree *if any arc of G which is not included in E' is a back-arc, namely it connects a node to an ancestor.*

2.1 AND/OR Search Tree

Given a constraint network $\mathcal{R} = \langle X, D, C \rangle$, its constraint graph G and a pseudo tree T of G, the associated AND/OR search tree, denoted $S_T(\mathcal{R})$, has alternating levels of

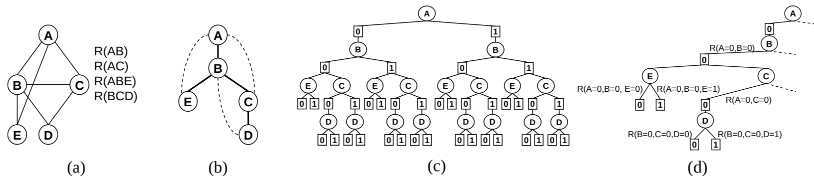


Fig. 1. AND/OR Search Tree

AND and OR nodes. The OR nodes are labeled X_i and correspond to the variables. The AND nodes are labeled $\langle X_i, x_i\rangle$ and correspond to the value assignments in the domains of the variables. The structure of the AND/OR search tree is based on the underlying backbone tree T. The root of the AND/OR search tree is an OR node labeled with the root of T. The children of an OR node X_i are AND nodes labeled with assignments $\langle X_i, x_i\rangle$ that are consistent with the assignments along the path from the root, $path(x_i) = (\langle X_1, x_1\rangle, \langle X_2, x_2\rangle, \ldots, \langle X_{i-1}, x_{i-1}\rangle)$. The children of an AND node $\langle X_i, x_i\rangle$ are OR nodes labeled with the children of variable X_i in the pseudo tree T. The arcs from X_i to $\langle X_i, x_i\rangle$ are associated with appropriate *labels* derived from the constraints in C.

Example 1. Figure 1a shows a constraint network. Figure 1b shows a pseudo tree of the constraint graph, together with the back-arcs (dotted lines). Figure 1c shows the AND/OR search tree based on the pseudo tree, for binary variables, and Figure 1d shows a portion of the labeled tree.

Theorem 1 ([5, 1, 2]). *Given a constraint network $\mathcal{R}$ and a pseudo tree T of depth m, the size of the AND/OR search tree based on T is $O(n \cdot \exp(m))$. A constraint network of tree-width w^* has an AND/OR search tree of size $O(n \cdot \exp(w^* \cdot \log n))$.*

Definition 2 (backtrack-free AND/OR search tree). *Given an AND/OR search tree $S_T(\mathcal{R})$ whose internal nodes have already been assigned consistency values of $0/1$, the* backtrack-free AND/OR search tree *of $\mathcal{R}$ based on T, denoted $BF_T(\mathcal{R})$, is obtained by pruning from $S_T(\mathcal{R})$ all inconsistent subtrees.*

2.2 AND/OR Search Graphs

It is often the case that a search space that is a tree can become a graph if identical nodes are merged, because identical nodes root identical search subspaces. Sometimes, two nodes may not look identical, but they might still root the same search subtrees.

Minimal AND/OR Search Graphs We next characterize the smallest search graph that may result from merging nodes.

Definition 3 (merging subtrees). *Assume a given arc-labeled AND/OR search tree $S_T(\mathcal{R})$ and assume two paths in the AND/OR search tree ending by nodes annotated by the same assignment $(\langle X_i, x_i\rangle)$, $s_1 = \pi_1(\langle X_i, x_i\rangle)$ $s_2 = \pi_2(\langle X_i, x_i\rangle)$. s_1 and s_2 are* unifiable *at $\langle X_i, x_i\rangle$ iff the arc-labeled search subgraphs rooted at s_1 and s_2 are identical. The* merge *operator , $merge_{(s_1,s_2)}$, transforms S_T into a graph S'_T by merging s_1 with s_2 at $\langle X_i, x_i\rangle$.*

Proposition 1. *Given an AND/OR search tree, its closure under AND-merge yields a unique fix point called* minimal AND/OR search graph*.*

Note that we can accommodate an even more general definition of merging of two AND nodes that are assigned different values from their domain, as long as they root identically labeled subtrees. Or, we can allow merging of OR nodes annotated by the same variable name if they root identical subtrees. While such extensions are straightforward, for clarity we will restrict ourselves to merging nodes having an identical variable-value assignment.

The Context-Based AND/OR Graph We will now present a general generative rule for unifying nodes in the AND/OR search graph that yields the size bound above. We denote by $d_{dfs}(T)$ a linear DFS (depth first search) ordering of a tree T.

Definition 4 (induced-width of a pseudo tree). *The induced width of G relative to pseudo tree T, $w_T(G)$, is the induced-width of the extended graph of G relative to T, G^T, along the $d_{dfs}(T)$ ordering.*

Proposition 2. *1. The minimal induced-width of G over all pseudo trees is identical to the induced-width (tree-width) w^* of G. 2. The minimal induced-width restricted to chains is identical to its path-width pw^*.*

Let G^{T^*}, be the induced graph of G^T. Clearly, each variable and its parent set in G^{T^*} is a clique. We associate each variable with its parent-separators.

Definition 5 (parent-separators). *Given the induced-graph, G^{T^*}, the parent-separators of X denoted psa_X, are formed by X and its earlier neighbors in G^{T^*} that are connected to a descendent of X in T.*

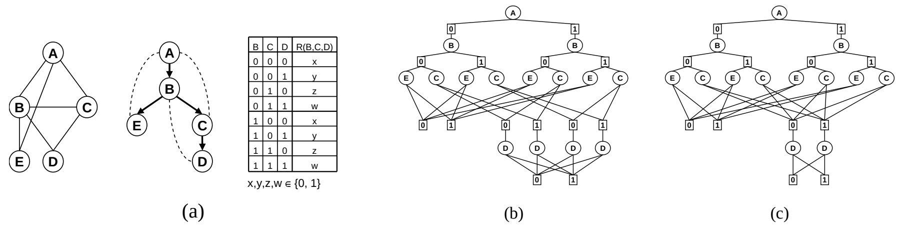
For every node X_i in G^{T^*}, the parent-separators of X_i separate, in G^{T^*}, the descendants of X from the rest of the graph. Therefore,

Theorem 2. *Given G^{T^*}, Let $s_1 = \pi_1(\langle X_{i+1}, x_{i+1}\rangle)$ and $s_2 = \pi_2(\langle X_{i+1}, x_{i+1}\rangle)$ be two partial paths of assignments in its AND/OR search tree S_T ending with two AND nodes, n_1 and n_2 both annotated by $\langle X_{i+1}, x_{i+1}\rangle$. If $s_1[psa_{X_{i+1}}] = s_2[psa_{X_{i+1}}]$, then the AND/OR search subtrees rooted at s_1 and s_2 are identical and n_1 and n_2 can be merged. $\pi(x_i)[psa_{X_i}]$ is called the context of $\pi(x_i)$.*

Definition 6 (context minimal AND/OR graph). *The AND/OR search graph of $\mathcal{R}$ relative to the backbone tree T that is closed under context-based merge operator is called* context minimal AND/OR graph *and is denoted $S_T^c(\mathcal{R})$.*

Example 2. Consider the example given in Figure 2a to the left. Relative to the (DFS) pseudo tree given, the context of A is A, of B it is AB, of C it is only BC since its parent A is not connected to a descendant of C in the pseudo tree. The context of D is D and of E it is E. The context minimal AND/OR search graph is given in Figure 2b.

Theorem 3. *Given a constraint network $\mathcal{R}$, its primal graph G, and a pseudo tree T having $w = w_T(G)$, the size of the context minimal AND/OR search graph based on T, $S_T^c(\mathcal{R})$, is $O(n \cdot k^w)$, when k bounds the domain size.*



B	C	D	R(B,C,D)
0	0	0	x
0	0	1	y
0	1	0	z
0	1	1	w
1	0	0	x
1	0	1	y
1	1	0	z
1	1	1	w

x,y,z,w ∈ {0, 1}

Fig. 2. Context minimal vs. minimal AND/OR graphs

Note that the criterion of merging only identical contexts is cautious. There could be many paths whose contexts are not identical, yet they might root identical subgraphs.

Example 3. Consider the constraint network given in Figure 2a, where $R(B, C, D)$ is given in the table. We can see that $R(B = 0, C, D) = R(B = 1, C, D)$. This can in fact shrink the context of C in the pseudo tree from BC to C. Figure 2b shows the context minimal graph, and Figure 2c shows the minimal graph, where each value of C annotates a single AND node only.

Theorem 4. *The context minimal AND/OR search graph S^c_T of a graph-models having a backbone tree with bounded tree-width, w can be generated in output polynomial time and space $O(\exp w)$.*

3 On the Uniqueness of the Minimal AND/OR Graph

We have already seen that the minimal AND/OR graph is unique for a constraint network, given a backbone pseudo tree. We will now prove a much more significant property of backtrack-free minimal AND/OR graphs. Namely that they are unique for all equivalent constraint networks given a backbone tree.

Definition 7 (strongly minimal AND/OR graph). *A* strongly minimal *AND/OR graph of $\mathcal{R}$ relative to pseudo tree T is the minimal AND/OR graph, $M_T(\mathcal{R})$, that is backtrack-free, denoted by $SM_T(\mathcal{R})$.*

We ask: given two equivalent constraint networks representing the same set of solutions, each having a different constraint graph, are their strongly minimal AND/OR search graphs identical? The above question is not well defined because an AND/OR graph for $\mathcal{R}$ is defined only with respect to a backbone pseudo tree. We can have two equivalent constraint networks having two different graphs where a pseudo tree for one graph may not be a pseudo tree for the other. We ask therefore a different question: given two equivalent constraint networks and given a backbone tree that is a pseudo tree for both, is the minimal AND/OR graph relative to T unique?

We will answer this question positively quite straightforwardly. We first show that equivalent networks that share a backbone tree have identical backtrack-free AND/OR search trees. Since the backtrack-free search trees uniquely determine their strongly minimal graphs, the claim follows.

Definition 8 (shared pseudo trees). *Given a collection of graphs on the same set of nodes, we say that the graphs share a pseudo tree T, if T is a pseudo tree of each of these graphs. A set of constraint networks on the same set of variables share a tree T, iff their respective constraint graphs share T.*

Theorem 5. *If $\mathcal{R}_1$ and $\mathcal{R}_2$ are two equivalent networks that share T, then $BF_T(\mathcal{R}_1) = BF_T(\mathcal{R}_2)$, and it follows that $SM_T(\mathcal{R}_1) = SM_T(\mathcal{R}_2)$.*

Definition 9 (intersection of search trees). *Lets R_1 and R_2 be two networks (not necessarily equivalent) sharing a tree T. The intersection of their AND/OR search trees relative to T, $S_T(R_1) \wedge S_T(R_2)$ is an AND/OR search tree based on T that includes all and only the common partial paths in both $S_T(R_1)$ and $S_T(R_2)$.*

We know that the notion of intersection of constraint networks is well defined. Moreover, intersecting two equivalent constraint networks yields an equivalent constraint network whose constraint graph is the union of both input networks' graphs.

Corollary 1. $S_T(R_1 \wedge R_2) = S_T(R_1) \wedge S_T(R_2)$.

Therefore, similar to Montanari's notion of minimal binary network of constraints [7] we can extend this notion relative to pseudo-trees.

Definition 10 (intersection AND/OR search tree for T). *The intersection AND/OR search tree of $\mathcal{R}$ for T, denoted $IS_T(\rho)$, when ρ is the set of solutions of R, is the intersection of all AND/OR search trees that share T and are equivalent to $\mathcal{R}$.*

Proposition 3. *Given a constraint network R that has a pseudo tree T, its backtrack-free AND/OR search tree is identical to its intersection AND/OR search tree for T. Namely $BF_T(R) = IS_T(R)$.*

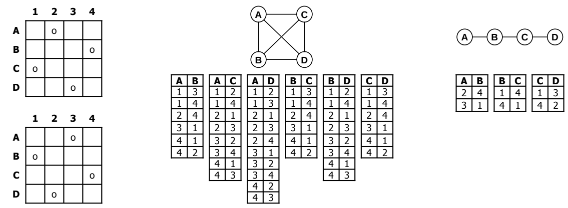
This now motivates a new semantic width concept.

Definition 11 (semantic width). *The* semantic-width of $\mathcal{R}$ relative to a pseudo tree T, *denoted $sw_T(\mathcal{R})$, is defined by $sw_T(\mathcal{R}) = min_{G \in G_T(R)} w_T(G)$ where $G_T(\mathcal{R})$ are all the constraint graphs of networks equivalent to $\mathcal{R}$ that share the pseudo-tree T. The* semantic width *of a constraint network, $\mathcal{R}$, is the minimal semantic-width over all the pseudo-trees that can express this network.*

Computing the semantic width is obviously a hard problem. However, the semantic width can explain why sometimes the minimal AND/OR graph or tree are much smaller than the upper bounds exponential in tree-width or path-width. In many cases, there could be a huge disparity between the tree-width of $\mathcal{R}$ and the semantic width along T.

Example 4. Figure 3a shows the two solutions of the 4-queen problem. The problem is expressed by a complete graph of tree-width 3, given in Figure 3b. Figure 3c shows an equivalent problem, which has tree-width 1. The semantic-width of the 4-queen is 1.

Clearly, if a network has a single solution, its semantic width is "0". Also, for any $\mathcal{R}$, $w_T(\mathcal{R})$ may be far larger than $sw_T(\mathcal{R})$ because there may be far sparser than R representations that are equivalent to $\mathcal{R}$. In particular, an OBDD's size is governed by the best graph representation and not by the input constraint graph. We can conclude:



(a) (b) (c)

Fig. 3. The 4-queen problem

Proposition 4. *The size of the strongly minimal AND/OR search graph of a constraint network $\mathcal{R}$ derived by a pseudo tree T is exponential in $sw_T(\mathcal{R})$ rather than $w_T(\mathcal{R})$.*

Example 5. Consider a constraint network on n variables such that every two variables are the equality constraint ($X = Y$). One graph representation is a complete graph, another is a chain and another is a tree. If the problem is specified as a complete graph, and if we use a linear order, the OBDD will have a linear size because there exists a representation having a path-width of 1 (rather than n).

4 Conclusion

This paper specializes the AND/OR idea [3, 4, 6] to constraint networks and elaborates the properties of AND/OR search graphs. The AND/OR search tree can be bounded exponentially by the depth of its pseudo tree. The AND/OR search tree can be turned into a graph at the expense of using more memory, by merging identical subtrees. The size of the context minimal AND/OR search graph is exponential in the tree-width. We also introduced a new graph parameter, the *semantic width*, which characterizes the size of the *minimal AND/OR graph*. This parameter can explain why sometimes huge disparities are noted between the actual result of compilation schemes (e.g. OBDDs) and the usual bound exponential in path-width.

References

1. R. Bayardo and D. Miranker. A complexity analysis of space-bound learning algorithms for the constraint satisfaction problem. In *AAAI'96*, 1996.
2. A. Darwiche. Recursive conditioning. *Artificial Intelligence Journal*, 125(1-2):5–41, 2001.
3. R. Dechter and R. Mateescu. Mixtures of deterministic-probabilistic networks and their AND/OR search space. In *UAI'04*, 2004.
4. R. Dechter and R.Mateescu. The impact of AND/OR search spaces on constraint satisfaction and counting. In *CP'04*, 2004.
5. E. C. Freuder and M. J. Quinn. Taking advantage of stable sets of variables in constraint satisfaction problems. In *IJCAI'85*, 1985.
6. R. Mateescu and R. Dechter. AND/OR cutset conditioning. In *International Joint Conference on Artificial Intelligence (IJCAI'05). To appear.*, 2005.
7. U. Montanari. Networks of constraints: Fundamental properties and applications to picture processing. *Information Science*, 7(66):95–132, 1974.