Towards a Definition of the Constraint Probability of Non-Binary Constraint Networks

Yaling Zheng and Berthe Y. Choueiry
Constraint Systems Laboratory
Department of Computer Science and Engineering
University of Nebraska-Lincoln
{yzheng|choueiry}@cse.unl.edu

Working Note of November 28, 2003
(WN-Nov-28-03)

Abstract

The constraint probability of a constraint network is the ratio of the number of constraints in the network over that of all possible constraints. While this parameter can be easily computed for binary networks, it is not clear how to compute it for non-binary networks. In a non-binary network, a constraint C_x of arity k induces constraints between any combination of the variables in the scope of C_x. These induced constraints need to be taken into consideration when computing the constraint probability of a non-binary network. In this report we propose a method for computing the constraint probability of a non-binary Constraint Network (CN), and illustrate the use of this new definition on a simple example.

1 Introduction

The constraint probability p of a Constraint Network (CN) is an important parameter for the empirical evaluation of the algorithms used to process and solve such networks. When the network has exclusively binary constraints, the constraint probability is defined in a straightforward manner as the ratio of the number of binary edges in the constraint network to the total number of possible edges:

$$p_1 = \frac{e}{e_{max}} \tag{1}$$

$$e_{max} = \frac{n(n-1)}{2} \tag{2}$$

where n is the number of nodes in the constraint graph, e is the number of binary edges, and e_{max} is the total number of possible edges.

When non-binary constraints exist in the network, no formal definition for this parameter exists to the best of our knowledge. The difficulty stems from the fact that a constraint C_x of arity k naturally induces constraints of arity i for all combinations of the $i \leq k$ variables in the scope of C_x. In order to specify unequivocally the constraint probability, we use the concept of an *induced* CN.

This report is structured as follows. Section 2 gives the definitions this study relies on and the ones we propose. Section 3 describes an algorithm to compute constraint probability of a non-binary constraint network. Section 4 explores alternative definitions of constraint probability. Section 5 gives the definition of constraint tightness of a non-binary CSP. Section 6 draws our conclusions and the issues that remain to be solved.

2 Definition

In this section, we propose a definition of constraint probability of a non-binary constraint networks. We first recall the definition of a Constraint Satisfaction Problem (CSP). Then we introduce the concept of the *sub-constraints* of a given constraint, which we use to define and discuss the constraint probability of a non-binary CSP.

2.1 CSP

A *Constraint Satisfaction Problem* (CSP) is defined as the tuple

$$\mathcal{P} = (\mathcal{V}, \mathcal{D}, \mathcal{C}), \text{where :} \tag{3}$$

- $\mathcal{V}$ is a finite set of n variables $\{V_1, V_2, \ldots, V_n\}$,
- $\mathcal{D}$ is a finite set of domains $\{D_1, D_2, \ldots, D_n\}$ for variables in $\mathcal{V}$, and
- $\mathcal{C} = \{C_1, C_2, \ldots, C_q\}$ is a finite set of constraints [Montanari 1974].

Each constraint C_i is defined by a pair (S_i, r_i), where

- S_i, a subset of $\mathcal{V}$, is called the *the constraint scope*,
- $m_i = |S_i|$ is the *constraint arity*, and
- r_i is the *constraint definition* defined as an m_i-ary relation over the constraint scope.

A *solution* to $\mathcal{P}$ is a set of values $\{a_1, a_2, \ldots, a_n\}$ assigned to $\{V_1, V_2, \ldots, V_n\}$ respectively such that all the constraints in $\mathcal{C}$ are satisfied.

A CSP is represented graphically by a *constraint network* (CN). In this network, the variables in $\mathcal{V}$ and the constraints in $\mathcal{C}$ are represented by two different types of nodes (i.e., variable node and constraint node). An edge links the node representing a constraint and the nodes representing the variables in the scope of the constraint. Figure 1 shows a constraint network with 4 variables $\{V_1, V_2, V_3, V_4\}$ and 2 constraints $\{C_{123}, C_{34}\}$. C_{123} is a ternary constraint over V_1, V_2, and V_3, and C_{34} is a binary constraint over V_3 and V_4.

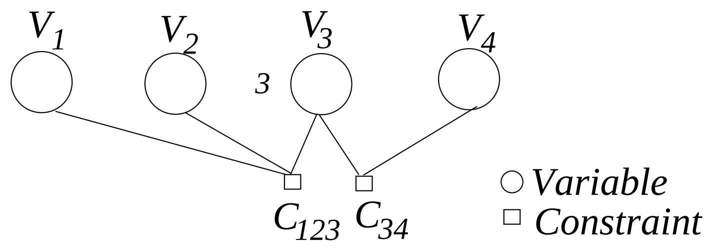

Figure 1: *Constraint network of a simple CSP.*

In our study, we assume that the constraint network CN is *simple*, *connected*, and *reduced*. We define these terms as follows (see Figure 2):

- *Simple:* A CN is said to be simple when no more than one constraint is defined over a given combination of variables (i.e., no two constraints have exactly the same scope).
- *Connected:* A CN is said to be connected when there is a path between every pair of variables in the network, where a path is a sequence of edges alternating between variables nodes and constraint nodes.
- *Reduced:* A CN is said to be reduced when there is no constraint whose scope is a subset of the scope of another constraint.

Naturally, a reduced CN is also simple. A reduced CSP can be achieved by repeatedly subsuming each constraint C_i whose scope is a subset of the scope

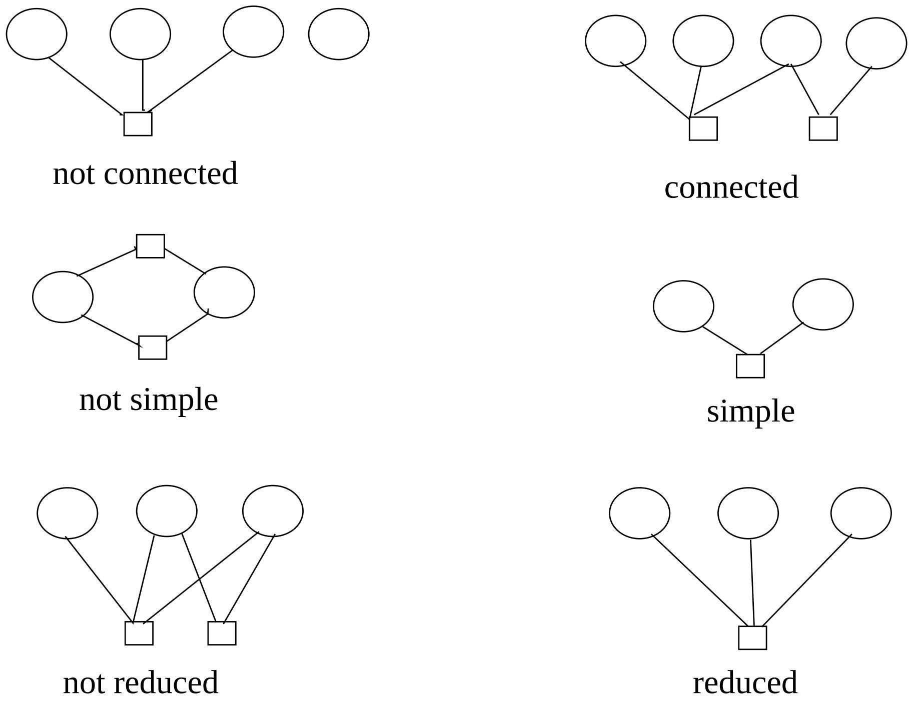

Figure 2: *Simple, connected, and reduced constraint networks.*

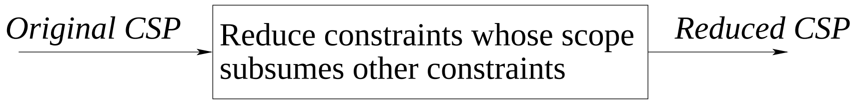

Figure 3: *Computing the reduced CSP.*

of another constraint C_j into C_j until the CSP is reduced. Figure 3 illustrates this reduction. Algebraically, the subsumption operation is obtained by a natural join operation in relational algebra $C_j \leftarrow C_i \bowtie C_j$.

2.2 Induced CSP

First, we define the set of sub-constraints of a given constraint:

Definition 2.1. The *sub-constraints* of a constraint $C_k = (S_k, r_k)$ is the set of the constraints defined over every non-empty subset of S_k, the scope of C_k:

$$\text{Sub-constraints}(C_k) = \{C_i = (S_i, r_i) \mid \forall\, 1 \leq i \leq (2^{|S_k|} - 1)\}, \qquad (4)$$

where S_i is a non-empty subset of S_k and the definition r_i of the constraint

C_i is given by the projection[1] of r_k over S_i:

$$r_i = \pi_{S_i}(r_k). \tag{5}$$

It is obvious that C_k is equivalent to the natural join of the constraints in Sub-constraints(C_k). For the example in Figure 1, the following are the sub-constraints of C_{123}:

- 1 ternary constraint over (V_1, V_2, V_3),
- 3 binary constraints between (V_1, V_2), (V_1, V_3), and (V_2, V_3), and
- 3 unary constraints on V_1, V_2, and V_3.

When computing the number of constraints in a network, we only consider constraints that have distinct scopes. That is, the constraints in the network are all made simple. In the example of Figure 1, the unary constraint induced by C_{34} on V_3 and the one induced by constraint C_{123} are replaced by a unique unary constraint C'_3 as shown in Figure 4.

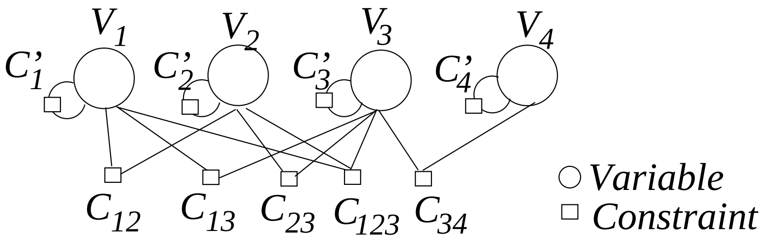

Figure 4: *Constraint network of a simple CSP.*

Given Definition 2.1, it appears that the total number of sub-constraints for a constraint C_k with arity $K = |S_k|$ is the size of the power-set of S_k excluding the non-empty set. This number is given by:

$$C_K^1 + C_K^2 + \cdots + C_K^K = 2^K - 1. \tag{6}$$

In general, a constraint network is not necessarily simple, connected, or reduced. However, for the purpose of computing the constraint probability of the network, we require that the network be simple. That is, we consider that the number of constraints in the network is equal to the number of its *simple* constraints, as defined above. For this purpose, we introduce the definition of an *induced* CSP.

[1]As a relational algebra operator.

Definition 2.2. The *induced problem* of a given CSP $\mathcal{P}$= ($\mathcal{V}$, $\mathcal{D}$, $\mathcal{C}$) is the CSP $\mathcal{P}'$= ($\mathcal{V}$, $\mathcal{D}$, $\mathcal{C}'$) obtained by first computing all the sub-constraints of the constraints in $\mathcal{C}$, then replacing the resulting constraints that have the same scope by their intersection.

This process is illustrated in Figure 5. For example, Figure 4 shows the

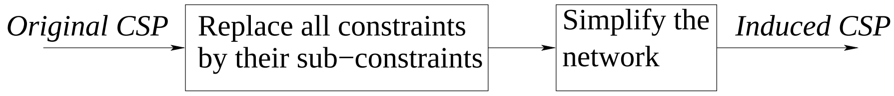

Figure 5: *Computing the induced CSP.*

network of the induced by the CSP shown in Figure 1. Given the above definition, the following holds:

Proposition 2.3. The maximal number of simple constraints in a non-binary network is equal to:

$$C_n^1 + C_n^2 + ... + C_n^n = 2^n - 1 \tag{7}$$

where $n = |\mathcal{V}|$ is the number of variables in the network.

2.3 Constraint probability

We now define the constraint probability of a non-binary CSP:

Definition 2.4. The *constraint probability* of a non-binary CSP $\mathcal{P}$=($\mathcal{V}$, $\mathcal{D}$, $\mathcal{C}$) is the ratio of number of constraints in the induced CSP $\mathcal{P}'$=($\mathcal{V}$, $\mathcal{D}$, $\mathcal{C}'$) over the maximum number of constraints in $\mathcal{P}'$, which is equal to the maximal number of constraint in $\mathcal{P}$, since $\mathcal{P}$ and $\mathcal{P}'$ have the same set of variables:

$$\text{Constraint probability} = p_1 = \frac{|\mathcal{C}'|}{2^n - 1}. \tag{8}$$

According to Definitions 2.2 and 2.4, the range of the constraint probability of a non-binary CSP depends on the value of the highest constraint arity k_{max} in the problem. This range can be specified as follows:

Proposition 2.5. In a CSP $\mathcal{P} = (\mathcal{V}, \mathcal{C})$ with maximum constraint arity $2 \leq k_{max} \leq n$, the range of the constraint probability is in the interval:

$$\left[\frac{2^{k_{max}} - 1}{2^n - 1}, \frac{C_n^1 + C_n^2 + ... + C_n^{k_{max}}}{2^n - 1} \right] \tag{9}$$

where $n = |\mathcal{V}|$. Note that the constraint network is necessarily connected.

Proof: When the maximum constraint arity in a CSP is k_{max}, then the network has at least one constraint (lower bound), which must be a constraint k_{max}-arity, and at most $C_n^{k_{max}}$ constraints with arity k_{max} (upper bound). When the CSP has a unique constraint of arity k_{max} then the number of constraints of the induced CSP is exactly:

$$C^1_{k_{max}} + C^2_{k_{max}} + \cdots + C^{k_{max}}_{k_{max}} = 2^{k_{max}} - 1. \tag{10}$$

Conversely, when the CSP has $C_n^{k_{max}}$ constraints with arity k_{max}, the total number of constraints in the induced CSP is exactly:

$$C^1_n + C^2_n + \cdots + C^{k_{max}}_n. \tag{11}$$

Hence Equation (9) holds. □

Using Proposition 2.5, we can easily compute the range of the constraint probability of a CSP of 10 variables given various values of the maximum constraint arity in the CSP as shown in Table 1:

k_{max}	**Range of p_1**
10	[1, 1]
9	[0.499511, 0.999022]
8	[0.249266, 0.989247]
7	[0.124144, 0.945259]
6	[0.063538, 0.827957]
5	[0.030303, 0.622678]
4	[0.014662, 0.376344]
3	[0.006842, 0.171065]
2	[0.002932, 0.053763]
1	[0, 0.009775]

Table 1: *Range of the constraint probability of a CSP with 10 variables.*

Proposition 2.6. The constraint probability of a CSP and that of its reduced CSP are equivalent.

Proof: Obviously follows from Definition 2.4. □

3 An algorithm to compute probability of a CSP

The computation of the constraint probability of a given CSP requires us to compute the total number of constraints in the induced network, see Equation 8. In this section, we propose an algorithm for computing this number directly from the original network without building the induced network.

Input: $\mathcal{P} = (\mathcal{V}, \mathcal{D}, \mathcal{C})$, a CSP
Output: p_1, the constraint probability of $\mathcal{P}$

$\mathcal{S} \leftarrow \{\text{scope}(C_i), \forall C_i \in (\mathcal{C})\}$
if $|\mathcal{S}| = 1$ where $\mathcal{S} = \{S_1\}$ **then**
  #-of-induced-constraints $\leftarrow 2^{|S_1|} - 1$
else
  if $|\mathcal{S}| = 2$ where $\mathcal{S} = \{S_1, S_2\}$ **then**
    #-of-induced-constraints $\leftarrow (2^{|S_1|} - 1) + (2^{|S_2|} - 1) - (2^{|S_1 \cap S_2|} - 1)$
  else
    $\mathcal{S}' \leftarrow \emptyset$
    for each constraint scope $S_i \in \mathcal{S}$ **do**
      for each set S_j in Power-set$(S_i)\backslash\emptyset$ **do**
        $\mathcal{S}' \leftarrow \mathcal{S}' \cup \{S_j\}$
      end for
    end for
    #-of-induced-constraints $\leftarrow |\mathcal{S}'|$
  end if
end if
$p_1 \leftarrow \frac{\text{\#-of-induced-constraints}}{2^{|\mathcal{V}|}-1}$

Algorithm 1: Computing the constraint probability of a CSP

The worst-case time complexity of this algorithm is $O(|\mathcal{C}| \cdot 2^{k_{max}})$, where $|\mathcal{C}|$ is the total number of constraints, and k_{max} is the maximum constraint arity in the CSP. When the constraint network is not connected and has isolated sub-networks, this algorithm can be improved by first splitting the constraint network into its non-connected components, calculating the number of constraints in each isolated part separately, and then making the summation of the number of induced constraints in each isolated parts.

4 Alternative definitions of constraint probability

During the lab meeting, an animated discussion occurred about the appropriateness of the definition of constraint probability as proposed in Equation (2.4). We summarize below these various proposals. We have not yet addressed the evaluation of these various definitions, which remains to be carried out. Note that all these definitions, in contrast with the one introduced in Equation (8), do not account for constraints with arity larger than k_{max}. Whatever definition one adopts, it is important to keep in mind that many real-world applications typically exhibit a large number of low-arity constraints and small number of a high-arity constraints.

1. *Constraint ratio* (CR): The constraint ratio is a vector CR of size n, where n is the number of variables in the CSP. The indices of this vector are in $[1, n]$. The entry of index k is the constraint ratio of arity k, denoted CR_k. CR_k is defined as the ratio of the number of constraints of arity k in the CSP to the total number of possible constraints of arity k:
$$\mathrm{CR}_k = \frac{\text{number of constraints of arity } k}{C_n^k} \tag{12}$$
The advantage of this definition is that it is consistent with the known definition of constraint probability of binary CSPs. The inconvenience of this definition is that it does not provide a global value for the constraint probability.

2. Mansour proposed to define constraint probability as follows:
$$k_{max}\text{–CR} = \frac{\text{number of constraints in the reduced CSP}}{C_n^{k_{max}}} \tag{13}$$
where k_{max} is the largest constraint arity in the CSP. This method assumes every constraint whose scope is less than k_{max} can be regarded as a constraint with scope k_{max} by extending this constraint like this: constraint C_{12} can be extended to C_{1234}, where $C_{1234} = C_{12} \bowtie D_3 \bowtie D_4$.

   The advantage of this definition is that it is easy to compute the constraint probability. Its disadvantage is that it does not reflect the distribution in the network of constraints with arity strictly lower than k_{max}.

3. Another definition is:
$$\mathrm{CR}_{k_{max}} = \frac{\text{number of constraints in the induced CSP}}{C_n^1 + C_n^2 + ... + C_n^{k_{max}}} \tag{14}$$
where k_{max} is the largest existing constraint arity in the constraint network. In contrast to Equation (8), this equation excludes from its denominator constraints with arity greater than k_{max}.

The study and evaluation of the individual and relative merits of these various definitions remain to be carried out.

5 Constraint Tightness

Constraint tightness is another important parameter for the empirical evaluation of algorithms used to process and solve constraint networks. As for the binary case, we consider that the tightness of a non-binary constraint is equal to the ratio of the number of forbidden tuples to the number of all possible tuples (which is the size of the Cartesian product of the domains of the variables in the scope of the constraint). We may have in the future to define the tightness of the constraints of a reduced CSP in terms of the tightness values of the original CSP. Note that the concept of constraint tightness is defined for CSPs with finite domains. Generalizing this notion over CSPs with continuous domains may require non-obvious hyper-volume calculations.

6 Conclusion

In this report, we propose a definition for the constraint probability of a non-binary CSP. This definition takes into consideration the sub-constraints of each constraint in the network. Further, we propose an algorithm to compute this constraint probability. We also list a number of alternatives definitions of constraint probability. The significance of these definitions remains to be studied.

References

Montanari, Ugo 1974. Networks of Constraints: Fundamental Properties and Application to Picture Processing. *Information Sciences* 7:95–132.