Reward-based region optimal quality guarantees

Meritxell Vinyals[1], Eric Shieh[2], Jesus Cerquides[1], Juan Antonio Rodriguez-Aguilar[1], Zhengyu Yin[2], Milind Tambe[2], and Emma Bowring[3]

[1] Artificial Intelligence Research Institute (IIIA),
Campus UAB, Bellaterra, Spain
{meritxell, cerquide,jar}@iiia.csic.es
[2] University of Southern California, Los Angeles, CA 90089
{eshieh, zhengyuy,tambe}@usc.edu
[3] University of the Pacific, Stockton, CA 95211
ebowring@pacific.edu

Abstract. Distributed constraint optimization (DCOP) is a promising approach to coordination, scheduling and task allocation in multi agent networks. DCOP is NP- hard [6], so an important line of work focuses on developing fast incomplete solution algorithms that can provide guarantees on the quality of their local optimal solutions.

Region optimality [11] is a promising approach along this line: it provides quality guarantees for region optimal solutions, namely solutions that are optimal in a specific region of the DCOP. Region optimality generalises k- and t-optimality [7, 4] by allowing to explore the space of criteria that define regions to look for solutions with better quality guarantees.

Unfortunately, previous work in region-optimal quality guarantees fail to exploit any a-priori knowledge of the reward structure of the problem. This paper addresses this shortcoming by defining reward-dependent region optimal quality guarantees that exploit two different levels of knowledge about rewards, namely: (i) a ratio between the least minimum reward to the maximum reward among relations; and (ii) the minimum and maximum rewards per relation.

1 Introduction

Distributed Constraint Optimization (DCOP) is a popular framework for cooperative multi-agent decision making. It has been applied to real-world domains such as sensor networks [12], traffic control [3], or meeting scheduling [8]. In real-world domains, and particularly in large-scale applications, DCOP techniques have to cope with limitations on resources and time available for reasoning. Because DCOP is NP-Hard [6], complete DCOP algorithms (e.g. Adopt [6], OptAPO [5], DPOP [8]) that guarantee global optimality are unaffordable for these domains due to their exponential costs. In contrast to complete algorithms, incomplete algorithms [12, 2, 10, 7, 4] provide better scalability.

Unfortunately, an important limitation for the application of incomplete algorithms is that they usually fail to provide quality guarantees on their solutions.

The importance of quality guarantees is twofold. First, they help guarantee that agents do not converge to a solution whose quality is below a certain fraction of the optimal solution (which can have catastrophic effects in certain domains). Secondly, quality guarantees can aid in algorithm selection and network structure selection in situations where the algorithmic cost of coordination must be weighed up against solution quality (trade-off cost versus quality).

To the best of our knowledge, region optimal algorithms [11] are the only incomplete DCOP algorithms that can provide guarantees on the worst-case solution quality of their solutions at design time and exploit the available knowledge, if any, about the DCOP(s) to solve regarding their graph structure. The region optimal framework, that generalises the k- and t- optimal frameworks proposed in [7, 4], defines quality guarantees for region optimal solutions, namely solutions that are optimal in specific region of the DCOP. Thus, region optimality allows to explore the space of local optimality criteria (beyond size and distance) looking for those that lead to better solution qualities. To assess region optimal quality guarantees, in our previous work [11] we propose two methods with different computational costs: (1) a first one, based on solving an LP, that guarantees tightness; and (2) a second one that requires linar time but does not ensure tightness.

Unfortunately, previous work in region-optimal bounds fail to exploit some a-priori knowledge of the reward structure of the DCOP problem, if available, and only the knowledge of graph structure is exploited so far. As argued in [1] for the particular case of k-optimality, this limits the applicability of these approaches to many domains for which some information about the range of rewards is available. For example, in sensor networks [12], we may know the maximum reward of observing a phenomenon and the minimum reward when we have no observations.

This is the shortcoming we address in this paper and at this aim we extend the region optimal bounds to be able to exploit two different kinds of knowledge about the reward structure. Concretely, we show how to tight region optimal quality guarantees by assuming: (1) a ratio between the least minimum reward to the maximum reward among relations, along the lines of [1] (e.g. the ratio between the maximum reward of observing a phenomenon and the minimum reward when no observation is known); and (2) that the minimum and maximum rewards per relation are known (e.g. the maximum reward of observing a phenomenon and the minimum reward when no observation are known).

This paper is organised as follows. Section 2 provides some background on DCOPs and on the region optimal framework. Section 3 extends region optimal quality guarantees to exploit some a-priori knowledge about the reward structure. Section shows the average improvement of the proposed reward-based guarantees with respect to the guarantees formulated in [11]. Finally, section 5 draws conclusions.

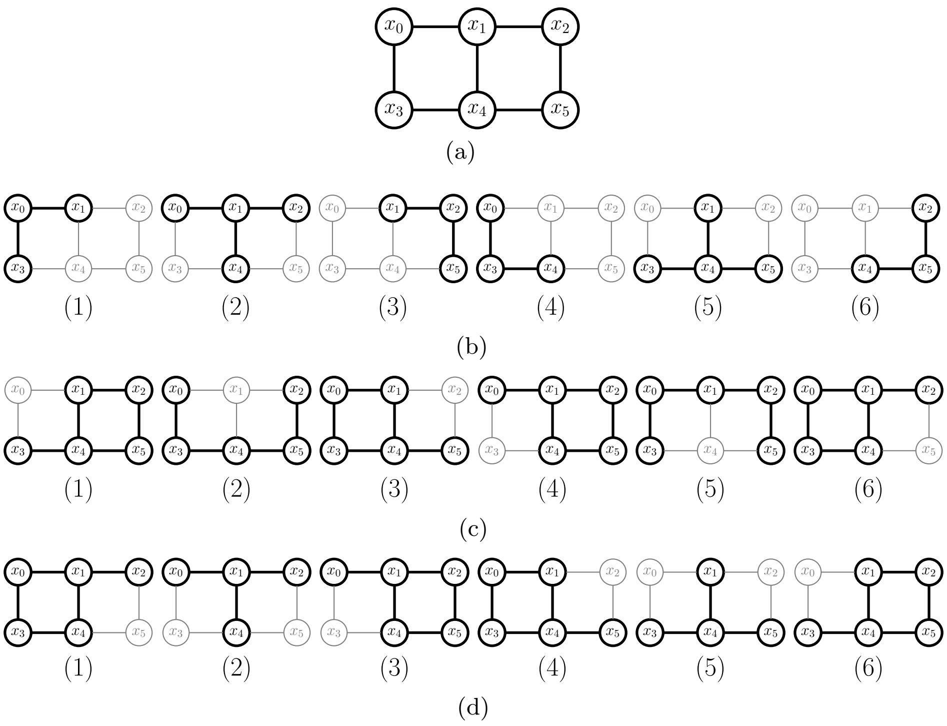


Fig. 1. Example of (a) a DCOP graph, (b) its 1-distance region, (c) its 5-size region and (d) its 5-size-distance-bounded region

2 Background

2.1 DCOP

A Distributed Constraint Optimization Problem (DCOP) consists of a set of variables, each assigned to an agent which must assign a discrete value to the variable: these values correspond to individual actions that can be taken by agents. Constraints exist between subsets of these variables that determine rewards to the agent team based on the combinations of values chosen by their respective agents, namely relations. Let $\mathcal{X} = \{x_1, \ldots, x_n\}$ be a set of variables over domains $\mathcal{D}_1, \ldots, \mathcal{D}_n$. A relation on a set of variables $V \subseteq \mathcal{X}$ is expressed as a reward function $S_V : \mathcal{D}_V \rightarrow \mathbb{R}^+$, where $\mathcal{D}_V$ is the joint domain over the variables in V. This function represents the reward generated by the relation over the variables in V when the variables take on an assignment in the joint domain $\mathcal{D}_V$. Whenever there is no need to identify the domain, we simply use S to note relations.

In a DCOP each agent knows all the relations that involve its variable(s). In this work we assume that each agent is assigned a single variable, so we will use the terms “agent” and “variable” interchangeably.

Formally, a DCOP is a tuple $\langle \mathcal{X}, \mathcal{D}, \mathcal{R} \rangle$, where: $\mathcal{X}$ is a set of variables (each one assigned to a different agent); $\mathcal{D}$ is the joint domain space for all variables; and $\mathcal{R}$ is a set of reward relations. The solution quality for an assignment $d \in \mathcal{D}$ to the variables in $\mathcal{X}$ is the sum of the rewards for the assignment over all the relations in the DCOP, namely:

$$R(d) = \sum_{S_V \in \mathcal{R}} S_V(d_V) \tag{1}$$

where $d_V \in \mathcal{D}_V$ contains the values assigned by d to the variables in V. With slight abuse of notation we allow to write equation 1 as $R(d) = \sum_{S \in \mathcal{R}} S(d)$.

Solving a DCOP amounts to choosing values for the variables in $\mathcal{X}$ such that the solution quality is maximized. A binary DCOP (each relation involves a maximum of two variables) is typically represented by its constraint graph, whose vertices stand for variables and whose edges link variables that have some direct dependency (appear together in the domain of some relation). An example of a constraint graph is depicted in figure 1(a).

2.2 Region optimality

In [11] we introduce region optimality, a framework that generalise the k- and t-optimal frameworks [7, 4] by providing reward-independent quality guarantess for optima in regions characterised by any arbitrary criterion. Region optimality allows to explore the space of criteria (beyond size and distance) looking for those that lead to better solution qualities. Next we give a brief overview of the region optimal framework by defining the concepts of neighbourhood, region and region optimal solution.

Formally, a neighbourhood, A, is a subset of variables of $\mathcal{X}$. For instance, figure 1(b)(1) depicts a neighbourhood for the DCOP in figure 1(a) where boldfaced nodes in the constraint graph stand for variables included in the neighbourhood, namely $\{x_0, x_1, x_3\}$. Given two assignments x and y, we define $D(x, y)$ as the set containing the variables whose values in x and y differ. Then given a neighbourhood A, we say that x is a *neighbour of y in A* iff x differs from y only in variables that are contained in A, thus $D(x, y) \subseteq A$.

Given some assignment x, we say that it is optimal in a neighbourhood A if its reward cannot be improved by changing the values of some of the variables in the neighbourhood. That is, for every assignment y such that x is a *neighbour of y in A*, we have that $R(x) \geq R(y)$. Thus, an assignment x is optimal in the neighbourhood of figure 1(b)(1) if any other assignment that maintains the values of x_2, x_4 and x_5 as in x receives at most the same reward as x.

Given two neighbourhoods $A, B \subseteq \mathcal{X}$ we say that B completely covers A if $A \subseteq B$. We say that B does not cover A at all if $A \cap B = \emptyset$. Otherwise, we say that B covers A partially. For each neighbourhood A we can classify each relation S_V

in a DCOP into one of three disjoint groups, depending on whether C^α covers V completely ($T(A)$), partially ($P(A)$), or not at all ($N(A)$). For example, given the neighbourhood $\{x_0, x_1, x_3\}$ in figure 1(b)(1) we can classify the relations of the DCOP in figure 1(a) as : $T(\{x_0, x_1, x_3\}) = \{S_{\{x_0,x_3\}}, S_{\{x_0,x_1\}}\}$, $P(\{x_0, x_1, x_3\}) = \{S_{\{x_1,x_2\}}, S_{\{x_3,x_4\}}, S_{\{x_1,x_4\}}\}$, $N(\{x_0, x_1, x_3\}) = \{S_{\{x_2,x_5\}}, S_{\{x_4,x_5\}}\}$.

A region $\mathcal{C}$ is a multi-set[1] of subsets of $\mathcal{X}$, namely a multi-set of neighbourhoods of $\mathcal{X}$. For instance, figure 1(b) show a region composed of six neighbourhoods (b)(1)-(b)(6). Given a region $\mathcal{C}$, we say that x is *inside region $\mathcal{C}$ of y* iff x differs from y only in variables that are contained in one of the neighbourhoods in $\mathcal{C}$, that is, if there is a neighborhood $C^\alpha \in \mathcal{C}$ such that x is neighbour of y in C^α. Then, we can claim optimality for x in a region $\mathcal{C}$ (noted as $x^{\mathcal{C}}$) whenever it is optimal in each neighbourhood $C^\alpha \in \mathcal{C}$, that is if it cannot be improved by any other assignment inside region $\mathcal{C}$. For instance, an assignment x will be optimal in the region depicted in figure 1(a) if it is optimal in each of its six neighbourhoods.

Finally, for each relation $S_V \in \mathcal{R}$ we define $cc(S_V, \mathcal{C}) = |\{C^\alpha \in \mathcal{C} \text{ s.t } V \subseteq C^\alpha\}|$, that is, the number of neighbourhoods in $\mathcal{C}$ that cover the domain of S_V completely. We also define $nc(S_V, \mathcal{C}) = |\{C^\alpha \in \mathcal{C} \text{ s.t } V \cap C^\alpha = \emptyset\}|$, that is, the number of neighbourhoods in $\mathcal{C}$ that do not cover the domain of S_V at all. For instance, in the region of figure 1(b) there are two neighbourhoods that totally cover the relation $S_{\{x_0,x_1\}}$, namely neighbourhoods (1) and (2). Thus, $cc(S_{\{x_0,x_1\}}, \mathcal{C}) = 2$. Moreover, there is only one neighbourhood that do not cover $S_{\{x_0,x_1\}}$ at all, namely neighbourhood (6). Thus, $nc(S_{\{x_0,x_1\}}, \mathcal{C}) = 1$.

In [11] we show how region optimality generalises $k-$ and $t-$optimality by observing that: (i) both criteria are based on the definition of a region over the constraint graph; and (ii) given any assignment, checking for either k-size or t-distance optimality amounts to checking for optimality in that region. For example, figure 1(b) shows the neighbourhoods corresponding to the 1-distance region of the DCOP in figure 1(a), where each neighbourhood corresponds to one variable and its direct neighbours (e.g. neighbourhood (1) includes variable x_0 and its direct neighbours) whereas figure 1(c) shows the neighbourhoods corresponding to the 5-size region of the DCOP in figure 1(a), where each neighbourhood stands for a set of five connected variables.

Furthermore, region optimality allows to explore the space of arbitrary criteria to generate regions that otherwise will never be explored by size or distance criteria. For example, figure 1(d) shows a region created by the size-bounded-distance criteria proposed in [11]. Observe that the 5-size-bounded-distance region will never be created by either size or distance criteria because: (1) regarding size, it includes regions of size 4 and size 5; and (2) regarding distance, the region includes neighbourhoods different from the 1-distance region (shown in figure 1(b)) and from the 2-distance region which includes neighbourhoods that contain all the variables in the DCOP.

[1] A multi-set is a generalisation of a set that can hold multiple instances of the very same element.

In the next section we describe the methods to calculate bounds for a $\mathcal{C}$-optimal assignment, namely an assignment that is optimal in an arbitrary region $\mathcal{C}$.

2.3 Region optimal reward-independent quality guarantees

In this section we review the methods proposed in [11] to calculate reward-independent quality guarantees for any region optima. In [11] we propose two methods to calculate region optimal reward-independent quality guarantees each one with a different computational cost: (1) a first one, based on solving an LP, that guarantees tightness; and (2) a second one, that requires linear time but does not ensure tightness. Assuming no knowledge of reward structure (except from they are non-negatie) but exploiting the knowledge of the graph structure when available, these methods provide worst case quality of a $\mathcal{C}$-optimal solution as a fraction of the global optimal, where $\mathcal{C}$ is an arbitrary region.

We say that we have a bound δ when we can state that the quality of any $\mathcal{C}$-optimal assignment $x^{\mathcal{C}}$ is larger than δ times the quality of the optimal x^*. Hence, having a bound δ means that for every $x^{\mathcal{C}}$ we have that $\frac{R(x^{\mathcal{C}})}{R(x^*)} \geq \delta$.

Tight region optima quality guarantees. For a given set of relations $\mathcal{R}$, let $x^{\mathcal{C}}_{-}$ be the $\mathcal{C}$-optimal assignment with smallest reward, then $\frac{R(x^{\mathcal{C}}_{-})}{R(x^*)}$ provides a tight bound on the quality of any $\mathcal{C}$-optimal for the specific rewards $\mathcal{R}$.

In we show how to calculate a tight bound on a $\mathcal{C}$-optimal assignment independently from the specific DCOP rewards by directly search the space of reward values to find the set of rewards $\mathcal{R}^*$ that minimizes $\frac{R^*(x^{\mathcal{C}}_{-})}{R^*(x^*)}$.

The assessment of this bound involves to solve the following program:

Find $\mathcal{R}$, $x^{\mathcal{C}}$ and x^* that
minimize $\frac{R(x^{\mathcal{C}})}{R(x^*)}$
subject to $x^{\mathcal{C}}$ being a $\mathcal{C}$-optimal for $\mathcal{R}$

Given the definition of region optimality, the condition of being a $\mathcal{C}$-optimal for $\mathcal{R}$ can be expressed as: for each x inside region $\mathcal{C}$ of $x^{\mathcal{C}}$ we have that $R(x^C) \geq R(x)$. However, instead of considering all the assignments for which x^C is guaranteed to be optimal, we consider only the subset of assignments such that the set of variables that deviate with respect to x^C take the same value than in the optimal assignment. If we restrict to this subset of assignments, then each neighbourhood covers a $2^{|C^\alpha|}$ assignments, one for each subset of variables in the neighbourhood. Let 2^{C^α} stand for the set of all subsets of the neighbourhood C^α. Then for each $A^k \in 2^{C^\alpha}$ we can define an assignment x^{α_k} such that for every variable x_i in a relation completely covered by A^k we have that $x_i^{\alpha_k} = x_i^*$, and for every variable x_i that is not covered at all by A^k we have that $x_i^{\alpha_k} = x_i^{C}$. Then, we can write the value of x^{α_k} as :

$$R(x^{\alpha_k}) = \sum_{S \in T(A^k)} S(x^*) + \sum_{S \in P(A^k)} S(x^{\alpha_k}) + \sum_{S \in N(A^k)} S(x^{\alpha_k}) \tag{2}$$

Now, the definition of $\mathcal{C}$-optimal can be expressed as:

$$R(x^C) \geq \sum_{S \in T(A^k)} S(x^*) + \sum_{S \in P(A^k)} S(x^{\alpha_k}) + \sum_{S \in N(A^k)} S(x^C) \quad \forall A^k \in \{2^{C^{\alpha_k}} | C^{\alpha_k} \in \mathcal{C}\} \quad (3)$$

that, by setting partially covered relations to the minimum possible reward (0 assuming non-negative rewards), results in:

$$R(x^C) \geq \sum_{S \in T(A^k)} S(x^*) + \sum_{S \in N(A^k)} S(x^C) \quad \forall A^k \in \{2^{C^{\alpha_k}} | C^{\alpha_k} \in \mathcal{C}\} \qquad (4)$$

where $T(A^k)$ is the set of completely covered relations, $P(A^k)$ the set of partially covered relations and $N(A^k)$ the set of relations not covered at all.

This program can be simplified into a linear program (LP), detailed in [9], whose solution defines a tight bound on the quality of a $\mathcal{C}$-optimal solution for the graph structure represented by $\mathcal{R}$. Let M be the number of variables of the largest neighbourhood in $\mathcal{C}$. The LP has $2 \cdot |\mathcal{R}|$ variables and $\mathcal{O}(2^M \cdot |\mathcal{C}|)$ constraints, and hence it is solvable in time polynomial in $|\mathcal{R}|$ and in $2^M \cdot |\mathcal{C}|$.

Faster quality guarantees for region optima. Because the computational complexity of the LP method can be high as the number of relations $|\mathcal{R}|$, the number of neighbourhoods $|\mathcal{C}|$ or its size M grows in we propose a faster alternative method to compute region optimal bounds. This faster method can compute a bound, by means of proposition 1, in time $\mathcal{O}(|\mathcal{R}||\mathcal{C}|)$ but, as a counterpart, we lose the tightness of the bound.

Proposition 1. *Let $\langle \mathcal{X}, \mathcal{D}, \mathcal{R} \rangle$ be a DCOP with non-negative rewards and $\mathcal{C}$ a region. If $x^{\mathcal{C}}$ is a $\mathcal{C}$-optimal assignment then*

$$R(x^{\mathcal{C}}) \geq \frac{cc_*}{|\mathcal{C}| - nc_*} R(x^*) \qquad (5)$$

where $cc_* = \min_{S \in \mathcal{R}} cc(S, \mathcal{C})$, $nc_* = \min_{S \in \mathcal{R}} nc(S, \mathcal{C})$, and x^* is the optimal assignment.

Proposition 1 directly provides a simple algorithm to compute a bound. Given a region $\mathcal{C}$ and a graph structure, we can directly assess cc_* and nc_* by computing $cc(S, \mathcal{C})$ and $nc(S, \mathcal{C})$ for each relation $S \in \mathcal{R}$ and taking the minimum. This will take time $\mathcal{O}(|\mathcal{R}||\mathcal{C}|)$, that is linear in the number of relations of the DCOP and linear in the number of neighbourhoods in the region.

Despite its complexity improvements, the bound assessed by proposition 1 is not guaranteed to be tight and can return worse bounds than the ones provided by the LP-based mechanism.

Both the LP and proposition 1 assess bounds that depend on the graph structure but are independent of the specific reward values. We can always use them to assess bounds independently of the graph structure by assessing the bound for the complete graph, since any other structure is a particular case of the complete graph with some rewards set to zero.

3 Reward-based region optimal bounds

In the previous section we review two methods that provide bounds on any $\mathcal{C}$-optimal, characterized by an arbitrary $\mathcal{C}$ criterion, that are independent of the specific reward structure. However, as shown in [1] for the specific criterion of group size, if some knowledge of the reward structure of the problem is available then it can be exploited to assess more accurate bounds. Here we show how to incorporate reward structure knowledge in the region optimality framework and formulate two different improvements based on:

- the knowledge that the minimum reward is guaranteed to be a certain factor of the maximum reward on any relation (the so-called *minimum fraction reward*); and
- the knowledge of the minimum and maximum rewards per relation.

We also compare these two improvements showing that the bounds that use minimum and maximum rewards per relation are tighter than the ones using the minimum fraction reward. Hence, more knowledge about the reward structure helps obtain tighter bounds.

3.1 Based on the minimum fraction reward

In this section we show how to refine $\mathcal{C}$-optimal bounds when we know that the minimum reward is a certain factor β $(0 < \beta \leq 1)$ of the maximum reward on any relation. Thus, this refinement is a generalitzation of the improvements in tightness for size-optimal bounds proposed in [1].

In order to obtain a tighter bound, we will employ the partially covered relations by a neighborhood.

First, we show how assuming a minimum fraction reward of β we can improve the LP-based $\mathcal{C}$-optimal bounds, overviewed in section 2.3.

In this case, instead of setting the values of all relations $\sum_{S\in P(C^\alpha)} S(x^{\alpha_k})$ to 0, as in equation 4, we can exploit the knowledge that $S(x^{\alpha_k}) \geq \beta{\cdot}S(x^*)$ $\forall S \in \mathcal{R}$. Then, $\forall A^k \in \{2^{C^{\alpha_k}} | C^{\alpha_k} \in \mathcal{C}\}$, the restriction for assignment x^{α_k} is rewritted as:

$$\sum_{S\in\mathcal{R}} S(x^C) \geq \sum_{S\in T(C^\alpha)} S(x^*) + \sum_{S\in P(C^\alpha)} \beta \cdot S(x^*) + \sum_{S\in N(C^\alpha)} S(x^C) \tag{6}$$

Notice that it is the only change we need to do to incorporate the knowledge of the minimum fraction reward and that the program can be simplified to an LP, following analogous operations as the ones detailed in [9], with the same number of variables than in the reward independent case. The solution of such LP defines a tight bound on the quality of a $\mathcal{C}$-optimal solution for the graph structure represented by $\mathcal{R}$ and rewards with a minimum fraction reward of β.

Second, we show how assuming a minimum fraction reward of β we can improve the faster $\mathcal{C}$-optimal bounds, overviewed in section 2.3. Before that we define, for each relation $S \in \mathcal{R}$, $pc(S, \mathcal{C}) = |\mathcal{C}| - nc(S, \mathcal{C}) - cc(S, \mathcal{C})$ as the number of neighbourhoods in region $\mathcal{C}$ that partially cover relation S. Then, the following proposition shows how to exploit the minimum fraction reward along with the partially covered relations to obtain a bound tighther than the one in equation 5.

Proposition 2. *Let $\langle \mathcal{X}, \mathcal{D}, \mathcal{R} \rangle$ be a DCOP, $\mathcal{C}$ a region and β the minimum fraction reward. If $x^{\mathcal{C}}$ is a $\mathcal{C}$-optimal assignment then:*

$$\mathcal{R}(x^{\mathcal{C}}) \geq \left(\frac{cc_*}{|\mathcal{C}| - nc_*} + \beta \frac{pc_*}{|\mathcal{C}| - nc_*} \right) \mathcal{R}(x^*) \tag{7}$$

where $cc_* = \min_{S \in \mathcal{R}} cc(S, \mathcal{C})$, $nc_* = \min_{S \in \mathcal{R}} nc(S, \mathcal{C})$, $pc_* = \min_{S \in \mathcal{R}} pc(S, \mathcal{C})$, and x^* is the optimal assignment.

Proof. The proof is analagous to the one for the general bound of equation 5 formulated in [11], but without disregarding partially covered relations. For every $C^\alpha \in \mathcal{C}$, consider an assignment x^α such that: $x_i^\alpha = x_i^{\mathcal{C}}$ if $x_i \notin C^\alpha$, and $x_i^\alpha = x_i^*$ if $x_i \in C^\alpha$. Since $x^{\mathcal{C}}$ is $\mathcal{C}$-optimal, for all $C^\alpha \in \mathcal{C}$, $\mathcal{R}(x^{\mathcal{C}}) \geq \mathcal{R}(x^\alpha)$ holds, and hence:

$$\mathcal{R}(x^{\mathcal{C}}) \geq \frac{\sum_{C^\alpha \in \mathcal{C}} \left(\sum_{S \in T(C^\alpha)} S(x^*) + \sum_{S \in N(C^\alpha)} S(x^{\mathcal{C}}) + \sum_{S \in P(C^\alpha)} S(x^\alpha) \right)}{|\mathcal{C}|}. \tag{8}$$

Using β we can express the third term in equation 8 in terms of $\mathcal{R}(x^*)$ considering that the knowledge for any relation $S \in \mathcal{R}$ satisfies $S(x^\alpha) \geq \beta \cdot S(x^*)$. Therefore, the following inequalities hold:

$$\sum_{C^\alpha \in \mathcal{C}} \sum_{S \in P(C^\alpha)} S(x^\alpha) \geq \sum_{S \in \mathcal{R}} pc(S, \mathcal{C}) \cdot \beta \cdot S(x^*) \geq pc_* \cdot \beta \cdot \mathcal{R}(x^*) \tag{9}$$

From the proof of the general bound of equation 5 in [11] we know that the first and the second sets of relations of equation 8 can be also expressed in terms of $\mathcal{R}(x^{\mathcal{C}})$ and $\mathcal{R}(x^*)$. Therefore, after substituting these results in equation 8 and rearranging terms, we obtain equation 7.

3.2 Based on the minimum/maximum relation rewards

In this section we show how to tighten the reward-independent $\mathcal{C}$-optimal bound δ for a DCOP problem when we know the values of the minimum and maximum rewards for each relation $S \in \mathcal{R}$, namely $l_S = \min_{d_V \in \mathcal{D}_V} S(d_V)$ and $u_S = \max_{d_V \in \mathcal{D}_V} S(d_V)$ respectively. Because we do not make any assumption over the technique used to calculate δ rather tham it is reward-independent bound, this improvement applies to both: the reward-independent LP-bounds introduced in section and the faster reward-independent bounds of equation introduced in section.

Proposition 3. *Let $\langle \mathcal{X}, \mathcal{D}, \mathcal{R} \rangle$ be a DCOP and $\mathcal{C}$ a region. If $x^{\mathcal{C}}$ is a $\mathcal{C}$-optimal assignment then:*

$$\mathcal{R}(x^{\mathcal{C}}) \geq \frac{1}{U} \left((U - L) \cdot \delta + L \right) \cdot \mathcal{R}(x^*) \tag{10}$$

where $U = \sum_{S \in \mathcal{R}} u_S$, $L = \sum_{S \in \mathcal{R}} l_S$.

Proof. Let $\hat{\mathcal{R}}$ be a distribution defined as $\hat{\mathcal{R}}(x) = \mathcal{R}(x) - L = \sum_{S \in \mathcal{R}} (S(x) - l_S)$. Notice that the rewards of $\hat{\mathcal{R}}$ are non-negative because after substracting the minimum of each non-negative relation of $\mathcal{R}$ we obtain new relations in which the minimum value is 0. Moreover, because the value of any assignment in $\hat{\mathcal{R}}$ is equal to the value in $\mathcal{R}$ plus a constant, any $\mathcal{C}$-optimal $x^{\mathcal{C}}$ in $\mathcal{R}$ is also $\mathcal{C}$-optimal in $\hat{\mathcal{R}}$. Thus, by definition of $\mathcal{C}$-optimal bounds the following inequality holds:

$$\hat{\mathcal{R}}(x^{\mathcal{C}}) \geq \delta \cdot \hat{\mathcal{R}}(x^*) \tag{11}$$

Then by expressing $\hat{\mathcal{R}}$ in terms of $\mathcal{R}$ and isolating $\mathcal{R}(x^{\mathcal{C}})$ we obtain:

$$\mathcal{R}(x^{\mathcal{C}}) \geq \delta \cdot \mathcal{R}(x^*) + (1 - \delta) \cdot L \tag{12}$$

Now multiplying and dividing the right equation side by $\mathcal{R}(x^*)$:

$$\mathcal{R}(x^{\mathcal{C}}) \geq \left(\frac{\delta \cdot \mathcal{R}(x^*) + (1 - \delta) \cdot L}{\mathcal{R}(x^*)} \right) \cdot \mathcal{R}(x^*) \tag{13}$$

Since the bound provided by equation 13 above increases as the value of the optimum, $\mathcal{R}(x^*)$, decreases, we can get rid of $\mathcal{R}(x^*)$, which is in general unknown, by replacing it with an upper bound. By definition, U is an upper bound of $\mathcal{R}(x^*)$. Hence, we can substitute U for $\mathcal{R}(x^*)$ to obtain equation 10.

Equation 10 places the reward-independent bound in equation 5 within the $[L, U]$ scale, and subsequently assesses the fraction of upper bound U that it represents.

3.3 Comparing reward-dependent bounds

By analysing equations 7 and 10 we observe that reward-dependent bounds are guaranteed to be greater or, in the worst-case equal, than the general bound of equation 5. However, the relation between these two reward-dependent bounds themselves is not so clear. Next, we show that if the specific minimum/maximum values of each relation are known one can assess better bounds, by means of equation 10 when $\delta = \frac{cc_*}{|\mathcal{C}| - nc_*}$, than by only using the minimum fraction bound, by means of equation 7.

Proposition 4. *Let $\langle \mathcal{X}, \mathcal{D}, \mathcal{R} \rangle$ be a DCOP, $\mathcal{C}$ a region, β the minimum fraction reward, where $l_{S_V} = \min_{d_V \in \mathcal{D}_V} S_V(d_V)$, $u_{S_V} = \max_{d_V \in \mathcal{D}_V} S(d_V)$:*

$$\frac{cc_*}{|\mathcal{C}| - nc_*} + \beta \frac{pc_*}{|\mathcal{C}| - nc_*} \leq \frac{U - L}{U} \frac{cc_*}{|\mathcal{C}| - nc_*} + \frac{L}{U} \tag{14}$$

Proof. After rearranging terms and simplifying, we obtain that equation 14 is equivalent to:

$$pc_* \cdot \beta \leq (|\mathcal{C}| - nc_* - cc_*) \cdot \frac{L}{U} \tag{15}$$

First, from the definition of partial covering, $pc(S,C) = |\mathcal{C}| - nc(S,C) - cc(S,C)$, we observe that $pc(S,C)$ increases as $nc(S,C)$ and $cc(S,C)$ decrease. Since nc_*, cc_* are the minimum values that functions nc, cc can take on respectively, then $pc_* \leq |\mathcal{C}| - nc_* - cc_*$ holds. Therefore, proving that $\beta \leq \frac{L}{U}$, namely that $min_{S\in\mathcal{R}} \frac{l_S}{u_S} \leq \frac{\sum_{S\in\mathcal{R}} l_S}{\sum_{S\in\mathcal{R}} u_S}$, is enough to prove that equation 15 holds. We build the proof by induction of the number of relations $n = |\mathcal{X}|$. Consider without loss of generality a problem with n relations such that $\frac{l_{r_1}}{u_{r_1}} \leq \ldots \leq \frac{l_{r_{n-1}}}{u_{r_{n-1}}} \leq \frac{l_{r_n}}{u_{r_n}}$ holds. If $n = 2$, then $min(\frac{l_{r_1}}{u_{r_1}}, \frac{l_{r_2}}{u_{r_2}}) \leq \frac{l_{r_1}+l_{r_2}}{u_{r_1}+u_{r_2}}$ simplifies to $\frac{l_{r_1}}{u_{r_1}} \leq \frac{l_{r_2}}{u_{r_2}}$, which by problem definition is true. When $n > 2$ we must prove that $\frac{l_{r_1}}{u_{r_1}} \leq \frac{l_{r_1}+\sum_{j\geq 2} l_{r_j}}{u_{r_1}+\sum_{j\geq 2} u_{r_j}}$ holds, or equivalently that $\frac{l_{r_1}}{u_{r_1}} \leq \frac{\sum_{j\geq 2} l_{r_j}}{\sum_{j\geq 2} u_{r_j}}$. By recursively applying the expression for $n = 2$ to $\frac{\sum_{j\geq 2} l_{r_j}}{\sum_{j\geq 2} u_{r_j}}$ we obtain that:

$$\frac{\sum_{j\geq 2} l_{r_j}}{\sum_{j\geq 2} u_{r_j}} = \frac{l_{r_2}}{u_{r_2}} + \frac{\sum_{j\geq 3} l_{r_j}}{\sum_{j\geq 3} u_{r_j}} \geq \min(\frac{l_{r_2}}{u_{r_2}}, \frac{\sum_{j\geq 3} l_{r_j}}{\sum_{j\geq 3} u_{r_j}}) \geq \ldots$$

$$\geq \min(\frac{l_{r_2}}{u_{r_2}}, \min(\frac{l_{r_3}}{u_{r_3}}, \ldots, \min(\frac{l_{r_{n-1}}}{u_{r_{n-1}}}, min(\frac{l_{r_n}}{u_{r_n}}) \ldots) = \frac{l_{r_2}}{u_{r_2}}.$$

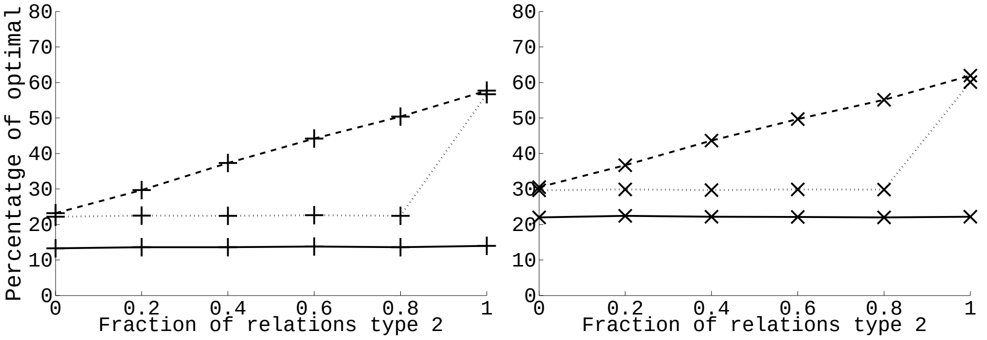
Thus, $\frac{\sum_{j\geq 2} l_{r_j}}{\sum_{j\geq 2} u_{r_j}} \geq \frac{l_{r_2}}{u_{r_2}}$ and consequently, $\frac{\sum_{j\geq 2} l_{r_j}}{\sum_{j\geq 2} u_{r_j}} \geq \frac{l_{r_1}}{u_{r_1}}$ holds.

4 Empirical results

In this section we provide with some results that show the average-improvement of the reward-dependent bounds defined in section 3 for some particular DCOP instances.

Figures 2(a)(b) show the values of the $\mathcal{C}$-optimal reward-based bounds for random DCOPs with 100 agents and density 4 using as a criterion neighborhoods of size 3 and of distance 1 respectively. All results are averaged over 50 sample instances. Bounds are calculated using the LP method. Because intuitively exploiting the knowledge about the minimum/maximum rewards should provide tighter bounds on heterogeneous reward structures than exploiting the knowledge about the minimum fraction reward, we generate DCOPs with two types of relations: (1) type 1, relations whose rewards are integers drawn from a uniform distribution $U[2500, 10000]$ and (2) type 2, relations whose rewards are integers drawn from a uniform distribution $U[5000, 10000]$.

The dotted lines show the reward-dependent bounds based on the minimum fraction reward (MFR), the dashed lines show the reward-dependent bounds based on the minimum/maximum relation rewards (MMR) and the solid lines show the original bounds as presented in [11]. The x-axis represents the fraction of the total relations of type 2 with respect to the default relations of type 1.



(a) Size 3 bounds

(b) Distance 1 bounds

Fig. 2. Reward-based bounds on 100 agent random DCOPs with density 4 using as a criterion (a) node size 3 and (b) distance 1.

Observe that both reward-dependent bounds provide significant improvements with respect the reward-independent bound. For example, the size 3 reward independent bound is around 12% whereas reward-dependent bounds are around 22% when using type 1 relations (x-axis= 0) and around 60% when using type 2 relations (x-axis= 1). Moreover, see that when all relations are of the same type both reward-based bounds, MMR and MFR, are very close. In contrast, in mixed instances, when both type of relations are present, graphs show that MFR can not improve the bound further by taking advantatge of type 2 relations. Indeed MFR bound does not get a significant improvement with the introduction of type 2 relations until the type 1 relations disappear. In contrast MMR bound significantly increases with the fraction of type 2 relations.

In summary, these results show how exploiting more knowledge about the reward structure, such as the minimum maximum reward per relation, help obtain tighter bounds for a wider range of reward distributions, particularly in heterogeneous distributions composed of rewards of different kind.

5 Conclusions

In this paper we extended the region optimal region bounds in [11] to exploit some a-priori knowledge of the reward structure of the problem, if available. To that end, this paper provided reward-dependent bounds that incorporate as available prior knowledge: (i) a ratio between the least minimum reward to the maximum reward among relations; (ii) a minimum and maximum rewards per relation.

References

1. Bowring, E., Pearce, J.P., Portway, C., Jain, M., Tambe, M.: On k -optimal distributed constraint optimization algorithms: new bounds and algorithms. In: AA-

MAS (2). pp. 607–614 (2008)

2. Farinelli, A., Rogers, A., Petcu, A., Jennings, N.R.: Decentralised coordination of low-power embedded devices using the max-sum algorithm. In: AAMAS (2). pp. 639–646 (2008)
3. Junges, R., Bazzan, A.L.C.: Evaluating the performance of DCOP algorithms in a real world, dynamic problem. In: AAMAS. pp. 599–606 (2008)
4. Kiekintveld, C., Yin, Z., Kumar, A., Tambe, M.: Asynchronous algorithms for approximate distributed constraint optimization with quality bounds. In: AAMAS. pp. 133–140 (2010)
5. Mailler, R., Lesser, V.R.: Solving distributed constraint optimization problems using cooperative mediation. In: AAMAS. pp. 438–445 (2004)
6. Modi, P.J., Shen, W.M., Tambe, M., Yokoo, M.: Adopt: asynchronous distributed constraint optimization with quality guarantees. Artif. Intell. 161(1-2), 149–180 (2005)
7. Pearce, J.P., Tambe, M.: Quality guarantees on k-optimal solutions for distributed constraint optimization problems. In: IJCAI. pp. 1446–1451 (2007)
8. Petcu, A., Faltings, B.: A scalable method for multiagent constraint optimization. In: IJCAI. pp. 266–271 (2005)
9. Vinyals, M., Shieh, E., Cerquides, J., Rodriguez-Aguilar, J.A., Yin, Z., Tambe, M., Bowring, E.: LP formulation of regional-optimal bounds. Tech. Rep. TR-IIIA-2011-01 (2011), at: http://www.iiia.csic.es/files/pdfs/TR201101.pdf
10. Vinyals, M., Pujol, M., Rodriguez-Aguilar, J., Cerquides, J.: Divide and Coordinate: solving DCOPs by agreement. In: AAMAS. pp. 150–156 (2010)
11. Vinyals, M., Shieh, E., Cerquides, J., Rodriguez-Aguilar, J.A., Yin, Z., Tambe, M., Bowring, E.: Quality guarantees for region optimal algorithms. In: AAMAS (2011)
12. Zhang, W., Wang, G., Xing, Z., Wittenburg, L.: Distributed stochastic search and distributed breakout: properties, comparison and applications to constraint optimization problems in sensor networks. Artif. Intell. 161(1-2), 55–87 (2005)