How Local Is That Optimum? *k*-Optimality for DCOP*

Jonathan P. Pearce, Rajiv T. Maheswaran and Milind Tambe
University of Southern California
Los Angeles, CA 90089
{jppearce, maheswar, tambe}@usc.edu

ABSTRACT

In multi-agent systems where sets of joint actions (JAs) are generated, metrics are needed to evaluate these sets and efficiently allocate resources for the many JAs. For the case where a JA set can be represented by multiple solutions to a DCOP, we introduce *k*-optimality as a metric that captures desirable properties of diversity and relative quality, and apply results from coding theory to obtain upper bounds on cardinalities of *k*-optimal JA sets. These bounds can help choose the appropriate level of *k*-optimality for settings with fixed resources and help determine appropriate resource allocation for settings where a fixed level of *k*-optimality is desired.

Categories and Subject Descriptors

I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence; I.2.8 [**Artificial Intelligence**]: Problem Solving, Control Methods, and Search

General Terms

Design, Theory

Keywords

constraint reasoning, DCOP, multiagent systems, k-optimality

1. INTRODUCTION

We consider a cooperative multi-agent system that generates a set of joint actions (JAs). The outcome for a single JA, a combination of individual actions, can be captured by a DCOP [2], a model which decomposes the system into a fixed interaction and reward structure. Motivating domains include a team of troops that generate many sorties which consume supplies or a team of rescue units that generate many potential plans (for a disaster rescue commander) which consume human decision time. JA sets can be (i) a sequence of JAs to execute or (ii) a set of choices, but in either case, they consume resources as a function of set size.

*This work is supported by the Defense Advanced Research Projects Agency (DARPA), through the Department of the Interior, NBC, Acquisition Services Division, under Contract No. NBCHD030010.

AAMAS'05 July 25-29, 2005, Utrecht, Netherlands.

Most work in DCOPs and similar methods generates a single JA with high absolute reward, but reward alone is a poor metric for domains with JA sets, as it often yields clustered solutions. Clustering is undesirable, as diversity (the difference among JAs) is often a key property for evaluating JA sets, e.g., commanders want varied options, not tiny perturbations of essentially the same option. However, diversity alone is undesirable, as we want to ensure a level of relative quality (each JA is best among a group of similar JAs and cannot be improved by simple changes). To do this, we define a metric, *k*-optimality, that naturally captures diversity and relative quality: a *k*-optimal JA has the highest reward within a neighborhood of JAs differing from it by at most *k* individual actions; *k*-optimality quantifies the neighborhood in which a local optimum is optimal. A *k*-optimal JA set (a collection of *k*-optimal JAs) then guarantees a level of relative quality (each JA is better than all JAs in a neighborhood of radius *k*) and diversity (JAs in the set must be separated by at least *k* individual actions).

Because each JA may consume resources, and the number of generated JAs may not be known *a priori*, resource allocation is a critical problem. Unfortunately, we cannot predict this number because the exact rewards in our domains are not known in advance. For example, supplies need to be allocated to troops executing multiple sorties before exact numbers and locations of adversaries are known. However, reward-independent upper bounds can be obtained on the size of *k*-optimal JA sets (i.e. to safely allocate enough supplies) by applying results from coding theory.

These bounds are useful in two ways: (i) If a particular level of *k*-optimality is desired, bounds indicate the maximum resource requirement for any *k*-optimal JA set. Thus, bounds provide a safe number of resources that can be allocated *a priori* to ensure enough for all *k*-optimal JAs. (ii) If resource availability is fixed, bounds help us identify instances where resources are guaranteed to be wasted. Because fewer *k*-optimal JAs can exist as *k* increases, if the resource level is above the bound on *k*-optimal JAs for the chosen *k*, then the resources above the bound are guaranteed to go unused.

2. *k*-OPTIMALITY

We introduce the notion of a *k-optimal joint action* as a metric that captures both relative quality and diversity when selecting JA sets. The fitness of *k*-optimality for evaluating JA sets is illustrated in Figure 1, which shows a deployment of troops where each unit can advance or hold. Decision-support agents (assigned to each unit) coordinate to generate multiple sortie plans to be executed. Using reward alone as a metric to generate a JA set leads to a cluster of near-identical solutions (essentially, all troops hold). Using diversity alone (ensuring all JAs differ by more than two actions) leads to a JA set where many JAs can be improved with deviations of only two agents (shown by the arrows). With *k*-optimality (*k*=2),

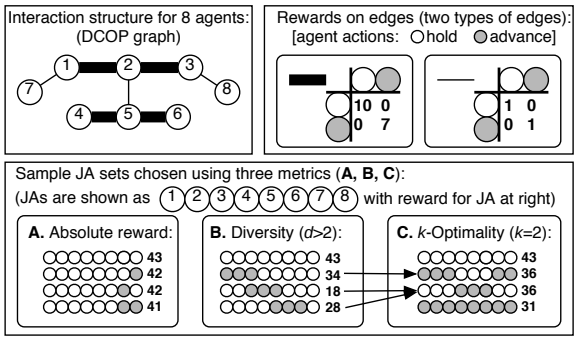

Figure 1: Generating JA sets under various metrics

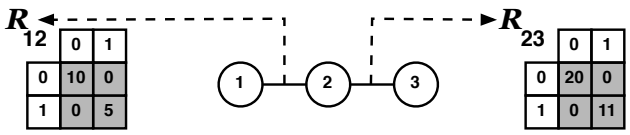

Figure 2: DCOP example

we generate a set of diverse JAs (all troops hold; front advances, rear holds; front holds, rear advances; all advance) where no JA can be improved with a two-agent deviation.

To define k-optimality more formally, we begin with our model of the multi-agent team problem. For a set of agents $\mathcal{I} := \{1, \ldots, I\}$, the i^{th} agent takes action $a_i \in \mathcal{A}_i$. We denote the joint action of a subgroup of agents $S \subset \mathcal{I}$ by $a_S := \times_{i \in S} a_i \in \mathcal{A}_S$ where $\mathcal{A}_S := \bigtimes_{i \in S} \mathcal{A}_i$ and the joint actions (JAs) of the entire multi-agent team by $a = [a_1 \cdots a_I] \in \mathcal{A}$ where $\mathcal{A} := \bigtimes_{i \in \mathcal{I}} \mathcal{A}_i$. The team reward for taking a particular JA, a, is an aggregation of the rewards obtained by subgroups in the team: $R(a) = \sum_{S \in \mathcal{S}} R_S(a) = \sum_{S \in \mathcal{S}} R_S(a_S)$ where S is a minimal subgroup that generates a reward (or incurs a cost) in an n-ary DCOP or cost network (i.e. a constraint), $\mathcal{S}$ is the collection of all such minimal subgroups for a given problem and $R_S(\cdot)$ denotes a function that maps $\mathcal{A}_S$ to $\mathbb{R}$. By minimality, we mean that the reward component R_S cannot be decomposed further: $\forall S \in \mathcal{S}, R_S(a_S) \neq R^1_{S_1}(a_{S_1}) + R^2_{S_2}(a_{S_2})$ for any $R^1_{S_1}(\cdot) : \mathcal{A}_{S_1} \to \mathbb{R}$, $R^2_{S_2}(\cdot) : \mathcal{A}_{S_2} \to \mathbb{R}$, $S_1, S_2 \subset \mathcal{I}$ s.t. $S_1 \cup S_2 = S$, $S_1, S_2 \neq \emptyset$. It is important to express the team reward in minimal form to accurately represent the dependencies and independencies among agents. Thus, $\mathcal{S} \subseteq \mathcal{P}(\mathcal{I})$ (where $\mathcal{P}(\cdot)$ denotes the power set) captures these local interactions.

To evaluate JA sets, specifically JAs with respect to each other, we need notions of neighborhood and distance among JAs. For two JAs, a and $\tilde{a}$, we define the following terms. The *deviating group* is $D(a, \tilde{a}) := \{i \in \mathcal{I} : a_i \neq \tilde{a}_i\}$, the set of agents whose individual actions differ. The *distance* is $d(a, \tilde{a}) := |D(a, \tilde{a})|$ where $|\cdot|$ denotes the cardinality of the set. The *relative reward* is $\Delta(a, \tilde{a}) := R(a) - R(\tilde{a}) = \sum_{S \in \mathcal{S}: S \cap D(a,\tilde{a}) \neq \emptyset} [R(a_S) - R(\tilde{a}_S)]$. Given the above, we can now classify a as a *k-optimal joint action* if $\Delta(a, \tilde{a}) > 0 \ \forall \tilde{a}$ s.t $d(a, \tilde{a}) \leq k$. Every JA can be given a k, identifying the size of the neighborhood where it is locally optimal. A collection of k-optimal JAs will be mutually separated by a distance greater than k as they each have the highest reward within a radius of k. Thus, a higher k-optimality of a collection implies a greater level of relative reward and diversity. Let $A_k = \{a \in \mathcal{A} : \Delta(a, \tilde{a}) > 0 \ \forall \tilde{a}$ s.t $d(a, \tilde{a}) \leq k\}$ be the set of all k-optimal JAs. It is straightforward to show $A_{k+1} \subseteq A_k$.

EXAMPLE 1. *Figure 2 is a binary DCOP in which agents choose actions from* $\{0, 1\}$*, with rewards shown for the two constraints (minimal subgroups)* $\mathcal{S} = \{\{1, 2\}, \{2, 3\}\}$*. The JA* $a = [1\ 1\ 1]$ *is* 1*-optimal because any single agent who deviates reduces the team reward. However,* $[1\ 1\ 1]$ *is not* 2*-optimal because if the group* $\{2, 3\}$ *deviated, making the JA* $\tilde{a} = [1\ 0\ 0]$*, team reward would increase from 16 to 20. The optimal JA,* $a^* = [0\ 0\ 0]$ *is k-optimal for all* $k \in \{0, 1, 2, 3\}$.□

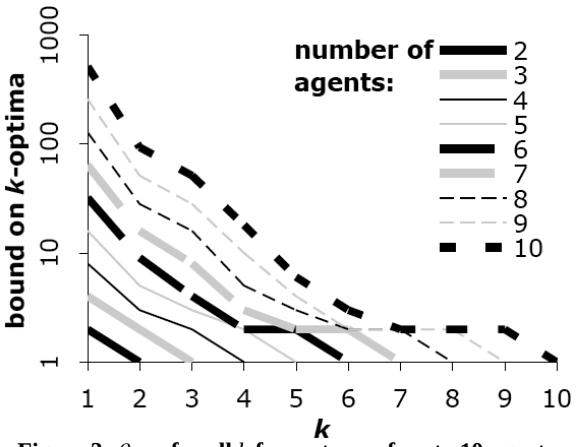

Figure 3: β_{HSP} for all k for systems of up to 10 agents

3. BOUNDS ON k-OPTIMA

Bounds on $|A_k|$ can help allocate resources when a particular level of k-optimality is desired, and can help identify guaranteed resource waste in fixed-resource settings. To find upper bounds on the number of k-optimal JAs, we discovered a correspondence to coding theory [1]. We assume every subgroup G has a unique optimal (subgroup) joint action a^*_G for any context a_{G^C} (if $G \subset \mathcal{I}$ where $G \neq \emptyset$ and $G \neq \mathcal{I}$, then $\exists\ a^*_G \in \mathcal{A}_G$ s.t. $R(a^*_G; a_{G^C}) > R(a_G; a_{G^C})$ for all $a_G \neq a^*_G$; G^C denotes the complement of set G). Finding the maximum possible number of k-optimal JAs can be mapped to finding the maximum number of codewords in a space of q^I words where the minimum distance between any two codewords is $d = k + 1$. We can map words to JAs and codewords to k-optimal JAs as follows: A joint action a taken by I agents each with an action space of cardinality q is analogous to a word of length I from an alphabet of cardinality q. The distance $d(a, \tilde{a})$ can then be interpreted as a Hamming distance between two words. Then, if a is k-optimal, and $d(a, \tilde{a}) \leq k$, then $\tilde{a}$ cannot also be k-optimal because that implies the subgroup $D(a, \tilde{a})$ has two optimal (subgroup) joint actions to the context $D(a, \tilde{a})^C$, violating our assumption. Thus, any two k-optimal JAs must be separated by distance greater than k.

Three well-known bounds on codewords are Hamming[1]: $\beta_H = q^I / \left(\sum_{n=0}^{\lfloor k/2 \rfloor} \binom{I}{n} (q-1)^n\right)$, Singleton: $\beta_S = q^{I-k}$, and Plotkin[2]: $\beta_P = \left\lfloor \frac{k+1}{k+1-(1-q^{-1})I} \right\rfloor$ [1]. Thus, $|A_k|$, the number of k-optimal JAs for a given I and q, can be bounded by $\beta_{HSP} := \min\{\beta_H, \beta_S, \beta_P\}$. For example, to find a reward-independent bound on 1-optimal JAs for 3 agents with $q = 2$, (e.g., the system in Figure 2), we obtain $\beta_{HSP} = 4$, without knowing R_{12} and R_{23} explicitly. If all 1-optimal JAs are to be executed, each requiring a resource, then the agents must be equipped with 4 resources to be safe. But, if they are equipped with more, the extra resources are guaranteed to go unused. Figure 3 shows β_{HSP} for all k for up to 10 agents, with $q = 2$.

4. REFERENCES

[1] S. Ling and C. Xing. *Coding Theory: A First Course*. Cambridge U. Press, 2004.

[2] P. J. Modi, W. Shen, M. Tambe, and M. Yokoo. Adopt: Asynchronous distributed constraint optimization with quality guarantees. *Artificial Intelligence*, 161(1-2):149–180, 2005.

[1]For even k. For odd k, with $q = 2$, $\beta_H(I, k, q) = \beta_H(I - 1, q, k - 1)$ can be used to obtain a tighter bound. [1]

[2]only valid when $(1 - 1/q)n < k + 1$