A Decision Network based Framework for Multiagent Coalition Formation *

Sayan D. Sen
Department of Electrical Engineering and Computer Science
Vanderbilt University
sayan.d.sen@vanderbilt.edu

Julie A. Adams
Department of Electrical Engineering and Computer Science
Vanderbilt University
julie.a.adams@vanderbilt.edu

ABSTRACT

Novel systems allocating teams of humans and unmanned heterogeneous vehicles are necessary for future applications. An intelligent framework is presented that reasons over a library of coalition formation algorithms to select the most appropriate algorithm(s) to apply to complex missions. The framework is based on decision networks to handle uncertainties in dynamic environments. A group of features is used to identify the most suitable algorithm(s). The proposed framework uses principal component analysis to extract the most significant features that are crucial for making decisions. A technique based on link analysis calculates the utility values for each feature-value pair and algorithm in the library. Experimental results demonstrate that the presented framework accurately chooses the most appropriate coalition formation algorithm(s) based on multiple specified mission criteria and requirements.

Categories and Subject Descriptors

I.2.11 [**Distributed Artificial Intelligence**]: Multiagent Systems

General Terms

Design, Experimentation

Keywords

Coalition Formation; Decision Networks; Multiagent Systems; Link Analysis

1. INTRODUCTION

Teaming humans and robots for tasks requires efficient coalition formation and coordination. Task requirements are often greater than the capabilities of a single robot. Therefore, efficient coalition formation of agents is necessary to perform tasks collectively. Coalition formation is an NP-complete problem [20] that is also hard to approximate within a reasonable factor [24]. Exponential search spaces lead to heuristic based solutions. Several algorithms leverage different heuristics to obtain solutions in a reasonable computation time [14, 25, 33, 36]. The use of heuristics does not guarantee solution quality and can result in sub-optimal solutions. Heuristics work well only when relevant information is available. Real world environments do not always guarantee the availability of the heuristic knowledge; therefore, rendering heuristic-based algorithms less applicable. It has been shown experimentally that heuristics are effective only when applicable to compatible missions [2].

The primary contribution of this paper is the framework, **i**ntelligent **C**oalition **F**ormation for **H**umans **a**nd **R**obots (*i-CiFHaR*) that reasons over a library of coalition formation algorithms, each employing separate heuristics, and chooses the most appropriate algorithm(s) to apply to a given situation. Such a system will be more robust and adaptable in dynamic, real-time environments than a system with a single heuristic-based coalition formation algorithm. The proposed framework incorporates a decision making module leveraging decision networks to select the most appropriate algorithm(s). Experimental results demonstrate that *i-CiFHaR* successfully chooses the most suitable algorithm(s) for various mission scenarios.

i-CiFHaR leverages a coalition formation algorithm taxonomy [23] that classifies the algorithms along multiple dimensions, or features. The features characterize real-world uncertain domains, and can be used to choose appropriate algorithm(s). A total of seventeen features (e.g., *Agent Capability*, *Agent Structure*, *Task Preemption*) are used.

Uncertainty is an important concern for complex missions. *i-CiFHaR*'s decision making module is based on decision networks to address these uncertainties. Decision networks are extensions of Bayesian networks that are used to model decision making under multiple, uncertain criteria [12]. The decision network renders the proposed framework scalable and flexible by allowing the addition of new taxonomy features as random variable nodes to the network. *i-CiFHaR* handles the combinatorial explosion associated with an increase in the number of features by extracting a subset of prominent features that play a significant role in determining the appropriate algorithm(s), thereby reducing the problem dimensionality. The extraction of the prominent features is based on Principal Component Analysis.

i-CiFHaR is similar to the autonomous squadron formation system for unmanned aerial vehicles [2], which uses three coalition formation algorithms (e.g., *Depth-First Iterative Deepening*, *Depth-First*, and *Genetic Algorithm*). The brute force algorithms search through all possible $(2^n - 1)$

*This research has been supported by an ONR DEPSCOR Award # N000140911161.

Appears in: *Proceedings of the 12th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2013), Ito, Jonker, Gini, and Shehory (eds.), May, 6–10, 2013, Saint Paul, Minnesota, USA.*

combinations of n vehicles to compute the best coalitions, rendering the framework impractical when n is large. Coalitions are computed by all the three algorithms, and the best coalition is selected based on a utility metric [2]. *i-CiFHaR* differs in two ways: (1) a library of diverse and intelligent coalition formation algorithms is used; rendering the framework more adaptable to a wide range of situations, and (2) *i-CiFHaR* chooses the most appropriate algorithm(s) to apply instead of utilizing all available algorithms, some of which may not be applicable in many scenarios.

Section 2 provides an overview of related work. Section 3 describes the taxonomy features and coalition formation algorithms. The framework design and construction of the decision network are detailed in Section 4. Experimental results are provided in Section 5. Section 6 and Section 7 present the discussion and conclusions, respectively.

2. RELATED WORK

Several multi-robot coordination architectures exist in the literature. One of the earliest fault-tolerant architectures for adaptive, distributed multi-robot task allocation is ALLIANCE [19] that leverages behaviors such as *impatience* and *acquiescence* to motivate robots to perform tasks. Coalitions of heterogeneous robots are autonomously generated by ASyMTRe [30] that leverages schema based sensor sharing. Schemas represent basic robot behaviors that are preprogrammed into the robot at design time. TraderBots [3] is a market-based multi-robot coordination architecture developed for completing tasks in dynamic environment. DEMiR-CF [21] is a generic framework designed for multi-robot coordination that uses standard auction rules. A novel framework, CoMutaR [26] has been proposed to tackle two important multi-robot task allocation issues: task allocation among robots, and efficient coordination among robots to complete a task. The Multiagent Adjustable Autonomy Framework (MAAF) is an innovative proxy based architecture that has been proposed for multi-robot and multi-human teams for complex missions [6]. MAAF is based on a state-of-the-art proxy framework that facilitates effective role allocation decisions for robots and humans [22].

Systems dealing with real-world scenarios must accommodate uncertainties, which can often be captured using certainty factors [27], fuzzy sets [10], and probability [34]. Bayesian networks have been used to perform sound inferences and reasoning in many medical systems [34]. Decision networks have proven to be an effective decision making tool in behavioral animation of virtual humans [37]. Yu and Terzopoulos [37] use a number of uncertain human qualities and environmental factors to construct a hierarchical decision network that triggers appropriate human actions during an emergency situation.

3. TAXONOMY AND ALGORITHMS

A well-defined taxonomy is crucial for multi-agent systems that defines a list of dimensions along which coalition formation algorithms can be classified. A three axes taxonomy [9] exists defining multi-robot task allocation problems as: (1) single-task vs. multi-task robots, (2) single-robot vs. multi-robot tasks, and (3) instantaneous vs. time-extended assignments of tasks. A taxonomy [4] for multi-robot systems has been formulated based on factors, such as: (1) number of robots, (2) communication topology, range, and bandwidth of robots, (3) processing capabilities of robots, and (4) group reconfigurability. Lao and Zhang's taxonomy is more task orientated and considers task demands, task resource requirements, and profit objectives [15].

i-CiFHaR employs a coalition formation algorithm taxonomy [23]. The taxonomy features used to classify algorithms are: *Agent Orientation* (F1), *Agent Heterogeneity* (F2), *Agent Capability* (F3), *Agent Awareness* (F4), *Agent Structure* (F5), *Inter-Task Constraint* (F6), *Task Preemption* (F7), *Task Requirement* (F8), *Intra-Task Constraint* (F9), *Task Coupling* (F10), *Performance Criteria* (F11), *Communication Requirement* (F12), *Task Allocation* (F13), *Domain Constraint* (F14), *Overlapping Coalition* (F15), *Algorithm Technique* (F16), and *Computation Method* (F17).

i-CiFHaR implements ten coalition formation algorithms. Shehory and Kraus' heuristic algorithm [25] addresses task allocation in a Distributed Problem Solving environment. Vig and Adams [33] extended Shehory and Kraus' algorithm [25] to apply to multi-robot domains. The modified algorithm models resources as non-transferable and includes fault tolerance by incorporating balance and fault-tolerance coefficients. Only Shehory and Kraus' algorithm [25] allows overlapping coalitions. RACHNA [32] employs a market-based auction procedure, where tasks use utilities as bid amounts to bid for agents. RACHNA allows task preemption by rerunning the bidding process. Abdallah and Lesser's algorithm [1] uses an underlying organization hierarchy to compute coalitions in polynomial time. The leaf nodes in the organization represent the agents and the non-leaf nodes represent managers acting as computational units. The managers use Q-learning to optimize local decision making within the organization. A clique-based distributed coalition formation algorithm [31] exploits the communication topology network among the agents and computes coalitions of modest sizes that form maximal cliques. Low inter-agent communication is required if the network is sparse. Weerdt et al.'s distributed algorithm [35] based on Contract Net Protocol computes coalitions that are connected in a social network based on inter-agent communication. The agent (manager) receiving a task recruits neighboring agents in the network to form coalitions. Gaston and desJardins' algorithm [7] also employs a social network based on inter-agent communication to calculate effective coalitions. Sujit et al.'s two-stage, distributed algorithm [29] specific to unmanned aerial vehicles concentrates on: (1) minimizing coalition sizes, and (2) minimizing task completion time. Service and Adams' resource-model based algorithm [24] is very similar to previous algorithms [25, 33] in that it employs the same greedy task allocation procedure and limits coalition sizes; however, it attempts to maximize system utility, rather than minimize cost. Finally, the auction based multi-agent task allocation system MURDOCH employs a resource-centric, publish/subscribe communication protocol to generate task coalitions [8].

4. SYSTEM DESIGN

i-CiFHaR incorporates a three tiered architecture (see Figure 1). A human provides mission requirements via a *User Interface*. The *Decision Making Module* chooses the most appropriate coalition formation algorithm(s) to apply, given the mission requirements. The *Utility Calculation* module calculates the utilities of the feature-value pairs and algorithms that are essential for creating the decision network's utility table (see Section 4.1). The *Feature Extraction* module extracts the most important taxonomy features that play a significant role in discriminating the coalition formation algorithms, which is necessary for dimensionality reduction (see Section 4.2). Decision networks form the core of the decision making process. The *Decision Network* module builds *i-CiFHaR*'s decision network dynamically at run-time based on the important features that are extracted by the *Feature Extraction* module (see Section 4.3).

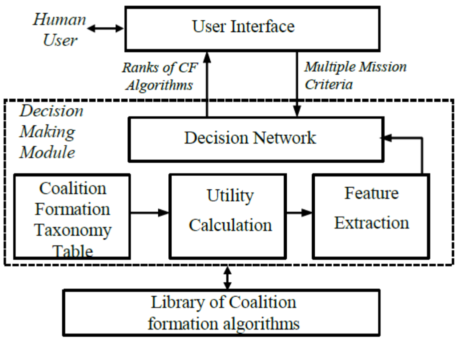

Figure 1: The *i-CiFHaR*'s framework.

4.1 Utility Calculation

A decision network contains *chance nodes* that represent random variables, *decision nodes*, and a single *utility node*. The *utility node* has an associated utility table containing utility values (degree of preference) for all possible configurations of the parent nodes. The parents of the *utility node* in *i-CiFHaR*'s decision network are: (1) a set of *chance nodes* representing the subset of important taxonomy features, and (2) a *decision node* comprising the coalition formation algorithms (see Section 4.3). The generation of the utility table entries, thus requires the utilities of the feature-value pairs and the algorithms.

The utility calculation is based on link analysis, which is used by HITS [13] and PageRank [18] to compute composite scores for web pages. Query web pages (called *hubs*) are linked to multiple query relevant web pages (called *authorities*) in a hyperlinked environment (WWW) [13]. Similarly, each coalition formation algorithm can be linked to a subset of related features governing its applicability (Figure 2). Thus, coalition formation algorithms and features can be visualized as *hubs* and *authorities*, respectively. Let F be a taxonomy feature set containing n features and *Domain* be a feature-domain set containing the domain of each of the n taxonomy features, then

$$\forall F_i \in F, \exists D_i \in Domain \mid 1 \leq i \leq n, D_i \neq \emptyset \qquad (1)$$

where D_i is feature F_i's domain of possible values. Let V be a set of size d containing all possible feature-value pairs, such that

$$V = \{(f_x, d_i) \mid f_x \in F, d_i \in D_x\} \qquad (2)$$

$$d = \sum_{i=1}^{n} |D_i|, \qquad (3)$$

where $|D_i|$ represents the domain size of $F_i \in F$.

For example, the *i-CiFHaR* feature (f_x) *Agent Structure*, with domain $D_x = \{$*organization hierarchy, social network, none*$\}$ results in three possible feature-value pairs: (1) {*Agent Structure, organization hierarchy*}, (2) {*Agent Structure, social network*}, and (3) {*Agent Structure, none*}. The entire set, V of size d can be derived from all the features and their corresponding domains. Let C be a set of size m containing the coalition formation algorithms. A bipartite directed graph $G = (C, V, E)$ is constructed using C and V as two disjoint node sets of G (Figure 2). A directed edge $e_k \in E$ from a coalition formation algorithm, $C_l \in C$ to a feature-value pair, $V_j \in V$ indicates that C_l is associated to the feature-value pair V_j.

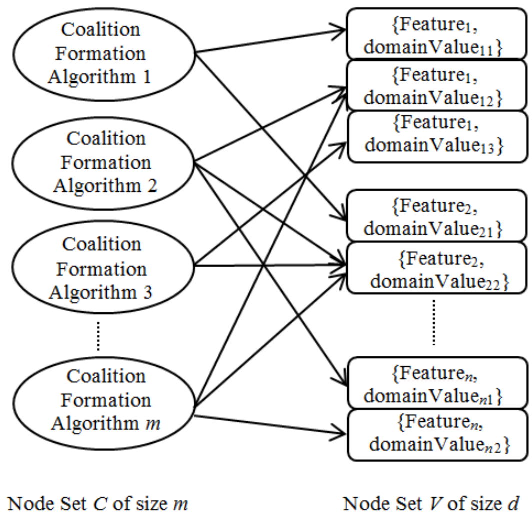

Figure 2: The link structure connecting coalition formation algorithms to feature-value pairs.

Algorithm 1 Utility Calculation algorithm

Input: mxd matrix A; iteration count k; x, x <k; const. c
Output: *algoUtil* of size 1xm; *featureUtil* of size 1xd

$classVector1 \leftarrow$ 1 x m vector of 1s
$featureVector1 \leftarrow$ 1 x d vector of 1s
$classVectorNormalized \leftarrow k$ x m matrix of 0s
$featureVectorNormalized \leftarrow k$ x d matrix of 0s
$classVectorNormalized_{1,:} \leftarrow$ 1 x m vector of $(1/m)$
$featureVectorNormalized_{1,:} \leftarrow$ 1 x d vector of $(1/d)$
for i = 2 to k **do**
  $featureVector_i \leftarrow classVector_{i-1} * A$
  $featureVectorNormalized_{i,:} \leftarrow \frac{featureVector_i}{\sum featureVector_i}$
  $classVector_i \leftarrow featureVector_i * A^T$
  $classVectorNormalized_{i,:} \leftarrow \frac{classVector_i}{\sum classVector_i}$
end for
for i = 1 to m **do**
  $algoUtil_i \leftarrow c$*mean$(classVectorNormalized_{k-x:k,i})$
end for
for i = 1 to d **do**
  $featureUtil_i \leftarrow c$*mean$(featureVectorNormalized_{k-x:k,i})$
end for

The Utility Calculation algorithm (Algorithm 1) is motivated by the HITS algorithm [13] and computes the utility

Table 1: The algorithm utility values

Coalition formation (CF) algorithms	Utility Value
CF Algorithm 1 [25]	10.51
CF Algorithm 2 [33]	10.44
CF Algorithm 3 [32]	10.09
CF Algorithm 4 [1]	10.22
CF Algorithm 5 [31]	10.24
CF Algorithm 6 [35]	10.48
CF Algorithm 7 [7]	9.44
CF Algorithm 8 [29]	9.75
CF Algorithm 9 [24]	9.88
CF Algorithm 10 [8]	8.91

Table 2: Utility values of some feature-value pairs

Features	Feature-Value Pairs	Utility Values
Agent Capability (F3)	{F3, Resource}	4.07
	{F3, Service}	0.99
Agent Structure (F5)	{F5, Organization}	0.52
	{F5, Social Network}	1.52
	{F5, None}	3.02
Task Requirements (F8)	{F8, Resource}	4.07
	{F8, Service}	0.99
Performance Criteria (F11)	{F11, Max.Utility}	4.00
	{F11, Min.Cost}	1.07

value of each feature-value pair and algorithm. A m x d matrix, A is required, which is computed based on the link structure (Figure 2). The rows of A represent the coalition formation algorithms, while the columns represent all possible feature-value pairs. An element, $a_{ij} \in A$ is defined as,

$$a_{ij} = \begin{cases} 1 & \text{if coalition formation algorithm } i \text{ is associated with feature-value pair } j \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

The constant c is used to scale the normalized utilities. The node weights in Figure 2 are initialized to 1 and are updated iteratively until they converge to steady-state utility values. The generated node utility values are purely a function of the link structure (Figure 2). The utility values of all the implemented coalition formation algorithms and some of the feature-value pairs are shown in Tables 1 and 2, respectively.

4.2 Feature Extraction

i-CiFHaR uses seventeen features to classify the coalition formation algorithms; however, many features do not contribute in classifying the algorithms. For example, the feature *Agent Orientation* is assigned the same value (*Group Rational*) for all implemented algorithms, and does not contribute to the classification. Thus, feature extraction, or selection is essential to remove redundant features and reduce the problem dimensionality. Principal component analysis has been demonstrated for feature selection [11, 28]. *i-CiFHaR* extracts prominent features that discriminates between algorithms using the Feature Extraction algorithm

Algorithm 2 Feature Extraction algorithm

Input: mxn utility matrix U; Eigen Value Threshold $eigValThreshold$; weight Threshold $wtThreshold$
Output: Prominent features list, $prominentFeatureList$

Subtract the mean of each feature of U from each corresponding feature column, resulting in a matrix U_k
$covC \leftarrow$ covariance of the matrix U_k
$eigVec \leftarrow n$ x n matrix having n Eigen vectors of $covC$
$eigVal \leftarrow 1$ x n vector of Eigen values of n Eigen Vectors
$index1 \leftarrow$ vector of 1 through n
$index2 \leftarrow \emptyset$
for i=1 to n **do**
  if $eigVal_i > eigValThreshold$ **then**
    for j=1 to n **do**
      if $|eigVec_{j,i}| < wtThreshold$ **then**
        $index2 \leftarrow index2 \cup j$
      end if
    end for
    $index1 \leftarrow index1 \cap index2$
    $index2 \leftarrow \emptyset$
  end if
end for
$featureIndex \leftarrow$ vector of 1 through n
$prominentFeatureList \leftarrow featureIndex \setminus index1$

(Algorithm 2), which is based on principal component analysis. Principal component analysis linearly transforms a set of variables into a much smaller set of uncorrelated variables, containing most of the information from the original data set [5]. Each of the principal components accounts for variances in the original data set, with the first component accounting for the maximum possible variance. Principal component analysis permits a mapping from a high dimensional feature space to a lower dimensional space. The principal components are expressed as a linear combination of the n system features. The k^{th} principal component, pc_k is defined by

$$pc_k = a_{k1}F_1 + a_{k2}F_2 + ... + a_{kn}F_n = \sum_{i=1}^{n} a_{ki}F_i \quad (5)$$

where F_i represents the i^{th} feature variable.

The primary statistics resulting from the k^{th} principal component are the weight vector $(a_{k1}, a_{k2}, a_{k3}, , a_{kn})$ and the associated variance (λ_k). The relative sizes of the elements in the weight vector indicate the relative contributions of the corresponding variables in the original data set to the variance of the principal component [5]. If any variable produces very small weight factors consistently for all the major principal components, then the variable has very little variance and contributes negligibly towards the variance of the entire data set. These variables (features) can be deemed as less significant and the remaining prominent features constitute the decision network's *chance nodes*.

The Feature Extraction algorithm (Algorithm 2) uses a m x n matrix U, where m is the number of algorithms in *i-CiFHaR*'s library and n is the number of taxonomy features. $u_{ij} \in U$ is the utility value (computed by the Utility Calculation algorithm) of the specific feature-value pair for feature j associated with algorithm i.

Seventeen *Eigen Vectors* representing the principal components are computed. Six components (the EigVec$x$ in

Table 3: Weight factors of the first six principal components or Eigen Vectors (EigVec$x$)

	Weight Factors of Feature Variables																
	F1	**F2**	**F3**	**F4**	**F5**	**F6**	**F7**	**F8**	**F9***	**F10***	**F11**	**F12**	**F13**	**F14**	**F15**	**F16**	**F17**
EigVec1	0	0	.528	0	.045	.030	.257	.528	0	0	-.344	.102	.005	0	-.345	.353	0
EigVec2	0	0	.186	0	-.405	-.166	.221	.186	0	0	.419	-.358	-.018	0	.506	.358	0
EigVec3	0	0	.357	0	.278	.094	-.061	.357	0	0	.257	.385	-.044	0	.432	.502	0
EigVec4	0	0	-.161	0	.376	-.113	.034	-.161	0	0	-.346	.347	.013	0	.541	.508	0
EigVec5	0	0	-.055	0	.257	-.367	.056	-.055	0	0	.654	.372	-.017	0	-.378	.282	0
EigVec6	0	0	.089	0	-.401	.199	-.778	.089	0	0	.050	.302	.035	0	-.01	.286	0

**Weight factors are very small (approximately between 10^{-16} and 10^{-18}), and thus considered 0*

Table 4: Variances or Eigen Values of the first six principal components or Eigen Vectors (EigVec$x$)

	Eigen Vectors					
	EigVec1	**EigVec2**	**EigVec3**	**EigVec4**	**EigVec5**	**EigVec6**
Eigen Values	4.69	3.04	2.04	0.75	0.48	0.31

Table 3, where $1 \leq x \leq 6$) are selected that account for approximately 98% of the total variance in the original data set. These six principal components and their associated variances (Eigen Values) are listed in Tables 3 and 4, respectively. Each component consists of a weight vector containing seventeen weight factors. The weight factors correspond to the feature variables used by *i-CiFHaR* to classify the algorithms. The weight factors of seven feature variables (F1, F2, F4, F9, F10, F14, and F17 in Table 3) are zero for all the major principal components and, thus have negligible variances and do not contribute significantly towards classifying the algorithms. The remaining ten prominent features (shaded grey in Table 3) become the *chance nodes* of *i-CiFHaR*'s decision network.

4.3 Decision Network Construction

Once the Feature Extraction algorithm (Algorithm 2) identifies the most prominent taxonomy [23] features, the decision network is built dynamically at run-time using the extracted features as *chance nodes* (Figure 3). The extracted prominent features are: *Agent Capability* (F3), *Agent Structure* (F5), *Inter-Task Constraint* (F6), *Task Preemption* (F7), *Task Requirement* (F8), *Performance Criteria* (F11), *Communication Requirement* (F12), *Task Allocation* (F13), *Overlapping Coalition* (F15), and *Algorithm Technique* (F16). The extracted features are the most influential, uncertain criteria required to determine the most applicable algorithm, given the specified mission requirements.

The *decision node* contains every decision alternative from which *i-CiFHaR* selects the appropriate algorithm(s). Since the decision making module selects the best possible algorithm(s), the *decision node*'s domain (rectangular node in Figure 3) comprises the coalition formation algorithms. There is one *utility node* in the network. The *chance nodes* (representing the extracted prominent taxonomy features) and the *decision node* are parents of the *utility node* (Figure 3). The *utility node* houses a table containing the utility values for all possible configurations of the parent nodes. The utility table values are usually obtained by consulting domain experts or through intuitions and preferences of the system designer [37]; however, *i-CiFHaR* uses a mathematical expression (see Equation 6 in Section 4.4) to calculate the utility table values.

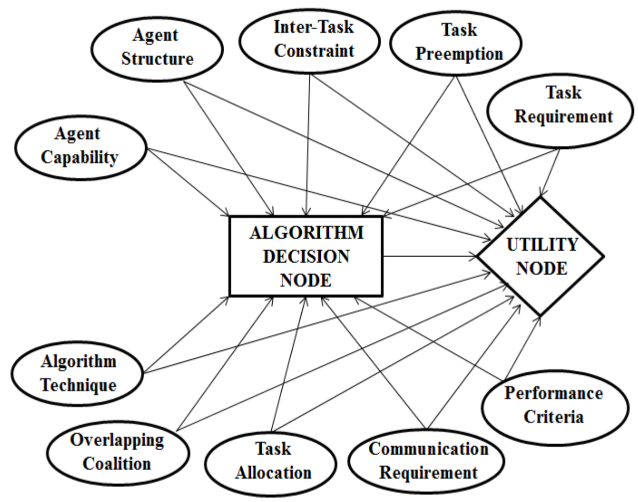

Figure 3: *i-CiFHaR*'s Decision Network.

4.4 Mathematical expression definition

It is often difficult to calculate the utility table values based on designer preferences or intuitions. The number of table entries is exponential to the number of parents of the *utility node*. Two approaches are implemented to address this problem. First, the most prominent features are used to construct the decision network, reducing the problem dimension. Second, the utility table entries are calculated using Equation (6), and as a result *i-CiFHaR* is more flexible, maintainable, and reusable.

$$\forall S_k \in W, U(S_k|Act_i) = UC_i * \sum_{j=1}^{N} a_{ij} * UV_j \qquad (6)$$

Equation 6 requires a m x d adjacency matrix A, where $a_{ij} \in A$ is defined by Equation 4 in Section 4.1. The number of prominent taxonomy features extracted by *i-CiFHaR*'s *Feature Extraction* module is N. $U(S_k|Act_i)$ is the utility value of the k^{th} state S_k of hypothetical world W, when an action Act_i is taken. Act_i indicates that the *decision node* chooses the i^{th} coalition formation algorithm C_i. The utility values of the i^{th} coalition algorithm, C_i and the j^{th} feature-value pair, V_j are denoted by UC_i and UV_j, respectively, both calculated by the Utility Calculation algorithm.

5. EXPERIMENTS

i-CiFHaR and the ten coalition formation algorithms are implemented on a Linux platform (Ubuntu-12.04, 64-bit) using C++ and Qt framework [16]. The system implements the decision network using the Netica-C API [17], a Bayesian network development software tool that uses a junction tree algorithm to evaluate decision networks. The following section describes the scenarios for which the decision network framework was evaluated.

5.1 Experimental Settings

A large number of mission scenarios can be generated randomly depending on the possible feature-value combinations. System users can establish domain and mission appropriate probabilities based on domain knowledge, prior mission deployments, intelligence, etc. Feature domain values are mutually exclusive; therefore, the sum of the feature-value probability assignments is 1. Table 5 delineates a subset of random scenarios for which *i-CiFHaR*'s decision making was evaluated. The prominent features (F3, F5, F6, F7, F8, F11, F12, F13, F15, and F16) are described in Section 4.2. Uncertain and dynamic scenarios for real-world missions were simulated in two ways. First, the domain value assigned to each of the prominent features that characterizes the mission domain was varied, denoted by Feature-Value Pairs (FVP) in Table 5. Secondly, the uncertainty related to each mission was varied by regulating the probability value associated with each feature-value pair, denoted by $PrVal$ in Table 5. Consider mission scenarios 1 and 4 that differ in many aspects. For example, the feature *Agent Capability* (F3) was assigned the domain value *Resource* (p = 0.8) in Scenario 1 and *Service* (p = 0.9) in Scenario 4. Scenario 1 does not require *Task Preemption* (F7) (p = 0.8), while Scenario 4 needs preemption (p = 0.9). The feature *Performance Criteria* (F11) was assigned to domain values *Minimize Cost* (p = 0.75) and *Maximize Utility* (p = 0.85) for Scenarios 1 and 4, respectively. Given the uncertain scenarios, *i-CiFHaR* effectively selected the most appropriate coalition formation algorithm(s) for each scenario by maximizing the expected utility scores. The constant, c in the Utility Calculation algorithm (see Section 4.1) was selected to be 100. The impact of constant c was assessed by varying the value of c from 50 to 200, in increments of 25. Eight trials were conducted for each value of c and each mission scenario. The differing c values resulted in no change in the algorithm rankings (Table 6).

5.2 Experimental Results

The coalition formation algorithms that are applicable to each mission scenario are highlighted in Table 6. The algorithms for each scenario are listed by decreasing expected utility scores, with the most appropriate algorithms highlighted in grey (Table 6). The algorithm with the highest expected utility is ranked as the best fit for the given scenario. A high expected utility score indicates how well the corresponding algorithm satisfies the mission requirements. The highlighted subset contains algorithms whose expected utility scores are within a 15% lower bound of the maximum expected utility score, thus the cutoff thresholds for the scenarios are 255.05, 225.04, 249.11, and 175.27, respectively. The algorithms' expected utility scores do not vary for a given set of feature-value pair probabilities, thus multiple trial runs generate the same results as a single run. *i-CiFHaR* generates algorithm rankings for each mission scenario with an average time of 0.082 seconds and standard deviation of 0.01 seconds over ten trials (40 trials in total).

6. DISCUSSION

The experiments demonstrated the ability of *i-CiFHaR* to select appropriate algorithms for a series of missions.

Mission Scenario 1 represented a random domain for a real-world mission. The mission's criteria can be best satisfied by Shehory and Kraus' algorithm (CFA1) [25], which was selected with the highest expected utility score. Vig and Adams' algorithm (CFA2) [33] and Service and Adams' algorithm (CFA9) [24] were also selected as the next most appropriate algorithms. Both CFA2 and CFA9 are extensions of algorithm CFA1 and are equally applicable to the scenario. The relative lower score of CFA2 is because it does not allow overlapping coalitions, while CFA1 allows both overlapping and non-overlapping coalitions. CFA9 does not permit overlapping coalitions and maximizes system utility, instead of minimizing the cost requirement.

The requirements of Mission Scenario 2 remained the same as that of Scenario 1, except that the mission required overlapping coalitions to curtail resource expenditure. *i-CiFHaR* selected CFA1 as the best algorithm and CFA2 as the second most suitable algorithm. The noticeable difference between the expected utility scores was due to the fact that CFA1 is the only algorithm that permits overlapping coalitions. CFA2 satisfied the remaining mission criteria; therefore, it was ranked as the second best for the scenario. Service and Adams' algorithm (CFA9) ranked third.

Mission Scenario 3 simulated a scenario that required a low communication footprint due to constrained bandwidth. The agents form a communication topology based on limited inter-agent communication, giving rise to a sparse social network. The system selected Tošić and Agha's algorithm (CFA 5) [31] as the most appropriate algorithm. As described in Section 3, this algorithm uses a social network and requires low communication bandwidth when the underlying topology graph is sparse. Weerdt et al.'s algorithm (CFA6) [35] was the second best algorithm, as it satisfied most of the mission criteria, but it requires higher communication bandwidth.

There was a high probability that the last scenario included a number of high priority tasks requiring task preemption. RACHNA (CFA3) [32] was selected with a high expected utility score as the most appropriate algorithm, which is justified given that RACHNA is the only algorithm that allows task preemption. Service and Adams' algorithm (CFA9) does not allow task preemption, but it satisfied many of the mission requirements and was ranked the second best algorithm.

Consider an urban bomb blast incident, where reports suggest multiple injured victims, and a high probability of unknown bombs in the area. Before the human first responders enter the hazardous area, coalitions of robots are sent in to assess the situation and report the victim locations and any potential threats (e.g., bombs, hazardous materials). This mission can trigger a number of high-priority tasks, similar to Mission Scenario 4. Since this example mission requires task preemption, *i-CiFHaR* will select Vig and

Table 5: Feature-Value Pair Probabilities for various mission scenarios

Mission Scenario 1	**FVP**	F3,Res	F5,None	F6,Yes	F7,No	F8, Res	F11,MC	F12,High	F13,IA	F15,No	F16, Gr
	PrVal	0.8	0.8	0.75	0.8	0.85	0.75	0.8	0.85	0.8	0.85
Mission Scenario 2	**FVP**	F3,Res	F5,None	F6,Yes	F7,No	F8, Res	F11,MC	F12,High	F13,IA	F15,Yes	F16, Gr
	PrVal	0.8	0.8	0.75	0.8	0.85	0.75	0.8	0.85	0.8	0.85
Mission Scenario 3	**FVP**	F3,Res	F5,SNet	F6,No	F7,No	F8,Res	F11,MU	F12,Low	F13,IA	F15,No	F16, Gr
	PrVal	0.8	0.85	0.85	0.8	0.85	0.75	0.9	0.85	0.75	0.85
Mission Scenario 4	**FVP**	F3,Ser	F5,None	F6,Yes	F7,Yes	F8,Ser	F11,MU	F12,High	F13,IA	F15,No	F16,Auc
	PrVal	0.9	0.8	0.75	0.9	0.8	0.85	0.8	0.85	0.8	0.8

***Key** Res: Resource, Ser: Service, MC: Min. Cost, MU: Max. Utility, IA: Instantaneous, Gr: Greedy, Auc: Auction, SNet: Social Nets

Table 6: Coalition formation algorithms (CFA) by mission scenarios by expected utility scores

Mission Scenario1	**Algorithms**	CFA1	CFA2	CFA9	CFA8	CFA6	CFA5	CFA4	CFA10	CFA3	CFA7
	Utility Score	300.02	296.98	283.54	244.19	238.71	225.97	222.06	208.90	154.28	130.86
Mission Scenario2	**Algorithms**	CFA1	CFA2	CFA9	CFA8	CFA6	CFA5	CFA4	CFA10	CFA3	CFA7
	Utility Score	264.75	251.32	245.94	207.15	198.89	187.08	185.22	175.09	115.94	95.43
Mission Scenario3	**Algorithms**	CFA5	CFA6	CFA4	CFA9	CFA8	CFA1	CFA2	CFA7	CFA10	CFA3
	Utility Score	293.06	267.57	245.21	243.11	236.46	227.54	225.25	192.73	172.55	108.45
Mission Scenario4	**Algorithms**	CFA3	CFA9	CFA1	CFA2	CFA10	CFA8	CFA6	CFA4	CFA5	CFA7
	Utility Score	206.19	187.39	168.37	166.46	158.50	144.51	131.58	122.14	118.73	101.31

Adams' algorithm (CFA3), because it permits task preemption. The first task (*search area*) is to search the area for victims and bombs. Once victims are found, a second, more important task (*triage victim*) is triggered. The *search area* task is preempted, and a new coalition of robots is formed to triage victims. The remaining robots are assigned to the *search area* task. When a bomb is found, the higher priority task (*bomb diffuse*) is triggered, the other two tasks are preempted and a new coalition of robots is allocated to diffuse the bomb. The wireless connectivity in the region has diminished due to the original bomb's release of radiation, rendering CFA3 inapplicable owing to its high communication requirements. *i-CiFHaR* dynamically re-evaluates the algorithm selection procedure, accounting for the mission's new low-communication requirements (similar to Mission Scenario 3), and selects Tošić and Agha' algorithm [31] to form new task coalitions.

7. CONCLUSIONS

An intelligent framework is presented that reasons online over a library of coalition formation algorithms to select the most appropriate coalition formation algorithm(s) to apply to a given mission scenario. *i-CiFHaR* is the first coalition framework to use a library of algorithms, rather than a single heuristic based algorithm. The framework leverages a decision network to make decisions online under multiple, uncertain mission criteria. A link analysis based algorithm calculates the utility values of the algorithms and the feature-value pairs. The framework uses a number of features to select the most suitable coalition formation algorithm(s). The curse of dimensionality is addressed by extracting prominent features that discriminate the coalition algorithms. These prominent features are utilized to dynamically create the decision network at run-time. The experimental results show that *i-CiFHaR* selects the appropriate algorithm(s), given multiple mission criteria. When a single best fit algorithm is unavailable, *i-CiFHaR* selects a subset of suitable algorithms that are applicable to form coalitions. *i-CiFHaR* is applicable to missions with frequent contingency occurrences that introduce changing mission requirements (e.g., overlapping coalitions resulting from robot failures, task preemption). The likelihood of handling diverse situations increases with the inclusion of a broad set of algorithms in the system. *i-CiFHaR* provides a more robust approach to allocate task coalitions for dynamic, real-world scenarios. Future work includes incorporating additional coalition formation algorithms (e.g., anytime algorithms, etc.).

8. REFERENCES

[1] S. Abdallah and V. Lesser. Organization-Based Cooperative Coalition Formation. In *Proceedings of the IEEE/WIC/ACM International Conference on Intelligent Agent Technology*, pages 162–168, 2004.

[2] P. DeJong. Coalition Formation in Multi-Agent UAV Systems. MS Thesis, University of Central Florida, 2005.

[3] M. B. Dias. *TraderBots: A New Paradigm for Robust and Efficient Multirobot Coordination in Dynamic Environments*. PhD Thesis, Carnegie Mellon University, 2004.

[4] G. Dudek, M. Jenkin, and E. Milios. A taxonomy of multirobot systems. In *Robot Teams: From Diversity to Polymorphism*, pages 3–22, 2002.

[5] G. H. Dunteman. *Principal Components Analysis.* Sage Publications, 1989.

[6] A. Freedy, O. Sert, E. Freedy, J. McDonough, G. Weltman, M. Tambe, T. Gupta, W. Grayson, and P. Cabrera. Multiagent Adjustable Autonomy Framework (MAAF) for Multi-Robot, Multi-Human Teams. In *Proceedings of the International Symposium on Collaborative Technologies and Systems*, pages 498–505, 2008.

[7] M. E. Gaston and M. desJardins. Agent Organized Networks for Dynamic Team Formation. In *Proceedings of the 4th International Joint Conference*

on Autonomous Agents and Multiagent Systems, pages 230–237, 2005.

[8] B. Gerkey and M. J. Matarić. Sold!: Auction Methods for Multirobot Coordination. *IEEE Transactions on Robotics and Automation*, 18(5):758–768, 2002.

[9] B. P. Gerkey and M. J. Matarić. A formal analysis and taxonomy of task allocation in multi-robot systems. *International Journal of Robotics Research*, 23(9):939–954, 2004.

[10] C. Herrmann. A Hybrid Fuzzy-Neural Expert System for Diagonosis. In *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*, pages 494–502, 1995.

[11] I. Jolliffe. Discarding Variables in a Principal Component Analysis. I: Artificial Data. *Journal of the Royal Statistics Society. Series C (Applied Statistics)*, 21(2):160–173, 1972.

[12] U. B. Kjærulff and A. L. Madsen. *Bayesian Networks and Influence Diagrams: A Guide to Construction and Analysis*. Springer Publishing Company, 2008.

[13] J. M. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46(5):604–632, 1999.

[14] S. Kraus, O. Shehory, and G. Taase. Coalition Formation with Uncertain Heterogeneous Information. In *Proceedings of the International Conference on Autonomous Agents and Multiagent Systems*, pages 1–8, 2003.

[15] H. C. Lau and L. Zhang. Task Allocation via Multi-Agent Coalition Formation: Taxonomy, Algorithms and Complexity. In *Proceedings of the 15th IEEE International Conference on Tools with Artificial Intelligence*, pages 346–350, 2003.

[16] Nokia. Qt. http://www.qt.nokia.com, January 2012.

[17] NORSYS. Netica application: A complete software package to solve problems using bayesian belief networks and influence diagrams. http://www.norsys.com/netica.html, August 2012.

[18] L. Page, S. Sergey, R. Motwani, and T. Winograd. The PageRank Citation Ranking: Bringing Order to the Web. Technical report, Stanford University, 1999.

[19] L. E. Parker. ALLIANCE: An Architecture for Fault Tolerant Multi-Robot Cooperation. *IEEE Transactions on Robotics and Automation*, 14(2):220–240, 1998.

[20] T. Sandholm, K. Larson, M. Andersson, O. Shehory, and F. Tohmé. Coalition structure generation with worst case guarantees. *Artificial Intelligence*, 111(1-2):209–238, 1999.

[21] S. Sariel. *An Integrated Planning, Scheduling and Execution Framework for Multi-Robot Cooperation and Coordination*. PhD Thesis, Istanbul Technical University, 2007.

[22] P. Scerri, D. Pynadath, L. Johnson, P. Rosenbloom, M. Si, N. Schurr, and M. Tambe. A Prototype Infrastructure for Distributed Robot-Agent-Person Teams. In *Proceedings of the 2nd International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 433–440, 2003.

[23] T. C. Service and J. A. Adams. Coalition Formation Algorithm Taxonomy. Technical Report HMT-10-03, Vanderbilt University, 2010.

[24] T. C. Service and J. A. Adams. Coalition formation for task allocation: theory and algorithms. *Journal of Autonomous Agents and Multi-Agent Systems*, 22(2):225–248, 2011.

[25] O. Shehory and S. Kraus. Methods for task allocation via agent coalition formation. *Artificial Intelligence*, 101:165–200, 1998.

[26] P. M. Shiroma and M. F. M. Campos. CoMutaR: A framework for multi-robot coordination and task allocation. In *Proceedings of the IEEE International Conference on Intelligent Robots and Systems*, pages 4817–4824, 2009.

[27] E. H. Shortliffe. *Computer-Based Medical Consultations: MYCIN*. Elsevier Science Ltd, 1986.

[28] F. Song, Z. Guo, and D.Mei. Feature Selection Using Principal Component Analysis. In *Proceedings of the International Conference on System Science, Engineering Design and Manufacturing Informatization*, pages 27–30, 2010.

[29] P. B. Sujit, J. M. George, and R. W. Beard. Multiple UAV Coalition Formation. In *Proceedings of the American Control Conference*, pages 2010–2015, 2008.

[30] F. Tang and L. E. Parker. ASyMTRe: Automated Synthesis of Multi-Robot Task Solutions through Software Reconfiguration. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1501–1508, April 2005.

[31] P. T. Tošić and G. A. Agha. Maximal Clique Based Distributed Coalition Formation for Task Allocation in Large-Scale Multi-agent Systems. In *Proceedings of the Massively Multi-Agent Systems I*, pages 104–120, 2005.

[32] L. Vig and J. A. Adams. Market-based multi-robot coalition formation. In *Proceedings of the 8th International Symposium on Distributed Autonomous Robotic Systems*, pages 227–236, 2006.

[33] L. Vig and J. A. Adams. Multi-Robot Coalition Formation. *IEEE Transactions on Robotics*, 22(4):637–649, August 2006.

[34] X. Wang, H. Qu, P. Liu, and Y. Cheng. A self-learning expert system for diagnosis in traditional Chinese medicine. *Expert Systems with Applications*, 26:557 – 566, 2004.

[35] M. Weerdt, Y. Zhang, and T. Klos. Distributed Task Allocation in Social Networks. In *Proceedings of the 6th International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 17–24, 2007.

[36] K. Westwood and V. H. Allan. Heuristics for Co-opetition in Agent Coalition Formation. In *Proceedings of the Scandinavian Conference on Artificial Intelligence*, pages 143–150. Helsinki University of Technology, 2006.

[37] Q. Yu and D. Terzopoulos. A Decision Network Framework for the Behavioral Animation of Virtual Humans. In *Proceedings of the 2007 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 119–128, 2007.