51st IEEE Conference on Decision and Control
December 10-13, 2012. Maui, Hawaii, USA

Distribution of Agents with Multiple Capabilities in Heterogeneous Multiagent Networks – A Graph Theoretic View

Waseem Abbas and Magnus Egerstedt

Abstract— **In this paper, we investigate how to distribute agents of different types in heterogeneous multiagent systems. Heterogeneity can for instance be related to various resources and capabilities agents may have. We insist that every agent can find all different resources available in the network in its closed neighborhood. The total number of different resources that can be accommodated within a system under this setting depends on the underlying graph structure of the network. This paper provides an analysis of the assignment of multiple resources to nodes and the effect of these assignments on the overall heterogeneity of a network. We extend our analysis to proximity graphs, a widely used interaction model in multi agent and wireless systems. In addition, we perform qualitative and quantitative studies regarding the roles of individual agents and their interactions in such heterogeneous networks.**

I. INTRODUCTION

One of the challenges in heterogeneous networks is to optimally distribute agents with different capabilities within a network. In fact, the interconnection topology becomes a significant factor in determining the system's overall performance and capability when the agents are non-homogeneous and equipped with different resources. Several applications of such systems have been studied in the literature, ranging from energy efficient sensor networks (e.g., [1], [2]), coverage and optimization problems (e.g., [3], [4]), surveillance and monitoring systems (e.g., [5], [6]), facility location problems in operations research (e.g., [7]), and topology control in wireless networks (e.g., [8], [9]), to name a few. All of these problems can be studied in terms of this broader issue of how to use the underlying network structure to optimally perform various complex group level tasks by distributing nodes with various capabilities across the network.

In [10], we employed graph theoretic methods to characterize heterogeneity in multiagent systems from a network topology view point. The analysis was performed under the setting where each agent belongs to only one of the L different types available, and they are distributed such that every agent can find all different types in its closed neighborhood. In this paper, we continue to characterize the distribution of agents in heterogeneous multiagent systems in a more general framework, allowing each individual agent to have multiple capabilities. In order to deal with such situations, we can utilize the concept of assigning multiple types of resources to a node instead of one [11].

In this paper, we investigate this multiple resource assignment problem over a graph in the context of heterogeneous multiagent systems. Each agent is equipped with at most s different "capabilities" (resources or facilities) from a given set of r unique capabilities. Every node performs a task that needs to utilize all r facilities or resources within a network by interacting with its neighbors only. For a given system, how can we get such a distribution of agents? An even more fundamental concern is if such a distribution is possible at all for a given network topology. We address these issues by analysing the role of individual nodes and interactions in the context of heterogeneous distributions of capabilities among agents within a network. In terms of the network topology of multiagent system, these constraints can be related to a so-called (r, s)-configuration of an underlying graph structure [11]. Here, the goal is to assign s unique colors (or labels) to each vertex in a graph such that every vertex has r unique colors in its closed neighborhood.

If k different resource types can be accommodated in a network under the setting where every agent can find all resources in its closed neighborhood by having only one resource by itself, then clearly ks different resources can be accommodated in the same network if each agent is allowed to have s resource types. But it is shown in [11] that it may be possible to incorporate *more* than ks resource types in a network under a similar setup. Thus, the ability of the network structure to accommodate heterogeneous entities may improve significantly with the leverage of assigning multiple resources to the nodes.

The organization of the rest of the paper is as follows: In Section II, we introduce the notations used in the paper. Section III provides an analysis of the (r, s)-configuration property of graphs for multiagent systems. A suffcicient condition for a graph to have an (r, s)-configuration is presented in Section IV. In Section V, a special case of $(5, 2)$-configuration for R-disk proximity graphs is discussed. Finally we present our conclusions in Section VI.

II. PRELIMINARIES

In this section, we introduce the terms that are used throughout the paper. Also, we state the problem along with some preliminary results from [11].

Throughout this paper, a *graph* $G(V, E)$, with a vertex set V and an edge set E, is a simple undirected graph. An edge between the nodes v_i and v_j is denoted by $v_i \sim v_j$. The *open neighborhood* of a vertex $v \in V(G)$, denoted by $\mathcal{N}(v)$, is the set of vertices adjacent to v. Its *closed neighborhood*, denoted by $\mathcal{N}[v]$, is $\mathcal{N}(v) \cup \{v\}$. The degree of a vertex v, $deg(v)$, is the cardinality of $\mathcal{N}(v)$. The minimum degree of a graph, $\delta(G)$, is $min\{deg(v) \mid v \in V\}$ and the maximum

W. Abbas and M. Egerstedt are with a School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA wabbas@gatech.edu, magnus@gatech.edu

978-1-4673-2066-5/12/$31.00 ©2012 IEEE

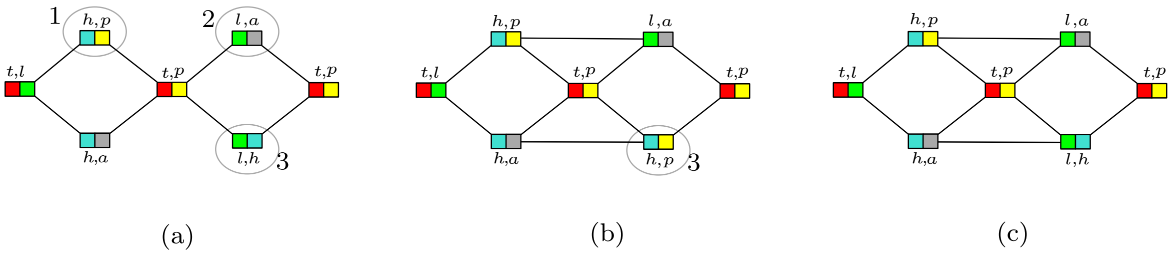


Fig. 1. (a) The closed neighborhoods of nodes 1 and 3 are missing label a, while node 2 does not have label h in its closed neighborhood. (b) Node 3 is missing l in $\mathcal{N}[3]$. (c) All of the five labels are distributed to get a $(5, 2)$-configuration.

degree of a graph, $\Delta(G)$, is $max\{deg(v) \mid v \in V\}$. A set S is a *dominating set*, if for each $v \in V(G)$, either $v \in S$ or v is adjacent to some $w \in S$. In other words, the closed neighborgood of each $v \in V(G)$ must contain at least a vertex in S for it to be a dominating set. The *domination number* is the cardinality of a dominating set with the minimum number of vertices. A *domatic partition*, $\mathcal{D}$, is a partition of $V(G)$ into subsets, $\mathcal{D} = \{V_1, V_2, \cdots, V_k\}$, such that, $\bigcup_{i=1}^{k} V_i = V(G)$, and each $V_i \in \mathcal{D}$ is a dominating set in G. The maximum cardinality of $\mathcal{D}$ is the *domatic number* of G, denoted by $dom(G)$. The difference of sets A and B, denoted by $A - B$, is the set of elements in A that are not in B. Also, a s-subset of a set $\mathcal{R}$, is a subset of $\mathcal{R}$ containing at most s elements.

Let us model the underlying network topology of a multiagent system by a graph $G(V, E)$, where V represents the set of agents and E represents the inter-connections among agents. Assume there are r possible types of resources or facilities that need to be distributed among the nodes, such that every node gets a maximum of s different types. Each node is assumed to be performing a "task" that needs to utilize all r different resources by interacting with its neighbors only. For a given G, how can we get such a distribution of resources among the agents, if it is possible at all? In mathematical terms, we can state this as follows

Let $\mathcal{R} = \{1, 2, \cdots, r\}$ be a set of labels. A function f,

$$f : V \to [\mathcal{R}]_s$$

is called an (r,s)-configuration of a graph G, where $[\mathcal{R}]_s$ is a collection of all s-subsets of $\mathcal{R}$, such that $\bigcup_{u \in \mathcal{N}[v]} f(u) = \mathcal{R}$, $\forall\, v \in V(G)$.

(r, s)-configurations of graphs are useful in studying heterogeneity in multiagent systems from a network topology view point. In such networks, agents may be different from each other in terms of their resources or capabilities (for instance, sensing, actuation, software, computational complexity). A unique label (or a color) can be associated with each resource type available in the network. All of the vertices in an underlying graph of the network are then assigned labels (or colors) in accordance with the resources contained by the corresponding agents. A vertex may have multiple labels if the agent has more than one resource types. In terms of (r, s)-configurations, s is the maximum number of resources any agent can have, and r is the maximum number of resource types available within a network such that every agent can find all these resource types in its closed neighborhood.

Various network topology related aspects of heterogeneity in multiagent systems can be studied in terms of the framework of (r, s)-configurations. For instance, does there exist an (r, s)-configuration of a graph G for a given r and s? For a given s, what is the maximum value of r for an (r, s)-configuration of G? How can we extend a given labelling of a graph to an (r, s)-configuration by adding edges? We address these problems in Seccions III and IV. Firstly, we illustrate these issues through an example below.

Example:

Consider an industrial location where some manufacturing process depends on environmental conditions, including temperature (t), light (l), humidity (h), air pressure (p), and air flow (a). A specific environmental condition, say $\omega(t, l, h, p, a)$, related to all of the above parameters needs to be maintained. Sensors for each of the above parameters t, l, h, p, a are mounted at various data collection points where $\omega(t, l, h, p, a)$ is computed. Let there be a constraint (e.g., hardware) that a maximum of two sensors can be mounted at every data collection point. Since, all five parameters are needed for the computation of $\omega(t, l, h, p, a)$, sensors need to be distributed such that all five sensor types are available in the closed neighborhood of every data collection point. So, the problem is to determine a $(5, 2)$-configuration of the underlying network topology. Three different cases are shown in the Fig. 1 for this set up. In the first case, the required distribution of sensors is not possible as a $(5, 2)$-configuration does not exist for that particular graph. In the second case, although a $(5, 2)$-configuration exists, the sensors are not distributed to achieve it. In the last case, the right distribution of sensors is shown.

Domination in graphs (see e.g., [14]) provides a basic tool for studying (r, s)-configurations of graphs. A graph G, having $dom(G) = \gamma$ has a $(s\gamma, s)$-configuration. But interestingly, it may also have an (r, s)-configuration, for some $s\gamma < r$. So, an (r, s)-configuration of a graph allows us to explore and utilize its structure in a more profitable way. For example, there are cycle graphs with n nodes, denoted by C_n, that have a domatic number of 2. So, they always have a $(4,2)$-configuration, but it is shown in [11] that every cycle C_n, where $n \neq 4,7$, has a $(5,2)$-configuration. Similar is the case with the cubic graphs [1]. Every cubic graph has a $(5,2)$-configuration [11], although there are infinite number of cubic graphs with a domatic number of 2. So, assigning multiple facilities to a node in a network may possibly increase the overall capacity of the network to accommodate a larger number of facilities.

Throughout the paper, we will use the terms labels and colors interchangeably, depending on the context. Also, the node and agent terms are equivalent here.

III. ANALYSING NETWORKS FOR (r,s)-CONFIGURATIONS

In heterogeneous multiagent systems, agents with different resources or capabilities (for instance sensing, actuation) are interconnected with each other. Local tasks performed by an agent depends on the resources available in its neighborhood. Thus, we need a way to analyse how various types of agents are distributed within a network? This information will be useful to figure out the missing resources in the neighborhood of an agent along with the interactions needed to make these resources available to that agent. In mathematical terms, we need a formal way to get an (r,s)-configuration from a given labelling of an underlying graph. This can be achieved by first identifying missing labels from the neighborhoods of its vertices, and then determining extra edges required to make these missing labels available in the vertices' neighborhoods. In addition, it is also valuable to characterize redundant edges, i.e. edges whose removal will not matter for the sake of the (r,s)-configuration. We address all these issues in this section.

Given a graph with n nodes, where each node has at most s distinct labels from a set of r labels. Let the *color matrix*, $C \in \mathbb{R}^{n\times r}$ be given by,

$$C_{ij} = \begin{cases} 1 & \text{if } j \in f(v_i), \quad \text{where } j \in \{1,2,\cdots,r\} \\ 0 & \text{otherwise.} \end{cases}$$

Here $f(v)$ indicates the colors assigned to a vertex v. The column index of C indicates the label (or the color), thus $C_{ij} = 1$ means that color j has been assigned to the vertex v_i. Note that if a maximum of s different colors can be assigned to a vertex, then there can be at most s number of 1's in each row of C.

We also define a *color distribution matrix*, Φ as follows,

$$\Phi = AC + C$$

where A is the adjacency matrix of the graph and C is the color matrix. Here, $\Phi \in \mathbb{R}^{n\times r}$,.

The color distribution matrix gives information regarding the distribution of various colors within a network. In fact, it tells us about the exact number of various colors available in the closed neighborhood of any node in a network, as stated in the following lemma.

[1]A graph whose every vertex has a degree 3 is a cubic graph.

Lemma 3.1: [10] Φ_{ij} is the number of nodes with color j in the closed neighborhood of node v_i.

Thus, for a given coloring, $\Phi_{ij} = 0$ means that v_i is missing the color j in its closed neighborhood. Thus, an extra edge is needed to connect v_i with some v_u, with a color j. The upper and lower bounds on the number of extra edges required to get an (r,s)-configuration from a given coloring of G are presented in the following result.

Theorem 3.2: The number of extra edges $\mathcal{E}$, needed to get an (r,s)-configuration from a given coloring of G is,

$$\left\lceil \frac{z(\Phi)}{2s} \right\rceil \leq \mathcal{E} \leq z(\Phi) \tag{1}$$

where, $z(\Phi)$ is the number of 0's in the color distribution matrix, Φ, for the given coloring.

Proof : Let $v_i \sim v_j$ be an extra edge connecting vertex v_i with colors $\kappa_1, \kappa_2, \cdots, \kappa_s$, to vertex v_j with colors $\tau_1, \tau_2, \cdots, \tau_s$. Since, every vertex can have at most s distinct colors, so $v_i \sim v_j$ can add at most s missing colors in $\mathcal{N}[v_i]$ and also at most s missing colors in $\mathcal{N}[v_j]$. This is possible whenever v_i is missing colors $\tau_1, \tau_2, \cdots, \tau_s$ in $\mathcal{N}[v_i]$ given by $\Phi_{i\tau} = 0\ \forall \tau \in \{\tau_1, \cdots, \tau_s\}$, and v_j is missing $\kappa_1, \kappa_2, \cdots, \kappa_s$ in $\mathcal{N}[v_j]$, given by $\Phi_{j\kappa}\ \forall \kappa \in \{\kappa_1, \cdots, \kappa_s\}$. In this case, the $v_i \sim v_j$ edge will change $2s$ zero entries in the Φ matrix to ones. In any other case, i.e., v_i has at least one of the $\tau_1, \tau_2, \cdots, \tau_s$ colors in its closed neighborhood or v_j has at least one of the $\kappa_1, \kappa_2, \cdots, \kappa_s$ colors in $\mathcal{N}[v_j]$, the number of zeros in Φ that will be converted to 1 will be less than $2s$. Thus, $\left\lceil \frac{z(\Phi)}{2s} \right\rceil \leq \mathcal{E}$.

The upper bound is straight forward as $\Phi_{i\tau} = 0$ means that v_i is missing a color τ in $\mathcal{N}[v_i]$, and the color τ can always be made available in $\mathcal{N}[v_i]$ through the addition of a single edge $v_i \sim v_j$, where v_j is any vertex with color τ. ■

As an example, consider G shown in Fig. 2, with a given labelling of the nodes. Here, each node can have at most two labels from a set of five labels, given by $\{1,2,3,4,5\}$. The corresponding C and Φ matrices are,

$$C = \begin{pmatrix} 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \end{pmatrix} \qquad \Phi = \begin{pmatrix} 1 & 1 & 1 & 1 & 2 \\ 2 & 2 & 1 & 1 & 2 \\ 1 & 1 & 2 & 2 & 2 \\ 1 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 \end{pmatrix}$$

Since $\Phi_{43} = \Phi_{51} = 0$, v_4 is missing label 3 in $\mathcal{N}[v_4]$ and v_5 is missing label 1 in its closed neighborhood. By adding $\mathcal{E}$ number of edges, where $1 \leq \mathcal{E} \leq 2$ (by Theorem 3.2), a $(5,2)$-configuration of G can be obtained. Note that by adding a single edge, $v_4 \sim v_5$, we get a $(5,2)$-configuration, where every node has set of five distinct labels in its closed neighborhood.

A. Redundant Edges

In dynamic multiagent systems, edges may be lost. These edge deletions may take away certain resources from

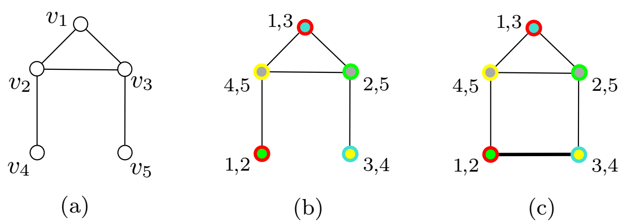


Fig. 2. (a) A graph G. (b) Labelling of the nodes, where v_1 is assigned the labels 1 and 3, v_2 is assigned the labels 4 and 5, and so on. (c) $v_4 \sim v_5$ edge is needed as label 3 is missing from $\mathcal{N}[v_4]$ and label 1 is missing from $\mathcal{N}[v_5]$.

the neighborhood of an agent, thus affecting the (r, s)-configuration of the underlying graph. So, we need to characterize all such edges whose deletion is not critical in the sense that their removal will preserve the number of resources available in the neighborhood of any agent. Let us define the *deficiency of a node* v in a network as the number of colors from a coloring set $\{1, 2, \cdots, r\}$ missing in $\mathcal{N}[v_i]$. Similarly, the *deficiency of a network* is the sum of all node deficiencies. Now, based on this notion, we can define a *redundant edge* to be one whose deletion does not increases the deficiency of a network.

If $\Phi_{ij} = 1$, it means that v_i has only one neighbor with color j, and thus, an edge between v_i and that j colored node is *not* redundant. Similarly, $\Phi_{ij} > 1$ will imply that v_i has more than one node with color j in $\mathcal{N}[v_i]$. So, there may be a redundant edge between v_i and some of its neighbors.

Theorem 3.3: Let v_i be a node with colors $\kappa_1, \kappa_2, \cdots, \kappa_s$, and v_j be its neighbor with colors $\tau_1, \tau_2, \cdots, \tau_s$. An edge $v_i \sim v_j$ is redundant if and only if $\Phi_{i\tau_1}, \Phi_{i\tau_2}, \cdots, \Phi_{i\tau_s}$ and $\Phi_{j\kappa_1}, \Phi_{j\kappa_2}, \cdots, \Phi_{j,\kappa_s}$, are all greater than 1 at the same time.

Proof: ($\Leftarrow$) Let $v_i \sim v_j$ be a redundant edge. Then, by definition, it means that v_i has at least two neighbors for each of the colors $\tau_1, \tau_2, \cdots, \tau_s$ in $\mathcal{N}[v_i]$, i.e. $\Phi_{i\tau_1}, \Phi_{i\tau_2}, \cdots, \Phi_{i\tau_s}$ are all greater than 1. Similarly, for v_j , the redundancy of a $v_i \sim v_j$ edge implies that for each of the colors, $\kappa_1, \kappa_2, \cdots, \kappa_s$, vertex v_j has at least two neighbors in $\mathcal{N}[v_j]$, implying that $\Phi_{j\kappa_1}, \Phi_{j\kappa_2}, \cdots, \Phi_{j,\kappa_s}$ are all greater than 1.

($\Rightarrow$) Now assume $v_i \sim v_j$ is not redundant, then at least one of the following is true.

(a) there exists a $\tau \in \{\tau_1, \tau_2, \cdots, \tau_s\}$, such that v_i has only v_j as a τ colored vertex in $\mathcal{N}[v_i]$, i.e., $\Phi_{i\tau} = 1$ for some $\tau \in \{\tau_1, \tau_2, \cdots, \tau_s\}$.

(b) there exists a $\kappa \in \{\kappa_1, \kappa_2, \cdots, \kappa_s\}$, such that v_j has only v_i as a κ colored vertex in $\mathcal{N}[v_j]$, i.e., $\Phi_{j\kappa} = 1$ for some $\kappa \in \{\kappa_1, \kappa_2, \cdots, \kappa_s\}$.

In both cases, $\Phi_{i\tau_1}, \Phi_{i\tau_2}, \cdots, \Phi_{i\tau_s}$ and $\Phi_{j\kappa_1}, \Phi_{j\kappa_2}, \cdots, \Phi_{j,\kappa_s}$, are not all greater than 1 simultaneously, proving the required result. ∎

Consider again the example shown in Fig. 2. Here, v_2 has labels $4, 5$, and v_3 has labels 2 and 5. Also, note that in the color distribution matrix, $\Phi_{22}, \Phi_{25}, \Phi_{34}$ and Φ_{35} are all greater than 1. Thus, by Lemma 3.3, the $v_2 \sim v_3$ edge is redundant and its deletion is not increasing the deficiency of any node in the network, as shown in Fig. 3.

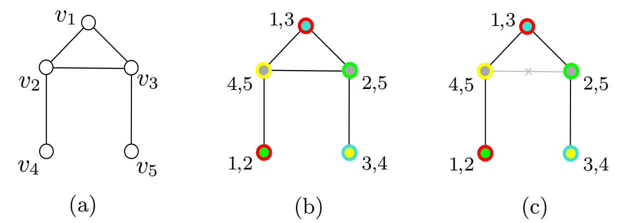


Fig. 3. (a) A graph G. (b) Labelling of the nodes of G. (c) $v_2 \sim v_3$ edge is redundant. Removing this edge will not increase the deficiency of any node.

IV. SUFFICIENT CONDITION AND A LABELLING SCHEME FOR AN (r, s)-CONFIGURATION

The underlying network topology of a system determines the number of various capabilities or resource types that can be incorporated within a system under the constraint that every node v can find every resource type in $\mathcal{N}[v]$. The domatic number is the maximum number of disjoint dominating sets in a graph. Thus, under the restriction that every node can have only one resource type, the maximum number of resource types that can be distributed in the network is the domatic number of the underlying graph. In other words, the maximum value of r in an $(r, 1)$-configuration is the domatic number of the graph. Thus, a graph with a domatic number of at least γ, always has an (r, s)-configuration for $r = s\gamma$. However, there are many graphs with $dom(G) = \gamma$ that have (r, s)-configurations for $r = s\gamma + 1$. For example, cycle graphs C_n, where n is not a multiple of 3 have $dom(C_n) = 2$, but they still have a $(5, 2)$-configuration. Thus, the structure of the network can be used to incorporate more heterogeneous resources. Here, we present a sufficient condition for a graph with domatic number γ to have an (r, s)-configuration with $r = s\gamma + 1$. This will also outline a procedure to get a labelling scheme for an (r, s)-configuration, for $r = s\gamma + 1$.

Firstly, we define some terms that will be used to prove Theorem 4.1, which is the main result of this section.

Definition 4.1: (Minimal Partition of G): Let G be a graph with domatic number γ and vertex set V. A *minimal partition* of G, denoted by Π, is a partitioning of V into $\gamma + 1$ disjoint sets such that,

$$\Pi = D_1 \cup D_2 \cup \cdots D_\gamma \cup V_\Pi \tag{2}$$

where D_i is a minimal dominating set, $\forall i \in \{1, 2, \cdots, \gamma\}$, and $V_\Pi = V - (\cup_{i=1}^{\gamma} D_i)$ is the set of vertices that are not included in any minimal dominating set D_i. ∎

We term V_Π in (2) as the set of *non-critical vertices* with respect to a minimal partition Π, and we note that $V_\Pi \cap (\cup_{i=1}^{\gamma} D_i) = \emptyset$.

Now, consider a minimal partition Π of G and let $D_{\gamma+1}$ be a dominating set such that $V_\Pi \subseteq D_{\gamma+1}$. Since $dom(G) = \gamma$ and V_Π is not a dominating set, we have

$$D_{\gamma+1} = V_\Pi \cup I_\Pi$$

where $I_\Pi \subset (\cup_{i=1}^{\gamma} D_i)$. We term a set I_Π with the smallest cardinality, a set of *common vertices* with respect to a minimal partition Π.

The notions of minimal partition, Π, set of non-critical vertices with respect to a minimal partition Π, and a set of common vertices with respect to Π are shown in Fig. 4.

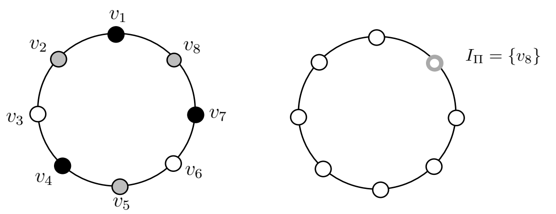


Fig. 4. A cycle graph, C_8 having a domatic number $\gamma = 2$. A minimal partition $\Pi = D_1 \cup D_2 \cup V_\Pi$, where $D_1 = \{v_1, v_4, v_7\}$ and $D_2 = \{v_2, v_5, v_8\}$ are minimal dominating sets, and $V_\Pi = \{v_3, v_6\}$ is the set of non critical vertices with respect to Π. We can take another dominating set D_3 as $D_3 = V_\Pi \cup I_\Pi$, where $I_\Pi = \{v_8\}$ is a set of common vertices with respect to a minimal partition Π.

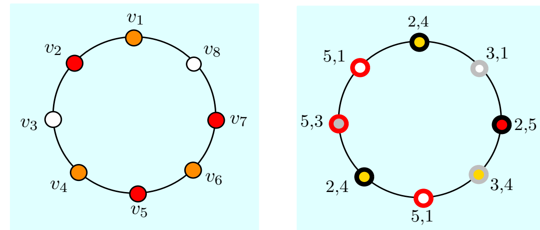
Theorem 4.1: Let G be a graph with domatic number γ. Let Π be a minimal partition of G and I_Π be a set of common vertices with respect to Π. If there exists another minimal partition of G, say $\tilde{\Pi} \neq \Pi$, such that $I_\Pi \subseteq V_{\tilde{\Pi}}$, where $V_{\tilde{\Pi}}$ is the set of non-critical vertices with respect to $\tilde{\Pi}$, then G has an (r, s)-configuration with $r = s\gamma + \lfloor \frac{s}{2} \rfloor$.

Proof: Let $\Pi = \bigcup_{i=1}^{\gamma} D_i \cup V_\Pi$, where V_Π is the set of non-critical vertices with respect to a minimal partition Π. Also, let $D_{\gamma+1}$ be a dominating set with $D_{\gamma+1} = V_\Pi \cup I_\Pi$, where I_Π is a set of common vertices with respect to Π. Assign $\lfloor \frac{s}{2} \rfloor$ distinct labels to all the vertices in a dominating set D_i, for every $i \in \{1, 2, \cdots, \gamma + 1\}$[2]. Under this labelling scheme, the vertices in I_Π will have $\left(2 \lfloor \frac{s}{2} \rfloor\right)$ distinct labels as they are included in two different dominating sets, including $D_{\gamma+1}$ and some other D_i for $i \in \{1, 2, \cdots, \gamma\}$. Note that the vertices in I_Π are the only ones with $\left(2 \lfloor \frac{s}{2} \rfloor\right)$ labels. Also, every $v \in V$ has the set of $\lfloor \frac{s}{2} \rfloor (\gamma + 1)$ labels in its closed neighborhood.

Now, consider another minimal partition of G, $\tilde{\Pi} = \bigcup_{i=1}^{\gamma} S_i \cup V_{\tilde{\Pi}}$, with $V_{\tilde{\Pi}}$ being the set of non-critical vertices with respect to $\tilde{\Pi}$, and each S_i being a minimal dominating set. Let $\tilde{\Pi}$ be such that $I_\Pi \subseteq V_{\tilde{\Pi}}$. It means that every vertex in $V - V_{\tilde{\Pi}}$ has $\lfloor \frac{s}{2} \rfloor$ labels. Since, $S_i \subseteq (V - V_{\tilde{\Pi}})$ for any $i \in \{1, 2, \cdots, \gamma\}$, every vertex $v \in S_i$ has $\lfloor \frac{s}{2} \rfloor$ labels. Now for every S_i, assign $\lceil \frac{s}{2} \rceil$ more unique labels to each vertex in S_i. Since each S_i is a dominating set, every $v \in V$ has a set of $\lceil \frac{s}{2} \rceil \gamma$ unique labels in $\mathcal{N}[v]$. Noting that $\lfloor \frac{s}{2} \rfloor (\gamma + 1)$ unique labels are already available in the closed neighborhood of every vertex, we get that all the vertices in V have now $\lfloor \frac{s}{2} \rfloor (\gamma + 1) + \lceil \frac{s}{2} \rceil \gamma = s\gamma + \lfloor \frac{s}{2} \rfloor$ distinct labels in their closed neighborhoods. Since each vertex is assigned at most s distinct labels, we have an (r, s)-configuration of G with $r = s\gamma + \lfloor \frac{s}{2} \rfloor$. ∎

[2] $\lfloor \frac{s}{2} \rfloor$ labels assigned to the vertices of D_i are different from the ones assigned to the vertices in D_j where $i \neq j$.

As an example, consider a $(5, 2)$-configuration of C_8. Since the domatic number of C_8 is 2, let us take $\gamma = 2$. We consider two minimal partitions of C_8, denoted by Π and $\tilde{\Pi}$ respectively. We take Π as shown in Fig. 4. For $\tilde{\Pi}$, we take $\tilde{\Pi} = S_1 \cup S_2 \cup V_{\tilde{\Pi}}$, as shown in Fig. 5(a). Since $I_\Pi \subseteq V_{\tilde{\Pi}}$, (r, s)-configuration exists for C_8, where $r = 5$ if we take $s = 2$.



(a) (b)

Fig. 5. (a) $\tilde{\Pi} = S_1 \cup S_2 \cup V_{\tilde{\Pi}}$, where $S_1 = \{v_1, v_4, v_6\}$ and $S_2 = \{v_2, v_5, v_7\}$ are disjoint minimal dominating sets, while $V_{\tilde{\Pi}} = \{v_3, v_8\}$ is the set of non-critical vertices with respect to $\tilde{\Pi}$. (b) A $(5, 2)$-configuration of C_8 is shown, where each vertex has two labels from a set of five labels $\{1, 2, 3, 4, 5\}$.

V. MULTIPLE RESOURCE ASSIGNMENT IN R-DISK GRAPHS

The R-disk proximity graph model is frequently employed to model inter-connections among nodes in multi agent networks. In such a model, a disk of radius R is associated with every node v that lies at the center of the disk. This disk represents the interaction range of a node, which is assumed to be same for all the nodes. A node forms an edge with others if and only if they exist within that R radius disk of the node [12]. Applications of such a model include ad hoc communication networks, wireless sensor networks (e.g. [18]), multi agent and multi robot systems (see e.g., [15]), and other broadcast networks with a limited range transmitters and receivers, to name a few.

The analysis of a $(5, 2)$-configuration of R-disk proximity graphs is of significance, particularly in the context of heterogeneous multiagent systems. Here, we show that R-disk graphs have a $(5, 2)$-configuration under certain mild conditions. It is assumed that the agents equipped with multiple capabilities or resources are lying in a plane, and the interactions among them are modelled by the R-disk proximity graph model.

We start by translating the geometric property of such graphs into a graph-theoretic one by first defining the following special graphs. A graph G is a *complete bi-partite graph* if there exists a partition of its vertex set $V = X \cup Y$, such that an edge $u \sim v$ exists whenever $u \in X$ and $v \in Y$. If $\mid X \mid = x$ and $\mid Y \mid = y$, then a complete bi-partite graph is denoted by $K_{x,y}$. Examples are shown in Fig. 6. We also define a *double cycle graph*, denoted by $C_4 \bullet C_4$, as the one obtained by identifying a vertex of C_4 with a vertex of another C_4, as shown in Fig. 6. Also, a gaph G is said to be an *H-free graph*, if H is not an induced subgraph of G.

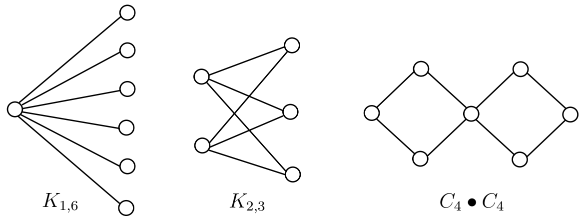


Fig. 6. Complete bi-partite graphs, $K_{1,6}$ and $K_{2,3}$. The double cycle graph, $C_4 \bullet C_4$, obtained by identifying a vertex of a C_4 with a vertex of another C_4.

It is shown in [17] that $K_{2,3}$ cannot be an R-disk graph. In the following Lemma, it is shown that R-disk graphs are always $K_{1,6}$-free.

Lemma 5.1: An R-disk proximity graph is $K_{1,6}$ free.

Proof: Let $G(V, E)$, be an R-disk proximity graph. Let $v \in V$ such that $\mathcal{N}(v) = \{v_1, v_2, \cdots, v_p\}$, where $p \geq 6$. Also, let $\theta_{(v_i v v_j)}$ be the angle v makes with v_i and v_j, as shown in Fig. 7. If $\|v_i, v_j\|$ is the euclidean distance between the nodes v_i and v_j, then it is easy to see that $\|v_i, v_j\| > R$, whenever $\theta_{(v_i v v_j)} > 60^o$. Thus $v_i, v_j \in \mathcal{N}(v)$ are non-adjacent if and only if $\theta_{(v_i v v_j)} > 60^o$. For G to have $K_{1,6}$ as an induced subgraph, there must be a subset $\tilde{\mathcal{N}}(v) \subseteq \mathcal{N}(v)$, with $\mid \tilde{\mathcal{N}}(v) \mid = q \geq 6$, such that $\theta_{(x_i v x_j)} > 60^o$, $\forall x_i, x_j \in \tilde{\mathcal{N}}(v)$. But this will give $\sum_{i=1}^{q-1} \theta_{x_i v x_{(i+1)}} + \theta_{x_q v x_1} > 360^o$, which is not possible. Thus an R-disk graph is $K_{1,6}$-free. ■

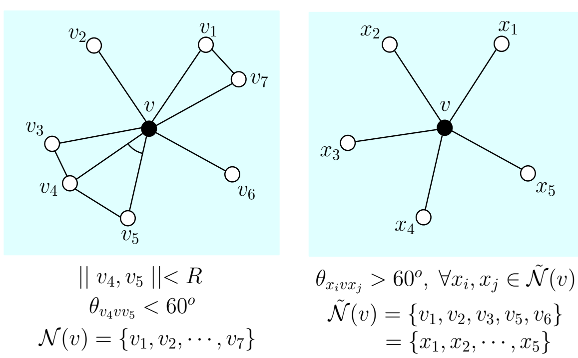


Fig. 7. An R-disk graph can never have $K_{1,6}$ as an induced subgraph.

A result regarding a $(5, 2)$-configuration of $K_{1,6}$-free graphs has been recently reported in [16].

Theorem 5.2: [16] A $K_{1,6}$-free graph G with a minimum degree of at least two has a $(5, 2)$-configuration, whenever G is not $C_4, C_7, K_{2,3}$ or $C_4 \bullet C_4$.

An $O(n^2)$ algorithm is also provided in [16] to achieve a $(5, 2)$-configuration of a graph, if it exists. Using Theorem 5.2, Lemma 5.1, and the fact that an R-disk graph can never be a $K_{2,3}$ graph, we get the following result directly,

Theorem 5.3: An R-disk proximity graph G with a minimum degree of at least 2 has a $(5, 2)$-configuration whenever $G \neq C_4, C_7$ or $C_4 \bullet C_4$.

VI. CONCLUSIONS

In this paper, we studied heterogeneity in multiagent systems from a network topology view point using concepts from graph theory. The notion of (r, s)-configurations of a graph is used to characterize the distribution of agents with multiple capabilities (or resources). In such a distribution, every agent can find all types of resources available in the network in its closed neighborhood. The role of individual agents and interactions in attaining (r, s)-configurations is also examined. This study not only analysed the role of network topology in the context of heterogeneous multiagent systems, but also provided ways to design network structures where agents equipped with various resources coordinate with each other to accomplish complex tasks.

REFERENCES

[1] M. Yarvis, N. Kushalnagar, H. Singh, A. Rangarajan, Y. Liu and S. Singh, "Exploiting heterogeneity in sensor networks", in *Proc. IEEE INFOCOM*, Miami, FL, March 2005, pp. 878–890.
[2] S. V. Pemmaraju, I. A. Pirwani,"Energy conservation via domatic partitions," in *Proc. 7th ACM Intl. Symp. Mobile Ad Hoc Networking and Computing*, New York, , 2006, pp. 143-154, , 2006. ACM.
[3] K. Laventall and J. Cortés, "Coverage control by robotic networks with limited-range anisotropic sensory," in *Proc. American Control Conf.*, Seattle, WA., 2008, pp. 2666–2671.
[4] Y. Wang, X. Wang, D.P. Agrawal and A.A. Minai, "Impact of heterogeneity on coverage and broadcast reachability in wireless sensor networks", in *International Conference on Computer Communications and Networks*, Arlington, VA, Oct. 2006
[5] D. A. Anisi, P. Ögren and X. Hu," *IEEE Trans. Autom. Control*, vol. 55, no. 12, pp. 2679–2691, 2010.
[6] J. Bay and T. Samad, "Multiagent Sensing and Control: Surveillance and Reconnaissance for Urban Operations", in *Intl. Symp. on Mathematical Theory of Networks and Systems*, Kyoto, Japan, 2006, pp. 891–900.
[7] S. Durocher, K. R. Jampani, A. Lubiw and L. Narayanan, "Modelling gateway placement in wireless networks: Geometric k-centres of unit disc graphs," *Comput. Geometry*, vol. 44, no. 5, pp. 286–302, 2011.
[8] P. Santi, "Topology control in wireless ad hoc and sensor networks," *ACM Computing Surveys*, vol. 37, no. 2, pp. 164–194, 2005.
[9] X.-Y. Li, "Algorithmic, geometric and graphs issues in wireless networks," in *Wireless Communications and Mobile Computing*, vol. 3, no. 2, pp. 119–140, 2003.
[10] W. Abbas and M. Egerstedt, "Distribution of Agents in Heterogeneous Multiagent Systems", in *Proc. IEEE Conf. Decision and Control*, Orlando, FL, 2011, pp. 976–981.
[11] S. Fujita, M. Yamashita and T. Kameda, "A study on r-configurations—A resource assignment problem on graphs," *SIAM J. Discrete Math.*, vol. 13, pp. 227–254, 2000.
[12] J. W. Jaromczyk and G. T. Toussaint, "Relative neighborhood graphs and their relatives," *Proc. IEEE*, vol. 80, no. 9, pp. 1502–1517, 1992.
[13] C. Löwenstein and D. Rautenbach, "Pairs of disjoint dominating sets and the minimum degree of graphs", *Graphs Combin.*, vol. 26, no. 3, 2010, pp. 407–424.
[14] T. W. Haynes, S. T. Hedetniemi and P. J. Slater, *Domination in graphs: Advanced topics*, Marcel Dekker, New York; 1998.
[15] J. Cortés, S. Martinéz and F. Bullo, "Robust rendezvous for mobile autonomous agents via proximity graphs in arbitrary dimensions," *IEEE Trans. Autom. Control*, vol. 51, no. 8, pp. 1289–1298, 2006.
[16] C. H. Liu, P. Whalen and R. Thomas,"$(5, 2)$-configuration on minimum degree at least two, $K_{1,6}$-free graphs," 36^{th} *Annual SIAM Southeastern Atlantic Section Conf.*, Huntsville, AL, Mar. 2012.
[17] E. J. van Leeuwen, "Approximation algorithms for unit disk graphs", in *Proc. IEEE Intl. Workshop on Graph Theoretic Algorithms*, Metz, France, 2005, pp. 351–361.
[18] X.-Y. Li, "Applications of computational geometry in wireless ad hoc networks," in *Ad Hoc Wireless Networking*, X.-Z. Cheng, X. Huang and D.-Z. Du, Eds. Kluwer, 2003, pp. 197–264.