Proceedings of the 2009 IEEE International Conference on Systems, Man, and Cybernetics
San Antonio, TX, USA - October 2009

Collaborative Group Assignment with Competency Trees

Wan Hok Man
Dept of Computing, The Hong Kong Polytechnic University, Hong Kong, China

Vincent TY Ng
Dept of Computing, The Hong Kong Polytechnic University, Hong Kong, China
cstyng@comp.polyu.edu.hk

***Abstract*— In collaborative commerce, task groups are often formed from experts of several companies to perform collaborative projects. The proper position assignments to different staff members from multi-organizations have become an important task for the success of projects. This paper proposes a collaborative framework with two components for companies to assign professionals to different roles in forming a group to complete a task. The first component is further divided into 2 parts. The first part uses a tree competency model to define competencies of staff (agents) as well as role requirements. The second part contains a tree similarity algorithm and bonus point algorithm to support the role assignment. The adoption of clustering techniques and negotiation protocol between team requester and multi-agent providers are the second component to support the group role assignments. Preliminary experiments have been performed with some interesting results.**

Keywords*— *Collaborative commerce, group roles assignments, human resources management, tree similarity

I. INTRODUCTION

The issue of role assignment problem is not new. Companies in the past have been doing it based on subjective judgment. Many studies lately have worked on creating a systematic methodology of assigning the right agents to the right roles [4]. In the field of computer communication networks, agents can be a set of sensor nodes and the assignment problem becomes finding the most suitable nodes to an unstable network environments [2].

One of the most obvious applications of role assignment is related to human resource systems [11]. In May 2008, the Hong Kong SAR Government has released the Qualifications Framework (QF) which has a 7–level quality or ability structure for different major occupations [6]. Since the release of the QF, government departments start to develop human resource policies and trainings with respect to the competencies described in the framework.

For the last two years, there has been a study to develop a competency model of police officers based on the QF structure. Among the many research issues, one major effort is to locate the right officer for a specific position. The problem becomes harder when it is required to set up a task team of several officers with different capabilities temporarily. Another challenge would be officers may need to come from different sections or units with special skills, while each section would have its own priority in releasing the officers. It is observed that this problem is not unique in the Police Force. In the commercial world, when there is a sizable project, companies would join together to work on the project collaboratively. Project teams from staff of different companies would be required and optimal costs locally to individual company as well as globally for the whole project are desired. Realizing this issue, we are interested to develop a collaborative framework to support group role assignment with agents (the term "agent" would also be used to represent the staff or personnel inside the company throughout this paper) from different parties. Some of the common role assignment questions in the area of human resource management are listed below.

(a) How to choose the most qualified personnel to fill in a vacancy? [10]
(b) Which models can be used to represent the competency of personnel and positions? [5]
(c) What methods can be used to determine the matching degree between personnel and position?
(d) How to select a group of people of which all of the candidates are also qualified to get the job done but with different costs?
(e) How to formulate a team (i.e. a group of agents working towards the same goal) from potential agents being to each of the positions in the team? [4]
(f) What criteria could be used in order to determine the combination of team members, such as qualification and the cost level of personnel?

With the six questions above, one can group (a), (b) and (c) into the *Simple Role Assignment* problem; while (d), (e) and (f) in the *Group Role Assignment* problem. For the simple role assignment problem, a modified tree similarity algorithm is proposed to determine whether a personnel or an agent is qualified to fulfill a specific position by calculating the "matching" score and the extent of bonus points gained by the agent when his overall qualification is better than required. The group role assignment problem can be further divided into 4 cases to be considered: i) roles are independent, ii) roles are independent but with significance rankings, iii) roles are inter-related with a common task to be fulfilled, and iv) roles are inter-related with a common task and have different rankings.

978-1-4244-2794-9/09/$25.00 ©2009 IEEE

II. RELATED WORKS

There have been a number of research works in role assignments in the past. Odell et al. [1] point out that a position's specification and description may not be constant over time. Supported by a case study, it states that these potential changes should be analyzed and classified in order to help on the role assignment decision making. Frank and Romer [2] present a role specification language and distributed algorithms for assigning sensor nodes on certain roles in the existing network. The contributions focus on getting use of role assignment concepts in computer communication networks.

Zhu et al. [4] state that role assignment is a process of agent evaluation, group role assignment and role transfer in order to maintain the role-based collaboration (RBC), such as facilitating an organizational structure. It uses the E-CARGO model to define role, agent, environment, group, role assignment, workable role and group. The requirements of a role are expressed as a vector and agents will form a capability matrix which can be applied to solve the simple, rated and weighted group role assignment problems efficiently.

Most of the previous related studies also used vectors to represent the requirements of roles and competencies of agents. Nonetheless, the subordinate and neighbor relationship between the competencies cannot be clearly demonstrated by vectors. For example, within the communication competence, there may have oral and written competences. We believe that, with the project experience of the Police Force, a hierarchical model would be better to represent the inter-relationships of competencies.

III. COLLABORATIVE FRAMEWORK

Group role assignment is not occurring alone in a single company. In collaborative commerce, companies pull their strength together to work on a sizeable project which needs expertise from different areas [3]. In this case, a project team would be set up temporarily with members from different companies as shown in Figure 1.

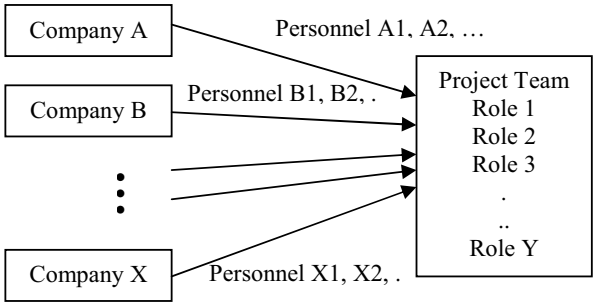

Figure 1: Collaborative framework for Project Team design

The criteria of choosing appropriate candidates in team formation can be varied by different factors. The first apparent factor is the direct matching method of the competencies of a personnel (agent) with the requirements of a job (role). Based on the requirements of the team, a human resource manager can fill up a selected role of the team by reviewing personnel profiles and repeating the process for the other roles one-by-one till completing all the assignments [14].

However, this simple method may not be optimal or possible when dealing with multi-companies. Even though companies form an alliance to work on a specific project, they are inclined to maintain their autonomy and retain a good access control of their resources [9]. Each personnel nominated would have a cost associated to the project as well as the respective company itself. Another factor is the potential of incomplete fulfillment of roles required for the team. A member is weak in communication while strong in organizing events may be able to compliment another team member whose role requires good organizational skills.

Furthermore, companies would like to nominate selected personnel as a sub-team to join the project because of their own business reasons. It is usually difficult to have a centralized body to decide who from what company partners to join the final project team. Also, during the assignment process, there can be different rounds of assignment because of personnel competencies, role requirements, and other company factors. In each round, the least appropriate personnel can be replaced in consideration of several attributes, such as cost, level of incompleteness, company priority and team experience. Negotiation in between the project requester and amongst project servers (companies) would be required.

Two parties (i.e. "requester" and "project servers - companies") can work collaboratively in order to form an optimal team to achieve the specific task that aligns with organizational strategic goals. The collaboration can be supported by a negotiation protocol. At present, we have been designing a collaborative framework on group role assignment for multi-companies. The framework has two components. The first component is a tree competency model with group role assignment algorithms. The second component is a negotiation protocol between different companies in formulating a project team. In this paper, we will mainly describe the competency tree model.

IV. COMPETENCY TREE MODEL

There are different ways to represent a role's requirements and an agent's capabilities. A common approach is to have a competency element to model a requirement or a capability. In our work, we adopt the HRXML [7] format to store personnel data and use a tree competence model for roles and agents (personnel). Apart from agents and roles, *clusters* are included to represent groups of similar roles or agents. Hence, the framework has three pieces of information, <R', A', G'>, in which R' is the set of roles/positions, A' is the set of agents/personnel and G' is the set of role or agent clusters.

Definition 1: Role. A role is defined as R :: = <n, C, W, L> with a tree model. R is consisted of:

- n: the identification of a role;

- C: a set of competency elements that the role required;
- W: a set of importance weightings for competencies of the role. They are in the range of [0,1].
- L: a set of level requirements of each competency for the role. They are in the range of [1,7]

The root node of a role tree has the name of "Role Competency" and followed by subordinate competency elements/nodes. Each node can have subordinate competency nodes and has two attributes, importance weighting and level requirement. An example of a role tree is shown in Figure 2.

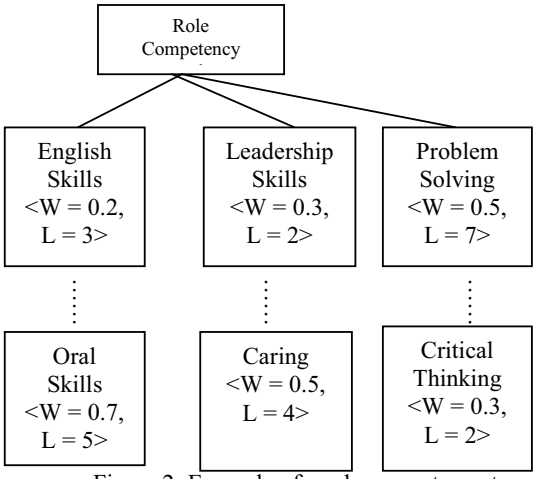

Figure 2: Example of a role competency tree

(Does it mean the SUM of the weightings of children with the same parent equal to 1?)

Definition 2: Agent. An agent is defined as A ::= <n, s, C, L> with a tree model, where

- n: the identification of the agent
- s: the cost level
- C: a set of competencies that the agent possesses
- L: a set of competency levels of the agent which are bounded to [1,7];

Both role and agent trees also contain the range from 1 to 7 to represent the level (L) of the competency so as to fit Qualifications Framework (QF)'s 7–level qualification hierarchy. The agent tree has a similar structure of a role tree except it does not have the importance weighting attribute in the nodes. The cost level is proportional to the number of competency elements/nodes that the agent has. That is, the more competencies an agent has, the higher cost level of the agent. An example of an agent tree is shown in Figure 3.

For direct agent and role matching, the computation would be expensive when dealing with multi-companies or a corporation with a large number of agents (personnel). For example, in the HK Police Force, there are more than 20,000 officers. Therefore, in order to have a faster assignment performance, preprocessing to get clusters of similar trees would be an asset. In [16], the RRSi index has been proposed to form groups of similar trees together. Here, there are two kinds of clusters, one for roles and the other for agents.

Definition 3: Cluster. A cluster is defined as G ::= <T, m>

- T: a set of role or agent competency trees within the cluster
- m: The cluster's meaning (i.e. the representative of the cluster)

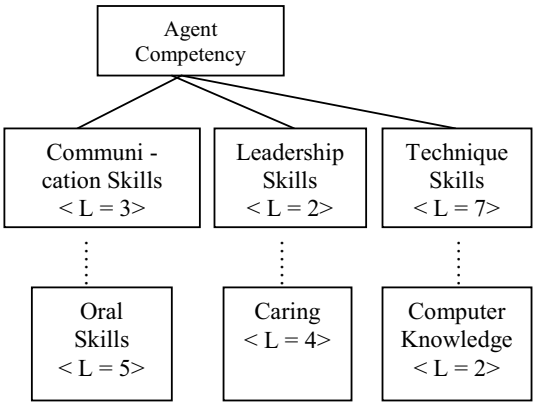

Figure 3: Example of an agent competency tree

V. THE SIMPLE ROLE ASSIGNMENT PROBLEM

We have reviewed several tree similarity calculation methods for easy adoption in HRXML [5, 12, 13]. At present the Bhavsar's method is adopted with 2 additional attributes in score calculation.

The S11RA algorithm is designed to solve the *Simple one-to-one role-agent assignment problem*. The algorithm traverses the role tree and the agent tree recursively. For each pair of corresponding nodes, it calculates the degree of matching by multiplying the importance weighting of a role, the matched level of the agent and the role, and their corresponding descendants' matching scores, recursively. The matched level is equal to the level of the agent divided by the level of the role. If the agent level is already greater than that of the role's, it returns a matched degree of value 1. The final value after executing the S11RA would be $D[P_i, Q_j] \rightarrow [0,1]$ to represent a similarity measure between role and agent competency trees, P_i, $Q_{j,}$, respectively. A value of 0 means the agent is totally unqualified to fulfill the role and 1 means the agent is totally matched or "over-qualified" the specifications of the role. Figure 4 shows the details of the S11RA algorithm where R and A are the root nodes of one role tree and an agent tree, respectively.

With S11RA algorithm, its results would indicate a set of agents suitable for a given role. However, it is also important to rank the suitable agents. Therefore, the idea of bonus point is added to return higher measures to better agents. The bonus point calculation is similar to S11RA except the node_sim expression is replaced by:

```
Bonus = matched weight *
                (agent_level - role_level)/role_level
                *  bonusPoints(R,A, i, j);
```

```
S11RA(R, A) {
  tree_sim = 0
  If (R and A do not match) return
  match_level = max(1, level_A/level_R)
  tree_sim = weight_R * match_level
  node_sim = 0
  if  (both R and A have child nodes) {
     Let P are the child nodes of R
     Let Q are the child nodes of A
     For each pair of nodes matching in P and Q {
          Let i be the node for the pair from P
          Let j be the node for the pair from Q
          node_sim = node_sim + S11RA(i,j)
          }// end for loop
   }
  tree_sim = tree_sim *((1- β ) +  β * node_sim)
  return tree_sim
```

Figure 4: The S11RA algorithm

VI. COMPETENCY TREES CLUSTERING

Clustering of roles or agents can bring different advantages. One application is to speed up the assignment performance when there are many agents present. The other application is to identify common patterns within a cluster for gap analysis or progression analysis. For our initial work, the single linkage clustering technique is adopted. It is also known as nearest neighbour technique, in which the distance between groups is defined as the closest pair of records from each group [8].

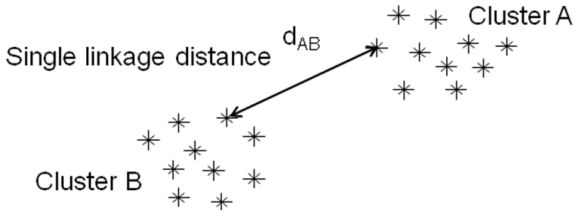

Figure 5. Single linkage clustering

The Competency Clustering (CC) algorithm has been developed to support the role assignments. The input is a set of agent competency trees or role competency trees. The output is a set of clusters and each cluster has grouped the most similar agents or roles within. The initial step of CC is to create a matrix of size n x n for n tree given. The S11RA algorithm between different pairs of trees is performed and the returned values are entered as an entry in the matrix. However, the node_sim calculation in S11RA algorithm is revised for clustering role competencies trees as shown below.

```
node_sim =minimum_weight * matched level *
                S11RA ( i, j,)
```

Note that the minimum_weight is equal to the minimum weight of two trees (i.e. matched weight).

The second step is to find the maximum similarity measures inside the matrix. The found measures are used to combine the trees or group of trees and a smaller matrix would be resulted. Similarity measures of the trees in the new matrix would be computed again for merging afterwards. The merging step is repeated until a termination criterion is reached. Figure 6 shows the two iterations of tree grouping.

	Tree1	Tree2	…..	TreeN
Tree1	1	**0.7**	.	0.4
Tree2	**0.7**	1	.	0.5
…..	.	.	.	.
TreeN	0.4	0.5	.	1

⇩

	Tree1, Tree2	…. ..	TreeN
Tree1, Tree2	1	.	0.5
……..	.	.	.
TreeN	0.5	.	1

Figure 6. Two iterations of Tree Merging

For typical clustering algorithms, they require a user to predefine the number of the clusters before performing the process. Here, we use a predined similarity thresold and compare whether the maximum similarity measures is lower than the thresold before each merging step of the CC algorithm for the termination criterion An input tree of high similarity to the existing cluster would be assigned to that specific cluster; otherwise, a new cluster will be created for it [9]. After forming clusters, their means are found with the CCM (Competence Cluster Mean) algorithm, as shown in Figure 7, as representatives of their respective clusters.

CCM(G) {

- Create a temporary tree
- Find out the common nodes of first two members in G and perform one of the following approaches for the corresponding nodes:
  - Take the minimum values
  - Take the maximum values
  - Take the average values
- Store the resultant nodes with the computed values into the temporary tree.
- Repeat the two steps above, but this time performs them with the temporary tree and the 3rd member in G, and so on, until all members in G are processed.
- Return the temporary tree as the representative tree of G

}

Figure 7. The CCM algorithm

VII. THE GROUP ROLE ASSIGNMENT PROBLEM

In the police force, there are different sections and officers in these sections possess different competencies. When an ad-hoc assignment or task is required, officers are asked to form a team from different sections. The scenario is similar to a multi-company project in which personnel from different offices are required. There are several criteria which can affect the decision of assignments, (i) individual matching of qualification between agents and roles, (ii) The extent of "over qualification" of an agent to a role and (iii) cost level of agents of different parties. With these considerations, we divide the group role assignments into the following cases:

1. Roles are independent;
2. Roles are independent but with significance rankings,
3. Roles are inter-related
4. Roles are inter-related and have different ranking requirements**.**

During group matching, an individual role may not able to find any agent (staff) for complete satisfaction or even over a threshold requirement. Rather failing the group formation, the requirement can be relaxed for allowing multiple agents to fulfill a set of role requirements. For example, agent A can possess some competencies that agent B is lacking to satisfy the intended role of B. With this relaxation, a better chance of group matching results and group members can be complimentary to each others. Within a team, a leader and a few sub-leaders are often needed to formulate a hierarchy of communication and authority. Therefore, a new attribute is included in defining a role, *rank*, to indicate the significance ranking required.

For case 1, it is considered that every role has to be completely satisfied by an agent. It is described as role independence. The proposed method is to group the number of agent competency trees into different clusters by the CC algorithm. Next, the cluster means of each agent cluster (i.e. the similar pattern or the representative of the cluster) are obtained. The role matching can then be done by calculating the tree similarity between the corresponding role tree to each cluster mean. At the same time, the bonus gained of each agent cluster mean to that role is calculated. When the cluster is identified, the process is repeated for the agents within the cluster. The final assignment decision is made by finding the agent which has the highest tree similarity and gained bonus. The steps are repeated for every role until all the assignments are completed. Figure 8 shows the case 1 (GRA1N) algorithm.

```
GRA1N {
  Let A be all the agents and R is the required roles
  G = CC(A)
  M = CCM (for each g in G)
  For each r in R {
    D = {d | s in M and d = S11RA (r,s) }
    g = {g | max(D) and m = CCM(g) }
    B = (b | b = Bonus_point(r,a) and (a in g))
    C = C ∪ {a | max(b) and (a in g)}
  }
  Return C
}
```

Figure 8. The GRA1N algorithm

The difference is the second case is to calculate the tree similarity between roles with ranking considerations. The assignment is similar to the first case but have the role ranking ordered in a descending order.

The third case relaxes the role requirements and allows multiple agents to compliment each other. Since there are many possible combinations for a feasible assignment solution. The decision can be based on the following choices:

i. Most qualified approach: Find the combination with the sums of the highest tree similarity scores and gained bonus points. Ie. find the most qualified personnel cluster to fit the team.
ii. Most cost-saving approach: Use the most cost-effective personnel cluster to fit the team – i.e. select the personnel cluster have minimum resource requirements
iii. Balancing approach: Combine the benefits of both Choice 1 and 2. Select the personnel cluster with sufficient qualification (e.g. 2nd highest of the similarity degree and gained bonus points) and acceptable cost level (e.g. 2nd highest of the salary and benefits level).

The forth case is the same as the third but the decisions start from the highest ranking team to the lowest one.

VIII. EXPERIMENTS

A series of experiments has been performed to try out the assignment methods. The algorithms are implemented by JSP on a PC. Based on some role requirements from a company, randomly generated datasets of roles and agents are created.

The initial experiment used the S11RA algorithm to test for the validity of the tree similarity. A dataset of 12 agents was generated and a role tree is edited based on the second agent. The result showed a close matching as indicated in Figure 9. Agent 11 has been reviewed and compared with the role tree which showed the difference of competencies is small also.

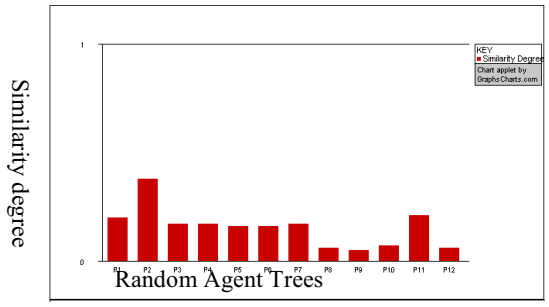

Figure 9. Simple Role Assignment Experiment

Another experiment was for the second case of the group role assignment with 50 agents and 3 roles. The work is focused on the qualification of the agent (i.e. highest similarity and bonus point degree). The results are shown in Figure 10 – Figure and the assignments are done in three iterations. The similarities and gained bonus for role 1 are shown in Figure 10 and Figure 11, respectively. Role 1 is assigned in Figure 12, and role 2 and 3 are in Figure 13 and 14, respectively. Hence, the final group role assignments for the team would be:

- Role 1: Agent 1 or 5
- Role 2: Agent 3
- Role 3: Agent 7

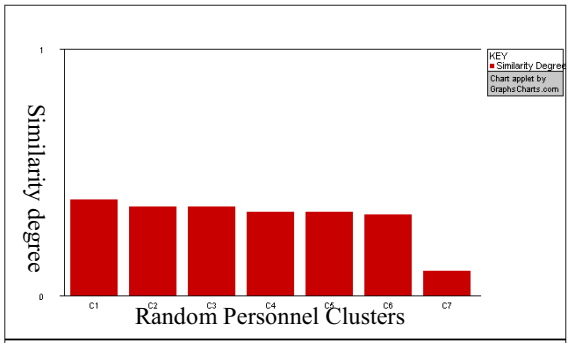

Figure 10. Group Role Assignment – Case 2a - Similarity

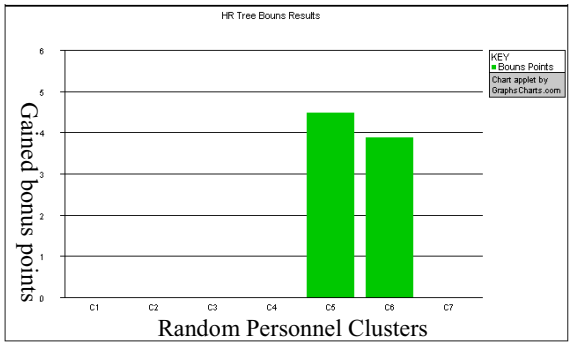

Figure 11. Group Role Assignment – Case 2a (gained bonus point)

However, if we just want to form a team in the most cost-saving way (i.e. lowest cost level, which is determined by the number of competencies that the agent have), from Figure 12-14, the group role assignment would be:

- Role 1: Agent 4
- Role 2: Agent 7
- Role 3: Agent 2, 3 or 5

If both qualification and cost level of agents are considered, the group role assignment would be changed to:

- Role 1: Agent 5
- Role 2: Agent 3
- Role 3: Agent 7

In summary, different criteria on the formation of a team can lead to different team combinations. Our framework can provide flexibility of manager to request a project team based on different role profiles (e.g. cost level and qualification of the agent).

Agent Cluster	Tree Similarity Degree	Gained Bonus Points	Cost Level
Cluster 1	**0.39**	0.0	37
Cluster 2	0.36	0.0	18
Cluster 3	0.36	0.0	18
Cluster 4	0.34	0.0	**17**
Cluster 5	0.34	**4.47**	18
Cluster 6	0.33	3.87	19
Cluster 7	0.1	0.0	**17**

Figure 12. Group Role Assignment – Case 2a (first role)

Agent Cluster	Tree Similarity Degree	Gained Bonus Points	Cost Level
Cluster 1	0.25	1.37	37
Cluster 2	0.33	1.22	18
Cluster 3	**1.0**	**3.39**	18

Cluster 4	**1.0**	1.45	**17**
Cluster 5	**1.0**	1.57	18
Cluster 6	**1.0**	0.0	19
Cluster 7	**1.0**	1.16	**17**

Figure 13. Group Role Assignment – Case 2a (second role)

Agent Cluster	Tree Similarity Degree	Gained Bonus Points	Cost Level
Cluster 1	0.19	0.0	37
Cluster 2	0.23	0.0	18
Cluster 3	0.23	0.0	18
Cluster 4	0.29	6.67	**17**
Cluster 5	0.22	6.67	18
Cluster 6	0.22	1.25	19
Cluster 7	**0.8**	**10**	**17**

Figure 14. Group Role Assignment – Case 2a (third role)

IX. CONCLUSTIONS

The correct assignments of positions to different staff members from multi-organizations have become an important task for the success of projects. This paper has introduced a collaborative framework with two components for companies to assign professionals to different roles in forming a group to complete a task. The first component is discussed in details on role and agent modeling, similarity calculations and assignment matching. Based on the requirements of the team, a HR manager can assign personnel by reviewing each agent 's profile and perform matching in several rounds between different roles in order to find out the most optimal team combination. Hence, two parties (i.e. "requester" and "service providers") can work collaboratively in order to form an optimal team to achieve the specific mission that aligns with organizational strategic goals. For more accurate of the assignment decision, more criteria can be concerned and applied into the definition of competencies model for agents (e.g. past performance) and roles requirements (e.g. Optimal Attitude). The dynamics of comparison of tree similarity algorithm to different organization and further control, guidance and measures of negotiation protocol between team requester and multi-agent providers can also be extended to our future works.

REFERENCES

[1] J.J. Odell, H. Van Dyke Parunak, S. Brueckner and J.Sauter, "Changing Roles: Dynamic Role Assignment", Journal of Object Technology, vol. 2, no. 5, September-October 2003, pp. 77-86.

[2] Christian Frank, Kay Romer, "Algorithms for Generic Role Assignment in Wireless Sensor Networks", 3*rd ACM Conference on Embedded Networked Sensor Systems (SenSys 2005), San Diego, 2-4 November 05*

[3] K. Nakatani, TT Chuang. "A Framework for the Development of Collaborative Commerce Applications", Issues in Information Systems, vol. 4, no. 2, 2005, pp. 3-9.

[4] Haibin Zhu, Senior Member, IEEE and Rob Alkins, "Group Role Assignment", Department of Computer Science and Mathematics, Nipissing University. Journal of Software, vol. 3, no. 6, June 2008, pp.69-78

[5] Virendar C. Bhavsar, Harold Boley and Lu Yang, "A Weighted –Tree Similarity Algorithm for Multi-Agent Systems in E-Business Environments", The 16th Canadian Conference on Artificial Intelligence AI 2003.

[6] Official website of The Hong Kong Qualification Framework http://www.hkqf.gov.hk/

[7] Chuck Allen, Lon Pilot, "HR-XML Enabling Pervasive Hr e-Business", XML europe 2001

[8] Kazuyuki Numata, Hideo Bannai, Yoshinori Tamada, Michiel de Hoon, Seiya Imoto and Satoru Miyanom, "Memory-Efficient Clustering Algorithms for Microarray Gene Expression Data", 2005

[9] E.Y. Li, T.C. Du, J.W. Wong. "Access control in Collaborative Commerce", Decision Support Systems, 43, 2007, pp. 675-685.

[10] Chen-Fu Chien a,*, Li-Fei Chen, "Data mining to improve personnel selection and enhance human capital: A case study in high-technology industry", Pergamon Press, Inc., Tarrytown, NY, USA, January 2008, pp.280-290

[11] Ernst Biesalski, "Knowledge Management and e-Human Resource Management", Workshop on Knowledge and Experience Management, FGWM 2003.

[12] Rui Yang, Panos Kalnis and Anthony K.H. Tung, "Similarity Evaluation on Tree – Structured Data", SIGMOD Conference 2005: pp.754-765

[13] Jin, J., Sarker, B.K., Bhavsar, V.C., Boley, H., and Yang, L., "Towards a Weighted-Tree Similarity Algorithm for RNA Secondary Structure Comparison", IEEE Computer Society Washington, DC, USA 2005, pp.639-644

[14] Haibin Zhu, Rob Alkins and Matthew Grenier, "Role-Based Chatting", Journal of Software, vol. 3, no. 6, June 2008, pp.69-78

[15] Zhu, H. and Zhou, M.C., "Role-Based Collaboration and its Kernel Mechanisms", *IEEE Trans. on Systems, Man and Cybernetics, Part C,* vol. 36, no. 4, July 2006, pp. 578- 589

[16] K. L. Ng and T. Y. Ng., "RRSi: Indexing XML Data for Proximity Twig Queries", KAIS, Vol 17, issue 2, November 2008, pp. 1930-216