Linear Programming and Genetic Algorithms Methods for Creation of Groups in Networks of Excellence

G. Guttman, M. Shpitalni (1)
Laboratory for CAD and LCE, Faculty of Mechanical Engineering
Technion - Israel Institute of Technology
32000 Haifa, Israel

Abstract
Creation of groups in Networks of Excellence (NoEs) based on knowledge mapping and expertise is a set covering problem known to be non-polynomial. Therefore it is usually approached by heuristic methods which yield good but not necessarily optimal coverage. Selecting teams to form a group within NoEs that are comprised of tens of teams can also be formulated and solved as an integer linear programming (ILP) problem whose solution is guaranteed to be optimal. This paper presents the ILP solution for team selection with typical objective functions. Several genetic algorithm-based methods are also compared to the optimal solution in terms of convergence (time and solution). The compared methods differ in selecting next-generation population schemes. The plain vanilla method is shown to be superior to both the roulette-based and the SUS methods.

Keywords: Knowledge management, Linear programming, Genetic algorithm

1 INTRODUCTION

Set covering problems as well as integer programming are non-polynomial (NP) problems, usually with exponential order of complexity [1]. For n binary (decision) variables the complexity order is 2^n. Advances in information technologies, such as computing speed, have dramatically enhanced the ability of computers to deal with NP problems, particularly with ILP (integer linear programming) problems. Common dedicated software packages and even general spreadsheets such Microsoft Excel's solver can deal with tens of binary decision variables. For many practical problems this is a reasonable number, so that ILP can replace the need for heuristic AI (artificial intelligence)-based solutions. Yet, in the majority of the cases, ILP cannot eliminate the need for heuristic algorithms.

These advances in information technologies, among them increased computing speed, communications and Internet, along with innovations in production systems have dramatically changed processes and engineering practices related to product development [2]. Today, an increasing number of projects, for instance multi-national cooperative projects, require cooperation among many teams of participants. Networks of excellence are a good example of such collaboration because they are specifically intended to overcome fragmentation of a particular research topic by networking resources and expertise. The teams in a network usually have different but overlapping fields of expertise. To accommodate a joint project, all the expertise needed for the project must be covered by the teams when forming a task-oriented group. The question is thus: how to select a set of teams to form a group that can optimally supply all the expertise needed for the project. Creating such groups and achieving the desired coverage constitutes a difficult, NP-problem that can become almost intractable very fast. To create the group, namely, to select research teams according to the knowledge and expertise required, an agreed-upon set of terms must be used to enable the algorithms to perform computerized searches and adjustments. This type of common terminology, when presented hierarchically and based on agreed-upon logic, forms the ontology to be used by the entire NoE [3]. This NoE ontology is indispensable for developing a common understanding of the information in a knowledge domain and for providing the means for automatic searching and decision-making. The creation of such groups based on their knowledge and expertise is a typical case for the application of genetic algorithms. Yet, for some common objective functions and for a reasonable number of teams (tens), selection can also be performed using LP methods.

In this paper we present the ILP formulation and solutions for several objective functions and for a variable number of teams. The optimal solutions obtained by the ILP will be used as reference for comparing the results obtained by several genetic algorithms. This situation in which the optimal solution is known also enables us to examine the convergence to the solution of some genetic algorithm schemes under various control parameters. Throughout the paper, the preliminary ontology and knowledge map developed for the product lifecycle field within the VRL-KCiP Network of Excellence will be used [4]. In Section 2 we formulate the problem and discuss several objective functions. Section 3 presents the ILP-based formulation and solutions. In Section 4 we examine several converging cases based on genetic algorithms, and Section 5 presents a summary and conclusions.

2 PROBLEM STATEMENT

Given:

- A NoE comprised of n teams: {1, 2, …n}
- Ontology (Hierarchical) {Design, Manufacturing, Assembly, Use…, K}
- Network knowledge map (matrix) n columns (teams) and K rows (ontology items)
- A Call for Project that requires k<K expertise items
- **Project:** {Expertise (Ontology {Design, Manufacturing, Assembly, Use, Maintenance, Recycling…k})}.

Thus, the knowledge map for the specific project is a n x k matrix. Figure 1 shows a partial ontology map (k=6) needed to accommodate a specific project and the relevant VRL-KCiP (n=25 teams) expertise a(I,j) (1≤i≤k, 1≤j≤n) of the various teams.

a(i,j)=1 if team i has an expertise in item j
a(i,j)=0 else

P		TEAM NUMBER																								
	Topic	Code	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
v	Design	1	1	1	1	0	1	1	1	1	0	1	1	1	1	0	1	0	0	1	0	1	1	1	1	0	1
v	Assem.	2	1	0	1	0	1	1	1	0	0	0	1	1	0	0	1	1	1	0	0	0	1	0	1	0	1
v	Mfg.	3	0	0	1	0	1	1	1	1	0	1	0	1	1	1	1	1	1	0	0	1	0	1	1	0	1
v	Use	4	1	1	1	0	0	1	1	0	1	0	1	1	0	0	0	1	1	0	1	0	1	1	1	0	0
v	Maint.	5	0	1	0	1	0	0	1	1	1	1	0	1	1	1	0	1	0	1	1	1	0	1	0	1	0
v	Recycle	6	0	1	1	1	1	1	1	1	0	0	0	1	1	1	1	1	1	1	0	0	0	1	1	1	1

Figure 1: Knowledge map of a network with a number of expertise areas.

The specific topics do not necessarily coincide with the main topics (top level) of the NoE knowledge map.

Required:

To form an **optimal group** consisting of m teams (m≤n) to handle a specific project. **An optimal group** is defined according to one of the objective functions typical to NoEs:

1. Minimum number of teams.
2. Minimum cost group which reflects the different costs charged by the different teams.
3. Minimum number of teams with constraints forcing or preventing participation of specific team(s) based on other considerations such as who 'brought' the project? current load on a member? etc.
4. Minimum number of teams, subject to minimal multiple coverage of each ontology topic.

3 ILP APPROACH

The set covering problem is first formulated as an ILP (integer linear programming) problem. Let k,n be natural numbers, c_j be positive integers (i.e. cost or level of expertise), and let $a_{i,j}$ be 0 or 1 for $(1 \le i \le k)$, $(1 \le j \le n)$. The set covering problem is defined as follows:

Determine the decision variables x_j which

$$\text{minimize} \quad m = \sum_{j=1}^{n} c_j x_j \qquad (1)$$

subject to

$$\sum_{j=1}^{n} a_{i,j} x_j \ge 1 \quad (1 \le i \le k) \qquad (2)$$

$$x_j \in \{0,1\} \qquad (3)$$

To create a group by selecting team j's, where expertise is binary (or cost is equal) cj≡1; thus, eq. (1) is replaced by:

$$\text{minimize} \quad m = \sum_{j=1}^{n} x_j \qquad (4)$$

For demonstration purposes, a small NoE consisting of five teams and six expertise items is used. Examples were executed on Excel's solver.

Case 1: Find a group consisting of a minimal number of teams that covers all six expertise items. The final result is given in Figure 2 under the constraints shown in Figure 3. The number of selected teams (two) is shown in the target cell $G$2; it was obtained by changing the decision cells $C$5:$G$5 (refer to Figure 4) which indicate that the selected teams were Team 1 and Team 4. All the constraints were met, as shown in cells $H$7:$H$13 as corresponding to $J$7:$J$13 by cells $I$7:$I$13.

Case 2: In this case, the aim is to create a minimal cost group. Thus the relative cost charged by the various teams must also be considered. Eq. (1) must be used with c_j costs, as shown in cells $C$14:$G$14 in Figure 5. To achieve minimal cost, three teams (2, 4 and 5) were selected due to the high cost of Team 1.

	A	B	C	D	E	F	G	H	I	J
1										
2		Number of participants=					2			
3										
4			Participants							
5			1	0	0	1	0			
6			1	2	3	4	5	Const.		
7	Expertise required	Design	1	0	0	0	1	1	≥	1
8		Production	0	1	1	1	1	1	≥	1
9		Mfg.	1	1	0	0	1	1	≥	1
10		Service	1	1	1	0	0	1	≥	1
11		Maintnance	0	1	0	1	0	1	≥	1
12		Support	1	1	0	0	0	1	≥	1
13		EOL	1	0	1	1	0	2	≥	1

Figure 2: A spreadsheet formulation and result for the coverage problem for the required project.

	H
6	Const.
7	=SUMPRODUCT(C$5:G$5,C7:G7)
8	=SUMPRODUCT(C$5:G$5,C8:G8)
9	=SUMPRODUCT(C$5:G$5,C9:G9)
10	=SUMPRODUCT(C$5:G$5,C10:G10)
11	=SUMPRODUCT(C$5:G$5,C11:G11)
12	=SUMPRODUCT(C$5:G$5,C12:G12)
13	=SUMPRODUCT(C$5:G$5,C13:G13)

Figure 3: The set of constraints.

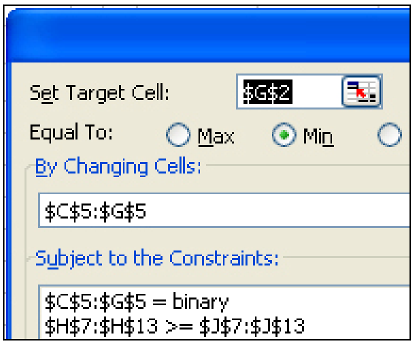

Figure 4: Spreadsheet formulation.

	A	B	C	D	E	F	G	H	I	J
1										
2			Project Price =				50			
3										
4			Participants							
5			0	1	0	1	1			
6			1	2	3	4	5	Const.		
7	Expertise required	Design	1	0	0	0	1	1	≥	1
8		Production	0	1	1	1	1	3	≥	1
9		Mfg.	1	1	0	0	1	2	≥	1
10		Service	1	1	1	0	0	1	≥	1
11		Maintnance	0	1	0	1	0	2	≥	1
12		Support	1	1	0	0	0	1	≥	1
13		EOL	1	0	1	1	0	1	≥	1
14			60	10	20	10	30			
15			Cost of participants							

Figure 5: A spreadsheet formulation and result for coverage problem for the required project of case 2.