COMPROMISE IN SCHEDULING OBJECTS PROCEDURES BASING ON RANKING LISTS

Henryk Piech, Wioletta Skibińska

Institute of Computer and Information Science, Czestochowa University of Technology, Poland
e-mail:hpiech@adm.pcz.czest.pl

Abstract. In our work possibility of support of ranking objects (tasks) is analyzed on base of group of lists. We can get these lists both from experts or with help of approximating and simple (according to complexity) algorithms. To support analyze we can use elements of neighborhood theory, preferential models, and rough sets theory. This supporting process is used for creation final list of tasks sequence. Usually, these problems are connected with distribution, classification, prediction, strategy of games as well as compromise searching operations. The utilization preference and domination models permits to crisp inferences and to force the chronological location of object. In some situations we have deal with dynamic character of filling lists resulting from continuous tasks succeeding and continuous their assigning to executive elements. The utilization the theory of neighborhood permits to locate objects in range of compromised solutions consist in closing to dominating proposal group. Main task for us is find the best compromise in aspect to final objects location. We want to defined advantages and drawback of methods basing on mention theories and analyze possibilities of their cooperation or mutual completions.

Introduction

There are many application of preference theory to solving the problems of decision supporting [1-4]. The dynamic scheduling using preferential models and rough sets theory does not introduce essential adopting changes in algorithms based on this theory but only adjusts to her parameters of data [5-8]. Preferences and dominations [9, 10] are used to comparing sequences of assigning tasks to realization but previously we should select data and define profiles which represent tasks (objects) in aspect of preference to execution (final location) [10, 11]. The domination in Pareto and Lorenz sense permits to settle basic relations between sequences of well ordered objects. The preferences of type "at least as good as..." estimated as interval (by low and upper their bounds) permit to define zone of uncertain solution. In such situations, for decision making we use additional criteria (on example the costs of reorganization [12]).

For defining the tags of localization one used elements of neighborhood theory [4, 13] such cooperation, toleration and collision in range of neighborhood [10, 14, 15]. They are named according to researched problems. It was connected, among others, with supporting or rejecting the thesis about task location in centre of given neighborhood. The closed neighborhoods confirm and support the decision (the thesis) about assignment the task to concrete location. The relation of tolerance have reflective and symmetrical character [13, 16]. The cooperating neighborhoods intensify the strength of domination and reduce the influence of passivity or small influence of tolerance. The cooperation (the supports of thesis) and the collision relation (the postponement of support of thesis, what means indirectly, the support of antithesis) crisp inference mechanisms [12, 17]. Cooperation has reversible character. This kind of dependence between relations should simplify creation of conclusion. According to theory of neighborhood, which we engage in procedure of establishing sequent, we increase the autonomy of studied tasks groups with reference their distribution. The symmetry of inference increases power of decision support at the same time [13, 18]. The next problem is connected with dynamic scheduling, and appointing the objective solutions (independent on sequent or set of criterions or experts opinions). Obviously, it is not always possible, but comfortably is to use interval solutions, particularly in situation, when solutions are on border of location classes according to given criterion [19].

1. Compromise estimation after process of creating final ranking list

Compromise is formed between ingredient judgment lists which was built with help of algorithms or on base of experts opinions. There is possibility of creating several type of compromise, for example:

1. minimum concessions and similar of their levels (minimum variance of concessions);

$$\{cmp1=\sum_{j=1}^{m}\ \sum_{i=1}^{n}(loc(i,j)-locf(i))^2\ \} \rightarrow\ min$$

$$var\{\sum_{i=1}^{n}(loc(i,j)-locf(i))^2\ \}\rightarrow min \qquad j=1,2,\ldots,m \tag{1}$$

or

$$var\{\sum_{j=1}^{m}(loc(i,j)-locf(i))^2\ \}\rightarrow min \qquad i=1,2,\ldots,n$$

where *var* - variance of concession according to ingredient list or to tasks.

2. minimum distances between center of neighborhoods with maximum powers (or concentration) and final tasks location;

$$\{cmp2=\sum_{i=1}^{n}(centre\_\max\_pow(i)-locf(i))^2\ \}\ \rightarrow min$$

or

$$\{cmp2 = \sum_{i=1}^{n} (centre\_\max\_concentration(i) - locf(i))^2\} \rightarrow min \qquad (2)$$

where:
*centre*_max_*pow* - centre of maximum power neighborhood,
*centre*_max_*concentration* - centre of maximum concentration (numbering) neighborhood,

3. minimum correction on final list according Lorenze preference location

$$\{cmp3 = \sum_{i=1}^{n} (Lorenz\_loc(i) - locf(i))^2\} \rightarrow min \qquad (3)$$

Generally we can describe compromise as follows:

$$\{cmp = \sum_{i=1}^{n} (criterion\_loc(i) - locf(i))^2\} \rightarrow min \qquad (4)$$

where *criterion_loc*(i) - location of i-th object suggested by chosen criterion.

We can to use different criterion or their composition for estimation of compromise. In result of using these criteria we often obtain the same location for different objects. In this case it needs to use auxiliary criteria, methods or heuristic rules. Sometimes we decided to use different criteria for compromise estimation and resign from based for creating final lists method (Fig. 1).

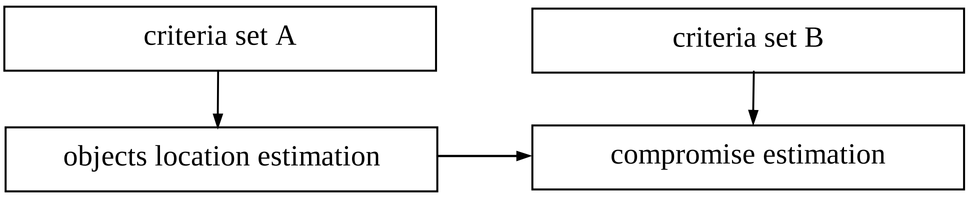

Fig. 1. Distinguished criteria set for creating final list and compromise: $A \cap B = 0$

In our convention (1)-(4) the best compromise refers the smallest value of parameter *cmp*. To compare compromise for several final lists we should keep the same criteria in set *B*.

2. Sets of criteria for creating final list

It is necessary to define several criteria because often results from using single criteria aren't unambiguously. It means that we have several objects pretending to one location on final list. We propose several composition of criteria:

1. $sup(\varphi \rightarrow \psi) \rightarrow max$
   $centre(\varphi, i) \rightarrow min$
2. $cnbh(\varphi, i) \rightarrow max$

$zone(\varphi,i) \rightarrow min$
$centre(\varphi,i) \rightarrow min$

3. $sup(* \rightarrow \psi) + sup(\varphi \leftarrow *) \rightarrow min$
$cnbh(\varphi,i) \rightarrow max$ (5)

where: $sup(\varphi \rightarrow \psi) \rightarrow max$ - maximal number of object in one placement in ingredient lists, we chose object φ_i and placed it on position ψ_j,
$centre(\varphi,i) \rightarrow min$ - minimal position of neighborhood centre, we chose object φ_i from this neighborhood, which is closed to begin of list and locate it in center of its neighborhood,
$cnbh(\varphi,i) \rightarrow max$ - maximal concentration neighborhood, we chose object φ_i with maximal neighborhood concentration (numbering) and locate them in its center,
$zone(\varphi,i) \rightarrow min$ minimal neighborhood distance from begin of list, we chose object φ_i with minimal neighborhood distance and locate them in its center,
$sup(* \rightarrow \psi) + sup(\varphi \leftarrow *) \rightarrow min$ - minimal number of objects pretending to position ψ_j and minimum positions to which pretended objects φ_i, we chose object φ_i and locate them on position ψ_j (intuition criterion).

We often obtain the same value of criteria estimators. In this case we should go to next criterion in hierarchy, considering the same object and searching next the best location for it. Similar situation appears when chosen location is occupied by previous located objects.

3. Methods and examples of creating final lists of scheduled objects

For scheduling objects we can use rules using in theories:
– neighborhoods,
– preferences,
– rough sets.

Beside of criteria set we can use specific methods using traditionally for classification, categorization, ordering objects [4]. We try to enrich every of proposed method by example. Above was described exploitation neighborhoods theory to define criteria set. It is possibility to combine elements of quoted theories in different variant:

1) neighborhoods + rough sets

We can create lower approximation $\underline{P}(O)$ [16] as set of maximal concentration (or power) neighborhoods and upper approximation $\overline{P}(O)$ as set of all objects locations. In this case main structure (O) is defined by sum of all neighborhoods.

2) neighborhoods + preferences

We can define preferences relation between neighborhoods (or maximal neighborhoods) using their characteristics (concentration, power).

3) neighborhoods + preferences + rough sets

From set of upper approximation we chose and remove extreme located neighborhoods and locate adequated them objects in their neighborhoods center.

For researches objects distribution it can be exploit rough sets theory (Pawlak theory).

Using Pawlak theory [20] we can adapt semantically dependence on physically sense of terminology, e.g. relative zone (O). In our case (in ordering objects by several algorithms simultaneously) we can define relative zone as a range of positions in which are included the most important neighborhoods representing all objects (lower approximation). Relative zone has common part with less important

$$\bigcup_{nbh(i,\max)\subseteq(O)} nbh(i,max) = \underline{P}(O) \qquad \text{(lower approximation)} \qquad (6)$$

$$\bigcup_{nbh(i,*<\max)\cap(O)\neq 0} nbh(i, * < max) = \overline{P}(O) \qquad \text{(upper approximation)} \qquad (7)$$

So in our case relative zone (O) can be named representative zone and it contained objects on all positions (O)=(1)+(2)+...+(8). This zone will be systematically cut off (from both sides) during extracting objects to final list (Tables 1). So this zone has dynamic length.

The draw back of above presented method (Tables 1) consist in preferring center neighborhoods location over their numbering.

When we use Lorenz preference rules [7] in simple way we can calculate average locations for all objects. In our example (Tables 1) we obtain next results:

$pL(1)=aver(loc(\varphi_1))=(1+1+3+8)/4=3,25$
$pL(2)=aver(loc(\varphi_2))=(1+2+4+7)/4=3,5$
$pL(3)=aver(loc(\varphi_3))=(1+2+2+3)/4=2$
$pL(4)=aver(loc(\varphi_4))=(1+1+3+8)/4=4,5$
$pL(5)=aver(loc(\varphi_5))=(4+6+6+8)/4=6$
$pL(6)=aver(loc(\varphi_6))=(2+3+3+5)/4=3,25$
$pL(7)=aver(loc(\varphi_7))=(6+7+8+8)/4=7,25$
$pL(8)=aver(loc(\varphi_8))=(5+6+7+7)/4=6,25$

where $pL(i)=aver(loc(\varphi_i))=1/m\sum_{j=1}^{m} loc(\varphi(i,j))$ - the strength of Lorenz preference characteristic.

After ordering we have final list of ranking

$pL(3)\succeq pL(1)\succeq pL(6)\succeq pL(2)\succeq pL(4)\succeq pL(5)\succeq pL(8)\succeq pL(7)$

or in form

Example 2

3	1	6	2	4	5	8	7

final list - preference in Lorenz sens

Tables 1

Using rough sets theory for creating final ranking list

1	3	6	2	4	5	8	7
3	6	1	4	8	5	2	7
1	2	3	4	6	7	8	5
2	3	6	5	4	8	7	1

sum of max. neigborhoods-->smn

(O)=(1)+(2)+...+(8)

1	3	6	2	4	5	8	7
3	6	1	4	8	5	2	7
1	2	3	4	6	7	8	5
2	3	6	5	4	8	7	1

smn=smn-nbh(1,max)-nbh(7,max)

(O)=(O)-(8)

1							**7**
1	3	6	2	4	5	8	7
3	6	1	4	8	5	2	7
1	2	3	4	6	7	8	5
2	3	6	5	4	8	7	1

final list –> first stage

smn=smn-nbh(3,max)-nbh(8,max)

(O)=(O)-(7)

1	**3**					**8**	**7**
1	3	6	2	4	5	8	7
3	6	1	4	8	5	2	7
1	2	3	4	6	7	8	5
2	3	6	5	4	8	7	1

final list –> second stage

smn=smn-nbh(2,max)-nbh(5,max)

(O)=(O)-(1)

1	**3**	**2**			**5**	**8**	**7**
1	3	6	2	4	5	8	7
3	6	1	4	8	5	2	7
1	2	3	4	6	7	8	5
2	3	6	5	4	8	7	1

final list –> third stage

smn=smn-nbh(6,max)-nbh(4,max)

(O)=0

Example 1

1	**3**	**2**	**6**	**4**	**5**	**8**	**7**

final list –> last stage

The drawback of this approach consist in regarding not essential date such single location e.g. to regarding essential information we use only neighborhoods and for it we prepare their characteristics:

Table 2

Characteristics of neighborhoods for all tasks

description of neighborhood	numbering	zone	centre	power
nbh(1,1)	2	1	1	2
nbh(2,1)	2	2	1	1
nbh(3,1)	4	3	2	1,33
nbh(41)	4	2	4	2
nbh(5,1)	2	1	6	2
nbh(6,1)	6	2	3	3
nbh(7,1)	4	3	8	1,33
nbh(8,1)	4	3	7	1,33

$$\mathrm{pn(i)}=\sum_{j=1}^{\ln(i)} numbering(nbh\_\varphi(i,j))*centre(\varphi(i,j))/\sum_{j=1}^{\ln(i)} numbering(nbh\_\varphi(i,j)) \quad (8)$$

where:
ln(*i*) - number of neighborhoods for *i*-th object
numbering(*nbh*(*i*,*j*)) - numbering (concentration) of *j*-th neighborhood for *i*-th object (table 2)
centre(φ(*i*,*j*)) - centre of of *j*-th neighborhood for *i*-th object (table 2)

pn(1)=2*1/2=1	*pn*(5)=2*6/2=6
pn(2)=2*1/2=1	*pn*(6)=6*3/6=3
pn(3)=4*2/4=2	*pn*(7)=4*8/4=8
pn(4)=4*4/4=4	*pn*(8)=4*7/4=7

Example 3

1	2	3	6	4	5	8	7

final list - gravity points for every object

To analysis and compare chosen methods we propose choice set of criterion (1), for example:

cnbh(φ,*i*) → φ *max*
zone(φ,*i*) → *min*
centre(φ,*i*) → *min*

and with help of them formulate final list. It give us solution with structure:

Example 4

1	3	6	4	2	5	8	7

final list - set of criteria

In this case we have next sequence of joining to final list:

1) $\varphi_4 \rightarrow \psi_4$ 2) $\varphi_8 \rightarrow \psi_7$ 3) $\varphi_7 \rightarrow \psi_8$ 4) $\varphi_3 \rightarrow \psi_2$

1) $\varphi_6 \rightarrow \psi_3$ 2) $\varphi_1 \rightarrow \psi_1$ 3) $\varphi_5 \rightarrow \psi_6$ 4) $\varphi_2 \rightarrow \psi_5$

According this method we use essential date and omit single object placement and deviation.

4. The example of exploitation compromise to judgment of set of final list

For choice compromise criteria we can lead the quantity of information which was use to define in estimation process. Such approach suggested to exploit Lorenz preferences as compromise criterion. In next step we estimate scale of differences between final lists and list created on base of Lorenz preference. According (4) we do it for all solution

1) $cmp = \sum_{i=1}^{n} (criterion\_loc(i) - locf(i))^2 = (3-1)^2 + (1-3)^2 + (6-2)^2 + (2-6)^2 + (4-4)^2 + (5-5)^2 + (8-8)^2 + (7-7)^2 = 48$

3) $cmp = \sum_{i=1}^{n} (criterion\_loc(i) - locf(i))^2 = (3-1)^2 + (1-2)^2 + (6-3)^2 + (2-6)^2 + (4-4)^2 + (5-5)^2 + (8-8)^2 + (7-7)^2 = 30$

4) $cmp = \sum_{i=1}^{n} (criterion\_loc(i) - locf(i))^2 = (3-1)^2 + (1-3)^2 + (6-6)^2 + (2-4)^2 + (4-2)^2 + (5-5)^2 + (8-8)^2 + (7-7)^2 = 16$

$min\ \{cmp(1); cmp(3); cmp(4)\} = min\ \{48; 30; 16\} = 16$

To find the nearest, to compromise solution, final list we have named additional parameter for defining method code, according which particular list was created. For example we extend location attribute name to form $locf_k(i)$, where k - code of using for creating final list method (which are adequated to presented above examples). Compromise expression stay to be simple and can has follow form:

$$\{cmp = \sum_{k=1}^{lm} \sum_{i=1}^{n} (criterion\_loc(i) - locf_k(i))^2\} \rightarrow min \qquad (9)$$

where lm - number of ordering methods basing on ingredient lists analysis.

The best compromise according Lorenz criterion we find in (4) example. Obviously when we chose different compromise criterion the best criterion will be different. Some time we dispose set of compromise criteria. In this case rules of searching compromise can be expressed by:

$$\{cmp=\sum_{j=1}^{lc}\sum_{k=1}^{lm}\ \sum_{i=1}^{n}(criterion(j)\_loc(i)-locf_k(i))^2\ \} \rightarrow min \qquad (10)$$

where *lc* - compromise criteria number.

If we use the same methods (criteria) for crating both final lists and compromise stencil list than ingredients $\sum_{i=1}^{n}(criterion(d)\_loc(i)-locf_d(i))^2$, where *d* respect choosing the same method (or criteria) for both task, will be obviously equal zero, but it doesn't influence, at all, on final compromise estimator level.

Conclusions

The experiences shows that combining methods of neighborhoods, preference and rough set for analysis ranking list is very comfortable and permit to exploit reach pat of information for crating final list and compromise solution.
The situation doesn't became more difficult even when we dispose the same set of methods for creating final lists and compromise list.
Exploiting neighborhood theory we use tools for eliminating inessential information in opposite to some variant of preference rules, but using preference methods we can create reference stencils.
Specific character of rough sets theory description permit not only to reject objects of inessential attributes values, but at the same time to dislocated objects using current compromise decisions.
Neighborhood estimators are less unambiguously but don't regard inessential date.

References

[1] Błażewicz J., Lenstra J.K., Rinnooy Kan A.H.G., Scheduling subject to resource constrains: Classification and complexity, Discrete Appl. Math. 1983, 5, 11-24.

[2] Greco S., Matarazzo B., Słowiński R., Axiomatic characterization of a general utility function and its particular cases in terms of conjoint measurement and rough sets decision rules, European J. of Operational Research 2003.

[3] Greco S., Matarazzo B., Słowiński R., Rough sets theory for multicriteria decision analysis, European J. of Operational Research 2001, 129, 1-47.

[4] Piech H., Analysis of possibilities and effectiveness of combine rough set theory and neibourhood theories for solving dynamic scheduling problem, IEEE Computer Society, P3674, Washington, Tokyo 2009, 296-302.

[5] Conway R.W., Maxwell, W.L., Miller L.W., Theory of Scheduling, Addision-Wesley, Reading, MA, 1954.

[6] Pawlak Z., Sugeno M., Decision Rules Bayes, Rule and Rough, New Decisions in Rough Sets, Springer-Verlag, Berlin 1999.

[7] Skowron A., Extracting lows from decision tables, Computational Inteligence 1995, 11(2), 371--388.

[8] Szwarc W., Permutation flow-shop theory revised, Math. Oper. Res. 1978, 25, 557-570.

[9] Greco S., Matarazzo B., Słowiński R., Extension of rough set approach to multicriteria decision support, INFOR 2000, 38, 161-196,.

[10] Jaroń J., Systemic Prolegomena to Theoretical Cybernetics, Scient. Papers of Inst. of Techn. Cybernetics 25, Wrocław 1975.

[11] Kohler W.H., A preliminary evolution of the critical path method for scheduling tasks on multiprocessor systems, IEEE Trans. Comput. 1975, 24, 1235-1238.

[12] Brzezińska I., Greco S., Słowiński R., Mining Pareto-optimal rules wth respect to support and anti-suport. Engeniering Applications of Artificial Inteligence 2007, 20(5), 587-600.

[13] Nikodem J., Autonomy and cooperation as factors dependability in wireless sensor network, IEEE Computer Society P3179, 406-413, 2008-08-09

[14] Finch H.A., Confirming power of observations metricized for decisions among hypotheses, Philosophy of Science 1999, 27, 391- 404.

[15] Kleinberg J., Navigation in a small words, Nature 2000, 406, 845.

[16] Greco S., Matarazzo B., Słowiński R., Stefanowski J., An algorithm for induction of decision rules with dominance principle, Rough Sets and Current Trends in Computing, LNAI, Springer-Verlag, Berlin 2005, 304-313.

[17] Sysło M.M., Deo N., Kowalik J.S., Algorytmy optymalizacji dyskretnej, WN PWN, Warszawa 1995.

[18] Słowiński R., Brzezińska I., Greco S., Application of Bayesian Confirmation Measures for Mining Rules from Suport Confidence Pareto Optimal Set, ICAISC LNSC, 4029, Springer, Heilderberg 2006, 1018-1026.

[19] Talbi E.D., Geneste L., Grabot B., Prévitali R., Hostachy P., Application of optimization techniques to parameter set-up in scheduling, Computers in Industry 2004, 55, 2, October, 105--124.

[20] Pawlak Z., Rough Sets. Theoretical Aspects of Reasoning about date, Kluwer, Dordrecht 1991.

[21] Kent R.E., Rough concept analysis: A synthesis of rough sets and formal concept analysis, Fundamanta Informaticae 1996, 27, 169-181.