Multi-objective optimization aided to allocation of vertices in aesthetic drawings of special graphs*

Audrius Varoneckas[a], **Antanas Žilinskas**[b], **Julius Žilinskas**[b]

[a]Vytautas Magnus University
Vileikos str. 8, LT-44404 Kaunas, Lithuania
a.varoneckas@if.vdu.lt

[b]Institute of Mathematics and Informatics, Vilnius University
Akademijos str. 4, LT-08663 Vilnius, Lithuania
antanas.zilinskas@mii.vu.lt; julius.zilinskas@mii.vu.lt

Received: 14 January 2013 / **Revised:** 13 July 2013 / **Published online:** 25 September 2013

Abstract. A problem of drawing specific graphs is considered emphasizing aesthetic appeal of the visualization. We focus on graphs related to the management of business processes. A particular problem of the aesthetic drawing is considered where the aesthetic allocation of vertices is aimed. The problem is stated as a problem of bi-objective optimization where the objectives are the length of connectors and the compatibility of the sequence flows with the favorable top-down, left-right direction. An algorithm based on the branch-and-bound approach is proposed.

Keywords: visualization of graphs, multi-objective optimization, business process diagrams.

1 Introduction

Graphs are very convenient models for different applications. In many cases the main advantage of graph models is their suitability for representation by a planar/spatial structure of connected shapes. Requirements to the geometric layout of graphs depend on the considered application, however frequently one of the most important requirements is formulated as optimality with respect to one or another criterion. In most cases some nonlinear constraints should be taken into account, e.g. formulated as logic expressions [1]. The requirements to optimize some technological criteria while searching for an appropriate layout is natural, e.g. in the problems of electronic design where the graph vertices and edges represent objects with geometric and electrical properties [2]. Graph drawing is reduced to the optimization problems also in many other cases where not only technical but also abstract objects are modeled by graph vertices and edges.

Plenty of publications on graph drawing as well as corresponding algorithms are available [3]; nevertheless special cases of the problem frequently cannot be solved by

*This research was supported by Agency for Science, Innovation and Technology (MITA) trough the grant No. 31V-145.

straightforward application of the known methods and algorithms. In the present paper we consider a particular problem of drawing graphs related to business process diagrams where criteria of aesthetics are crucial. According to the general opinion, the aesthetic attractiveness of the drawing of a business process diagram is especially important since the aesthetic layouts are also most informative and practical. The graph drawing aesthetics is comprehensively discussed, e.g. in [4–6]. However, the criteria of the aesthetic attractiveness not always guarantee the informativeness of the diagrams drawn; we cite [6]: “Few algorithms are designed for a specific domain, and there is no guarantee that the aesthetics used for generic layout algorithms will be useful for the visualization of domain-specific diagrams”. Such features of a layout as, for example, length of connectors, number of bends, number of crossings, and uniformity of distribution of shapes influence aesthetic of the considered layout. For different applications those features are of different importance to the aesthetic appeal of the layout, thus they can be considered aesthetic criteria. In the papers [7, 8] a method of quantitative evaluation of the importance of the aesthetic criteria is described where the attitude of the potential users to the aesthetic appeal of layouts was elicited by a psychological experiment.

The problem of the aesthetic graph drawing can be reduced to a problem of multi-objective optimization where the evaluated factors of importance are used. However, the multi-objective shortest path problems are NP-complete; for the analysis of the bi-objective case we refer to [9, 10]. To tackle the NP-complete problems various heuristic and meta-heuristic methods are well suitable [11–13]. For example, several algorithms of combinatorial multi-objective optimization related to the aesthetic drawing of connectors are proposed in [7] assuming that the location of shapes is fixed. Such a situation occurs in case a business process diagram is drawn in an interactive mode, and a user selects sites for shapes. After an interactive session is completed it is reasonable to draw the final aesthetically appealing diagram. In such a situation the complete drawing problem (allocation of shapes and drawing of connectors) could be considered. However, in such a statement the problem is too difficult. Therefore, we propose to decompose it into two stages: allocation of shapes and drawing of connectors. In the present paper the problem of aesthetic allocation of shapes is attacked by multi-objective optimization. At this stage two objectives (length of connectors and compatibility with the favorable sequence flow) are taken into account supposing that the other objectives will be optimized at the second stage of solution. In the second stage connectors may be drawn assuming that the locations are fixed. Multi-objective algorithms for drawing connectors [7, 14] minimize the length of lines, number of crossings, and number of bends.

2 A problem of the aesthetic allocation of shapes

A business process diagram consists of elements (e.g. activities, events, and gateways) which should be drawn according to the rules of Business Process Modeling Notation. The elements of diagrams are drawn as shapes which are allocated in a pool divided by the (vertical) swimlanes according to function or role. We consider a restricted set of shapes constituted by a rectangle, rhombus, and circle. An example of a business process

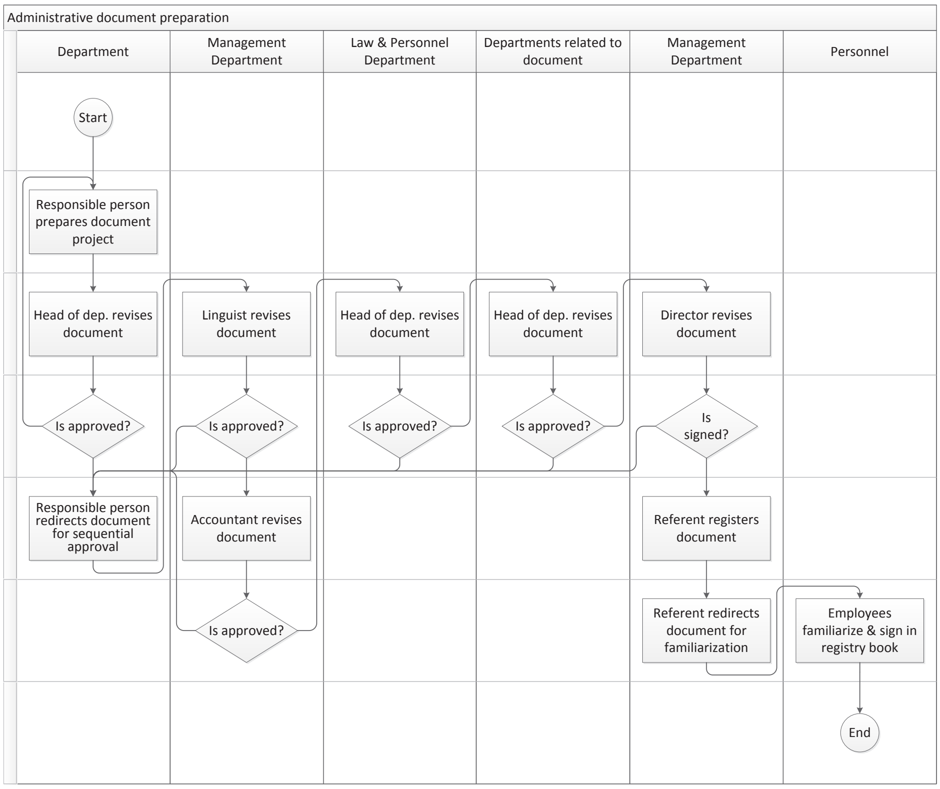

Fig. 1. An example of a business process diagram.

diagram is presented in Fig. 1. Two swimlanes are related to the Management Department since in the considered example the functions of planning (director) and organizing (accountant etc.) are explicitly separated. In the present paper we are interested only in the allocation of shapes, i.e. we ignore the interpretation of the diagram in terms of the visualized business process.

The input for the problem of the drawing a diagram is several lists containing shapes which should be allocated in the same swimlanes. The input data also define the pairs of shapes which should be connected. It is requested to allocate shapes in swimlanes, and the swimlines with regard to each other aiming at aesthetical appeal of the drawing. The problem is reduced to a problem of multi-objective combinatorial optimization. In the present paper the bi-objective problem is considered taking into account two objectives: length of connectors, and compatibility of the process flow with the top-down, left-right direction. For example, the business process diagram presented in Fig. 1 contains some rather long connectors as well as some sequence flows not compatible with the favorable top-down left-right direction. Figure 1 was used for the psychological experiment described in [7,8] as an example of a diagram of medium appeal of the considered criteria.

3 Allocation of shapes by means of multi-objective optimization

A multi-objective optimization problem is to minimize an objective vector $\mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_d(\mathbf{x}))$:

$$\min_{\mathbf{x}\in X} \mathbf{f}(\mathbf{x}),$$

where $\mathbf{x}$ is the decision vector and X is the search space. In most cases there is no single optimal solution to a given multi-objective optimization problem and the set of so-called non-dominated solutions is sought.

The decision vector $\mathbf{a}$ dominates the decision vector $\mathbf{b}$ (we denote $\mathbf{a} \succ \mathbf{b}$) if:

$$\forall i \in \{1, 2, \ldots, d\}\colon\ f_i(\mathbf{a}) \leqslant f_i(\mathbf{b}) \ \&\ \exists j \in \{1, 2, \ldots, d\}\colon\ f_j(\mathbf{a}) < f_j(\mathbf{b}).$$

A set of non-dominated (Pareto optimal) decision vectors is called Pareto set, and the set of corresponding objective vectors is called Pareto front.

In this paper we consider a multi-objective optimization problem for allocation of the shapes (flow objects) in business process diagram. The general aim of the problem is to produce aesthetic diagram, which we approach in two steps: allocation of the shapes and drawing of the orthogonal connectors. The shapes are allocated in a grid of predefined number of rows and columns (swimlanes). Let us denote the number of rows by n_r and the number of columns by n_c. The connectors show the sequence flow, two flow objects are connected if one directly precedes another. The shapes are allocated in such a way so that the connected shapes were close to each other and that the flow would direct from left to right and from top to bottom. Two objectives are simultaneously optimized:

- Minimization of total length of connectors: The sum of city block distances between connected shapes is minimized.
- Minimization of the number of right down flow violations: The number of times the preceding shape in the connection is not higher than and is to the right from the following shape is minimized.

The data of the problem are assignment of shapes to the roles (or functions) and the list of connections. Let us denote the number of shapes by n and the roles corresponding to shapes by $\mathbf{d}$, where $d_i,\ i = 1, \ldots, n$ define the role number of each shape. The connections are defined by $n_k \times 2$ matrix $\mathbf{K}$ whose rows define connecting shapes and k_{i1} precedes k_{i2}.

The shapes assigned to the same role should be shown in the same column (swimlane), however the columns may be permuted. Therefore, part of decision variables define assignment of roles to columns. Let us denote the assignment of roles to columns by $\mathbf{y}$ which is a permutation of $(1, \ldots, n_c)$ and y_i defines the column number of ith role.

Another part of decision variables define assignment of shapes to rows. Let us denote this assignment by $\mathbf{x}$, where x_i defines the row number of ith shape.

We define the objectives as following. The length of orthogonal connector cannot be shorter than the city block distance between the connected points. We model the potential length of connector as a city block distance between shapes. Therefore, the total length of connectors is calculated as

$$f_1(\mathbf{x},\mathbf{y}) = \sum_{i=1}^{n_k} |x_{k_{i1}} - x_{k_{i2}}| + |y_{d_{k_{i1}}} - y_{d_{k_{i2}}}|. \tag{1}$$

The number of right down flow violations is calculated as

$$f_2(\mathbf{x},\mathbf{y}) = \sum_{i=1}^{n_k} v_d(k_{i1}, k_{i2}) + v_r(k_{i1}, k_{i2}), \tag{2}$$

where down flow violation is

$$v_d(i,j) = \begin{cases} 1, & x_i \geqslant x_j, \\ 0, & \text{otherwise}, \end{cases}$$

and right flow violation is

$$v_r(i,j) = \begin{cases} 1, & y_{d_i} > y_{d_j}, \\ 0, & \text{otherwise}. \end{cases}$$

The connection of two shapes in the same row violates down flow since the bottom or side of preceding shape connects to the top of the following shape.

In such a definition objective functions are separable into two parts, one is dependent only on decision variables $\mathbf{x}$ and another on $\mathbf{y}$:

$$f_1(\mathbf{x},\mathbf{y}) = f_{1x}(\mathbf{x}) + f_{1y}(\mathbf{y}),$$

$$f_{1x}(\mathbf{x}) = \sum_{i=1}^{n_k} |x_{k_{i1}} - x_{k_{i2}}|, \tag{3}$$

$$f_{1y}(\mathbf{y}) = \sum_{i=1}^{n_k} |y_{d_{k_{i1}}} - y_{d_{k_{i2}}}|, \tag{4}$$

$$f_2(\mathbf{x},\mathbf{y}) = f_{2x}(\mathbf{x}) + f_{2y}(\mathbf{y}),$$

$$f_{2x}(\mathbf{x}) = \sum_{i=1}^{n_k} v_d(k_{i1}, k_{i2}), \tag{5}$$

$$f_{2y}(\mathbf{y}) = \sum_{i=1}^{n_k} v_r(k_{i1}, k_{i2}). \tag{6}$$

Therefore, the problem can be decomposed into two: find non-dominated vectors (f_{1x}, f_{2x}) and non-dominated vectors (f_{1y}, f_{2y}). The non-dominated solutions of two problems are then aggregated and non-dominated solutions of the whole problem are retained. The number of solutions of the first problem is

$$\prod_{i=1}^{n_c} \frac{n_r!}{(n_r - n_i)!},$$

where n_i is the number of shapes assigned to ith role. The number of solutions of the second problem is $n_c!$. For example, if we have 3 roles, there are 4 objects in one role and 6 objects in each other two roles, and we want to fit the diagram in 7 rows, the number of solutions of the second problem is $3! = 6$ and the number of solutions of the first problem is

$$\frac{7!}{3!} \times 7! \times 7! = 21\,337\,344\,000$$

which is a big number.

4 Branch and bound algorithm for multi-objective combinatorial optimization

The main concept of branch and bound is to divide sets of solutions and discard those which cannot contain optimal solutions. In single objective optimization a set of solutions cannot contain optimal solutions if the bound for the objective function over this set is worse than the known function value. In multi-objective optimization the set of solutions cannot contain Pareto optimal solutions if the bound vector $\mathbf{b}$ is dominated by at least one already known decision vector $\mathbf{a}$:

$$\forall i \in \{1, 2, \ldots, d\}\colon f_i(\mathbf{a}) \leqslant b_i \ \&\ \exists j \in \{1, 2, \ldots, d\}\colon f_j(\mathbf{a}) < b_j.$$

Bounds for objective functions over sets of solutions are needed. Construction of bounds depends on the type of sets of solutions over which the bounds are evaluated.

As it was described in the previous section, the number of solutions of the second problem is much smaller than that of the first problem. Therefore, we solve the second problem by enumeration of all solutions while the first problem is solved using branch and bound algorithm. In the case the number of columns is larger, branch and bound algorithm can be used for the second problem as well. We will represent a set of solutions of multi-objective problem for allocation of the shapes in business process diagrams as a partial solution where only some shapes are assigned to rows. Therefore, the partial solution is represented by the assignment $\mathbf{x}'$ of $n' < n$ shapes to rows. The bounds for objective functions of allocation of the shapes to rows in business process diagrams (3), (5) may be computed as follows. The total length of connectors cannot be smaller than the length of connectors among already assigned shapes:

$$b_1(\mathbf{x}') = \sum_{i=1,\, k_{i1} \leqslant n',\, k_{i2} \leqslant n'}^{n_k} |x_{k_{i1}} - x_{k_{i2}}|. \tag{7}$$

Similarly the number of down flow violations cannot be smaller than one of already assigned shapes:

$$b_2(\mathbf{x}') = \sum_{i=1,\, k_{i1} \leqslant n',\, k_{i2} \leqslant n'}^{n_k} v_d(k_{i1}, k_{i2}). \tag{8}$$

An iteration of the classical branch and bound algorithm processes an unexplored set of solutions. The iteration has three main components: selection of the candidate set

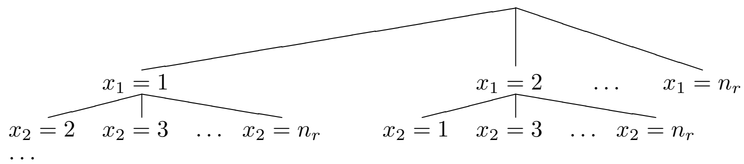

Fig. 2. Search tree of the branch and bound for allocation of the shapes in business process diagrams.

to be processed, subdivision of the set, and bound calculation. Performance of a branch and bound algorithm depends on selection strategy [15]. We build a branch and bound algorithm for multi-objective problem for allocation of the shapes in business process diagrams using the depth first selection. The advantage of this selection strategy is that it allows avoidance of storing of candidate sets [16], the number of the candidates may be very large similarly as the number of solutions is huge.

The search tree of the branch and bound is illustrated in Fig. 2. The levels of the tree represent different shapes. The branches of the tree represent assignment of the flow objects to rows of business process diagram. Of course the shape cannot be assigned to the row where another shape of the same role (swimlane) is already assigned.

The algorithm for multi-objective problem for allocation of the shapes in business process diagrams can be outlined in the following steps:

1. Form the first possible assignment of shapes to rows in $\mathbf{x}$. Set $n' \leftarrow n + 1$.
2. Repeat while $n' > 0$
   - If the current solution is complete ($n' \geqslant n$)
     - Set $n' \leftarrow n$.
     - Compute
       $$f_{1x}(\mathbf{x}) = \sum_{i=1}^{n_k} |x_{k_{i1}} - x_{k_{i2}}|, \qquad f_{2x}(\mathbf{x}) = \sum_{i=1}^{n_k} v_d(k_{i1}, k_{i2}).$$
     - If no solutions in the solution list dominate the current solution $\mathbf{x}$, add it to the solution list:
       $$\text{If } \nexists \mathbf{a} \in S\colon\ \mathbf{a} \succ \mathbf{x},\ \text{then } S \leftarrow S \cup \{\mathbf{x}\}.$$
     - If there are solutions in the solution list dominated by the current solution, remove them from the solution list:
       $$S \leftarrow S \setminus \{\mathbf{a} \in S\colon\ \mathbf{x} \succ \mathbf{a}\}.$$
   - Otherwise
     - Compute
       $$b_1(\mathbf{x}') = \sum_{i=1,\, k_{i1} \leqslant n',\, k_{i2} \leqslant n'}^{n_k} |x_{k_{i1}} - x_{k_{i2}}|,$$
       $$b_2(\mathbf{x}') = \sum_{i=1,\, k_{i1} \leqslant n',\, k_{i2} \leqslant n'}^{n_k} v_d(k_{i1}, k_{i2}).$$

  - If (b_1, b_2) is dominated by a solution from the solution list reduce n'.
- Update $x_{n'}$ by available number and increase n' or reduce n' if there are no further numbers available.

3. Find non-dominated solutions Q of the second problem.
4. Aggregate non-dominated solutions of two problems,

$$\begin{aligned} f_1(\mathbf{x},\mathbf{y}) &= f_{1x}(\mathbf{x}) + f_{1y}(\mathbf{y}), \\ f_2(\mathbf{x},\mathbf{y}) &= f_{2x}(\mathbf{x}) + f_{2y}(\mathbf{y}), \end{aligned} \quad \mathbf{x} \in S,\ \mathbf{y} \in Q,$$

   and retain non-dominated solutions of the whole problem.

5 Computational experiments

We perform computational experiments on problems of business process diagrams. A computer with Intel i7-2600 CPU 3.40GHz, 8GB RAM and Ubuntu 12.10 Linux was used for experiments. The branch and bound algorithm has been implemented in C/C++ and built with g++ 4.7.2 compiler.

The data of the first example business process are given in Table 1. There are 3 roles in this business process, 4 shapes are in one role and 6 shapes are in each other two roles. The smallest number of rows to fit the business process diagram is 6. In this case the number of different assignments of roles to columns is 6 and shapes to rows is 186 624 000. In the case of 7 rows the number is indicated in Section 3 and is 21 337 344 000.

Table 1. Data of example business process diagram.

Flow objects

No.	Role	Name	No.	Role	Name
1	Role1	Event1	9	Role2	Gateway4
2	Role1	Activity1	10	Role2	Activity6
3	Role1	Gateway1	11	Role2	Gateway3
4	Role1	Activity4	12	Role2	Event4
5	Role1	Activity5	13	Role3	Activity8
6	Role1	Activity7	14	Role3	Activity2
7	Role2	Event2	15	Role3	Gateway2
8	Role2	Activity3	16	Role3	Event3

Connections

i	k_{i1}	k_{i2}	i	k_{i1}	k_{i2}	i	k_{i1}	k_{i2}
1	1	2	7	10	9	13	14	11
2	2	3	8	5	11	14	14	13
3	3	4	9	9	12	15	8	15
4	3	5	10	11	12	16	15	11
5	7	8	11	5	13	17	14	15
6	6	9	12	13	9	18	16	14

Branch and bound leads to complete enumeration of all possible solutions in the worst case – when the bounds do not help to discard any sets of solutions. It is interesting to investigate how far is practical case from the worst case. Therefore, we compare the results of the developed branch and bound algorithm with complete enumeration, where all possible assignments of shapes to rows are enumerated and non-dominated are retained. The computational time (t) and the number of functions evaluations (NFE) for the algorithms solving the example problem are presented in Table 2. NFE for complete enumeration coincide with computed numbers of possible different assignments. Branch and bound is approximately 70 times faster for 6 rows problem, approximately 770 times faster for 7 rows problem, and approximately 3400 times faster for 8 rows problem. This enables the solution of the considered problem in an acceptable time.

Both algorithms produce the same sets of Pareto solutions as expected. Pareto solutions of the example problem with 7 rows are given in Table 3. Pareto fronts for problems with different number of rows are illustrated in Fig. 3. Pareto solutions of the problem with 6 rows are dominated by that of the problems with more rows. This means that the problem with 6 rows is too restrictive. Many objective vectors of Pareto solution of the problems with 7 and 8 rows coincide.

Solutions of the multi-objective problem of shape allocation in the example business process diagram with 7 and 8 rows are shown in Fig. 5. We do not consider the problem of drawing orthogonal connectors in this paper. In these diagrams connectors are drawn following the rules of the software developed by OrgSoft. The connectors may be drawn in several horizontal and vertical lanes located in the middle and on the sides of cells. Bends were rounded to enhance visibility and clarity. The number of crossings was minimized when drawing connectors.

Table 2. Experimental comparison of the branch and bound algorithm and complete enumeration.

Number	branch and bound		complete enumeration		Speed-up	
of rows	t, s	NFE	t, s	NFE	t	NFE
6	0.15	$2374407 + 6$	10.96	$186624000 + 6$	73	79
7	1.44	$28118029 + 6$	1112.63	$21337344000 + 6$	773	759
8	9.49	$192605603 + 6$	32476	$682795008000 + 6$	3422	3545

Table 3. Pareto solutions of the example business process diagram problem with 7 rows.

$\mathbf{x}$			$\mathbf{y}$	$f_1(\mathbf{x},\mathbf{y})$	$f_2(\mathbf{x},\mathbf{y})$
1 2 3 6 4 5	2 3 6 4 5 7	5 3 4 2	1 3 2	33	1
1 2 3 5 4 6	2 3 6 4 5 7	5 3 4 2	1 3 2	31	2
1 2 3 6 4 5	2 3 6 5 4 7	5 3 4 2	1 3 2	30	3
1 2 3 5 4 6	2 3 6 5 4 7	5 3 4 2	1 3 2	28	4
1 2 3 5 4 6	2 3 6 7 4 5	5 3 4 2	1 3 2	26	6
1 2 3 5 4 6	2 3 6 7 4 5	5 4 3 2	1 3 2	25	8
1 2 3 5 4 6	2 3 6 7 4 5	5 3 4 2	1 2 3	25	8
1 2 3 5 4 6	2 3 6 7 4 5	5 4 3 2	1 2 3	24	10

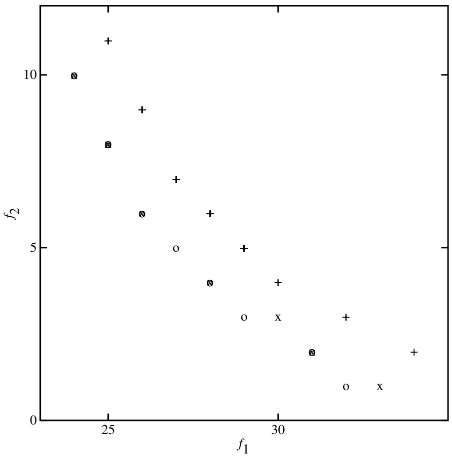

Fig. 3. Pareto fronts for the multi-objective optimization problems for allocation of the shapes in the example business process diagram: 6 rows (+), 7 rows (x), and 8 rows (o).

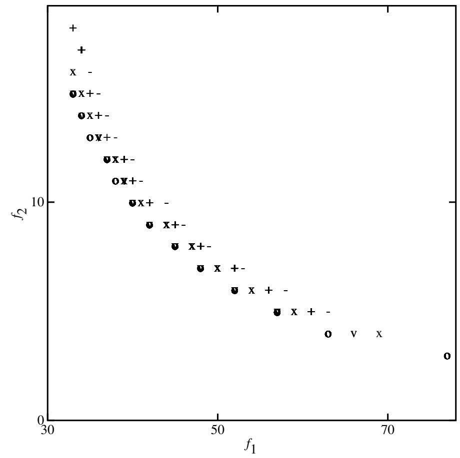

Fig. 4. Pareto fronts for the multi-objective optimization problems for allocation of the shapes in a middle size business process diagram: 5 rows (-), 6 rows (+), 7 rows (x), 8 rows (v), and more than 8 rows (o).

The shortest total length solution with objective vector $(24, 10)$ is shown in Fig. 5(a). It can be seen that connected flow objects are placed near by, often in the same row what violates direction down. The smallest number of violations in 7 rows solution with objective vector $(33, 1)$ and in 8 rows solution with objective vector $(32, 1)$ are shown in Figs. 5(c) and 5(d). There is only one violation coming from the last column to the previous one violating right flow. All the other connectors come down and in the same swimlane or directing to the right. The additional row enabled the move of flow objects in swimlane Role2 and Role3 down by one row and rearrangement of flow objects in swimlane Role1 to save total length. A non-dominated intermediate solution with the objective vector $(28, 4)$ is shown in Fig. 5(b). Balance between flow violations and the total length can be seen.

We continue our experiments with a middle size business process diagram shown in Section 2. The data of the problem are given in Table 4. There are 6 functions in this business process, there is one function with 5 shapes, two with 4 shapes and three with 2 shapes. The smallest number of rows to fit the business process diagram is 5. Even with this number of rows the number of different assignments of shapes to rows is $13\,824\,000\,000$.

The results of branch and bound algorithm are shown in Table 5. The time grows approximately twice when the number of rows is increased by one, but the growth speed decreases after 12 rows. Anyway the problem with the largest required number of rows (19) is solved in approximately two and a half hours which is acceptable for off-line improvement of business process diagrams.

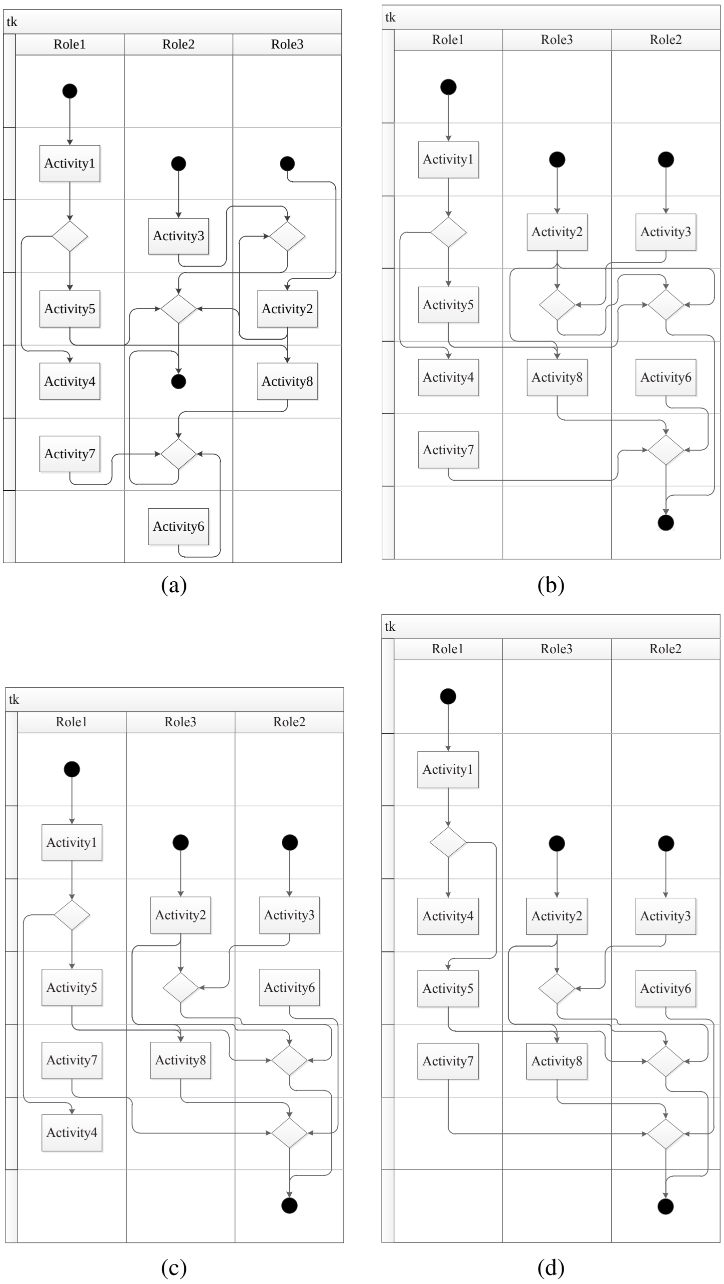

Fig. 5. Solutions of the multi-objective problem of shape allocation in the example business process diagram: (a) the shortest total length $(24, 10)$, (b) intermediate solution $(28, 4)$, (c) the smallest number of violations with 7 rows $(33, 1)$, (d) the smallest number of violations with 8 rows $(32, 1)$.

Table 4. Data of middle size business process diagram.

Flow objects

No.	Role	Description
1	Department	Start
2	Department	Responsible person prepares document project
3	Department	Head of dep. revises document
4	Department	Is approved1
5	Department	Responsible person redirects document for sequential approval
6	Management Department	Linguist revises document
7	Management Department	Is approved2
8	Management Department	Accountant revises document
9	Management Department	Is approved3
10	Law & Personnel Department	Head of dep. revises document
11	Law & Personnel Department	Is approved4
12	Department related to document	Head of dep. revises document
13	Department related to document	Is approved5
14	Management Department	Director revises document
15	Management Department	Is approved6
16	Management Department	Referent registers document
17	Management Department	Referent redirects document for familiarization
18	Personnel	Employees familiarize & sign in registry book
19	Personnel	End

Connections

i	k_{i1}	k_{i2}	i	k_{i1}	k_{i2}	i	k_{i1}	k_{i2}
1	1	2	9	7	8	17	13	5
2	2	3	10	8	9	18	13	14
3	3	4	11	9	5	19	14	15
4	4	2	12	9	10	20	15	5
5	4	5	13	10	11	21	15	16
6	5	6	14	11	5	22	16	17
7	6	7	15	11	12	23	17	18
8	7	5	16	12	13	24	18	19

Table 5. Results of the branch and bound algorithm solving multi-objective shape allocation problem in a middle size business process diagram.

Number of rows	t, s	NFE	Number of rows	t, s	NFE
5	0.87	$11846524 + 720$	13	1175.44	$8072969995 + 720$
6	8.76	$87341601 + 720$	14	1845.78	$11516056991 + 720$
7	20.10	$267553983 + 720$	15	2746	$15764528221 + 720$
8	37.05	$473246383 + 720$	16	3825	$20848903023 + 720$
9	76.96	$997982630 + 720$	17	5182	$26788986132 + 720$
10	193.69	$1946020628 + 720$	18	6817	$33597007137 + 720$
11	394.98	$3386280514 + 720$	19	8670	$41280000441 + 720$
12	751.75	$5496804470 + 720$			

Pareto fronts for problems with different number of rows are illustrated in Fig. 4. Pareto solutions of the problem with 5, 6 and 7 rows are dominated by that of the problems with more rows. This means that the problem with less than 8 rows are too restrictive. The non-dominated solution with the objective vector $(77, 3)$ appears when more than 12 rows are allowed.

The shortest total length solution with objective vector $(33, 15)$ is shown in Fig. 6. Connected flow objects are placed near by, but flow violations are quite often. The solution with the smallest number of flow violations $(77, 3)$ is shown in Fig. 7. The desirable flow in top-down left-right direction is clearly visible, only unavoidable violations appear due to the cycles in the process. However this diagram requires a larger number of rows than the one with the shortest total length.

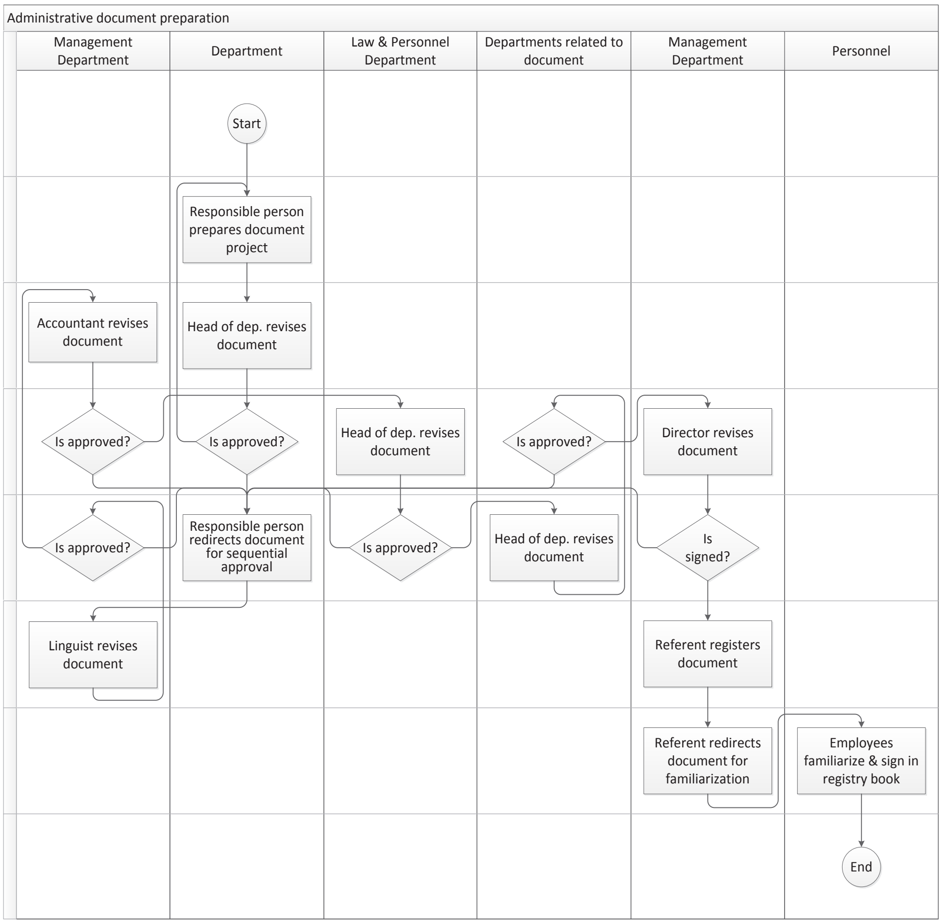

Fig. 6. Solution with the shortest total length $(33, 15)$ of the multi-objective problem of shape allocation in a middle size business process diagram.

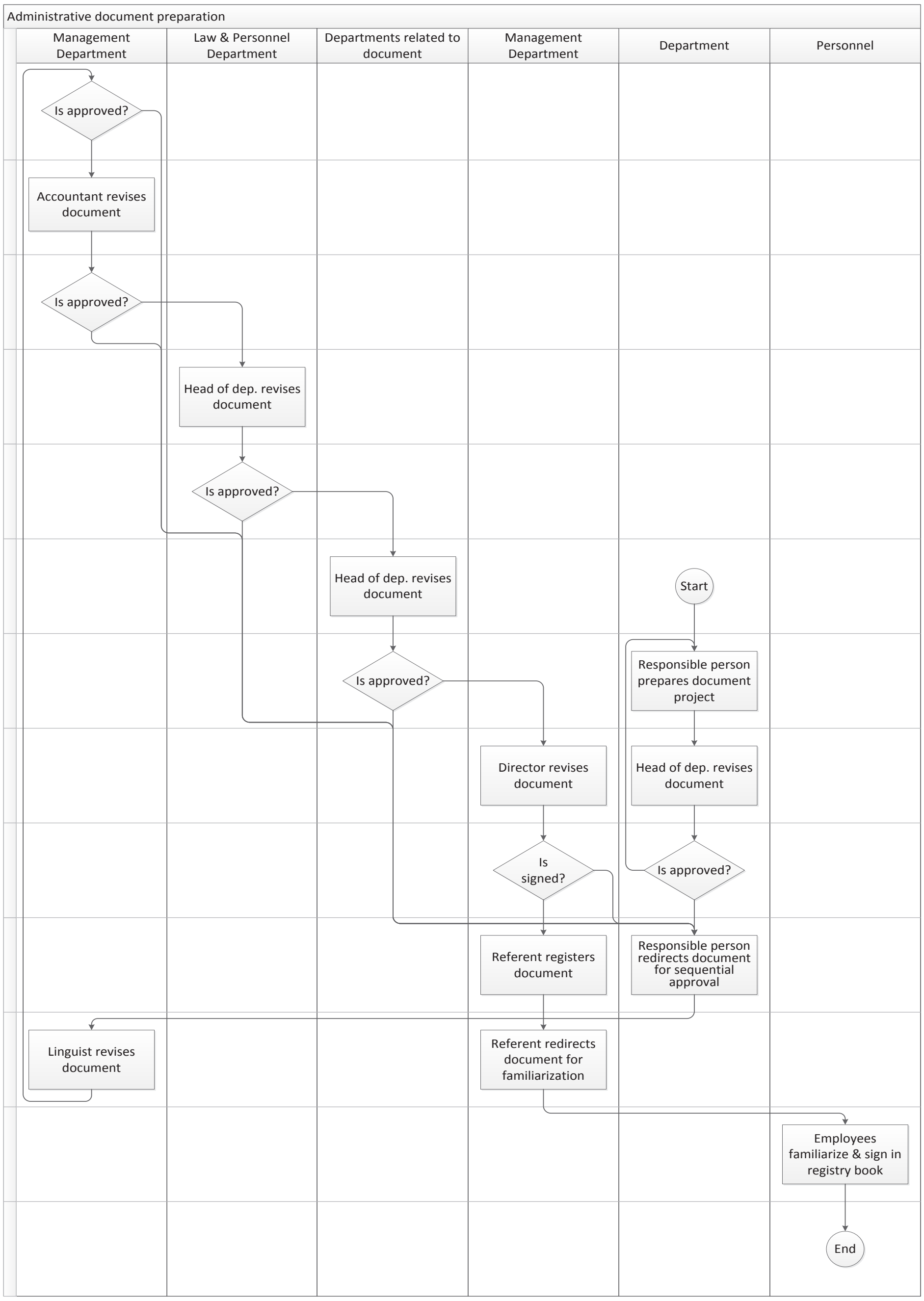

Fig. 7. Solution with the smallest number of flow violations $(77, 3)$ of the multi-objective problem of shape allocation in a middle size business process diagram.

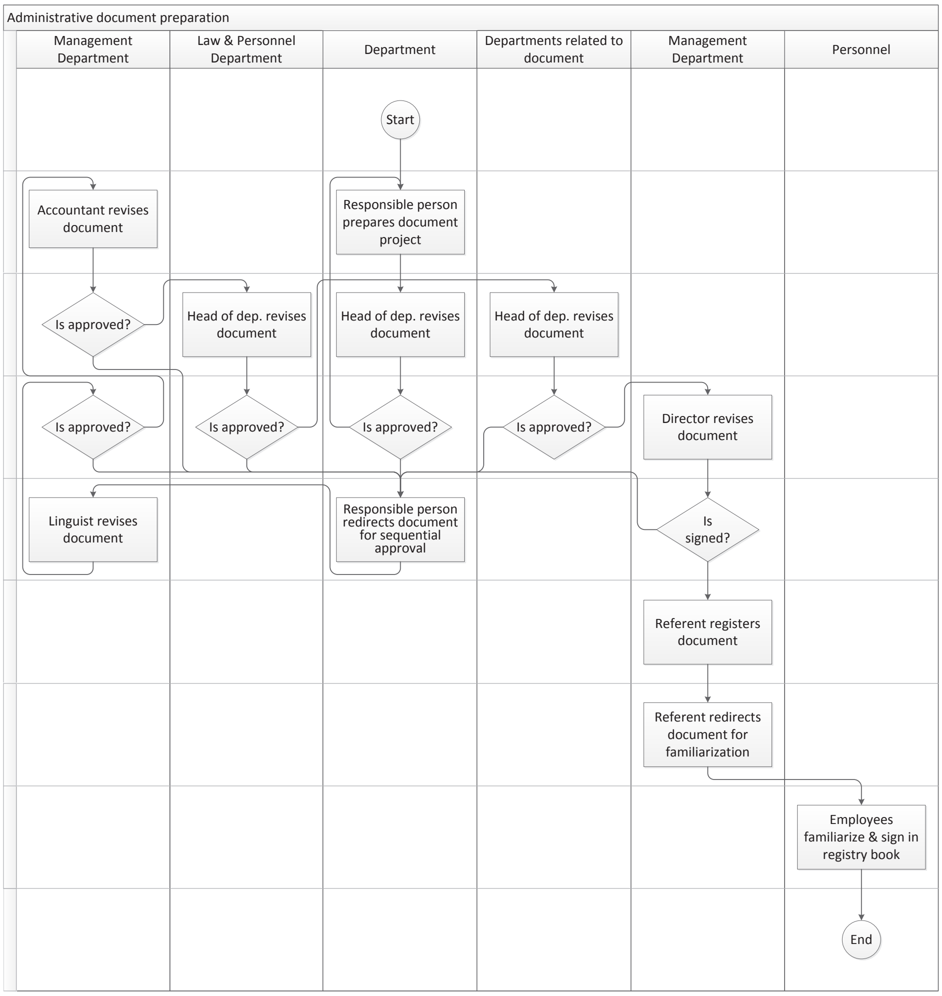

Fig. 8. Non-dominated intermediate solution with the objective vector $(38, 11)$ of the multi-objective problem of shape allocation in a middle size business process diagram.

One of the non-dominated solutions with the objective vector $(38, 11)$ is illustrated in Fig. 8. Similarly to the smaller problem a balance between flow violations and total length of connectors can be seen.

6 Conclusions

A multi-objective optimization algorithm is adapted to the solution of a particular problem of the aesthetic drawing of graphs related to business process diagrams. A problem of the aesthetic allocation of vertices in a planar region is considered, modeling the problem of the allocation of flow objects in swimlanes of the work pool, and allocation of the swimlanes with respect to each other. The advantage of the proposed algorithm is in the approximation of the whole Pareto set enabling a decision maker to settle an appropriate trade off between the considered objectives.

Acknowledgment. The remarks of two reviewers were very helpful to improve the readability of the paper.

References

1. J. Hooker, *Logic-Based Methods for Optimization: Combining Optimization and Constraint Satisfaction*, John Wiley & Sons, New York, NY, 2000.
2. H.-Y. Chen, Y.-W. Chang, Global and detailed routing, in: L.-T. Wang, Y.-W. Chang, K.-T. Cheng (Eds.), *Electronic Design Automation: Synthesis, Verification, and Testing*, Morgan Kaufmann, Burlington, MA, 2009, pp. 687–749.
3. G.D. Battista, P. Eades, R. Tamassia, I.G. Tollis, *Graph Drawing: Algorithms for the Visualization of Graphs*, Prentice Hall, Upper Saddle River, NJ, 1998.
4. Ch. Bennett, J. Ryall, L. Spalteholz, A. Gooch, The aesthetics of graph visualization, in: D.W. Cunningham, G. Meyer, L. Neumann (Eds.), *Computational Aesthetics in Graphics, Visualization, and Imaging*, Morgan Kaufmann, Burlington, MA, 2007, pp.1–8.
5. H.C. Purchase, Metrics for graph drawing aesthetics, *J. Visual Lang. Comput.*, **13**(5):501–516, 2002.
6. H.C. Purchase, M. McGill, L. Colpoys, D. Carrington, Graph drawing aesthetics and the comprehension of UML class diagrams: An empirical study. in: *Proceedings of the 2001 Asia-Pacific Symposium on Information Visualisation*, Vol. 9, 2001, pp. 129–137.
7. V. Jančauskas, A. Mackutė-Varoneckienė, A. Varoneckas, A. Žilinskas, On the multi-objective optimization aided drawing of connectors for graphs related to business process management. *Commun. Comput. Inf. Sci.*, **319**:87–100, 2012.
8. A. Žilinskas, A. Mackutė-Varoneckienė, A. Varoneckas, Weighting criteria of aesthetic attractiveness of layouts of business process diagrams, in: *Stochastic Programming for Implementation and Advanced Applications, International Workshop STOPROG, July 3–6, 2012, Neringa, Lithuania*, Technika, Vilnius, 2012, pp. 142–147.
9. M. Garey, D. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W.H. Freeman & Co, New York, NY, 1979.
10. P. Hansen, Bicriterion path problems, in: G. Fandel, T. Gal (Eds.), *Multiple Criteria Decision Making: Theory and Applications*, Lect. Notes Econ. Math. Syst., Vol. 177, Springer-Verlag, Berlin, New York, 1980, pp. 109–127.
11. S.M. Douiri, S. Elbernoussi, An unconstrained binary quadratic programming for the maximum independent set problem. *Nonlinear Anal. Model. Control*, **17**(4):410–417, 2012.

12. T.F. Gonzalez, *Handbook of Approximation Algorithms and Metaheuristics*, Chapman & Hall/CRC, Boca Raton, FL, 2007.

13. J. Hromkovic, *Algorithms for Hard Problems. Introduction to Combinatorial Optimization, Randomization, Approximation, and Heuristics*, Springer-Verlag, Berlin, Heidelberg, 2004.

14. V. Jančauskas, G. Kaukas, A. Žilinskas, J. Žilinskas, On multi-objective optimization aided visualization of graphs related to business process diagrams. in: *Databases and Information Systems. Tenth International Baltic Conference on Databases and Information Systems, July 8–11, 2012, Vilnius, Lithuania*, Žara, Vilnius, 2012, pp. 71–80.

15. R. Paulavičius, J. Žilinskas, A. Grothey, Investigation of selection strategies in branch and bound algorithm with simplicial partitions and combination of Lipschitz bounds, *Optim. Lett.*, **4**(2):173–183, 2010, doi: 10.1007/s11590-009-0156-3.

16. A. Žilinskas, J. Žilinskas, Branch and bound algorithm for multidimensional scaling with city-block metric, *J. Glob. Optim.*, **43**(2):357–372, 2009, doi: 10.1007/s10898-008-9306-x.