Trajectory Decision Making Framework

Mark Sh. Levin

Inst. for Information Transmission Problems, Russian Academy of Sciences, Moscow 127051, Russia

Abstract—The paper addresses a general view to trajectory (route) decision making framework (i.e., designing a trajectory). The view is based on four-part morphological scheme: (a) routing combinatorial problems (e.g., shortest path problem, minimum spanning tree problems, TSP), (b) assessment scales (i.e., quantitative, ordinal, poset-like), (c) graph/network based models as “solving space” (e.g., k-partite graph), and (d) node/vertex models/types. The following issues are considered: (i) structuring the “design/solving space”, (ii) problem statement, (iii) heuristics. A realistic university student trajectory problem (route from BS degree to PostDoc position) is examined.

Key words: decision making, combinatorial optimization, routing, trajectory design, heuristic, student trajectory

DOI: 10.53921/18195822.2021.21.4.265

1. INTRODUCTION

In recent decades, the role of dynamic decision making has been increased (e.g., [12, 32, 37]). Planning and scheduling problems play central roles in many domains, for example: (a) motion planning, navigation (e.g., [6, 15, 41, 83]), (b) digital system design (VLSI design) (e.g., [88]), (c) network routing (e.g., [14, 23, 50]), (d) information retrieval (e.g., [9]), (e) scenario-planning (e.g., [11, 90]), (f) path design in biomedicine (e.g., [53, 58–60]). Note, special advanced integrated planning and scheduling systems (APS) are widely used in manufacturing and supply management (e.g., [25, 65, 80]). Petri nets based approaches are often used for workflows modeling (e.g., in manufacturing systems) (e.g., [69, 96]). Multi-stage decision making systems for complicated problems have been discussed in [84]. In [86], reflexive game theory has been proposed for modeling of multistage decision processes. Many methods based on artificial intelligence have been developed for multistage planning, for example: (i) Hierarchical Task Network (HTN) planning [21, 29, 30], (ii) network languages for complex systems [81]. In general, the complexity of the methods above is very high.

This article focuses on a special trajectory (multistage, discrete) decision making (DM) framework that is based on a digraph with five-types of vertices (nodes). The following issues are considered: (a) brief description of routing problem(s), (b) structuring the “design/solving space”, and (c) heuristic solving schemes. A real world student trajectory problem (designing a student route from BS degree to PostDoc position) on the basis of a multicriteria orienteering model is examined (i.e., problem description and statement, solving heuristic, realistic numerical example). The paper is based on the preliminary electronic preprint [57] and the conference paper [58].

2. TRAJECTORY DECISION MAKING SCHEME

A generalized four-part scheme (morphological structure) of the examined domain (route/trajectory decision making problems) is depicted in Fig. 1.
Part 1. Some basic route-based combinatorial problems:
1. shortest path problem [8, 17, 26, 85];
2. minimum spanning tree problem [17, 26, 97];
3. minimum Steiner tree problem [16, 17, 26, 35, 38, 61];
4. travelling salesman problem (TSP) [17, 26, 40];
5. longest path problem [17, 26, 40, 94];
6. maximum leaf spanning tree [22, 24, 26, 45];
7. vehicle routing problem (VRP) [2, 47, 87, 91];
8. orienteering problem [1, 31, 33, 89];
9. traveling purchaser problem [67, 68, 76];
10. cluster routing problem [36, 95];
11. path planning (e.g., with task assignment) [7, 48].

Part 2. Assessment scales for problem parameters [56]:
1. quantitative scales; 2. ordinal scales; 3. vector scales;
4. poset-like scales (e.g., multiset estimates);
5. stochastic estimates; 6. fuzzy set based estimates.

Part 3. Models of “design/solving space” [56, 57]:
A. $k$-part graph/network:
   A1. one-part graph/network, A2. multi-part graph/network;
B. $k$-layer graph-network:
   B1. one-layer model (e.g., graph/digraph/network),
   B2. multi-layer models (e.g., two-layer network).

Part 4. Types of node/agent model [56–58, 66]:
1. node/vertex;
2. vertex & design alternatives (e.g., as in “and-or” graph;
   in multistage design of modular systems);
3. vertex & hierarchy of design alternatives;
4. two/three component node: (i) “analysis/diagnosis”,
   (ii) “design/implementation”, (iii) “analysis/decision”.

For example, two problem kinds can be pointed out as basic “physical” metaphors: (i) automobile
routing problems (e.g., [10, 46]), (ii) team orienteering problems (e.g., [1, 31, 33, 42, 63, 89]).
Table 1 contains a list of trajectory or route-like decision making problems. In the considered route DM
problems, it is necessary to do the following: (1) generation of “design/solving space” (i.e., states,
transmission operations); (2) specification of the goal as a resultant point (or a set of goal points);
(3) design of the route at the “design/solving ‘space”, and (4) analysis of the route implementation
and the route modification (if needed).

3. ROUTE DECISION MAKING PROBLEMS

3.1. Basic problems

An illustration for the basic routing decision making problem is depicted in Fig. 2. Here, directed
graph $G = (H, E)$ is given ($H$ is vertex set, $E$ is arc set); initial vertex (origin) $h^0 \in H$ and goal
vertex (destination) $h^g \in H$ are pointed out; each arc $e \in E$ has a length (i.e., nonnegative weight,
cost) $\lambda(e)$. The basic problem is:
Find the route (path) from vertex $h^0$ to vertex $h^g$ $L = \langle h^0, ..., h^g \rangle$ that minimizes the total length (cost) of the path (i.e., the sum of the path arcs weights).

<table>
<thead>
<tr>
<th>No.</th>
<th>Problem</th>
<th>Source(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Basic “reference” applied problems:</td>
<td></td>
</tr>
<tr>
<td>1.1</td>
<td>“Physical” route (e.g., car/robot route)</td>
<td>[1, 46]</td>
</tr>
<tr>
<td>1.2</td>
<td>Control trajectory in parameter space</td>
<td>[20, 71]</td>
</tr>
<tr>
<td>1.3</td>
<td>Search strategy in problem solving</td>
<td>[64, 70]</td>
</tr>
<tr>
<td>1.4</td>
<td>Urban route (choice, planning)</td>
<td>[5, 73, 77, 98]</td>
</tr>
<tr>
<td>1.5</td>
<td>Trajectory for mobile robot movement</td>
<td>[6, 41]</td>
</tr>
<tr>
<td>1.6</td>
<td>Mission of airplane/aerospace apparatus (e.g., unmanned aerial vehicles UAVs)</td>
<td>[39, 43, 44]</td>
</tr>
<tr>
<td>1.7</td>
<td>Ship trajectories</td>
<td>[82, 83]</td>
</tr>
<tr>
<td>1.8</td>
<td>Data path synthesis in digital systems (e.g., VLSI design)</td>
<td>[88]</td>
</tr>
<tr>
<td>1.9</td>
<td>X-cast routing (anycast, broadcast, multi-broadcast, multicast, unicast, geocast)</td>
<td>[14, 23, 50, 56, 62]</td>
</tr>
<tr>
<td>2.</td>
<td>Prospective problems:</td>
<td></td>
</tr>
<tr>
<td>2.1</td>
<td>Motion planning, navigation (urban traffic planning, automobile routing, robot motion planning, inspection path planning, etc.)</td>
<td>[15, 41]</td>
</tr>
<tr>
<td>2.2</td>
<td>Tourism route planning</td>
<td>[27, 28]</td>
</tr>
<tr>
<td>2.3</td>
<td>Search trajectory in information systems</td>
<td>[9, 56]</td>
</tr>
<tr>
<td>2.4</td>
<td>Scenario/multistage scenario</td>
<td>[11, 90]</td>
</tr>
<tr>
<td>2.5</td>
<td>Trajectory of educational objects (e.g., student plan, course improvement, trajectory, trajectory of research/educational center changes)</td>
<td>[52, 53]</td>
</tr>
<tr>
<td>2.6</td>
<td>Plans in biomedicine (e.g., medical treatment, immunological analysis)</td>
<td>[53, 56, 58–60]</td>
</tr>
<tr>
<td>2.7</td>
<td>Trajectory of modular system development</td>
<td>[53, 56]</td>
</tr>
<tr>
<td>2.8</td>
<td>Trajectory of organizational-economical objects (e.g., start-up company, teams, objects (e.g., start-up company, team,</td>
<td>[56]</td>
</tr>
<tr>
<td>2.9</td>
<td>Plans of monitoring processes (e.g., system testing/inspection, maintenance)</td>
<td>[43, 79]</td>
</tr>
</tbody>
</table>

Several polynomial algorithms were suggested for the problem (including polynomial algorithms for multi-objective versions) (e.g., [17, 26, 85]). Generally, the following support problems can be pointed out: (i) building the route (design), (ii) analysis of the route implementation and the online route modification (correction). Note, basic simplification approach consists in partitioning the initial solving “space” into series of “subspaces”. Evidently, other versions of route decision making problems can be examined as well, for example: (a) multi-goal, multi-route problem (Fig. 3), (b) orienteering problem (Fig. 4), (c) route problem over multi-layer “solving space” (Fig. 5).

Some basic route DM models (and their variants) are well-known, for example (Fig. 1) [17, 26, 47, 87]: (a) traveling salesman problem(s); (b) longest path problem; (c) minimum spanning tree problem(s); (d) maximum leaf spanning tree problem; (e) vehicle routing problem(s) (VRPs). Further, the orienteering problem and its modifications will be used as a basic one (main applied domains: logistics, sport, tourism) (e.g., Fig. 3) [1, 28, 31, 89]. In fact, the problem integrates knapsack problem and TSP. Here, digraph $G = (H, E)$ ($|H| = n$) is given, each vertex $h \in H$ has a nonnegative score (profit) $\theta(h)$, each edge/arc $e \in E$ has a nonnegative length (cost, travel time) $\lambda(e)$.

The problem is:

Find a route (a path from the start point $h^0 \in H$ to the end point $h^g \in H$) over a subset of the most important graph vertices that maximizes the sum of the scores of the selected vertices while taking into account a constraint for total route length (total cost) (i.e., combination of knapsack problem and TSP).
The mathematical model is formulated as follows: \( H = \{1, \ldots, i, \ldots, n\} \) is the set of vertex/nodes, vertex 1 is the start point of the route, vertex \( n \) is the end (goal) point of the route, binary variable \( x_{ij} = 1 \) if the built route (path) contains arc \((i, j)\) and \( x_{ij} = 0 \) otherwise (vertex \( i \) precedes \( j \), \( \theta_i \) is the vertex profit, \( \lambda_{ij} \) is the arc cost (if arc \((i, j) \in E\) ), \( d \) is a distance constraint for the built path). The model is:

\[
\begin{align*}
\max & \sum_{i=1}^{n} \sum_{j=1}^{n} \theta_i x_{ij} \\
\text{s.t.} & \sum_{j=2}^{n} x_{1j} = \sum_{i=1}^{n-1} x_{in} = 1; \\
& \sum_{i=2}^{n-1} x_{ik} = \sum_{j=2}^{n-1} x_{kj} \leq 1, \ k = 2, n-1; \\
& \sum_{i=1}^{n} \sum_{j=1}^{n} \lambda_{ij} x_{ij} \leq d; \\
& x_{ij} \in \{0, 1\}, \ i = 1, n, \ j = 1, n.
\end{align*}
\]

The problem is NP-hard [31]. Multicriteria problem statement can be examined as well (e.g., the score of each vertex is a vector estimate and the objective function is a vector based on the score components summarization).

3.2. Structuring of design space

The “design/solving space” is modeled as a digraph/network. The following possible extensions of “design/solving space” are used [56, 57]: 1. multi-layer structure of “design/solving space” (Fig. 5); 2. multi-domain case (or multi-part digraph/network) (Fig. 6); 3. combined case. Three-domain trajectory/route DM problem based on different basic combinatorial optimization route problems (TSP, orienteering problem, shortest path) is depicted in Fig. 6.
3.3. Types of nodes and illustration of routing

In general, the following five types of elements (i.e., node/vertex models) are considered (Fig. 7) [56–58, 66]:

1. Vertex/node ($\mu_i$) (Fig. 7a). This case corresponds to traditional situation when a digraph is used (e.g., in the shortest path problem).

2. Vertex/node ($\mu_i$) with corresponding design alternatives $\{A_i^1, ..., A_i^{q_i}\}$ (problem: selection of the best design alternative for the vertex) (Fig. 7b). This case can be used in routing in “and-or” digraphs [66], in network routing with selection of the best communication protocol at each network node [55, 56].

3. Vertex/node ($\mu_i$) and corresponding hierarchy of design alternatives $\Lambda^\mu_i$ (problem: composition of the best composite design alternative(s) on the basis of hierarchy above) (Fig. 7c). This case can be used in network routing with hierarchical modular design of the implemented communication protocol at each node, in combinatorial planning of immunoassay technology [53, 55, 56, 60].

4. Composite (multi-component) vertex (i), for example: two components as follows: (a) “design/implementation” part ($\mu_i$) (problem: composition of the best composite design alternative on the basis of hierarchy above to implement) and (b) “analysis/decision” part ($\alpha_i$) (to analyze the result of the implementation above and selection of next way/path, usage of logical rules) (Fig. 7d).
This case can be used in combinatorial planning of medical treatment (i.e., design/implementation and analysis) [57, 58].

5. Composite (multi-component) vertex (i), for example: three components as follows: (a) “diagnosis” part (βi) (problem: composition of the best composite diagnosis alternative on the basis of hierarchy, implementation of diagnosis procedure), (b) “design/implementation” part (μi) (problem: composition of the best composite design alternative on the basis of hierarchy above to implement), and (c) “analysis/decision” part (αi) (to analyze the result of the implementation above and selection of next way/path, based on logical rules) (Fig. 7e). This case can be used in system maintenance, in medical treatment [58].

Two numerical examples illustrate the routing processes for case 2 and for case 3.

**Example 1.** An illustrative example for routing based on design alternatives at each graph/network vertex is the following. For each vertex, the resultant design alternative can be selected in online mode or on the basis of off-line solving process [56, 57]. Here, each vertex of “design space” corresponds to Fig. 7b. The example involves the following (Fig. 8):

(i) \( G = (H, E) \) is a digraph, vertex set \( H = \{\mu_1, \mu_2, \mu_3, \mu_4, \mu_5, \mu_6, \mu_7, \mu_8\} \), arc set \( E = \{ (\mu_1, \mu_2), (\mu_2, \mu_3), (\mu_1, \mu_4), (\mu_2, \mu_5), (\mu_3, \mu_5), (\mu_3, \mu_6), (\mu_4, \mu_6), (\mu_3, \mu_7), (\mu_5, \mu_6), (\mu_5, \mu_7), (\mu_6, \mu_7), (\mu_6, \mu_8), (\mu_7, \mu_8) \} \);

(ii) \( \mu_1 \) is an initial point; \( \mu_8 \) is a goal point;

(iii) there exist three design alternatives for each vertex \( \mu_i \) (i = 1, 8): \( A^\mu_{1_1}, A^\mu_{1_2}, A^\mu_{1_3} \);

(iv) selected alternatives (for each vertex) are: \( A^\mu_{1_1}, A^\mu_{1_2}, A^\mu_{1_3}, A^\mu_{2_1}, A^\mu_{2_2}, A^\mu_{2_3}, A^\mu_{2_4}, A^\mu_{2_5}, A^\mu_{2_6}, A^\mu_{2_7}, A^\mu_{2_8} \) (in Fig. 9, the alternatives are pointed out by “oval”);

(v) the designed global route (by vertices) is: \( L = < \mu_1, \mu_2, \mu_5, \mu_6, \mu_8 > \); and

(vi) the resultant route consisting of design alternatives is: \( \hat{L} = < A^\mu_{1_1}, A^\mu_{1_2}, A^\mu_{1_3}, A^\mu_{2_1}, A^\mu_{2_2}, A^\mu_{2_3} > \).

In this problem, the selected design alternatives of neighbor path vertices have to be “good” compatible as in combinatorial synthesis approach (morphological clique problem) [53, 56].

**Example 2.** An illustrative example for routing based on hierarchy of design alternatives at each graph/network vertex is the following. For each vertex, the resultant set of design alternative can be designed in online mode or on the basis of offline solving process. Here, each vertex of “design space” corresponds to Fig. 7d. In [56], this problem is examined as multi-stage design of modular systems. The example involves the following (Fig. 9):

(i) \( G = (H, E) \) is a digraph, vertex set \( H = \{\mu_1, \mu_2, \mu_3, \mu_4, \mu_5, \mu_6, \mu_7\} \), arc set \( E = \{ (\mu_1, \mu_2), (\mu_2, \mu_3), (\mu_2, \mu_4), (\mu_3, \mu_4), (\mu_4, \mu_5), (\mu_5, \mu_6), (\mu_5, \mu_7) \} \);

(ii) \( \mu_1 \) is an initial (origin) point; \( \mu_6 \) is a goal (destination) point;

(iii) there exists a hierarchy of design alternatives for each vertex \( \mu_i \) (i = 1, 5): \( \Lambda^\mu_i \);

(iv) three design alternatives are composed for each vertex \( \mu_i \) (i = 1, 6): \( A^\mu_{1_1}, A^\mu_{1_2}, A^\mu_{1_3} \);
(v) selected alternatives (for each vertex) are: $A_1^{t_1}$, $A_3^{t_2}$, $A_1^{t_3}$, $A_2^{t_4}$, $A_1^{t_5}$, $A_2^{t_6}$ (in Fig. 9, the alternatives are pointed out by “oval”);

(vi) the designed global route (by vertices) is: $L = < \mu_1, \mu_2, \mu_4, \mu_5, \mu_6 >$; and

(vii) the resultant route consisting of design alternatives is: $\bar{L} = < A_1^{t_1}, A_3^{t_2}, A_2^{t_4}, A_1^{t_5}, A_2^{t_6} >$.

In this problem, the selected design alternatives of neighbor path vertices have to be “good” compatible (combinatorial synthesis based on morphological clique problem) [52,53,56].

4. BASIC SOLVING STRATEGIES

For basic routing problems, the following evident strategy can be used:

Strategy 0. Evident route strategy:
(0.1) design of a set of routes,
(0.2) selection of the best route.

In the case of complex nodes, the solving strategy has to contain special stages for analysis and selection/design of design alternatives at each node/vertex of the “solving space”. Let us consider a one-layer route DM problem with nodes as “vertex& alternatives” and “vertex& hierarchy alternatives”. The following two basic solving strategies can be pointed out for the problem type:

Strategy 1. “Global” route strategy:
(1.1) designing a “global” route over graph vertices,
(1.2) selection/design of the best design alternative for each graph vertex of the “global” route,
(1.3) composition of a resultant route from the best alternatives for each graph vertex of the “global” route.

Strategy 1 is illustrated in Example 1 (Fig. 8) [56,57]. The approach was used for multi-stage design of modular systems [56].

Strategy 2. Extended digraph strategy:
(2.1) transformation/extension of the initial digraph (e.g., additional arcs, modification/extension of the alternatives hierarchies for the node(s)),
(2.2) designing the best route over the obtained extended graph/network.

Clearly, this strategy increases the problem dimension.

5. STUDENT EDUCATIONAL TRAJECTORY

In recent years, the attention was paid for educational decision making problems (e.g., curriculum design, students performance analysis, classification of students, composition of modular courses, program evaluation, student careers analytics, analysis of paths in student databases, student team formation, student careers planning, educational data mining, etc.) (e.g., [3, 4, 13, 18, 19, 34, 53, 54, 72, 74, 75, 78, 92, 93]). Here a simplified personalized plan (educational trajectory) for a BS student of Moscow Inst. of Physics and Technology (State Univ.) (Faculty of Radio Engineering and Cybernetics) is examined [57]:

Route from BS degree point to PostDoc position.
The morphological scheme of the problem is: (i) orienteering problem, (ii) ordinal and vector estimates, (iii) six-part digraph, (iv) simple node model. A BS degree in Communication Engineering is considered as initial point \( a_1 \) (origin), a PostDoc position in an university is considered as the goal point (destination). Table 2 contains educational points/nodes of the “solving space” including their characteristics and estimates upon criteria (ordinal scale \([1, 5]\), 5 corresponds to the best level): (a) quality of educational program (i.e., a set of disciplines, basic lectures, seminars) \( C_1 \) (estimate \( \theta^1 \)), (b) possible research results (including publication activity) \( C_2 \) (estimate \( \theta^2 \)), and (c) integrated index of professional degree prestige (World University Rating, quality of professional education/research, scientific school(s), etc.) \( C_3 \) (estimate \( \theta^3 \)).

<table>
<thead>
<tr>
<th>Node/vertex</th>
<th>Degree level</th>
<th>Educational institution</th>
<th>Professional domain</th>
<th>Time (years) ( \tau )</th>
<th>Estimates ( \theta^1 )</th>
<th>( \theta^2 )</th>
<th>( \theta^3 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( a_1 )</td>
<td>BS</td>
<td>MIPT, Russia</td>
<td>Commun. Eng.</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( b_1 )</td>
<td>MS</td>
<td>MIPT, Russia</td>
<td>Commun. Eng.</td>
<td>2</td>
<td>5</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>( b_2 )</td>
<td>MS</td>
<td>MIPT, Russia</td>
<td>Appl. Math.</td>
<td>2</td>
<td>5</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>( b_3 )</td>
<td>MS</td>
<td>USA univ.</td>
<td>Commun. Eng.</td>
<td>2</td>
<td>3</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>( b_4 )</td>
<td>MS</td>
<td>USA univ.</td>
<td>Inform. Syst.</td>
<td>2</td>
<td>3</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>( b_5 )</td>
<td>MS</td>
<td>Canadian univ.</td>
<td>Commun. Eng.</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>( b_6 )</td>
<td>MS</td>
<td>UK univ.</td>
<td>OR/Algorithms</td>
<td>2</td>
<td>4</td>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td>( b_7 )</td>
<td>MS</td>
<td>German univ.</td>
<td>Inform. Syst.</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>( g_1 )</td>
<td>MS (2nd)</td>
<td>USA univ.</td>
<td>Commun. Eng.</td>
<td>1</td>
<td>3</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>( g_2 )</td>
<td>MS (2nd)</td>
<td>USA univ.</td>
<td>Inform. Syst.</td>
<td>1</td>
<td>3</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>( g_3 )</td>
<td>MS (2nd)</td>
<td>Canadian univ.</td>
<td>Commun. Eng.</td>
<td>1</td>
<td>3</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>( g_4 )</td>
<td>MS (2nd)</td>
<td>UK univ.</td>
<td>OR/Algorithms</td>
<td>1</td>
<td>4</td>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td>( g_5 )</td>
<td>MS (2nd)</td>
<td>German univ.</td>
<td>Inform. Syst.</td>
<td>1</td>
<td>3</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>( w_1 )</td>
<td>PhD</td>
<td>MIPT, Russia</td>
<td>Commun. Eng.</td>
<td>3</td>
<td>5</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>( w_2 )</td>
<td>PhD</td>
<td>MIPT, Russia</td>
<td>Appl. Math.</td>
<td>3</td>
<td>5</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>( w_3 )</td>
<td>PhD</td>
<td>USA univ.</td>
<td>Commun. Eng.</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>( w_4 )</td>
<td>PhD</td>
<td>USA univ.</td>
<td>Inform. Syst.</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>( w_5 )</td>
<td>PhD</td>
<td>Canadian univ.</td>
<td>Commun. Eng.</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>( w_6 )</td>
<td>PhD</td>
<td>UK univ.</td>
<td>OR/Algorithms</td>
<td>3</td>
<td>5</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>( w_7 )</td>
<td>PhD</td>
<td>German univ.</td>
<td>Inform. Syst.</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>( f_1 )</td>
<td>PhD (2nd)</td>
<td>USA univ.</td>
<td>Commun. Eng.</td>
<td>2</td>
<td>4</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>( f_2 )</td>
<td>PhD (2nd)</td>
<td>USA univ.</td>
<td>Inform. Syst.</td>
<td>2</td>
<td>4</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>( f_3 )</td>
<td>PhD (2nd)</td>
<td>Canadian univ.</td>
<td>Commun. Eng.</td>
<td>2</td>
<td>4</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>( p_1 )</td>
<td>PostDoc</td>
<td>USA univ.</td>
<td>Commun. Eng.</td>
<td>3</td>
<td>3</td>
<td>5</td>
<td>5</td>
</tr>
</tbody>
</table>

In Fig. 10, the examined simplified “design/solving space” (as series of subspaces; expert judgment) and five route examples are depicted:

\[
L^{ex_1} = \langle a_1, b_1, w_1, p_1 \rangle, \quad L^{ex_2} = \langle a_1, b_3, g_2, w_2, p_1 \rangle, \quad L^{ex_3} = \langle a_1, b_5, g_4, w_5, f_2, p_1 \rangle, \quad L^{ex_4} = \langle a_1, b_6, g_5, w_6, p_1 \rangle, \quad L^{ex_5} = \langle a_1, b_7, g_7, f_3, p_1 \rangle. \]

Further, four basic “generalized” educational trajectories/routes are examined \((i = \overline{1,7}, \, k = \overline{1,5}, \, v = \overline{1,7}, \, \xi = \overline{1,3})\) (expert judgment):

\[
L^1 = \langle a_1, b_1, w_v, p_1 \rangle \quad \text{(Fig. 11a)};
\]

\[
L^2 = \langle a_1, b_1, g_k, w_v, p_1 \rangle \quad \text{(Fig. 11b)};
\]

\[
L^3 = \langle a_1, b_1, w_v, f_\xi, p_1 \rangle \quad \text{(Fig. 11c)}; \quad \text{and}
\]

\[
L^4 = \langle a_1, b_1, g_k, w_v, f_\xi, p_1 \rangle \quad \text{(Fig. 11d)}. \]
Fig. 10. “Design/solving space” (student trajectories)

(a) trajectory $L^1$  
(b) trajectory $L^2$

(c) trajectory $L^3$  
(d) trajectory $L^4$

Fig. 11. Basic general educational trajectories

Table 3, Table 4, Table 5 contain ordinal complexity estimates of movement between model nodes (scale $[1, 5]$, 5 corresponds to the most complex movement; absence of estimate corresponds to impossible movement: the digraph arc is absent).

First, a modification of orienteering problem (three objective functions, constraint for maximum arc length, constraint for aggregated (summarized) time of visited vertices) is considered as follows: $H = \{1, ..., i, ..., n\}$ is the set of vertex/nodes, vertex 1 is the start point of the route (origin), vertex $n$ is the end (goal) point of the route (destination), binary variable $x_{ij} = 1$ if the built route (path) contains arc $(i, j)$ and $x_{ij} = 0$ otherwise (vertex $i$ precedes $j$), $\theta_i$ is the vertex profit, $\lambda_{ij}$ is the arc cost (if arc $(i, j) \in E$), $d^{\text{max}}$ is a distance constraint for movement between neighbor vertices in the built route/path, $T$ is a time constraint for the built route as summarization of time costs of path vertices.

The basic model is:

$$\max \sum_{i=1}^{n} \sum_{j=1}^{n} \theta_i x_{ij}, \quad \max \sum_{i=1}^{n} \sum_{j=1}^{n} \theta_i^2 x_{ij}, \quad \max \sum_{i=1}^{n} \sum_{j=1}^{n} \theta_i^3 x_{ij},$$

s.t. $\sum_{j=2}^{n} x_{ij} = \sum_{i=1}^{n-1} x_{in} = 1$; $\sum_{i=2}^{n-1} x_{ik} = \sum_{j=2}^{n-1} x_{kj} \leq 1$, $k = \frac{2}{3}n - 1$;

$$\lambda_{ij} x_{ij} \leq d^{\text{max}} \forall i, j; \quad \sum_{i=1}^{n} \sum_{j=1}^{n} \tau_i x_{ij} \leq T; \quad x_{ij} \in \{0, 1\}, \ i = 1, n, \ j = 1, n.$$
Here, the Pareto-efficient solutions have to be searched for. Clearly, the problem is NP-hard. In our case, $a_1$ is the start point (i.e., graph vertex), $p_1$ is the end (goal) point (i.e., graph vertex). The optimization model has to be solved for each generalized trajectory above (i.e., $L_1$, $L_2$, $L_3$, $L_4$).

Let $L = < l_1, ..., l_i, ..., l_q >$ be an admissible route solution (i.e., educational trajectory). The characteristics of the solution are as follows:

(a) $\theta^1(L) = \sum_{i=2}^{q} \theta_i^1$ is integrated quality of education;
(b) $\theta^2(L) = \sum_{i=2}^{q} \theta_i^2$ is integrated quality of research results (including publication results);
(c) $\theta^3(L) = \sum_{i=2}^{q} \theta_i^3$ is integrated parameter of resultant prestige of the obtained academic degrees;
(d) $\tau(L) = \sum_{i=2}^{q} \tau_i$ is integrated required time (years) of the educational trajectory;
(e) $d(L) = \sum_{i=1}^{q-1} \lambda(l_i \rightarrow l_{i+1})$ is integrated estimate of movement complexity (between neighbor educational institutions).

As a result, the problem can be formulated as the following:

Find the route $L$ (solution) such that

(1) it fulfils two constraints: $\tau(L) \leq T$ and $d(L) \leq d_{\text{max}}$;

Table 3. Ordinal estimates of movement complexity $\lambda(a_1 \rightarrow b_i)$

<table>
<thead>
<tr>
<th>$a_1$ \ $b_i$</th>
<th>$a_1$</th>
<th>$b_1$</th>
<th>$b_2$</th>
<th>$b_3$</th>
<th>$b_4$</th>
<th>$b_5$</th>
<th>$b_6$</th>
<th>$b_7$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a_1$</td>
<td>1</td>
<td>2</td>
<td>4</td>
<td>4</td>
<td>5</td>
<td>5</td>
<td>3</td>
<td></td>
</tr>
</tbody>
</table>

Table 4. Ordinal estimates of movement complexity: $\lambda(b_i \rightarrow g_k)$, $\lambda(b_i \rightarrow w_v)$

<table>
<thead>
<tr>
<th>$b_i$ \ $g_k$, $w_v$, $f_j/p_1$</th>
<th>$g_1$</th>
<th>$g_2$</th>
<th>$g_3$</th>
<th>$g_4$</th>
<th>$g_5$</th>
<th>$w_1$</th>
<th>$w_2$</th>
<th>$w_3$</th>
<th>$w_4$</th>
<th>$w_5$</th>
<th>$w_6$</th>
<th>$w_7$</th>
<th>$f_1$</th>
<th>$f_2$</th>
<th>$f_3$</th>
<th>$p_1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$b_1$</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$b_2$</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$b_3$</td>
<td></td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$b_4$</td>
<td></td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$b_5$</td>
<td></td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$b_6$</td>
<td></td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$b_7$</td>
<td></td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 5. Ordinal estimates of movement: $\lambda(g_k \rightarrow w_v)$, $\lambda(w_v \rightarrow f_j)$, $\lambda(w_v \rightarrow p_1)$, $\lambda(f_j \rightarrow p_1)$

<table>
<thead>
<tr>
<th>$g_k$, $w_v$, $f_j/p_1$</th>
<th>$g_1$</th>
<th>$g_2$</th>
<th>$g_3$</th>
<th>$g_4$</th>
<th>$g_5$</th>
<th>$w_1$</th>
<th>$w_2$</th>
<th>$w_3$</th>
<th>$w_4$</th>
<th>$w_5$</th>
<th>$w_6$</th>
<th>$w_7$</th>
<th>$f_1$</th>
<th>$f_2$</th>
<th>$f_3$</th>
<th>$p_1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$g_1$</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$g_2$</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$g_3$</td>
<td>3</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$g_4$</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$g_5$</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$w_1$</td>
<td></td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$w_2$</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$w_3$</td>
<td></td>
<td></td>
<td>1</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$w_4$</td>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$w_5$</td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$w_6$</td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$w_7$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3</td>
<td>0</td>
<td>2</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$f_1$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$f_2$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$f_3$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
(2) it is a Pareto-efficient one by four criteria (objective functions):
max $\theta^1(L)$, max $\theta^2(L)$, max $\theta^3(L)$, min $d(L)$.

Note, the usage of educational generalized trajectories ($L^1$, $L^2$, $L^3$, $L^4$) leads to simplified/partitioned “solving space(s)”. As a result, the optimization problem can be transformed into a version of the multicriteria shortest path problem or multicriteria multiple choice problem. Thus, a “concurrent” general solving framework is used (Fig. 12).

In the example, a simplified heuristic is considered:

Stage 1. Searching for a solution with minimum $d(L)$ for each generalized trajectory ($L^1$, $L^2$, $L^3$, $L^4$). The resultant solutions and their estimates are presented in Table 6.

Stage 2. Selection of Pareto-efficient solutions:
$L^1_1 = <a_1, b_3, w_3, p_1>$, $L^2_1 = <a_1, b_1, g_1, p_1>$, $L^3_1 = <a_1, b_2, w_1, f_1, p_1>$, and $L^4_3 = <a_1, b_2, g_2, w_4, f_1, p_1>$.

Stage 3. Selection of the best solution (i.e., expert judgment).

In the example, solution $L^3_1 = <a_1, b_2, w_2, f_1, p_1>$ can be selected while taking into account obtained additional skills in applied mathematics (it may be crucial for the student future).

Table 6. Pareto-efficient solutions and their parameters (vector estimate)

<table>
<thead>
<tr>
<th>Route L</th>
<th>$\theta^1(L)$</th>
<th>$\theta^2(L)$</th>
<th>$\theta^3(L)$</th>
<th>$r(L)$</th>
<th>$d(L)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$L^1_1$ = $&lt;a_1, b_3, w_3, p_1&gt;$</td>
<td>10</td>
<td>15</td>
<td>15</td>
<td>8</td>
<td>7</td>
</tr>
<tr>
<td>$L^2_1$ = $&lt;a_1, b_1, g_1, w_3, p_1&gt;$</td>
<td>15</td>
<td>17</td>
<td>19</td>
<td>9</td>
<td>5</td>
</tr>
<tr>
<td>$L^3_1$ = $&lt;a_1, b_2, w_1, f_1, p_1&gt;$</td>
<td>15</td>
<td>18</td>
<td>19</td>
<td>10</td>
<td>5</td>
</tr>
<tr>
<td>$L^4_3$ = $&lt;a_1, b_1, g_1, g_7, f_1, p_1&gt;$</td>
<td>19</td>
<td>21</td>
<td>23</td>
<td>11</td>
<td>6</td>
</tr>
<tr>
<td>$L^5_2$ = $&lt;a_1, b_1, g_3, w_3, f_2, p_1&gt;$</td>
<td>19</td>
<td>21</td>
<td>23</td>
<td>11</td>
<td>6</td>
</tr>
<tr>
<td>$L^6_3$ = $&lt;a_1, b_2, g_2, w_4, f_1, p_1&gt;$</td>
<td>19</td>
<td>22</td>
<td>24</td>
<td>11</td>
<td>6</td>
</tr>
</tbody>
</table>

Fig. 12. Solving framework for educational trajectory

6. CONCLUSION

The paper suggests an approach (framework) for a special kind of multistage decision making based on a network with multi-type nodes. The approach is examined as a digraph based “intelligent” routing at special “design/solving space(s)”. A realistic example for designing a student trajectory illustrates the framework.

Some future research directions are: 1. analysis, modeling and usage of various kinds of “design/solving spaces” (including series composition of subspaces, dynamical “design/solving space”); 2. usage of various basic combinatorial routing problems (e.g., spanning trees problems, versions of TSP, generalized orienteering problem) for construction of the corresponding trajectory/route DM
problems; 3. study of multi-layer (hierarchical) “design (solving) spaces” and trajectory (route) DM problems over them; 4. special investigation of multiple vehicle routing problems (i.e., multi-domain problems) including coordination solving modes (e.g., as in multi-robot motion planning problems, in cooperative path planning for multiple UAVs); 5. usage of multistage DM problems for testing (inspection, maintenance) of networked systems; 6. applications of the examined multistage DM problems in economics/ management, e.g., modeling of firm (or project) development, forecasting, scenario planning; 7. design of a special support computer-aided system for the considered multistage DM problems; and 8. usage of the considered multistage DM problems in education (CS, applied mathematics, engineering, management).

The author states that there is no conflict of interest.

REFERENCES


