Bus Transport Process Network Synthesis

Albert Nagy[1], Zsolt Ercsey[2], József Tick[3], Zoltán Kovács[4]

[1] Applied Informatics, Óbuda University, Bécsi út 96/b, 1034 Budapest, Hungary, e-mail: albert.nagy@me.com

[2] Department of System and Software Technology, Faculty of Engineering and Information Technology, University of Pécs, Boszorkány u. 2, 7624 Pécs, Hungary, e-mail: ercsey@mik.pte.hu

[3] Department of Applied Informatics, John von Neumann Faculty of Informatics, Óbuda University, Bécsi út 96/b, 1034 Budapest, Hungary, e-mail: tick@uni-obuda.hu

[4] Optin Ltd. Oroszlán u. 4. 6720 Szeged, Hungary, e-mail: zoltan.kovacs@optin.hu

Abstract: The current paper is about bus transport process network synthesis. Unlike previously discussed urban traffic modelling and solution methods, here, it is presented as a novel application of the p-graph methodology, while exploiting the peculiarities of the problem. The focus is on the synthesis step, where the set of potentially feasible solutions is determined, in other words, the maximal bus transport process structure is generated. The classical process network instances together with their properties are adapted to this new application field, i.e. to meet the special requirements of the bus transport. First, the meaning of the material type nodes and the operating unit type nodes are described in details. A new axiom is given to complete the set of p-graph's axioms. In addition, the utilization of the conventional maximal structure and solution structure generation algorithms, they are extended to gain advantage of the new axiom and to generate the potential solution structure in a more effective manner. Based on the solution structures a mathematical programming model is generated containing the constraints and the objective function of the bus transport problem. Thus, the generation of the bus launching list is prepared. The solution method presented for bus transport problems meets the high level expectations of decision-makers, i.e. the resulting system is complete, flexible and robust.

Keywords: bus transport; mathematical programming model; process network; p-graph; synthesis

1 Introduction

The importance of public transportation is indisputable. It contributes to the flourishing of cities, as well as, to a higher quality everyday life for citizens. A well-organized urban public transport system may offer more opportunities, including faster access to jobs, easier approach to recreational activities etc. Even though public transport now goes beyond buses, metros, boats, trams and trains, these traditional modes serve as a backbone in cities. This vitally important sector is constantly evolving as the hosting cities evolve, the number of passengers increases, and technology develops. Deployment of general digitalization, as well as, problem specific software solutions has become a norm. The arising novel information system development technologies, adapted sensor networks as well as info communication systems and devices renders it possible to implement more complex systems, like urban bus transport networks, as part of smart cities projects. Obviously, each bus transport system has its own peculiarities.

As companies started to focus on and invest in innovative solutions to optimize their services, striving for operational excellence is a general feature of public transportation companies. However, keeping this in a sustainable manner casts an immense burden on the sector players. Cost effectiveness remains one of the most fundamental goals. The major part of the costs of a public transport company arise from their vehicles, including the vehicle fleet, fuel, maintenance costs and drivers' salaries, in other words, a significant fraction of the costs corresponds to the operation of the vehicles, therefore, finding the optimal scheduling of the vehicles is a critical task to be performed. Nevertheless, this optimization problem is an immensely complex issue, including the scheduling of the vehicles and the drivers, while considering all sorts of labor standards etc. These subtasks may already lead to NP hard problems to be solved by themselves. Besides it complexity, it is of importance to note that the surrounding environment can change suddenly, immediately requiring intervention, which may only be realized should an algorithmic solution be available, as background support.

2 Materials and Methods

2.1 Vehicle Scheduling Problems

To overcome the difficulties this discipline has been extensively researched in the past decades, mainly focusing on the mathematical programming models and the corresponding solution of the mathematical models. The single depot vehicle scheduling problem equation was presented in Bodin and Golden [1] and later Bodin [2] presented the problem with minimum two depots, known as a multiple depot vehicle scheduling problem; while Bertossi [3] showed its NP-hardness.

There are applications available for the general cases [4], nevertheless, in practice there are always additional constraints that also have to be taken into consideration, which always bring the solution process to a more cumbersome situation. For example, Kliewer [5] considered the multiple depot vehicle scheduling problem, and assigned buses to cover a given set of timetabled trips with some practical requirements related to vehicle types, depot capacities etc. Later Li and Head [6] considered the bus-scheduling problem and evaluates new types of buses that use alternative energy sources to reduce emissions, including toxic air pollutants and carbon dioxide. Steinzen [7] discussed timetables in ex-urban bus traffic that consist of many trips serviced every day together with some exceptions that do not repeat daily. Dias [8] gave a genetic algorithm to the bus driver scheduling problem, allowing the simultaneous consideration of several complex criteria. Desaulniers and Hickman [9] reviewed state-of-the-art operations research models and approaches for solving public transit problems from the network design point of view. There are several other applicable models and solution methods available, for a detailed overview see [10].

2.2 P-Graph Methodology

The original idea of the P-graph methodology was published by Friedler [11]. This rigorously developed mathematical programming approach was first used to describe chemical engineering problems. The synthesis problem is defined by the (P, R, O) triplet, where *P* represents the set of products, *R* represents the set of raw materials and *O* represents the set of operating units of the problem. First, a directed bipartite graph is used to represent the process system, i.e. operating units serve as one type of the vertices of the graph, while the connecting materials, both raw materials, products and intermediate materials serve as the other type of vertices of the graph. Should a material be consumed or produced by an operating unit, there is a directed connection between the material and the operating unit, i.e. an arc of the graph. Each subgraph of the bipartite directed graph constructed before is a feasible process structure, p-graph in short, if and only if it satisfies the following axioms, representing necessary and sufficient combinatorial properties. Please note that the largest p-graph is called to be the maximal structure, while all other p-graphs are called to be solution structures. The axioms are as follows: (S1) Every final product is represented in the subgraph. (S2) A material type node in the subgraph has no input if and only if it represents a raw material. (S3) Every operating unit type node in the subgraph represents an operating unit defined in the synthesis problem. (S4) Every operating unit type node has at least one path leading to a product in the subgraph. (S5) If a material type node belongs to the subgraph it has to be an input to or an output from at least one operating unit type node in the subgraph. One of the main advantage of this methodology is that the maximal structure, as well as, all feasible solution structures can be algorithmically generated, i.e. MSG generates the maximal structure and SSG generates the solution structures, see [12]. The recursive algorithm exploits the advantages of a decision mapping, i.e. for a given material it is decided which operating units take part in the production of this given material and which operating units are excluded. Throughout the decision process, consistency has to be guaranteed, specifically, should an operating unit be excluded from the solution based on the decision at a given material, this operating unit cannot be chosen to be included in the solution later on based on the decision at another material. Moreover, should an operating unit be part of the solution structure, all of its input and output materials have to be included in the structure. With these algorithms all structures can be generated for the defined synthesis problem, that can be potentially optimal according to the objective.

The p-graph methodology is now used in various fields. For example, separation network synthesis problems with multiple feed streams and sharp separators were considered by Kovacs [13] and [14]. Bertok [15] investigated optimal design of supply chains. Barany [16] solved vehicle assignment problems to minimize cost and environmental impact of transportation. Garcia-Ojeda [17] solved the routing and scheduling of evacuees, facing a life-threatening situation. Lam [18] presented an extended implementation of the p-graph for an open structure biomass network synthesis. Tick [19], and [20] investigated workflow problems which were extended in the direction of business process modelling, called to be the robust Process Network Synthesis (PNS) problem by Almási [21]. An organization-based multi-agent system is modelled according to the framework of Organization Model for Adaptive Complex Systems and this design model is transformed into a process-network model by Garcia-Ojeda [22]. Heckl [23] considered multi-period operations, while Atkins [24] used the p-graph methodology to investigate economically feasibility of utilizing various wood processing residues in bio refineries. Vincze [25] transformed CPM problems to p-graphs to handle alternatives within one step. Ercsey [26] solved a clothing manufacturer's problem with p-graphs. Benjamin [27] proposed a methodology for criticality analysis of component units or plants in an integrated bioenergy system to increase the robustness against disruptions. Aviso [28] considered multi-period optimization of sustainable energy system to contribute reduction in greenhouse gas emissions.

2.3 Problem Definition

The organization and control of urban transport that system planners and transit operators have to cope with is immensely complex. Obviously, each country and city has its own peculiarities, and the resulting protocols may vary widely. The present paper considers bus transport problems from a novel point of view, namely the solution is based on the p-graph methodology, considering the below problem definition.

Timetables of buses highly depend on the specialties of the considered city, namely geographic locations, roadmaps, workplace distributions, residential densities of various districts, performance and disposition of other types of transport systems, passenger habits, as well as, social considerations have to also be taken into account. In this particular case, let us consider the timetable of buses as preliminary given, and the goal is its realization. Obviously, this primary goal has to be completed with optimality in a sense of economy, pollution, sustainability, rules and regulations etc.

Let us consider the situation, where the departure station and the terminal station are given together with the driving time of the bus within a turn. Other type of times should also be given beside the driving time of the turn, for example, time for technical tasks, changeover time, etc. Obviously, these types of times are important from the operation point of view, therefore exact values have to be precisely known, however, these are constant values, therefore their separate management is less important from the synthesis point of view, as it is discussed in details later.

Should the timetable be given, various parts of the day can be clearly identified when the launching density of the buses is the same, these will be considered as periods of the problem. Moreover, based on the timetable, the number of turns to be performed can also be calculated. Obviously, the size of the bus fleet, i.e. the number of buses is also known and it is assumed that there is a driver available for each bus. Moreover, it is assumed that changeover may occur once a day, in other words, the given driver may optionally leave the bus and hand it over to another driver. Labor standards and rules should also be considered. One of the simplest rules from the standards point of view is the minimal and maximal limit for the daily working hours. Rest period should be considered once a day in a way that the continuous driving time must not exceed a given number of working hours. Rest and changeover may only take place at the departure station. Time constants for the technical tasks are also assumed to be given, i.e. exact times for the stance, when the bus leaves the garage, entering the garage, changeover and discharge are given.

In summary:

- Periods, $P_1, P_2, \ldots, P_s$
- Number of turns to be performed during the periods, $r_1, r_2, \ldots, r_s$
- Launching density within a period, in minutes, $t_1, t_2, \ldots, t_s$
- Driving time within a route, in minutes, T
- Number of buses, B
- Number of drivers, D
- Time for the rest period, in minutes, RT
- Time for changeover, in minutes, HT
- Time for discharge, in minutes, LT
- Time for stance, in minutes, ST
- Time for entering the garage, in minutes, GT

- Minimum working hours, in minutes, NWH
- Maximum working hours, in minutes, XWH
- Minimum working hours to rest, in minutes, RWH

Beside the mandatory goal, namely, that the timetable should be fulfilled, some other type of goals may also be solved, for example, the mandatory goal should be solved with a minimal number of buses or a minimal number of drivers etc. This extension will be explained in detail in the the mathematical model of this paper.

3 Results

3.1 Solution Framework

The bus transport optimization and resource allocation problem specified in the previous section is proposed to be solved by an algorithmic method with three main phases as follows. Details of the phases are given in the following chapters of the paper with the focus on the synthesis part.

First phase: the structural model of the problem is generated. here a p-graph, namely the maximal structure of the problem is generated and the parameters of the arcs and nodes are set.

Second phase: the feasible solution structures are generated. here exploiting the peculiarities of the current problem, ssg algorithm is modified to enumerate all potentially feasible solution structures. for each solution structure, the corresponding mathematical programming model is generated, which contains linear constraints and it is mixed integer linear programming (milp) in case the considered cost function is linear and nonlinear programming (nlp) in case the considered cost function is nonlinear.

Third phase: the launching table of the buses including a scheduling is generated for the original problem. based on the solutions of the mathematical programming models generated in the second phase a scheduling algorithm generates the launching table of the buses.

3.2 Bus Transport Process Network, Concepts and Definitions

The p-graph framework aims to solve optimization problems from the synthesis point of view, originating in the chemical engineering industry. The terminology used related to the nodes and arcs also reflects these origins, i.e. the nodes of the bipartite, directed p-graphs are called to be of material types and of operating units; moreover, their operating rules also correspond to this interpretation. As graphs can be understood better and faster than mathematical equation, complex problems can be surveyed more easily when graphs illustrating them. In addition to the easier understanding, graphs serve as essential background for the algorithmic generation of mathematical models. In this paper, the p-graph methodology is applied to model, manage and optimize urban bus transport with respect to specific circumstances. Here, material type nodes are interpreted as some given states, or occurrence of some events (the bus is in the station), or a resource (time, driver, bus), or a physical object (garage). The operating unit type nodes are interpreted as some given activities (stance) or some transition between two states; moreover, this type of node may identify an activity which triggers an event or on the other hand, prevents another event from occurring. The operating unit type nodes for the current interpretation are as follows see Figure 1, represented horizontal bar.

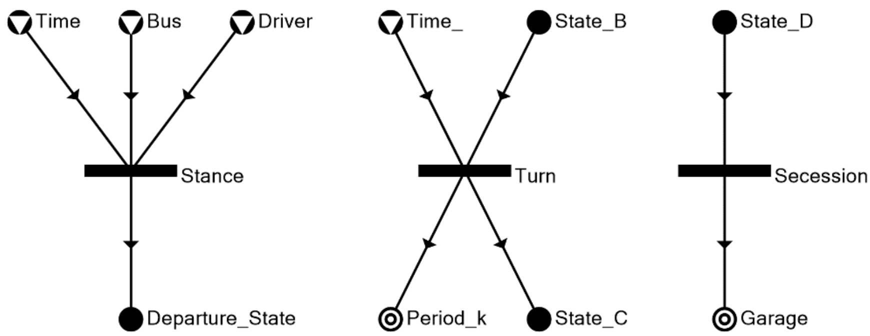

Figure 1

Operating units Stance, Turn and Secession of bus transport process networks

Stance: This operating unit type node represent the activity when a driver takes the bus from the garage to the departure station in *ST* minutes, i.e. from the driver's working hours point of view *ST* minutes is spent, i.e. the very event occurs that the z^{th} bus is driven by the k^{th} driver and they both get to the departure station, Departure_State event occurred.

Turn: This operating unit type node represents that a given bus and the corresponding driver left the departure station, went all the way to the terminal station and then returned from there back to the departure station, while obviously routed all intermediate stations in both directions. In other words, the state of the system has changed, i.e. the system transited from *State_B* to *State_C*, while *T* minutes passed from the available working hours of the driver; moreover, a turn is also performed in the k^{th} period. The period is again a material type node, moreover it is a product, since the timetable is prescribed by the problem definition, and as a result it is calculated how many turns have to be performed within a period.

Secession: Entering the garage, i.e. opposite to stance. This operating unit type node represents bus and the corresponding driver that returns to the garage, the

Garage event occurs and the overall process ends. Garage event is a material type node, moreover it is a product, since it serves as the desired aim to be achieved by the buses. Obviously, this activity may only happen should the given bus together with its driver(s) performed its daily tasks as well as the corresponding labor standards including the rest criteria were also satisfied.

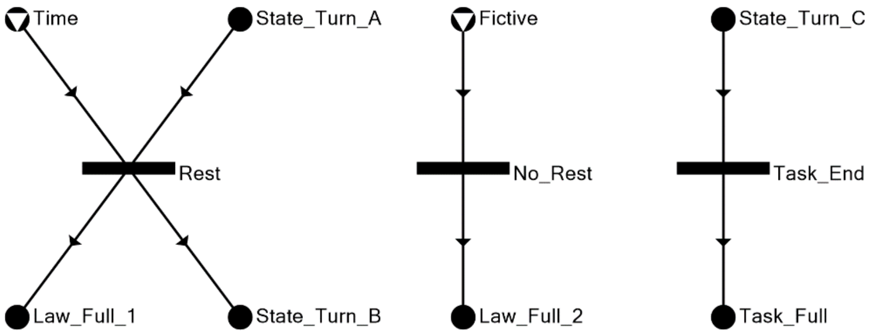

Figure 2
Operating units Rest, No_Rest and Task_End of bus transport process networks

Rest: This operating unit type node represents that a bus and the corresponding driver are at the departure station and the driver rests: after a performed number of turns (*State_Turn_A*) the driver pulls over and rests. This happens according to the labor standards and time regulations. The driver has to work minimum time to receive the right for rest, i.e. *RWH* has to be satisfied. The resting time, *RT* minutes, counts into the overall working hours of the driver. Should a driver rest, the corresponding labor standard is satisfied *Law_Full_1*. When the rest is over, the driver returns to the departure station (*State_Turn_B* occurs) and performs further turns. The operating unit Rest in the current interpretation is depicted in Figure 2, represented by a horizontal bar.

No_Rest: This operating unit type node represents the situation when the length of the working hours does not reach the limit, *RWH*, when rest time should be given to the driver. In other words, the given bus and the corresponding driver will return to the garage and finish work without a rest. Obviously, labor standards are met in this particular case also *Law_Full_2*. To meet the requirements of the axiom system, a raw material, Fictive, must be added to the p-graph. The operating unit No_Rest in the current interpretation is depicted in Figure 2, represented by a horizontal bar.

Task_End: This operating unit type node represents that a given bus and the corresponding driver are at the departure station and prepares to drive back to the garage to discharge and finish work. The operating unit Task_End in the current interpretation is depicted in Figure 2, represented by a horizontal bar.

Changeover: This operating unit type node represents that a given bus and the corresponding driver are at the departure station and the driver finishes work and hands the bus over to a new driver, who starts his work as if this was a stance within the garage. Please note that each driver has his/her own time frame therefore separate input material *Time* is part of the graph. The operating unit Changeover in the current interpretation is depicted in Figure 3, represented by a horizontal bar.

The material type nodes, namely raw materials for the current interpretation are as follows.

Time: This material type node represents the available working hours, the assigned parameter is the length of the working hours in minutes. Its minimal value is *NWH*, while its maximal value is *XWH*.

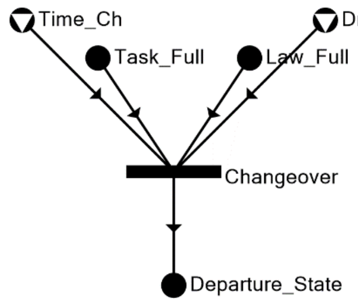

Figure 3
Operating unit Changeover of bus transport process networks

Bus: This material type node represents the available bus fleet, the assigned parameter is the number of buses, i.e. B.

Driver: This material type node represents the available drivers, the assigned parameter is the number of staff, i.e. D.

The material type nodes, namely products for the current interpretation are as follows.

Garage: This material type node represents the garages. This garage node has to be represented in the p-graph as many times as many garages are within the network of the buses. It is a product, since the process ends here and obviously, all buses have to return to one of the garages.

Period: This material type node represents the various periods identified within the timetable, each period is a product node. The assigned parameter is the number of turns to be performed within the period. Obviously, this has to be achieved.

Further nodes of the material type, which are neither raw materials, nor products for the current interpretation are as follows.

Departure_state: This material type node represents the state when a given bus and the corresponding driver are at the departure station and ready to perform turns.

State_Turn: This material type node represents the state when a given bus and the corresponding driver during the working hours are at the departure station and ready to i) perform further turns, ii) finish work, iii) go to rest or iv) perform a changeover.

Task_Full: This material type node represents the state when a given driver fulfilled his duties received for the day. After this state the driver either drives the bus to the garage for secession, or hands over the bus to another driver, for changeover. Please note that each driver corresponds to one and only one *Task_Full* node.

Law_Full: This material type node represents the labor standards corresponding to the working hours. Please note that each driver corresponds to one and only one *Law_Full* node.

Now following the p-graph methodology, the maximal structure is to be generated and then all different and feasible solution structures are to be derived. Essentially, during the generation of the p-graphs a relaxation of the original problem is performed, since only the main conditions are considered. For example, it is considered whether the rest is given or not, i.e. satisfying the labor standards, but the details of the rest are not yet considered, i.e. its exact rest duration, starting conditions etc. are not yet verified. These additional conditions will be handled in the second phase. Therefore, it may happen that a subgraph will be accepted as a feasible solution, but further on, for example in the second phase when further constraints are also added to the mathematical programming model, it turns out that this p-graph is infeasible from the original problem point of view. Conversely, it has to be clear that in this phase no structures are excluded which may result the correct solution of the original problem.

The result of the first phase is the generated maximal structure for the bus transport process network problem. The maximal structure of the following bus transport process network synthesis problems are given: a one period and one bus without changeover problem in Figure 4, a one period and one bus with changeover problem in Figure 5, and a two period and one bus without changeover problem in Figure 6.

When the aforementioned maximal structure is available for the problem, all feasible solution structures can be generated in the second phase. An early process, i.e. the *SSG* algorithm has been developed by Friedler [11] to generate these structures. To exploit all peculiarities of the current problem, it is advisable to extend the axiom system of the general p-graphs, therefore axiom (S6) is constructed and added to the set of axioms:

(S6) The IN degree, as well as the OUT degree is only one for all material type nodes, representing neither a raw material, nor a product.

Based on the set of axioms extended with axiom (S6) algorithm *SSG* is modified to exploit this special feature. A feasible solution of one bus and one period problem generated by the algorithm is given in Figure 7.

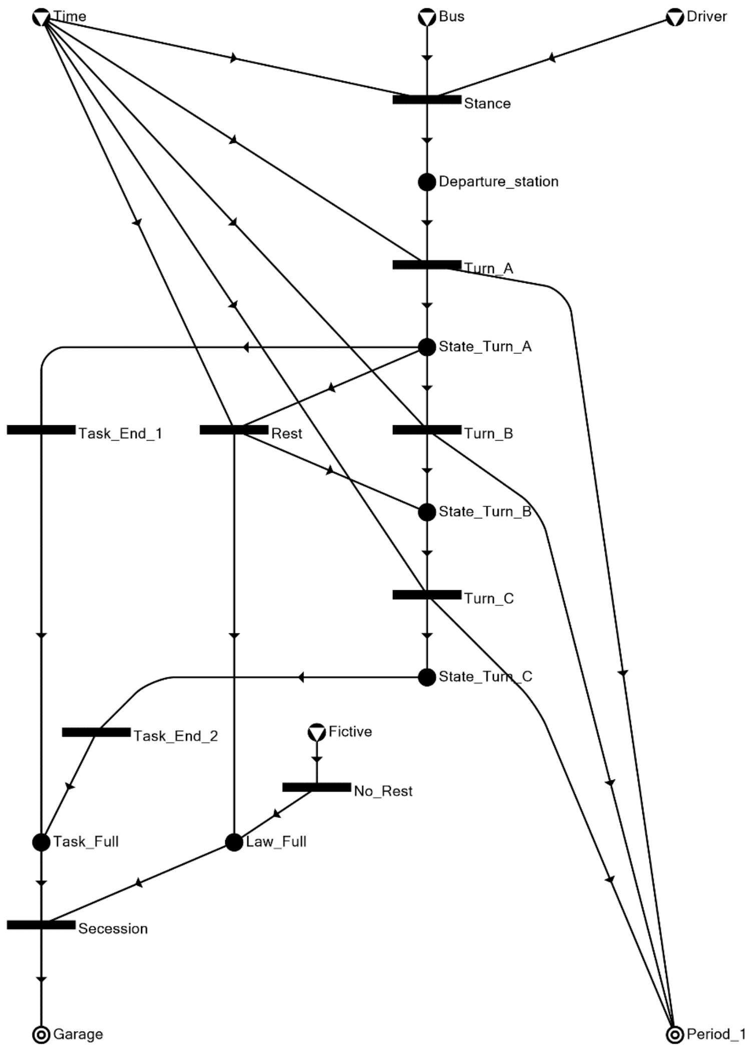

Figure 4

Bus transport process network synthesis

Maximal structure for a one period and one bus without changeover problem

The axiom reflects the following consideration. In reality, i.e. in a feasible solution structure any state should have only one predecessor, and similarly any state should have only one successor. See for example *State_Turn_*A in Figure 4, where the driver has various options: continue the turns: *Turn_B*, go to rest: Rest, or finish work: *Task_1_End*, but from all these options, obviously only Rest will be realized, see Figure 7.

3.3 Mathematical Programming Model

In the previous chapters it is illustrated, how the potentially feasible solution structures of a bus transport process network synthesis problem are developed. As mentioned before, these solution structures serve as relaxed solutions of the original problem, since some specific prescriptions are not yet considered.

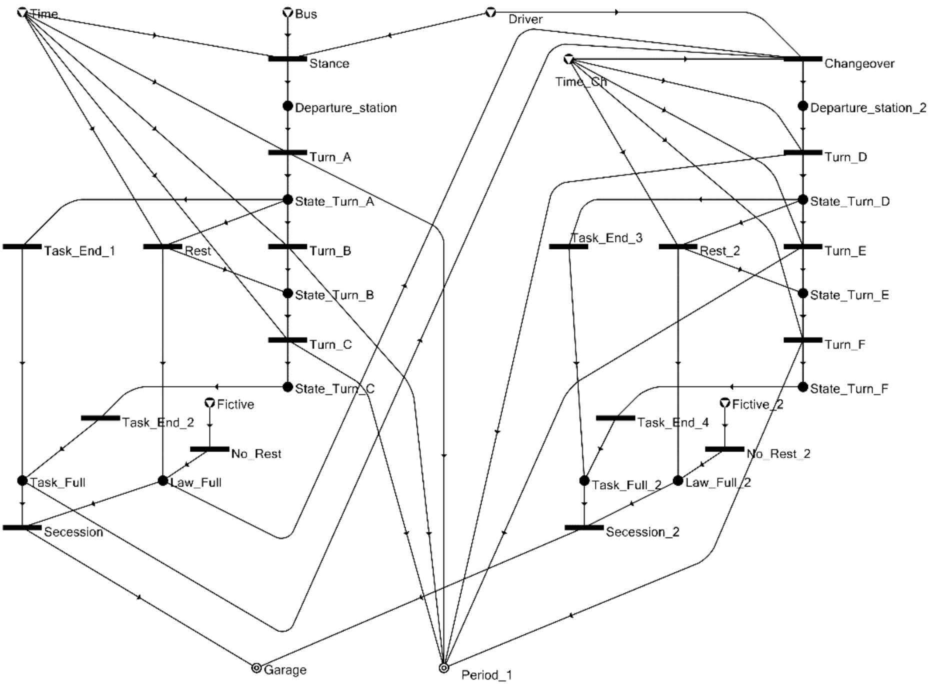

Figure 5

Bus transport process network synthesis.

Maximal structure for a one period and one bus with changeover problem

For example, the length of the rest period, when the rest may be taken and when it should be taken, the lengths of the working hours etc. are not considered with exact values when these solution structures are generated. The corresponding constraints will be satisfied through the MILP or NLP mathematical programming models assigned to the solution structures. Obviously, at this point, it may also turn out that there is no solution, i.e. the given solution structure together with the assigned mathematical programming model has to be discarded, and the next solution structure has to be considered.

Let *G* bipartite, directed graph be a solution structure generated by the modified SSG algorithm consider for example the network in Figure 7. Let an index set be generated, i.e. a non-negative integer is assigned to each node of the maximal structure.

One type of the constraints corresponds to maximal number of turns performed by every driver in every period at the nodes type *Turn_A*, *Turn_B*, *Turn_C*. The x_{ijA}, x_{ijB}, x_{ijC} non negative integer variables are assigned to these nodes, where *ijA* index corresponds to the variable representing the i^{th} driver, in the j^{th} period's *Turn_A*.

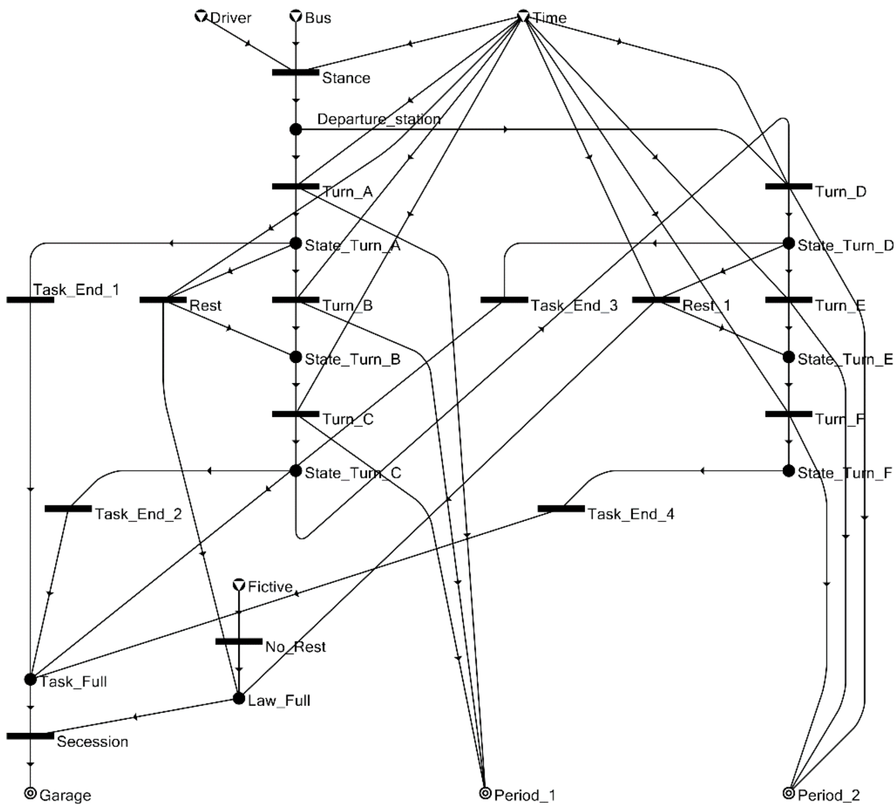

Figure 6
Bus transport process network synthesis
Maximal structure for a two period and one bus without changeover problem

Should the I^{th} driver rest in the j^{th} period, then p_{ij} binary variable is one, otherwise it is zero, which can be seen from the solution structure.

$$p_{ij} \cdot RT + T \cdot \left(x_{ijA} + x_{ijB} + x_{ijC}\right) \leq (r_j - 1) \cdot t_j + T \tag{1}$$

equation (1) describes the obvious expectation that the length of the period cannot be shorter than the number of turns performed by the driver multiplied by the driving time of the bus within a turn, plus the time for the potential rest period. This inequality has to be given for each *ij* pairs which are included in the solution structure.

The minimal and maximal length of the working hours are also specified by the original problem. Should the solution structure under consideration contain no changeover, then the corresponding inequalities are specified in the following way, see equation (2) and (3). These inequalities have to be given for each driver without changeover.

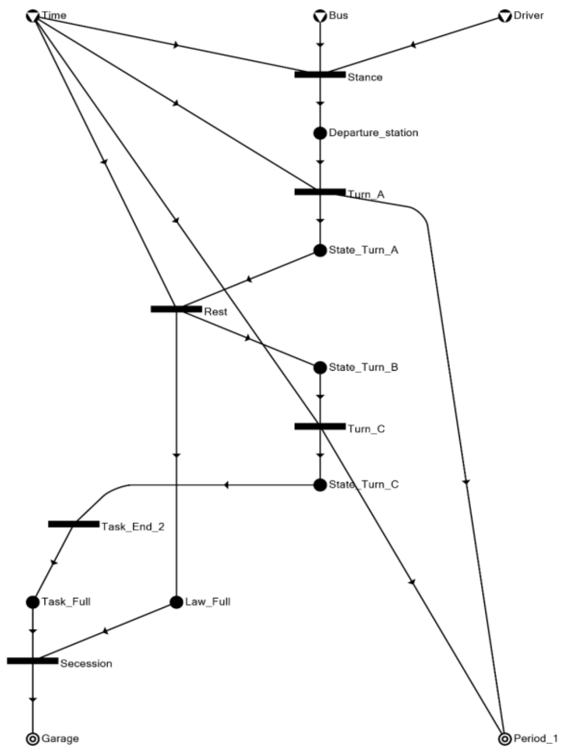

Figure 7
Bus transport process network.
A feasible solution of one bus and one period problem

$$ST + GT + RT \cdot \sum_{j=1}^{s} p_{ij} + T \cdot \sum_{j=1}^{s} (x_{ijA} + x_{ijB} + x_{ijC}) \leq XWH \tag{2}$$

$$ST + GT + RT \cdot \sum_{j=1}^{s} p_{ij} + T \cdot \sum_{j=1}^{s} (x_{ijA} + x_{ijB} + x_{ijC}) \geq NWH \tag{3}$$

Should the solution structure contain a changeover, and it is considered exactly at the i th driver in the k th period after the state *State_Turn_B*, then the following inequalities have to be given for the driver who started from the garage, equation (4) and (5). Obviously, these inequalities have to be given for each driver where changeover takes place.

$$ST + RT \cdot \sum_{j=1}^{k-1} p_{ij} + T \cdot \sum_{j=1}^{k-1} (x_{ijA} + x_{ijB} + x_{ijC}) + x_{ijA} + x_{ijB} \leq XWH \tag{4}$$

$$ST + RT \cdot \sum_{j=1}^{k-1} p_{ij} + T \cdot \sum_{j=1}^{k-1} (x_{ijA} + x_{ijB} + x_{ijC}) + x_{ijA} + x_{ijB} \geq NWH \tag{5}$$

On the other hand, the other part of the changeover situation, i.e. the case of the new driver, has the following inequalities, equation (6) and (7).

$$HT + RT \cdot \sum_{j=k+1}^{s} p_{ij} + x_{ikC} + T \cdot \sum_{j=k+1}^{s} (x_{ijA} + x_{ijB} + x_{ijC}) \leq XWH \tag{6}$$

$$HT + RT \cdot \sum_{j=1}^{k-1} p_{ij} + x_{ikC} + T \cdot \sum_{j=1}^{s} (x_{ijA} + x_{ijB} + x_{ijC}) \geq NWH \tag{7}$$

With the help of the above equation minimal and maximal length of the driving times for both drivers are specified. This inequality has to be given for each driver.

Let the non-negative, real variable w_{ij} for every driver and every period be given to illustrate the waiting time of the given driver in the given period, see equation (8).

$$w_{ij} = (r_j - 1) \cdot t_j - (x_{ijA} + x_{ijB} + x_{ijC}) \cdot T - p_{ij} \cdot RT \tag{8}$$

Now, the cost function of the mathematical programming model can be given, which in this particular case corresponds to the effectiveness of the human workforce, i.e. minimizing the overall waiting time of the drivers, see equation (9).

$$\sum_{i} \sum_{j} w_{ij} \quad \rightarrow \quad min \tag{9}$$

Obviously, the above given cost function may be extended or changed according to the desired goal. In some cases, the target to be optimized can even be specified directly from the solution structure; for example, in cases where the number of buses or the number of drivers are to be minimized. If so, the inequality system is to be solved in order to guarantee whether the solution is feasible or not, moreover, to determine the values of all the variables.

4 Discussion

As a result of the previously described method, namely when a solution of the assigned mathematical programming model is generated, the only remaining step is the generation of the actual bus launching list, i.e. creation of the bus schedule. As an advantage of the structural representation, this is a tightly constrained problem. It is tightly constrained, in the sense that each period is to be sharply separated, the turns are given for the buses separately, and the corresponding times are given separately.

Conclusion

Determination of the optimal solution of a complex system is a cumbersome task. In general, methods focus on the automatic generation of the mathematical programming model, with an immense number of variables. The classical process network synthesis originating in the chemical industry proved that should the focus be on the synthesis step, i.e. should the first step of the solution method be the exact generation of the potentially feasible process structures, then the resultant mathematical programming model together with its solution can be generated with relative ease. In this paper the urban bus transport process network synthesis problem is presented as a novel application of the classical p-graph methodology. First, the urban public transport is briefly reviewed, then the vehicle scheduling problems and the methodology are discussed. The model is focusing on the synthesis step of the problem solution. Both the concept, notations, definitions, axioms and algorithms are applied to exploit the peculiarities of the problem, i.e. the bus transport process network is defined. The meaning of the material type nodes and the operating unit type nodes are described in details. A new axiom is given to complete the set of classical p-graph’s axioms. The mathematical programming model of the bus transport process network is detailed resulting in the schedules. This mathematical programming model can algorithmically be produced from the generated solution structures. The presented solution method prepares the generation of the bus launching list, i.e. the solution of the bus scheduling problem can be achieved with relative ease due to the resultant tight conditions. This novel approach can be used to solve real-world problems with the existing constraints. Research on advanced process network synthesis for transport management systems proves to be a relevant area of further examinations. The expected results hold the promise to improve the operational planning activities of public transport systems.

References

[1] Bodin, Lawerence and Golden, Bruce. Classification in vehicle routing and scheduling. Networks, 11(1) pp. 97-108, 1981

[2] Bodin, Lawerence, Golden, Bruce, Assad, Arjang, and Ball, Mark. Routing and scheduling of vehicles and crews: The state of the art. Computers and Operations Research, 10(1) pp. 63-212, 1983

[3] Bertossi, Alan A, Carraresi, Paolo, and Gallo, Giorgio. On some matching problems arising in vehicle scheduling models. Networks, 17(1) pp. 271-281, 1987

[4] Meilton, M.: Selecting and implementing a computer aided scheduling system for a large bus company, Algorithms: Combinatorial Analysis. In Computer-Aided Scheduling of Public Transport, (eds. S. Voss and J. R. Daduna), 203-214, Springer-Verlag, Berlin, 2001

[5] Kliewer N., T. Mellouli, and L. Suhl. A time-space network based exact optimization model for multi-depot bus scheduling, European Journal of Operational Research, 175, pp. 1616-1627, 2006

[6] Li, J-Q and K. L. Head: Sustainability provisions in the bus-scheduling problem, Transportation Research, Part D, 49, 50-60, 2009

[7] Steinzen I., L. Suhl, N. Kliewer: Branching Strategies to Improve Regularity of Crew Schedules in Ex-Urban Public Transit. OR Spectrum (2009) 31: 727. DOI 10.1007/s00291-008-0136-5

[8] Dias, T. G., JP de Sousa, and JF Cunha. Genetic algorithms for the bus driver scheduling problem: a case study. Journal of the Operational Research Society, 53 (3), pp. 324-335, 2002

[9] Desaulniers, G. and M. D. Hickman: Public Transit, In Handbook in OR & MS, (eds. C. Barnhart and G. Laporte), Vol. 14, Chapter 2, Elsevier B.V., 2007

[10] Bunte, Stefan and Kliewer, Natalia. An overview on vehicle scheduling models. Journal of Public Transport, 1(4) pp. 299-317, 2009

[11] Friedler, F., K. Tarjan, Y. W. Huang, and L. T. Fan. Combinatorial Algorithms for Process Synthesis, Computers Chem. Engng, (1992b) 16, S313-320

[12] Friedler, F., K. Tarjan, Y. W. Huang, and L. T. Fan. Graph-Theoretic Approach to Process Synthesis: Axioms and Theorems, Chem. Engng Sci., (1992a) 47, 1973-1988

[13] Kovacs Z, Z Ercsey, F Friedler and L. T. Fan, Exact super-structure for the synthesis of separation-networks with multiple feed-streams and sharp separators; (1999) Computers and Chemical Engineering 23:(Supplement 1) pp. S1007-S1010

[14] Kovacs Z, Z. Ercsey, F. Friedler, L. T. Fan. Separation-network synthesis: global optimum through rigorous super-structure. (2000) Computers and Chemical Engineering 24:(8) pp. 1881-1900

[15] Bertok, B., R. Adonyi, F. Friedler. Optimal Design of Supply Chains by the P-Graph Framework. Computer-Aided Chemical Engineering, 25: pp. 1445-1450 (2009)

[16] Barany, M., B. Bertok • Z. Kovacs, F. Friedler • L. T. Fan. Solving vehicle assignment problems by process-network synthesis to minimize cost and environmental impact of transportation. Clean Techn Environ Policy. (2011) DOI 10.1007/s10098-011-0348-2

[17] Garcia-Ojeda, JC., B. Bertok, F. Friedler, Planning evacuation routes with the P-graph framework, Chemical Engineering Transactions, (2012) 29, 1531-1536

[18] Lam H. L., J. J. Klemeš, P. S. Varbanov, Z. Kravanja. P-graph synthesis of open-structure biomass networks. Ind. Eng. Chem. Res. (2012) 52, 172-180

[19] Tick, József, Csanád Imreh, Zoltán Kovács, Business Process Modeling and the Robust PNS Problem, Acta Polytechnica, Volume 10, Issue Number 6, (2013) 193-204. DOI: 10.12700/APH.10.06.2013.6.11

[20] Tick, J, P-Graph-based Workflow Modelling, Acta Polytechnica Hungarica, 4:1 pp. 75-88., 14 p. (2007)

[21] Almási, D; Cs, Imreh; T, Kovács; Tick, J, Heuristic Algorithms for the Robust PNS Problem, Acta Polytechnica Hungarica, 11: 4 pp. 169-181, 13 p. (2014)

[22] Garcia-Ojeda, JC; B. Bertok, F. Friedler; A. Argoti, L. T. Fan. A preliminary study of the application of the P- graph methodology for organization-based multiagent system designs: Assessment (2015) Acta polytechnica hungarica 12:2 pp. 103-122

[23] Heckl, I, L. Halász, A. Szlama, H. Cabezas, F. Friedler. Process synthesis involving multi-period operations by the P-graph framework. (2015) Computers and Chemical Engineering 83 pp. DOI 157-164. 10.1016/j.compchemeng.2015.04.037

[24] Atkins, M. J., Walmsley, T. G., Ong, B. H. Y., Walmsley, M. R. W., & Neale, J. R. (2016) Application of P-graph techniques for efficient use of wood processing residues in bio refineries. Chemical Engineering Transactions, 52, 499-504, https://doi.org/10.3303/CET1652084

[25] Vincze, N., Z. Ercsey, T. Kovács, J. Tick, Z. Kovács. Process network solution of extended CPM problems with alternatives, Acta Polytechnica 13:3 pp. 101-117 (2016)

[26] Ercsey, Z. Process network solution of a clothing manufacturer’s problem. Pollack Periodica 12:(1) pp. 59-67 (2017) DOI 10.1556/606.2017.12.1.5

[27] Benjamin, M. F. D., C.D. Cayamanda, R. R. Tan, L. F. Razon. P-graph approach to criticality analysis in integrated bioenergy systems. Clean Techn Environ Policy (2017) 19: 1841-1854. https://doi.org/10.1007/s10098-017-1368-3

[28] Aviso, K. B., J-Y. Lee, J. C. Dulatre, V. R. Madria, J. Okusa, R. R. Tan. A P-graph model for multi-period optimization of sustainable energy systems. (2017) Journal of Cleaner Production. Volume 161, 10 September 2017, pp. 1338-1351, DOI 10.1016/j.jclepro.2017.06.044