Mechatronics and Intelligent Transportation Systems
https://www.acadlore.com/journals/MITS

Automated Development of Railway Signalling Control Tables: A Case Study from Serbia

Ivan Ristic*

Signalling Solutions, 11030 Belgrade, Serbia

* Correspondence: Ivan Ristic (ivan.ristic@signalling-solutions.com)

Received: 10-24-2023 **Revised:** 11-26-2023 **Accepted:** 12-01-2023

Citation: I. Ristic, "Automated development of railway signalling control tables: A case study from Serbia," *Mechatron. Intell Transp. Syst.*, vol. 2, no. 4, pp. 227–235, 2023. https://doi.org/10.56578/mits020404.

Abstract: The automation of railway signalling control table preparation, a task historically marked by labor-intensity and susceptibility to error, is critically examined in this study. Traditional manual methods of generating these tables not only demand extensive effort but also bear the risk of errors, potentially leading to severe consequences in subsequent project phases if overlooked. This research, therefore, underscores the imperative for automation in this domain. An extensive review of existing methodologies in the field forms the foundation of this investigation, culminating in the enhancement of a select approach with advanced automation capabilities. The outcome is a standardized procedure, adaptable with minimal modifications to the unique national signalling norms of various countries. This procedure promises to streamline project execution in railway signalling, reducing both time and error margins. Such a standardized, automated approach is particularly pertinent to the Republic of Serbia, where this study is situated, but its implications extend globally. Key technologies employed include AutoCAD and Mathematica, which facilitate the requirements-driven automation process. This research not only contributes to the academic discourse on railway signalling automation but also offers a practical blueprint for its implementation across diverse national contexts.

Keywords: Railway signalling; Control tables automation; Manual error reduction; Standardized procedures; Cross-national adaptability

1 Introduction

Railway signalling serves as a vital element in the safe and efficient operation of train networks. The development of signalling control (interlocking) tables is a crucial but often intricate process. These tables define the signalling logic necessary for managing signals, switches, and track section vacancies, integrating complex considerations of train positions, speeds, and potential route conflicts. For smaller stations, manual generation of these control tables is feasible. However, as the complexity of the station's topology increases, the task becomes exponentially more challenging and prone to errors.

Key challenges in control table preparation include:

-In stations with complex layouts, multiple routing variants between two signals may exist. This necessitates a thorough evaluation of potential routes, including scenarios where the station operator manually selects specific paths. It is not uncommon for initially designed control tables to be suboptimal from the end-users' perspective, necessitating late-stage modifications that lead to project delays.

-Ambiguities in interpreting certain signalling rules, such as those governing route protection (flank protection, dark signals, etc.), can lead to inaccuracies in the control tables. This further complicates the preparation process and emphasizes the need for clear, unambiguous rule definitions.

The introduction of automated methods for generating control tables in railway signalling systems marks a significant advancement in the field. Previous research has explored various formal tools for this purpose, including EURIS [1], Ladder logic [2], Petri Nets (PN) [3, 4], RailML [5], Controlled Natural Language (CNL) [6], Maple [7], B-method, Gröbner Bases (GB), Abstract State Machines (ASM), and Finite State Machines (FSM). These studies typically required the manual preparation of the railway station's topological layout using specialized input tools before generating the control table. A notable progression in this area is evident in the studies referenced as [8–10]. These papers introduced methods where the topology of a railway station is generated within a dedicated graphical editor. This innovation allows users to define the railway topology simply by drawing it in the user interface, thereby incorporating a degree of automation in the control table generation process.

This paper contributes to this evolving field by presenting a methodology for generating station control tables in the Republic of Serbia using the MATHEMATICA® package directly from the AutoCAD station signalling layout. Traditionally, the AutoCAD layout serves as the initial step in designing a railway signalling system for specific stations. The proposed approach enables designers who primarily use AutoCAD for signalling design to further automate the control table generation process. This eliminates the need for the dedicated graphical tools assumed necessary in the aforementioned studies [8–10], thus streamlining the process and potentially reducing the time and effort required for accurate control table generation.

2 Methodology

The methodology outlined in this paper is predicated on several key assumptions, which are crucial for the effective application of the proposed approach to generate control tables for railway signalling. These assumptions are as follows:

-The station signalling layout containing only lines has been already prepared in the AutoCAD (.dwg format) and then converted to .dxf format;

-The station contains only the main signals (no shunting signals installed);

-There are no double diamond switches in the station area;

-There are no level crossings in the station area.

A general algorithm showing the whole process is elaborated in Figure 1.

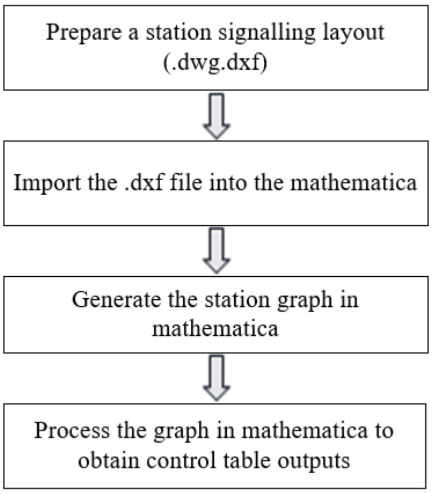

Prepare a station signalling layout (.dwg.dxf)
Import the .dxf file into the mathematica
Generate the station graph in mathematica
Process the graph in mathematica to obtain control table outputs

Figure 1. A general algorithm

The innovative algorithm proposed in this paper harnesses the versatile and powerful capabilities of MATHEMATICA for automating the generation of control tables in railway signalling. This method revolves around the construction and nuanced analysis of a graph model, meticulously drawn from the station's signalling layout within MATHEMATICA. The essence of the approach lies in the identification of key components in the graph model, namely vertices, edges, and edge labels, and effectively mapping these elements to their respective counterparts in the railway signalling system: vertices → sections, edges → tracks, and edge labels → signals in the route.

MATHEMATICA was chosen for its extensive suite of built-in functions adept at creating, manipulating, and visualizing complex graphs, as well as its incorporation of highly efficient algorithms for detailed graph analysis. Its ability to optimize operations such as finding the shortest paths, computing connectivity, or identifying critical graph properties, is invaluable, particularly when dealing with expansive and intricate graph models that mirror complex railway station layouts. The environment of MATHEMATICA ensures not only rapid processing but also the precision of results, which is paramount in the development of accurate control tables that are essential for the safety and efficiency of railway operations.

The AutoCAD signalling layout of a railway station, which is central to the methodology described in this paper, is illustrated in Figure 2.

As depicted in the previous figure, the railway station's signalling system comprises various essential elements, which are systematically catalogued in Table 1.

According to regulation [11], the aspects of the main signals in the railway signalling system are comprehensively detailed in Table 2.

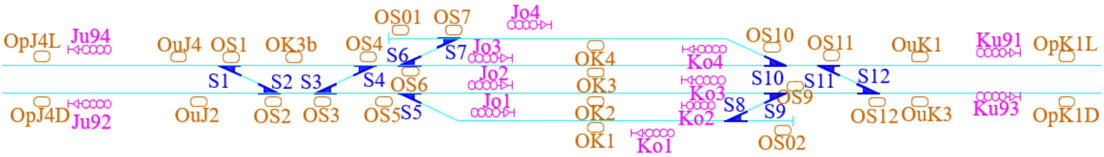


Figure 2. Station signalling layout prepared in the AutoCAD

Table 1. Legend of the essential elements for signalling system

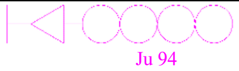
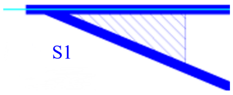
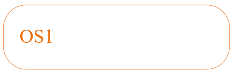
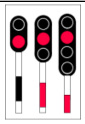
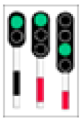
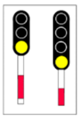
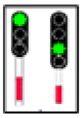
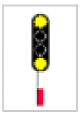
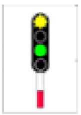
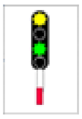
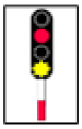
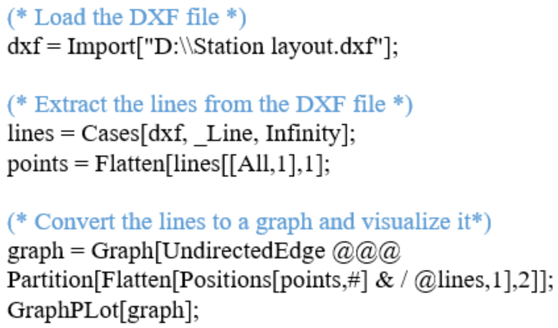
Element	Explanation
	Station main signal (entry or exit)
	Station switch (left or right)
	Station insulated section (for switch or track)
	Track section

Table 2. The aspects of main signals

Aspect No.	Display	Meaning
4		Unconditional stop
5a		Free passage
6		Caution: the next signal shows the stop aspect
7		Free passage, expect speed limit aspect on the next signal
8		Speed limit: expect the stop aspect on the next signal
9		Speed limit: expect free or caution aspect on the next signal
10		Speed limit: expect the speed limit aspect on the next signal
12a		Cautious entry into the station with speed 10km/h

A snippet of code in MATHEMATICA, which converts the .dxf file to the corresponding graph, is shown in Figure 3.



```
(* Load the DXF file *)
dxf = Import["D:\\Station layout.dxf"];

(* Extract the lines from the DXF file *)
lines = Cases[dxf, _Line, Infinity];
points = Flatten[lines[[All,1]],1];

(* Convert the lines to a graph and visualize it*)
graph = Graph[UndirectedEdge @@@
Partition[Flatten[Positions[points,#] & / @lines,1],2]];
GraphPLot[graph];
```

Figure 3. A snippet of the MATHEMATICA code for .dxf to graph conversion

3 Results

3.1 Implementation of the Signalling Rules

The comprehensive implementation of signalling rules, particularly those concerning the routes between main signals, forms a crucial part of this study. These rules encompass various parameters such as conditions for setting, locking, flank protection, and release of signals. A detailed exposition of these rules is available in reference [12]. Given the intricate nature of this subject matter, Table 3 provides a succinct overview of the basic route requirements as stipulated in the referenced document.

3.2 Allocation of Nodes and Edges in the Graph

Building upon the foundational research detailed in references [8–10], this paper adopts a structured approach to graph representation within the MATHEMATICA environment. In this model, the nodes of the graph are designated to represent distinct sections of the railway station, while the edges of the graph correspond to the tracks connecting these sections. Additionally, the labels on the graph edges are assigned to represent the appropriate signals governing the train movement along these tracks.

Further refining this graph representation involved transforming the initial undirected graph, as delineated in section 2, into two directed graphs. This transformation is critical for accurately depicting the directionality of train routes. Consequently, the resultant directed station graphs, as illustrated in Figure 4 and Figure 5, represent train routes in specific directions: Figure 4 for routes traveling from left to right, and Figure 5 for routes traveling from right to left.

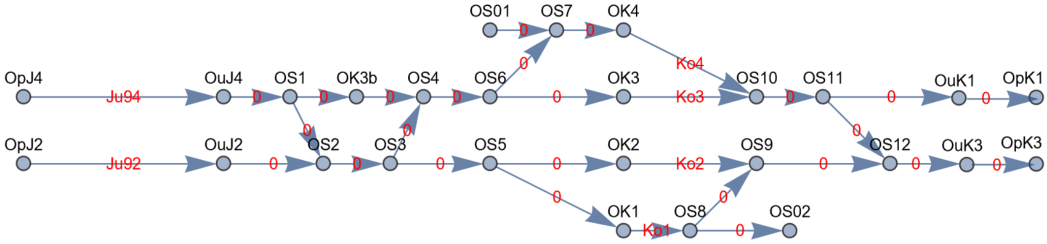


Figure 4. The corresponding station graph (train routes in the direction from left to right)

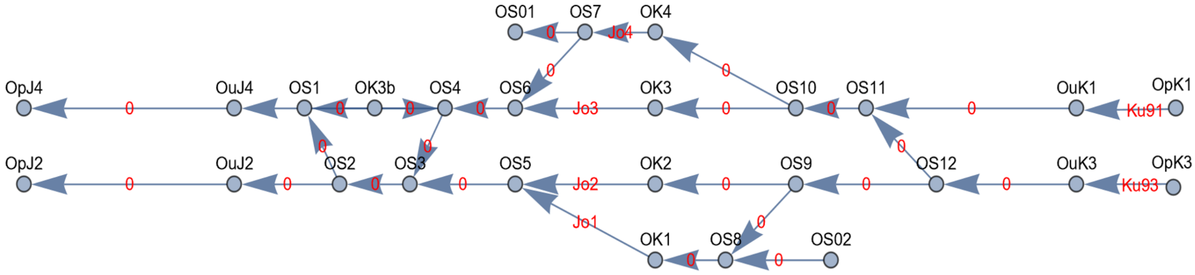


Figure 5. The corresponding station graph (train routes in the direction from right to left)

Table 3. An overview of the basic route requirements

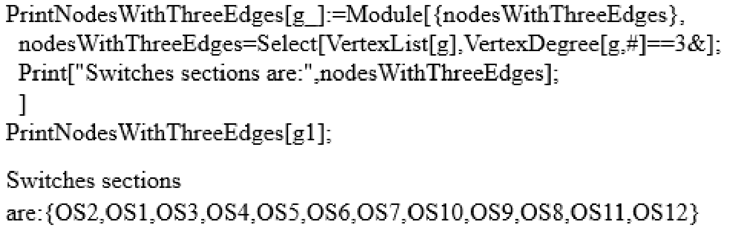
Route Requirement ID	Description
L1_SYS_ZE101	Train routes shall be settable: -from a main entry signal in the station to a main exit signal in the station (with overlap); -from a main signal in the station (entry or exit signal) to the open line (without overlap); -from entry signal to the limit track signal (with overlap).
L1_SYS_ZE12	Train routes must be settable individually or as a sequence, where main signals serve as both start and destination points in the route sequence.
L1_SYS_ZE13	Sequencing of train routes is allowed only if each individual route in the sequence is permitted.
L1_SYS_ZE18	Train routes require flank protection in both the route and overlap.
L1_SYS_ZE21	For points in the overlap, the overlap position is selectable.
L1_SYS_ZE22	All points (derailers) on the route and overlap require individual and automatic locking as dictated by the set route.
L1_SYS_ZE23	Overlaps are not settable individually.
L1_SYS_ZE25	Setting an exit route over an existing overlap is prohibited, except on the same track.
L1_SYS_ZE28	Route setting is allowed only if both the route and overlap are settable.
L1_SYS_ZE36	Signals in train routes should be activated sequentially from the destination to the start signal.
L1_SYS_ZE43	Route setting is prohibited if any part of the route is required for another route.
L1_SYS_ZE45	Route setting is prohibited if any track section or point in the route is blocked by the ”block” command.
L1_SYS_ZE46	Route setting is prohibited if any flank protection element is blocked by the ”block” command.
L1_SYS_ZE48	Using an overlap for a route is allowed if the route and overlap facilitate the same movement.
L1_SYS_ZE49	Points in a route should not be in demand as flank protection elements for another route if they are opposite to the position required by the route.
L1_SYS_ ZE50_1	Route setting is prohibited if any point within the route is occupied.
L1_SYS_ZE55	Setting a route over derailers is prohibited.
L1_SYS_ZE70Z	Points in an overlap should not be in demand as flank protection for another route in a position opposite to that required by the overlap.
L1_SYS_ZE82	Elements providing flank protection include points, derailers, main signals, and track limit signals.
L1_SYS_ZU16	For signals in the route: (1) Stop aspect is required for movements in the opposite direction. (2) Flank protection signals must show a stop aspect. (3) Opposite signals must show a stop aspect if routes protected by them overlap. (4) Opposite signals cannot be dark if at least one route protected by them does not overlap with the route.
L1_SYS_ZU17	Intermediate signals in the direction of movement should indicate a proceed aspect.
L1_SYS_ZU36_2	The start signal cannot show a proceed aspect if the destination signal is dark.
L1_SYS_ZU57_1	For signals located within the overlap and aligned with the direction of movement, it is imperative that they remain illuminated (i.e., not dark).
L1_SYS_ZU75	Sections between the route, overlap, and flank protection elements must be clear.
L1_SYS_ZS03	If the destination signal is dark, there will be no proceed aspect on the start signal and all intermediate signals on the route.
L1_SYS_ZS22	All signals in the route must show a stop aspect if flank protection conditions are not met.
L1_SYS_AW04	A route part is actively released if: (1) Cleared by the last axle. (2) The subsequent section is occupied (not due to a fault). (3) Occupied by a train (not due to a fault) since route setting. (4) The preceding part of the route has been released.
L1_SYS_AW05	The route is released by train movement one section at a time.
L1_SYS_AW06	Release of train route must begin at the start of the route.
L1_SYS_AW08	A destination section in the route can be released by train movement under specific conditions, even if it deviates from active release requirements.
L1_SYS_AD04	Overlap release occurs when: (1) The train completely enters the destination track. (2) Configured delay time expires. (3) All sections in the overlap are clear.
L1_SYS_AD06	Overlap release delay (90s) stops if a train occupies the overlap without a continually signaled route. Otherwise, it activates with the last axle's arrival to the destination track.
L1_SYS_AF01	Flank protection release for a demanding element occurs when the demand is canceled, which should happen as soon as the need for protection ceases (e.g., after route or overlap release).

3.3 Determination of the Routes from the Graph Topology

The process of route determination within the graph topology is a critical aspect of the control table generation methodology. It involves identifying all viable paths in the graph that start and end with a signal, representing the starting and destination points of a train route. Each row in the control table is then formulated to encapsulate the necessary conditions for setting a specific route in the railway station's signalling system.

A key aspect of this analysis is the treatment of each edge within the graph as a simple railway track section. However, a notable exception arises in scenarios where three edges converge at a single node, which is then identified as a switch point. The accurate detection and handling of such switch points pose a significant challenge in the route determination process. To address this, the orientation of the switch is determined based on the related track sections. Subsequently, the user is required to confirm the positions for each switch, a "+" signifying a straight route over the switch, and a "-" indicating a turnout route.

The methodology employed for detecting station switches in MATHEMATICA is not only intricate but also crucial for the accuracy of the route determination process. This is encapsulated in the pseudocode presented in Figure 6. This pseudocode provides a systematic approach to identify switch points within the station layout, ensuring that each route is accurately represented in the control table.



```
PrintNodesWithThreeEdges[g_]:=Module[{nodesWithThreeEdges},
  nodesWithThreeEdges=Select[VertexList[g],VertexDegree[g,#]==3&];
  Print["Switches sections are:",nodesWithThreeEdges];
  ]
PrintNodesWithThreeEdges[g1];

Switches sections
are:{OS2,OS1,OS3,OS4,OS5,OS6,OS7,OS10,OS9,OS8,OS11,OS12}
```

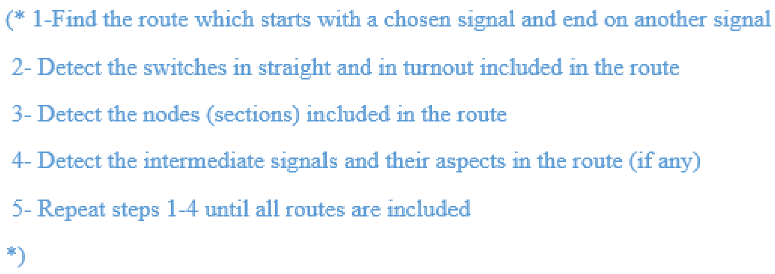
Figure 6. The pseudocode for the detection of the station switches

The route definition process is meticulously structured and hinges on the graph topology derived from the station layout. Routes are delineated between the entrance railway track section and the station tracks for entry routes, or between a station track and an exit railway track section for exit routes. This is achieved by utilizing the tags on the graph edges, which correspond to the relevant signals. In doing so, the methodology accurately identifies all track sections that comprise a route, including their sequential arrangement. Additionally, it determines the related switch points, along with essential information regarding their positions.

The determination of signal aspects is closely tied to the possible routes in the station topology. For instance, if a turnout is present in the route ahead of a signal, a flashing aspect (as detailed in options 7, 8, 9, or 10 from Table 2) is assigned to that signal. Conversely, in cases where the route proceeds in a straight direction, a steady signal aspect (options 5a or 6 from Table 2) is identified. This distinction is crucial for ensuring the accuracy and safety of signal operations within the station.

To implement this route determination process in MATHEMATICA, general graph searching algorithms, akin to those discussed in references [13–17], were employed. These algorithms are integral to navigating and analyzing the complex graph representations of station layouts.

Building upon the core graph theory functionalities of MATHEMATICA, as outlined in references [18–20], and aligning them with the signalling rules presented in Table 3, a specialized notebook was developed. This notebook serves as a dedicated tool for the determination of train routes within the station, incorporating the intricate signalling rules into the graph-based analysis. The general steps for generating routes in MATHEMATICA, encapsulating this comprehensive approach, are outlined in the pseudocode displayed in Figure 7.



```
(* 1-Find the route which starts with a chosen signal and end on another signal

 2- Detect the switches in straight and in turnout included in the route

 3- Detect the nodes (sections) included in the route

 4- Detect the intermediate signals and their aspects in the route (if any)

 5- Repeat steps 1-4 until all routes are included

*)
```

Figure 7. The pseudocode for generating the routes

The output of the route determination process, as applied to the station layout for train routes traveling from left to right, is systematically presented in Table 4.

The results obtained from the automated process of generating control tables in MATHEMATICA were subjected to manual verification and were found to be completely accurate. This outcome underscores a crucial advantage of automation: the significant reduction, or even elimination, of human error, which is often a risk factor in manual processes. The use of automated tools in railway signalling design and implementation not only enhances precision but also bolsters the reliability of control tables.

Table 4. Generated route table of routes in the direction from left to right

Route Start	Route End	Switches + Position	Switches - Position	Sections	Aspect-Start Signal	Aspect-End Signal
Ju92	Ko1	S2 S3	S5	OK1 OS2 OS3 OS5 OuJ2	8	4
Ju92	Ko2	S2 S3 S5	0	OK2 OS2 OS3 OS5 OuJ2	6	4
Ju92	Ko3	S2 S4 S6	S3	OK3 OS2 OS3 OS4 OS6 OuJ2	8	4
Ju92	Ko4	S2 S6	S3 S4	OK4 OS2 OS3 OS4 OS6 OuJ2	8	4
Ju94	Ko1	S3	S1 S2 S5	OK1 OS1 OS2 OS3 OS5 OuJ4	8	4
Ju94	Ko2	S2 S3 S5	S1	OK2 OS1 OS2 OS3 OS5 OuJ4	6	4
Ju94	Ko3	S1 S4 S6	0	OK3 OK3b OS1 OS4 OS6 OuJ4	8	4
Ju94	Ko4	S1 S4	S6 S7	OK3b OK4 OS1 OS4 OS6 OuJ4	8	4

4 Discussion

The results of this study highlight two main achievements that underscore the significance of automating the generation process of control tables in railway signalling:

-Efficiency and Error Reduction: Automation significantly shortens the duration of the overall project while concurrently minimizing the number of potential errors in the control table. This efficiency gain is contingent upon the accurate and unambiguous translation of specific signalling rules into the graph search algorithms used in the automation process. By streamlining the generation of control tables through automation, the time and resources invested in railway signalling projects can be considerably reduced, leading to more efficient project execution.

-Adaptability to Network Changes: Railway networks are inherently dynamic, often undergoing updates and modifications such as the addition of new switches or tracks. The automated processes showcased in this study excel in adapting to these changes. This adaptability ensures that control tables are always current, accurately reflecting the latest configuration of the railway network. The ability to quickly and accurately update control tables in response to changes in the network is a significant advantage, enhancing the responsiveness and reliability of railway operations.

Additionally, it's important to note the versatility of the proposed graphical representation of the results. The methodology allows for customization according to the specific requirements of different end users. The output does not need to be confined to a tabular form; it can also be translated into other formats such as specific software code or ladder diagrams. This flexibility in representation means that the results can be tailored to fit various applications and user preferences, further extending the utility and applicability of the automated process in diverse railway signalling contexts.

5 Conclusions

The traditional approach to preparing railway signalling control tables, integral to configuring signals and routes within a station, has historically been a meticulous and time-consuming task. The introduction of automation into this process marks a transformative advancement in the railway industry. The primary aim of this paper was to present a novel automated technique for generating railway signalling control tables, expanding upon existing methodologies found in the literature. This expansion involved a significant innovation: acquiring the station layout directly from an AutoCAD drawing, rather than relying on manual preparation in a dedicated editing tool.

The use of MATHEMATICA, a comprehensive computational software system, is central to this innovation. Its symbolic computation capabilities, combined with a broad spectrum of built-in functions, render it highly effective for managing complex mathematical operations and simplifying the coding of signalling logic. Furthermore, MATHEMATICA's advanced visualization tools enable users to create both informative and visually appealing representations of graphs and control tables. This graphical representation not only enhances the comprehension of signalling logic but also aids in easier troubleshooting and maintenance of the system.

A MATHEMATICA notebook developed as part of this research serves as a solid foundation for further automation and validation of railway signalling control tables. Future work is planned to integrate additional route conditions, such as the definition and generation of overlap and flank protection elements, which will be addressed in subsequent publications.

The potential for further enhancement of this tool lies in leveraging Natural Language Processing (NLP) and Large Language Models (LLM), particularly with the ongoing advancements in Artificial Intelligence (AI). One promising avenue is the integration of the Wolfram Language plugin for ChatGPT [19], combining the powerful capabilities of both platforms.

In conclusion, automating the preparation of railway signalling control tables not only streamlines the process but also results in significant long-term cost savings. By reducing manual labor and minimizing errors, it allows for more efficient resource utilization. This shift enables railway operators and infrastructure managers to reallocate personnel towards more strategic initiatives, thereby enhancing the overall cost-effectiveness and operational efficiency of control systems.

Data Availability

The data supporting the research results are included within the article or supplementary material.

Acknowledgments

The author wants to thank the authors of the papers [8–10] for their initial contribution and guidelines to this exciting topic, which served as a basis for the preparation of this paper.

Conflicts of Interest

The author declares no conflict of interest.

References

[1] F. van Dijk, W. Fokkink, G. Kolk, P. van de Ven, and B. van Vlijmen, "EURIS, a specification method for distributed interlockings," in *17th International Conference, SAFECOMP'98*, Heidelberg, Germany, 1998, pp. 296–305. https://doi.org/10.1007/3-540-49646-7_23

[2] P. James, A. Lawrence, F. Moller *et al.*, "Verification of solid state interlocking programs," in *Software Engineering and Formal Methods: SEFM 2013 Collocated Workshops: BEAT2, WS-FMDS, FM-RAIL-Bok, MoKMaSD, and OpenCert*, Madrid, Spain, 2014, pp. 253–268. https://doi.org/10.1007/978-3-319-05032-4_19

[3] S. Vanit-Anunchai, "Experience using coloured petri nets to model railway interlocking tables," in *2nd French Singaporean Workshop on Formal Methods and Applications EPTCS 156*, 2014, pp. 17–28. https://doi.org/10.4204/EPTCS.156.6

[4] I. Ustoglu, D. Topel, M. S. Durmus, R. Y. Tsarev, and K. Y. Zhigalov, "Obtaining generic petri net models of railway signaling equipment," in *Intelligent Systems Applications in Software Engineering: Proceedings of 3rd Computational Methods in Systems and Software*, Czech Republic, 2019, pp. 174–188. https://doi.org/10.1007/978-3-030-30329-7_17

[5] A. Nash, D. Huerlimann, J. Schutte, and V. P. Krauss, "RailML-A standard data interface for railroad applications," *WIT Trans. Built Environ.*, vol. 74, pp. 233–240, 2004. https://doi.org/10.2495/CR040241

[6] B. Luteberget, J. J. Camilleri, C. Johansen, and G. Schneider, "Participatory verification of railway infrastructure by representing regulations in RailCNL," in *15th International Conference, SEFM 2017*, Trento, Italy, 2017, pp. 87–103. https://doi.org/10.1007/978-3-319-66197-1_6

[7] A. Hernando, R. Maestre, and E. Roanes-Lozano, "A new algebraic approach to decision making in a railway interlocking system based on preprocess," *Math. Probl. Eng.*, vol. 2018, p. 4982974, 2018. https://doi.org/10.1155/2018/4982974

[8] I. Mutlu, U. Yıldırım, M. S. Durmuş, and M. T. Soylemez, "Automatic interlocking table generation for non-ideal railway yards," *IFAC Proc. Vol.*, vol. 46, no. 25, pp. 55–59, 2013. http://doi.org/10.3182/20130916-2-TR-4042.00035

[9] U. Yıldırım, M. S. Durmus, and M. T. Soylemez, "Automatic generation of the railway interlocking tables by using computer algebra toolbox," *Int. J. Adv. Comput. Technol.*, vol. 7, no. 3, pp. 109–121, 2015.

[10] M. S. Durmus and M. T. Soylemez, "Automatic interlocking table generation for railway stations using symbolic algebra," *IFAC Proc. Vol.*, vol. 45, no. 24, pp. 171–176, 2012. https://doi.org/10.3182/20120912-3-BG-2031.00035

[11] Directorate for Railways, "Signalling rulebook," 2020. https://www.raildir.gov.rs/doc/podzak/Pravilnik_o_vrstama_signala.docx

[12] JSC, "User requirements specifications for ETCS level 2 system on railway network in Republic of Serbia v1.2," 2020.

[13] E. Roanes-Lozano, “Looking for compatible routes in the railway interlocking system of an overtaking station using a computer algebra system,” in *22nd International Workshop on Computer Algebra in Scientific Computing*, Linz, Austria, 2020, pp. 528–542. https://doi.org/10.1007/978-3-030-60026-6_31

[14] E. Roanes-Lozano, J. Galan-Garcia, and G. Aguilera-Venegas, “A computer approach to overtaking station track layout diagram design using graphs: An alternative track diagram proposal for these stations,” *J. Comput. Appl. Math.*, vol. 391, p. 113455, 2021. https://doi.org/10.1016/j.cam.2021.113455

[15] E. Roanes-Lozano, J. A. Alonso, and A. Hernando, “An approach from answer set programming to decision making in a railway interlocking system,” *Rev. R. Acad. Cienc. Exactas Fís. Nat. Ser. A Mat.*, vol. 108, no. 2, pp. 973–987, 2014. http://doi.org/10.1007/s13398-013-0155-1

[16] T. Kavithaa, C. Liebchenb, K. Mehlhornc, D. Michaild, R. Rizzie, T. Ueckerdtf, and K. Zweig, “Cycle bases in graphs characterization algorithms, complexity, and applications,” *Comput. Sci. Rev.*, vol. 3, no. 4, pp. 199–243, 2009. https://doi.org/10.1016/j.cosrev.2009.08.001

[17] X. Chen, Y. He, and H. Huang, “A component-based topology model for railway interlocking systems,” *Math. Comput. Simul.*, vol. 81, no. 9, pp. 1892–1900, 2011. https://doi.org/10.1016/j.matcom.2011.02.007

[18] WOLFRAM U, “Graphs and networks: Concepts and applications,” WOLFRAM U Course. https://www.wolfram.com/wolfram-u/courses/data-science/graphs-and-networks-concepts-and-applications-dat021/

[19] WOLFRAM, “Wolfram plugin for chatgpt,” WOLFRAM Product Page. https://www.wolfram.com/wolfram-plugin-chatgpt/

[20] WOLFRAM, “Hands-on start to wolfram mathematica,” WOLFRAM U Course. https://www.wolfram.com/wolfram-u/courses/wolfram-language/hands-on-start-to-mathematica-wl005/