Mathematics

http://www.aimspress.com/journal/Math

AIMS Mathematics, 9(3): 7673–7710.
DOI: 10.3934/math.2024373
Received: 12 October 2023
Revised: 06 February 2024
Accepted: 19 February 2024
Published: 22 February 2024

Research article

A fast and general algebraic approach to Railway Interlocking System across all train stations

Antonio Hernando[1], **José Luis Galán-García**[2,*] **and Gabriel Aguilera-Venegas**[2]

[1] Depto. de Sistemas Informáticos, E.T.S.I. de Sistemas Informáticos, Universidad Politécnica de Madrid, Madrid, Spain

[2] Departamento de Matemática Aplicada, Escuela de Ingenierías Industriales, Universidad de Málaga, Málaga, Spain

* **Correspondence:** Email: jlgalan@uma.es.

Abstract: Railway interlocking systems are crucial safety components in rail transportation, designed to prevent train collisions by regulating switch positions and signal indications. These systems delineate potential train movements within a railway station by connecting sections into routes, which are further divided into blocks. To ensure safety, the system prohibits the simultaneous allocation of the same block or intersecting routes to multiple trains. In this study, we characterize the 'interlocking problem' as a safety verification task for a single real-time station configuration, rather than a 'command and control' function. This is a matter of verification, not solution, typically managed by an interlocking system that receives movement authority requests. Over the years, we have developed various algebraic models to address this issue, suggesting the potential use of computer algebra systems in implementing interlocking systems. However, some of these models exhibit limitations. In this paper, we propose a novel algebraic model for decision-making in railway interlocking systems that overcomes the limitations of previous approaches, making it suitable for large railway stations. Our primary objective is to offer a mathematical solution to interlocking problems in linear time, which our approach accomplishes.

Keywords: railway interlocking system; computer algebra; decision making;commutative algebra
Mathematics Subject Classification: 13P05

1. Introduction

Rail transportation is a crucial mode of transport globally, offering a reliable and efficient means of transporting goods and people over long distances. It plays a vital role in the economy and society, as highlighted in various studies on resilience in railway transport systems [1, 2].

A key component of rail transportation is the railway interlocking system, which is a safety-critical system designed to prevent train collisions. The compatibility of switch positions and signal indications at a railway station is crucial for safety. The interlocking system prevents improper changes to traffic signals and turnout switches. A railway station comprises sections connected by traffic signals and turnouts, defining possible train movements. A route is a sequence of connected sections that a train can travel along, partitioned into several blocks. To prevent collisions, two trains may never be in the same block of a route. Two intersecting routes (or their relevant blocks) must not be allocated to trains simultaneously.

In this paper, we define the 'interlocking problem' as follows: given a station topology composed of track sections, signals, and turnouts, and given a specific run-time configuration (which includes trains positioned in some sections, the positions of turnouts, and signal aspects), we must decide whether the signal aspects and turnout positions allow the trains present in the station to move in such a way that they could potentially collide. This definition refers to a 'safety check' for a single run-time configuration of a station, rather than a 'safe command and control' function, which is what existing interlocking systems implement. It is a verification problem, not a solution problem. In practice, an interlocking system usually receives movement authority requests from trains, a dispatcher, or another system. It then has to command the positions of turnouts and signal aspects to grant a movement authority to the train over a path that fulfills the request. This is done after checking that the path is free from other trains and is exclusively reserved for the requesting train.

This complex problem has been extensively studied, with recent research exploring artificial intelligence applications for fault detection [3] and comparing different safety verification methods [4]. There is also work on developing formal model-based methodologies to aid railway engineers in specifying and verifying interlocking systems [5].

Computer-based railway interlocking systems can be classified as follows:

- Route-table based: For every route request, an algorithm checks its feasibility using a "control table". The software consists of a single algorithm, independent of the topology.
- Geographical: The interlocking program is made up of instances of software objects that mimic the behavior of physical objects. The configuration of this program depends on the topology. Within this category, we can distinguish between:
  - Route-based: Routes are defined a priori, and their definitions constitute data shared by the instantiated objects.
  - On-demand route definition: There is no a priori definition of routes. Instead, a train's request to the interlocking system is given only as the final destination that the train must reach. Instantiated software objects are responsible for exploring possible routes to the destination and choosing one of them. Our approach lies in this category.

Classical railway interlocking systems are route-based and the compatibility of routes is decided in advance [6]. Some modern railway interlocking systems can make decisions on the fly. This makes them more flexible as routes are not predetermined. However, before authorizing any changes to signals or switch positions, it is necessary to check if two trains would travel on intersecting routes (including the same route) and potentially collide. All modern railway interlocking systems are computer-based, either geographical or route-table based. However, naive implementations of geographical algorithms may encounter efficiency problems when finding safe routes through the railway network. Efficient data validation for these geographical systems has been studied [7], and model checking using UMC has been proposed as another verification method [8].

Over the years, we have developed various models to study this problem [9, 10, 11, 12, 13, 14]. Some of these models are based on polynomials, ideals, and Groebner bases, similar to those used in artificial intelligence for implementing expert systems [15]. This approach bridges computational algebra and interlocking problems, suggesting that computer algebra systems can be used to implement interlocking systems.

Recently, we presented a groundbreaking algebraic model [16] that improves the implementation of interlocking systems. This model provides a linear algorithm that significantly outperforms previous models and is suitable for large-scale railway stations. However, this model has a limitation: the railway station must not have cycles in paths for any configuration of the switches of the turnouts and the indication of color lights. If this condition is not met, the model may produce incorrect conclusions. In this paper, we propose a new algebraic model that builds on the ideas of the previous work [16]. Unlike this model, our present approach does not require any specific conditions on the railway stations while retaining all its advantages over the earlier model [16].

The approach presented in [16] is entirely novel and offers intriguing theoretical perspectives. However, it has yet to gain acceptance within the general railway research community due to a significant limitation: the challenge of implementing it in actual interlocking systems. This is primarily due to the stringent certification requirements associated with such safety-critical applications. Despite this, the results could prove beneficial for simulations that do not require certification. The approach in [16] does not address the general problem and only works in cases where there are no cycles. From a mathematical perspective, overcoming the problem of cycles with the approach in [16] is not straightforward. Indeed, we have had to define new and highly abstract concepts to solve them, which are not immediately apparent nor derived from the paper [16]. We believe that the idea of extending the concept of interlocking problems to extended interlocking problems and defining a completely new relation between them is very novel from a mathematical perspective. We are confident that our manuscript offers fresh insights into the relationship between two seemingly disconnected fields, making it of considerable interest from a theoretical standpoint.

The paper is structured as follows: In Section 2, we explore the limitations of the framework presented in [16] and provide the motivation behind our proposal. In Section 3, we outline the main ideas used to address the limitation of [16]. In Section 4, we present formal definitions of interlocking problems, which are necessary to validate our approach. In Section 5, we present the key concept upon which our algebraic model is based: extended interlocking problems. In Section 6, we translate all previously described concepts into algebraic terms. The static parts of the railway stations are represented by polynomials while the dynamic part by means of a monomial. In Section 7, we demonstrate how an interlocking system can be implemented by means of a Computer Algebra System. In Section 8, we conduct a comprehensive comparison of the performance of our proposed approach against other existing methods. In Section 9, we set our conclusions. Finally, in the appendix, we provide an algorithm with linear complexity.

2. Motivation

In [16], the authors introduced a novel algebraic approach for detecting dangerous situations in railway stations. This method, which is based on a linear algorithm, significantly outperforms previous models presented by the authors or their direct collaborators. Its robustness and efficiency make it particularly suitable for application in large-scale railway stations.

The approach is based on representing a railway station by means of a set of polynomials. Then, the problem of determining if a railway station is in a dangerous state is translated into the algebraic problem of determining if the remainder of a monomial on division by a set of polynomials is zero.

However, while this method is significantly faster than previous algebraic models, it requires a strict condition within the railway station: for any given configuration, the paths within the railway station must not form cycles. Without this assumption, the approach in [16] does not work. For example, the algorithm does not work in the house layout depicted in Figure 1.

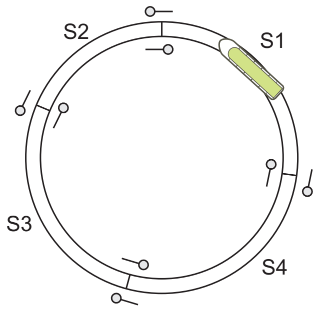

Figure 1. Example of a house layout in which the approach does not work.

It is evident that there is a single train present at this railway station, eliminating the possibility of a collision. Despite this, the algebraic model referenced in [16] inaccurately indicates that this situation is hazardous. This error arises because the model relies on a theorem that is only applicable to railway stations devoid of cycles.

Although the presence of cycles, as depicted in Figure 1 (we have provided this example solely to highlight the limitations of the preceding algebraic model), is not common in railway stations, such cycles may form when the station includes a specific type of turnout: trailable turnouts. A turnout, regardless of its type, is a common element in a railway station that connects three sections. Figure 2 depicts a turnout connecting three sections: S1, S2, and S3. A switch in the turnout determines the possible movement of the trains: if the switch is in the direct track position, then a train can move from section S1 to section S2 (and vice versa); if the switch is in the diverted track position, a train can move from section S1 to section S3 (and vice versa). We will distinguish between two types of turnouts:

Non-trailable turnouts. In case that the turnout is non-trailable, a train may derail if the switch is in the direct track position and it tries to pass from section S3, or if the switch is in the diverted track position and it tries to pass from section S2. For this reason non-trailable turnouts are usually properly protected by light signals.

Trailable turnouts. Trailable turnouts avoid potential derailment. For a trailable turnout, a train passes from section S3 to section S1 in the case that the switch is in direct track position, and it can also pass from section S2 to section S1 in the case that the switch is in diverted track position.

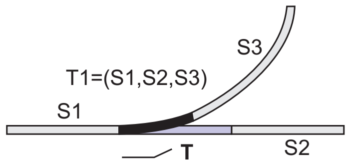

Figure 2. Trailable Turnout.

Trailable turnouts are commonly used in the configuration depicted in Figure 3 because trains can easily reverse directions. This configuration involves three trailable turnouts: the one connecting sections S1, S2, and S3 whose switch is in the diverted tuck position; the one connecting S4, S3, and S5 whose switch is in the diverted track position; and the one connecting S6, S2, and S5 whose switch is in the straight track position.

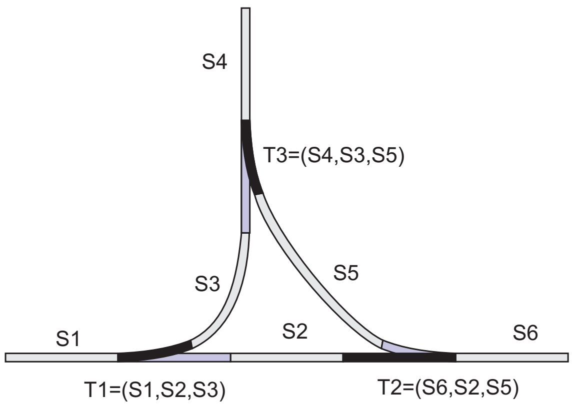

Figure 3. Configuration of three trailable turnout to reverse the direction of trains.

In Figure 4, we see the movement of the train to reverse the direction: the train moves from section S1 to section S3 because the switch of the turnout (S1,S2,S3) is in the diverted track position; next, the train moves from section S3 to section S4 because the turnout is trailable; next, the train stops and reverses from section S4 to section S5 since the switch of (S4,S3,S5) is in the diverted track position; next, the train moves backwards from section S5 to section S6 because the turnout is trailable; next, the train moves forward from section S6 to section S2 because the switch of (S6,S2,S5) is in the direct track position; and finally, the train moves forward from section S2 to section S1 because the turnout is trailable.

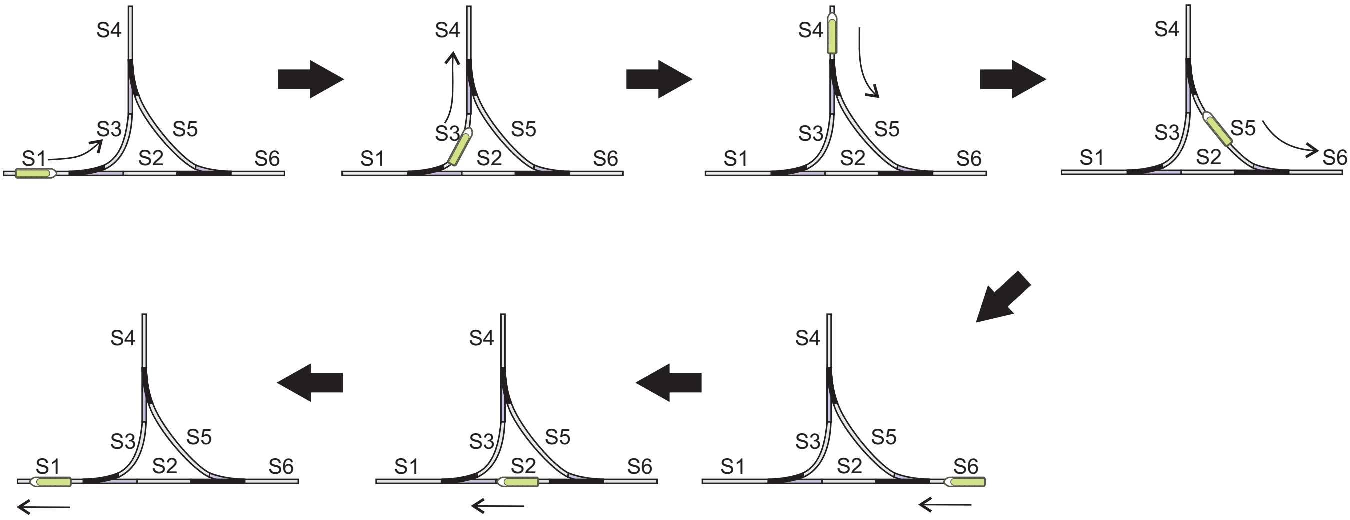

Figure 4. A train reversing direction.

Trailable turnouts may produce the presence of cycles in the railway station. Let us consider the simple railway station of Figure 5, considering that both turnouts are trailable. As may be seen, there is a cycle, and indeed the algorithm of the approach in [16] will output that the potential collisions may happen if only one train is placed in section S1 (which is not possible since there is only one train placed in the railway station).

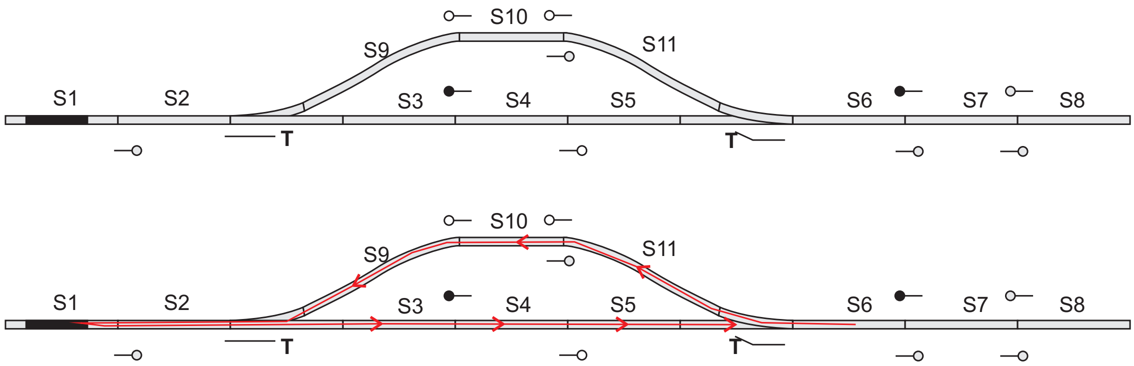

Figure 5. A cycle is formed by means of the trailable turnouts.

Consequently, the approach in [16] has a significant limitation: it requires that the railway station does not use trailable turnouts because its presence leads to the formation of cycles. This requirement prevents the approach from being used in all previous figures. Here, we present a new algebraic approach that overcomes this limitation by accounting for the possibility of cycles, while remaining extraordinarily fast and suitable for large stations.

3. Overview of our approach

Distinct from others, the approach presented in [16] is based on a specific definition of the "interlocking problem" and an analysis of the relation between different interlocking problems. We denote this relation as $\to$.

The relation $\to$ has a significant property: if $A \to B$, and if A is in a dangerous situation, then B is also in a dangerous situation.

The reciprocal property is not guaranteed in general, and indeed, for a general railway station it is possible that B is in a dangerous situation and A is not. However, if the railway station does not have cycles in its paths, the reciprocal property is also fulfilled (despite $\to$ not being symmetric).

Consequently, for the case of a railway station whose paths does not form cycles, we have that:

if $A \to B$, then A is in a dangerous situation if and only if B is also in a dangerous situation.

The strategy for determining whether an interlocking problem X_1 is in a dangerous situation in [16] is as follows: iteratively derive new interlocking problems $X_1 \to X_2 \to \ldots \to X_n$ until X_n can be solved immediately. Figure 6 illustrates the set of interlocking problems for a railway station and the relation $\rightsquigarrow$. Through a representation based on polynomials, this process of iteratively obtaining new derived interlocking problems can be automated using the division algorithm.

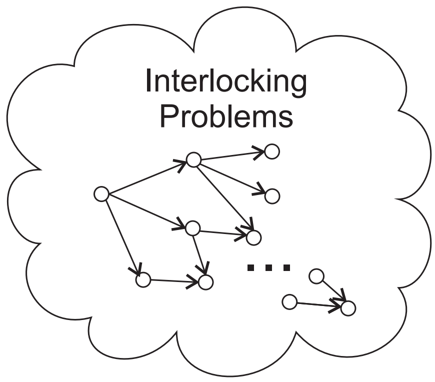

Figure 6. The set of interlocking problems and the relation $\to$.

However, this strategy only works under the condition that the railway station does not have cycles Otherwise, our strategy fails because it could be possible that X_n is in a dangerous situation, but X_1 is not. This is why the algorithm proposed in [16] does not work in railway stations with cycles.

In this paper, we will overcome this obstacle in the following way: we will extend the set of interlocking problems into a larger, more abstract set — the set of extended interlocking problems. Then, we will define a new relation, denoted by $\rightsquigarrow$, which, although not symmetric, fulfills the desired property:

if $A \rightsquigarrow B$, then A is in a dangerous situation if and only if B is also in a dangerous situation. (1)

We would like to emphasize that the relation $\rightsquigarrow$ is completely new and is not an extension of R_1. For two interlocking problems A and B, we could have that $A \to B$, but $A \not\rightsquigarrow B$.

Once we have defined this set of extended interlocking problems, and the relation $\rightsquigarrow$ between them, our strategy will be similar to [16]: we will iteratively derive new interlocking problems $X_1 \rightsquigarrow X_2 \rightsquigarrow \ldots \rightsquigarrow X_n$ until X_n can be solved immediately*. In this paper, we have found a completely

*As we will see, X_n can be solved immediately because trains are positioned within locked sections, meaning they are prohibited from moving.

new representation based on polynomials, so that this process of obtaining the extended interlocking problems $X_1, X_2, \ldots \to X_n$ can also be automatically done using the division algorithm.

Unlike [16], our current model works for any railway station because property (1) holds regardless of whether paths of the railway station may form cycles or not.

Figure 7 depicts the set of interlocking problems for a railway station and the relation 'is derived from'. Once the relation $\rightsquigarrow$ is defined, we can translate the problem into polynomials and detect if an interlocking problem is in a dangerous situation by means of the division algorithm.

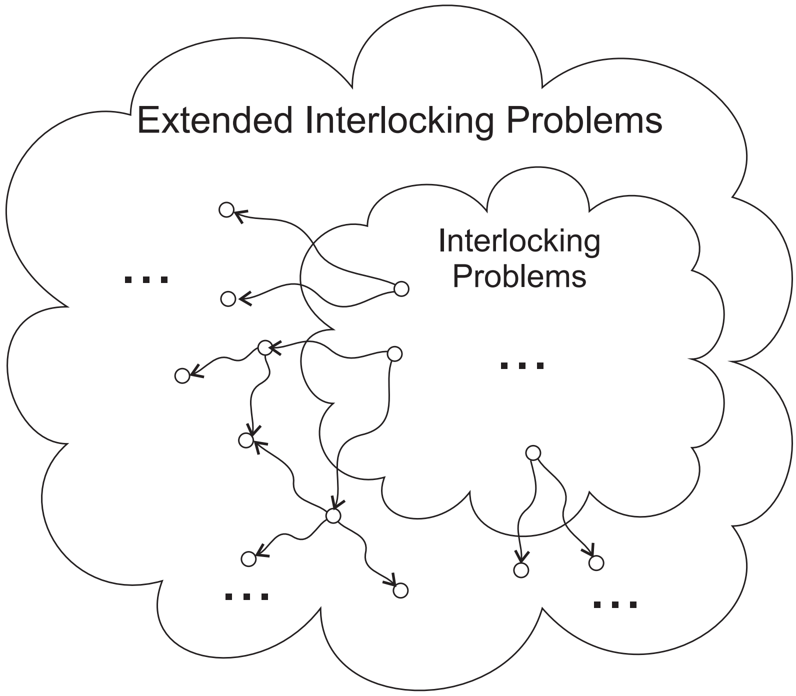

Figure 7. The set of extended interlocking problems and the new relation $\rightsquigarrow$.

4. Interlocking problems

In this section, we will formalize some key concepts related to railway stations and interlocking problem that allow us to prove the validity of our approach.

A railway station is defined by a finite set of sections $\{S_1 \ldots S_n\}$. Sections are physically connected by means of light signals and turnouts. We define this by means of a relation E.

Definition 4.1. *Given a railway station, we define the set $E \subset \mathbb{Z} \times \mathbb{Z}$ as:*

$$E = \{(i, j) | S_i \textit{ is connected to } S_j \textit{ or } S_j \textit{ is connected to } S_i \\ \textit{by means of a color light signal or a turnout}\}$$

Remark 4.1. *In a railway station, each section is connected to a maximum of two other sections on each extreme through a turnout. This means that each section can be connected to a maximum of four other sections. Therefore, the size of E is less than or equal to $4 \cdot N$ where N is the number of sections. As a result, the number of elements in E, is O(N).*

According to this definition, E is a symmetric relation. This relation informs whether there is physical contact between two sections. Figure 8 depicts an example of a railway with 11 sections. As may be seen, $(1, 2) \in E$ since S_1 is connected to S_2. We would like to emphasize here that the fact that $(1, 2) \in E$ does not indicate that a train can always pass from section S_1 to section S_2. Indeed, in this case, it depends on the indication of the light signal.

In this way, we define a relation P for indicating if a train can pass from a section S_i to another connected to section S_j in a particular configuration of the railway station.

Definition 4.2. *Given a railway station, we define the set $P \subset E$ as:*

$$P = \{(i, j) \in E |\ \textit{if a train can pass from section } S_i \textit{ to section } S_j\}$$

In the case of a turnout connecting S_i to S_j or S_k, we can analyze the difference in P of the section pairs regarding a normal turnout and those regarding a trailable turnout:

- If the turnout is trailable, both $(j, i) \in P$ and $(k, i) \in P$ hold true, regardless of the position of the switch.
- If the turnout is non-trailable, we have that:
  - In the case where the switch is in the direct track position, $(j, i) \in P$ and $(k, i) \notin P$.
  - In the case where the switch is in the diverted track position, $(j, i) \notin P$ and $(k, i) \in P$.

Figure 9 depicts a possible configuration of the railway station. As may be seen, since the switch of the turnout connecting sections S_2, S_3, and S_9 is in the direct track position, we have that $(2, 3) \in P$ and $(2, 9) \notin P$.

Just observe that $(5, 6) \in P$ and $(11, 6) \in P$ because the second turnout is trailable, whereas $(9, 2) \notin P$ and $(3, 2) \in P$ because the fist turnout is non-trailable.

There may be trains placed in the sections of a railway station. We will consider that each train is placed on just one section†. Since we assume that a collision occurs when more trains are placed in the same section, we will use multiset Q to represent occupancy of sections: a section belongs to Q if it is occupied by a train, and the section is repeated in Q once for each train placed in it.

Definition 4.3. *We define the multiset Q as the multiset of integers indicating the sections in which a train is placed: the number of times that element i appears in Q represents the number of trains located in section S_i.*

We will consider that Q is a multiset instead of a set, since Q will have repeated elements in case two different trains are placed in the same section (which is dangerous).

Figure 10 depicts a situation of the railway station in which there are two trains: one occupies section S_1; and another occupies section S_{10}. Consequently we have that $Q = \{1, 10\}$. If there are two trains occupying the same section S_1, then we would have that $Q = \{1, 1\}$.

Trains move through the railway network guided by the indications of light signals and turnouts. The concern arises when considering the possibility of a train colliding with another. We formally define an interlocking problem as an ordered pair (P, Q).

†In this paper, we will assume that each train occupies one section, as we showed in Section 8 of [16] that the general case where trains occupy multiple sections can be reduced to an equivalent interlocking problem where trains occupy a single section.

Definition 4.4. *An* interlocking problem *is an ordered pair (P, Q), where P is the set of Definition 4.2 and Q is the multiset of Definition 4.3.*

Given an interlocking problem (P, Q) we are interested in determining if movement of trains (following indications of the light signals and position of the switches) may result in a collision. We will formally define it:

Definition 4.5. *We will say that the railway interlocking problem (P, Q) is in a dangerous situation if and only if there are two lists of integers (we will call them* 'paths'*), $[u_1, \dots u_n]$ and $[v_1, \dots v_m]$ such that all the following conditions hold:*

(1) For all $0 < i < n$, we have that $(u_i, u_{i+1}) \in P$. That is to say, a train can pass from u_i to u_{i+1}.
(2) For all $0 < j < m$, we have that $(v_j, v_{j+1}) \in P$. That is to say, a train can pass from v_j to v_{j+1}.
(3) $\{u_1, v_1\} \subseteq Q$. That is to say, there must be a train in u_1 and v_1.
(4) $u_n = v_m$. That is to say, both paths reach the same section (there is a possible collision).
(5) For all $1 < i < n$ $u_i \notin Q$ and for all $1 < j < m$ $v_j \notin Q$. That is to say, intermediate sections in the list must be free of trains so that trains from section u_1 and section v_1 may reach section $u_n = v_m$.
(6) For all $1 \le i < n$ and for all $1 \le j < m$ we have $u_i \ne v_j$. That is to say, the possible collision happens at the end of the paths.
(7) For all $i \ne j$ we have $u_i \ne u_j$. That is to say, the path $[u_1, \dots u_n]$ does not contain cycles.
(8) For all $i \ne j$ we have $v_i \ne v_j$. That is to say, the path $[v_1, \dots v_m]$ does not contain cycles.

We will say that the railway interlocking problem (P, Q) is in a safe situation if and only if (P, Q) is not in a dangerous situation.

Example 4.1. *Let us consider the railway station of Figure 8.*

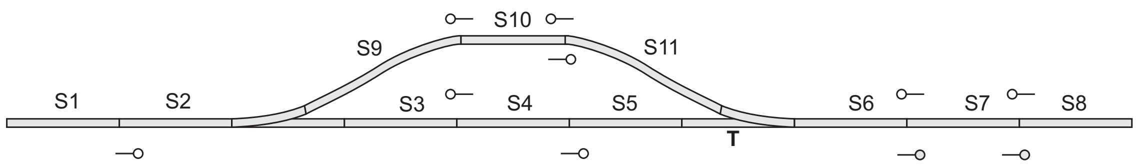

Figure 8. Track layout of a very simple railway station.

As may be seen, the railway station is divided into sections $S_1 \dots S_{11}$. There are ten color light signals (for example, there is one from S_1 to S_2) and two turnouts:

- *A turnout connecting sections S_2, S_3, and S_9.*
- *A turnout connecting sections S_6, S_5, and S_{11}. We have marked a turnout with 'T' in the figure to indicate that the turnout is trailable.*

According to Definition 4.1, the set E is:

$$E = \{(1,2),(2,9),(9,10),(10,11),(11,6),(2,3),(3,4),(4,5),(5,6),(6,7),(7,8)$$
$$(2,1),(9,2),(10,9),(11,10),(6,11),(3,2),(4,3),(5,4),(6,5),(7,6),(8,7)\}$$

Let us consider the configuration depicted in Figure 9. The position of the switches of the turnouts is represented by:

- *a small segment, if the switch is in the direct track position (see for instance the turnout between sections S_2 and S_3, S_9 in Figure 9), or*
- *a small angle, if the switch is in the diverted track position (see for instance the turnout between sections S_6 and S_5, S_{11} in Figure 9).*

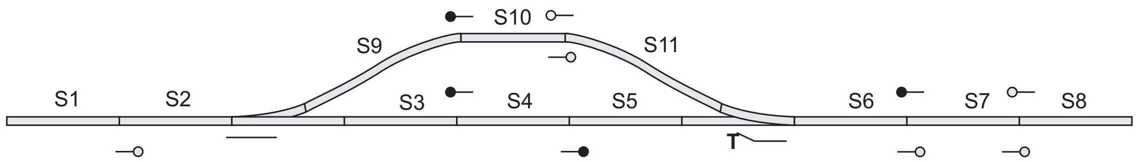

Figure 9. A possible configuration of the railway station of Figure 8.

*In view of the fact that the figures are printed in black and white, the aspect of the color light signals will be represented by a black circle (*stop *indication) or white circle (*proceed *indication).*

Now, we will determine the subset $P \subseteq E$ given by the indications of the light signals and the position of the switches of the turnouts in Figure 9:

$$P = \{(1,2),(9,10),(10,11),(2,3),(3,4),(11,6),(6,7),(7,8),(2,1),\\ (11,10),(5,4),(3,2),(6,11),(8,7),(5,6)\}$$

As may be seen, $(10,9) \notin P$ *because the indication of the color light signal connecting sections S_{10} and S_9 is* stop*. In the same way,* $(5,6) \in P$ *because* (S_6, S_5, S_{11}) *is a trailable turnout. However,* $(9,2) \notin P$ *because the switch of the non-trailable turnout* (S_2, S_3, S_9) *is in the direct track position.*

Figure 10 depicts the placement of one train in section S_1 and another train in section S_{10}. Therefore, in this case: $Q = \{1, 10\}$

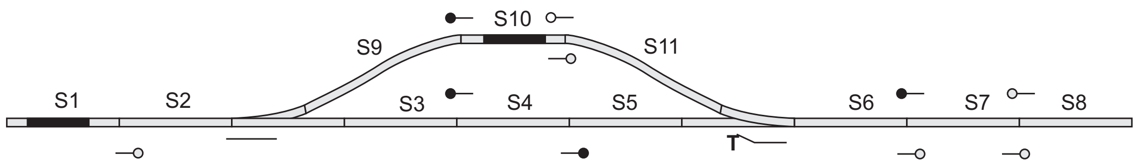

Figure 10. A possible placement of trains in the configuration of the railway station.

As may be seen in the figure, this situation is safe. However, if a new train were placed in section S_8, we would have that $Q = \{1, 8, 10\}$ *(Figure 11) and the situation would turn into a dangerous one: the trains situated in sections S_{10} and S_8 could collide in section S_7. As may be seen, we have that* $[10, 11, 6, 7]$ *and* $[8, 7]$ *fulfill the conditions in Definition 4.5.*

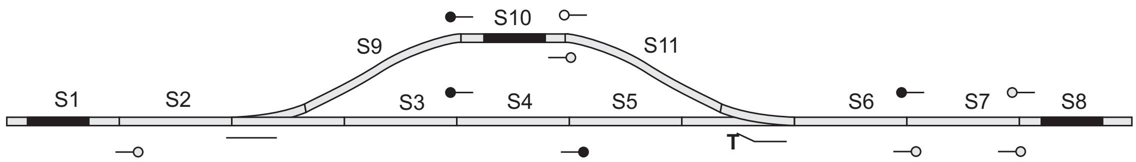

Figure 11. Another possible placement of trains in the configuration of the railway station. The proposed situation is dangerous.

5. Extended interlocking problems

In this section, we will extend the concept of the interlocking problem, and we will define a relation $\rightsquigarrow$ to such extension.

5.1. The concept of the extended interlocking problem

First, we will define the concept of a locked section. A locked section is one in which, if a train stayed there, it cannot move to another section. Formally,

Definition 5.1. *Given the set $P \subset E$, we say that a section i is locked in P if for every $(a, b) \in P$ we have that $a \neq i$.*

Intuitively, if a train is positioned on a locked section, it becomes immobilized and is unable to transition to another section within the railway station. Take, for instance, section S4 in Figure 9, which is a locked section. As observed in Example 4.1, neither $(4, 3)$ nor $(4, 5)$ belong to P. Consequently, if a train is placed on S4, it becomes 'locked' in place, as it is incapable of moving to either section S3 or S5.

Once locked sections are defined, we define the concept of the extended interlocking problem (Definition 5.2) and the conditions that it must fulfill to be in a dangerous situation (Defintion 5.3).

Definition 5.2. *Let E be a railway network. An extended interlocking problem is a (P, S, T, Q) where:*

- *P is a configuration of the railway network.*
- *S is a multiset of locked sections in P.*
- *T, Q are multisets of sections.*

Definition 5.3. *We will say that the railway interlocking problem (P, S, T, Q) is in a dangerous situation if and only if there are two lists of integers (we will call them* 'paths'*) $[u_1, \dots u_n]$ and $[v_1, \dots v_m]$ such that all the following conditions hold:*

(1) For all $0 < i < n$, we have that $(u_i, u_{i+1}) \in P$.
(2) For all $0 < j < m$, we have that $(v_j, v_{j+1}) \in P$.
(3) $\{u_1, v_1\} \subseteq S \cup T \cup Q$ and $u_1 \in S \cup Q$.
(4) $u_n = v_m$.
(5) For all $1 < i < n$ $u_i \notin S \cup Q$ and for all $1 < j < m$ $v_j \notin S \cup Q$.
(6) For all $1 < i < n$ and for all $1 < j < m$ we have $u_i \neq v_j$.
(7) For all $i \neq j$ we have $u_i \neq u_j$.
(8) For all $i \neq j$ we have $v_i \neq v_j$.

We will say that the railway interlocking problem (P, S, T, Q) is in a safe situation if and only if (P, S, T, Q) is not in a dangerous situation.

Remark 5.1. *As may be seen in Definition 4.5 and Definition 5.3, any interlocking problem (P, Q) can be regarded as the extended interlocking problem $(P, \emptyset, \emptyset, Q)$.*

Remark 5.2. *Referring to Definition 4.5 and Definition 5.3, we can deduce that an extended interlocking problem (P, S, T, Q) is in a dangerous situation if, and only if, there exists some $t \in T$ such that the interlocking problem $(P, S \cup \{t\} \cup Q)$ is also in a dangerous situation.*

5.2. *The relation* 'derived from' *in the set of extended interlocking problems*

Once we define the concept of the extended interlocking problem, we will analyze the set of extended interlocking problems for a specific railway station by means of a binary relation: '*derived from*'. This relation is the reflexive and transitive closure of another one, '*directly derived from*'.

Definition 5.4. *Let (P_1, S_1, T_1, Q_1) and (P_2, S_2, T_2, Q_2) be two extended interlocking problems of a railway network E. We say that (P_2, S_2, T_2, Q_2) is directly derived from (P_1, S_1, T_1, Q_1) if and only if one of these conditions holds:*

i) $T_1 = \emptyset; t \in Q_1$ *and we have that:*

$$P_2 = P_1; T_2 = \{t\}; S_2 = S_1; Q_2 = Q_1 - \{t\}$$

ii) $T_1 = T \cup \{t, t\}$ *and we have that:*

$$P_2 = P_1; T_2 = T \cup \{t\}; S_2 = S_1; Q_2 = Q_1$$

iii) If all sections in T_1 are locked in P_1, a section $t \in T_1$ is not repeated in T_1 and we have that:

$$P_2 = P_1; T_2 = T_1 - \{t\}; S_2 = S_1 \cup \{t\}; Q_2 = Q_1$$

iv) $S_1 = S_2$*;* $Q_1 = Q_2$*; and* $\exists (i, j) \in P_1$ *such that:*

- $i \in T_1$.
- $P_2 = P_1 - \{(i, j)\}$.
- $T_2 = T_1 \cup \{j\}$.

Definition 5.5. *We define the relation 'is derived from' as the reflexive and transitive closure of the relation 'is directly derived from'. That is to say, (P_2, S_2, T_2, Q_2) is derived from (P_1, S_1, T_1, Q_1), and we denote $(P_1, S_1, T_1, Q_1) \rightsquigarrow (P_2, S_2, T_2, Q_2)$, if and only if one of these conditions holds:*

i) $P_1 = P_2$*;* $S_1 = S_2$*;* $T_1 = T_2$*; and* $Q_1 = Q_2$*;*

ii) (P_2, S_2, T_2, Q_2) *is directly derived from* (P_1, S_1, T_1, Q_1)*,*

iii) There is an extended interlocking problem (P_3, S_3, T_3, Q_3) such that it is directly derived from (P_1, S_1, T_1, Q_1) and $(P_3, S_3, T_3, Q_3) \rightsquigarrow (P_2, S_2, T_2, Q_2)$.

Remark 5.3. *It is important to note that the relation $\rightsquigarrow$ is not symmetric. In other words, while it is possible for $(P_1, S_1, T_1, Q_1) \rightsquigarrow (P_2, S_2, T_2, Q_2)$ to hold true, the reverse, $(P_2, S_2, T_2, Q_2) \not\rightsquigarrow (P_1, S_1, T_1, Q_1)$, may also be true.*

5.3. *Some insights about the concept of extended interlocking problems and the 'Derived From' relation*

Although an extended interlocking problem is an abstract concept, it is indeed the cornerstone of our approach. We aim to provide some intuition about this abstract concept. As per Remark 5.2, an extended interlocking problem (P, S, T, Q) is associated with multiple interlocking problems, specifically one for each $t \in T$. An extended interlocking problem (P, S, T, Q) is in a dangerous situation if and only if there exists a $t \in T$ such that the interlocking problem $(P, S \cup \{t\} \cup Q)$ is also in a dangerous situation. Consequently, the extended interlocking problem (P, S, T, Q) provides insights into the solutions of numerous simple interlocking problems, one for each $t \in T$. If the extended interlocking problem (P, S, T, Q) is in a safe situation, then we can infer that, for every $t \in T$, the interlocking problem $(P, S \cup Q \cup \{t\})$ is also in a safe situation. Conversely, if (P, S, T, Q) is in a dangerous situation, it implies that there exists a $t \in T$ such that $(P, S \cup \{t\} \cup Q)$ is in a dangerous situation.

In this way, the multisets S, T, and Q represent sections of the railway station where trains are, or may potentially be, located:

- For each $s \in S$, it is assumed that a train is positioned on the section s. Additionally, we know that section a is locked.
- For each $q \in Q$, it is assumed that a train is positioned on section q.
- It is known that there is precisely one train positioned in one of the sections in T.

In Figure 12, we elucidate this concept by illustrating the extended interlocking problem $(P, S, T, Q) = (P, \{4\}, \{6, 7, 8\}, \{1, 10\})$ for the railway station shown in Figure 8, along with its associated simple interlocking problems. The components of this problem are defined as follows:

- $P = \{(1, 2), (9, 10), (10, 11), (2, 3), (3, 4), (11, 6), (6, 7), (7, 8), (2, 1), (11, 10), (5, 4), (3, 2), (6, 11), (8, 7), (5, 6)\}$
- $S = \{4\}$, represented by red rectangles. As can be observed, section S4 is a locked section, implying that a train cannot move from S4 to another section.
- $T = \{6, 7, 8\}$, represented by yellow rectangles in sections S6, S7, and S8.
- $Q = \{1, 10\}$, represented by black rectangles in sections S1 and S10.

Given that $T = \{6, 7, 8\}$, the extended interlocking problem $(P, \{4\}, \{6, 7, 8\}, \{1, 10\})$ is associated with the following simple interlocking problems: $(P, \{4, 8, 1, 10\})$, $(P, \{4, 7, 1, 10\})$, and $(P, \{4, 6, 1, 10\})$, as illustrated in Figure 13. Since the simple interlocking problem $(P, \{4, 6, 1, 10\})$ is in a dangerous situation, it follows that the extended interlocking problem $(P, \{4\}, \{6, 7, 8\}, \{1, 10\})$ is also in a dangerous situation.

In previous section we introduced a 'derived from' relation between extended interlocking problems. This relation plays a pivotal role in determining whether the extended interlocking problem (P, S, T, Q) is in a dangerous situation. As we will see in Proposition 5.4, if $(P, S, T, Q) \rightsquigarrow (P', S', T', Q')$, then (P, S, T, Q) is in a dangerous situation if and only if (P', S', T', Q') is in a dangerous situation.

In Figure 13, we illustrate this 'derived from' relation for the extended interlocking problem $(P, \emptyset, \emptyset, \{1, 4, 6, 8\})$, which is equivalent to the interlocking problem $(P, \{1, 4, 6, 8\})$.

- $(P, \emptyset, \emptyset, \{1, 4, 6, 8\}) \rightsquigarrow (P, \emptyset, \{8\}, \{1, 4, 6\})$ by i) in Definition 5.4. Just observe that $(P, \emptyset, \{8\}, \{1, 4, 6\}) \not\rightsquigarrow (P, \emptyset, \emptyset, \{1, 4, 6, 8\})$, as this relation is not symmetric (see Remark 5.3). As we will see in the next section, $(P, \emptyset, \emptyset, \{1, 4, 6, 8\})$ is in a dangerous situation if and only if $(P, \emptyset, \{8\}, \{1, 4, 6\})$ is.
- $(P, \emptyset, \{8\}, \{1, 4, 6\}) \rightsquigarrow (P_2, \emptyset, \{8, 7\}, \{1, 4, 6\})$ by iv) in Definition 5.4 where $P_2 = P - \{(8, 7)\}$. As we will see in the next section, $(P, \emptyset, \{8\}, \{1, 4, 6\})$ is in a dangerous situation if and only if $(P_2, \emptyset, \{8, 7\}, \{1, 4, 6\})$ is.

- $(P_2, \emptyset, \{8,7\}, \{1,4,6\}) \rightsquigarrow (P_3, \emptyset, \{8,8,7\}, \{1,4,6\})$ by iv) in Definition 5.4 where $P_3 = P_2 - \{(7,8)\}$. As we will see in the next section, $(P_2, \emptyset, \{8,7\}, \{1,4,6\})$ is in a dangerous situation if and only if $(P_3, \emptyset, \{8,8,7\}, \{1,4,6\})$ is.
- $(P_3, \emptyset, \{8,8,7\}, \{1,4,6\}) \rightsquigarrow (P_3, \emptyset, \{8,7\}, \{1,4,6\})$ by ii) in Definition 5.4. As we will see in the next section, $(P_3, \emptyset, \{8,8,7\}, \{1,4,6\})$ is in a dangerous situation if and only if $(P_3, \emptyset, \{8,7\}, \{1,4,6\})$ is.
- $(P_3, \emptyset, \{8,7\}, \{1,4,6\}) \rightsquigarrow (P_3, \{8,7\}, \emptyset, \{1,4,6\})$ by iii) in Definition 5.4 because{8,7} does not contain repeated elements and sections S8 and S7 are locked. As we will see in the next section, $(P_3, \emptyset, \{8,7\}, \{1,4,6\})$ is in a dangerous situation if and only if $(P_3, \{8,7\}, \emptyset, \{1,4,6\})$ is.
- $(P_3, \{8,7\}, \emptyset, \{1,4,6\}) \rightsquigarrow (P_3, \{8,7\}, \{6\}, \{1,4\})$ by i) in Definition 5.4. As we will see in the next section, $(P_3, \{8,7\}, \emptyset, \{1,4,6\})$ is in a dangerous situation if and only if $(P_3, \{8,7\}, \{6\}, \{1,4\})$ is.
- $(P_3, \{8,7\}, \{6\}, \{1,4\}) \rightsquigarrow (P_4, \{8,7\}, \{6,7\}, \{1,4\})$ by iv) in Definition 5.4 where $P_4 = P_3 - \{(6,7)\}$. As we will see in the next section, $(P_3, \{8,7\}, \{6\}, \{1,4\})$ is in a dangerous situation if and only if $(P_4, \{8,7\}, \{6,7\}, \{1,4\})$ is.
- $(P_4, \{8,7\}, \{6,7\}, \{1,4\}) \rightsquigarrow (P_4, \{8,7,6,7\}, \emptyset, \{1,4\})$ by iii) in Definition 5.4 because {6,7} does not contain repeated elements and sections S6 and S7 are locked. As we will see in the next section, $(P_4, \{8,7\}, \{6,7\}, \{1,4\})$ is in a dangerous situation if and only if $(P_4, \{8,7,6,7\}, \emptyset, \{1,4\})$ is.

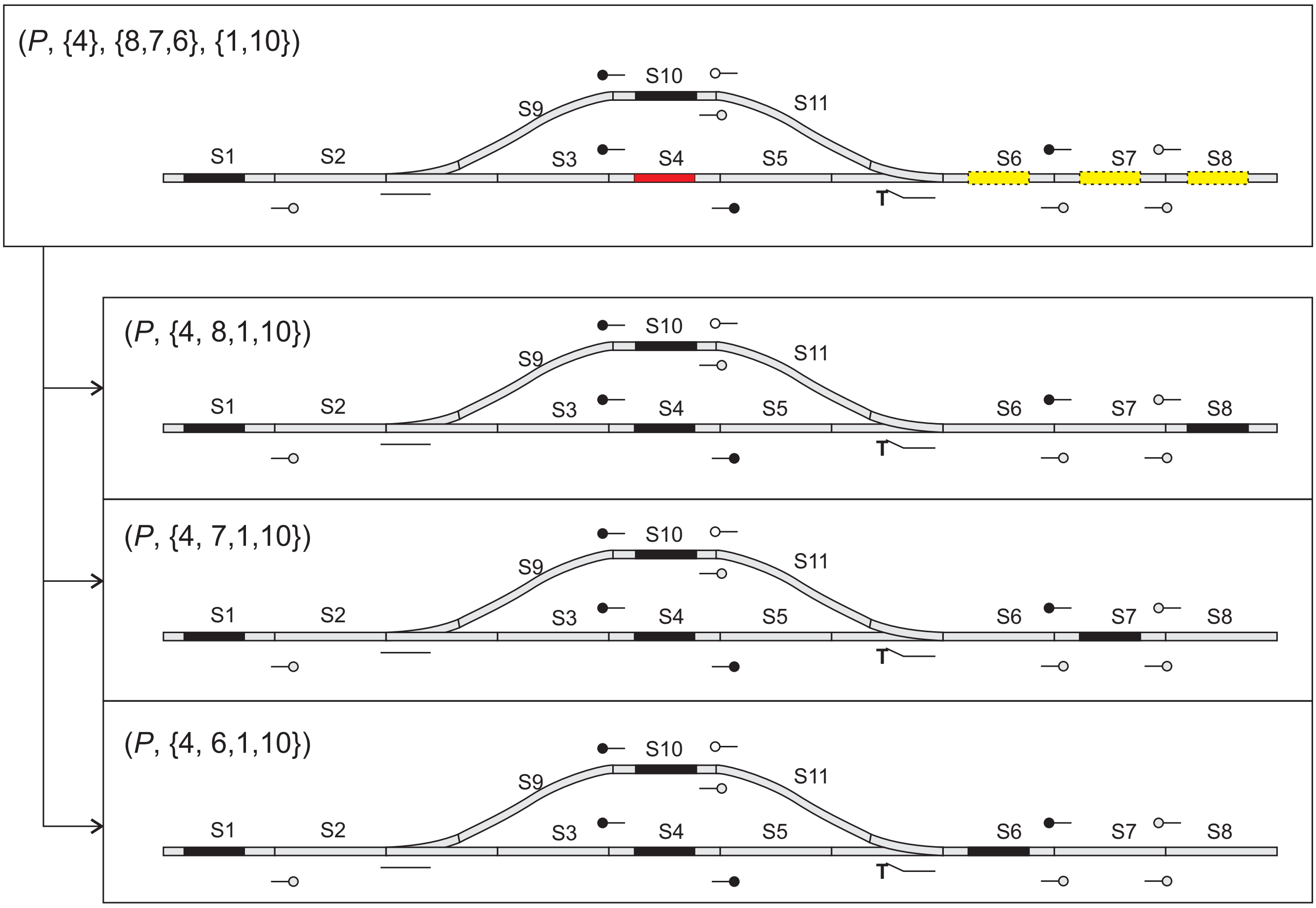

Figure 12. Intuition of the concept extension interlocking problems.

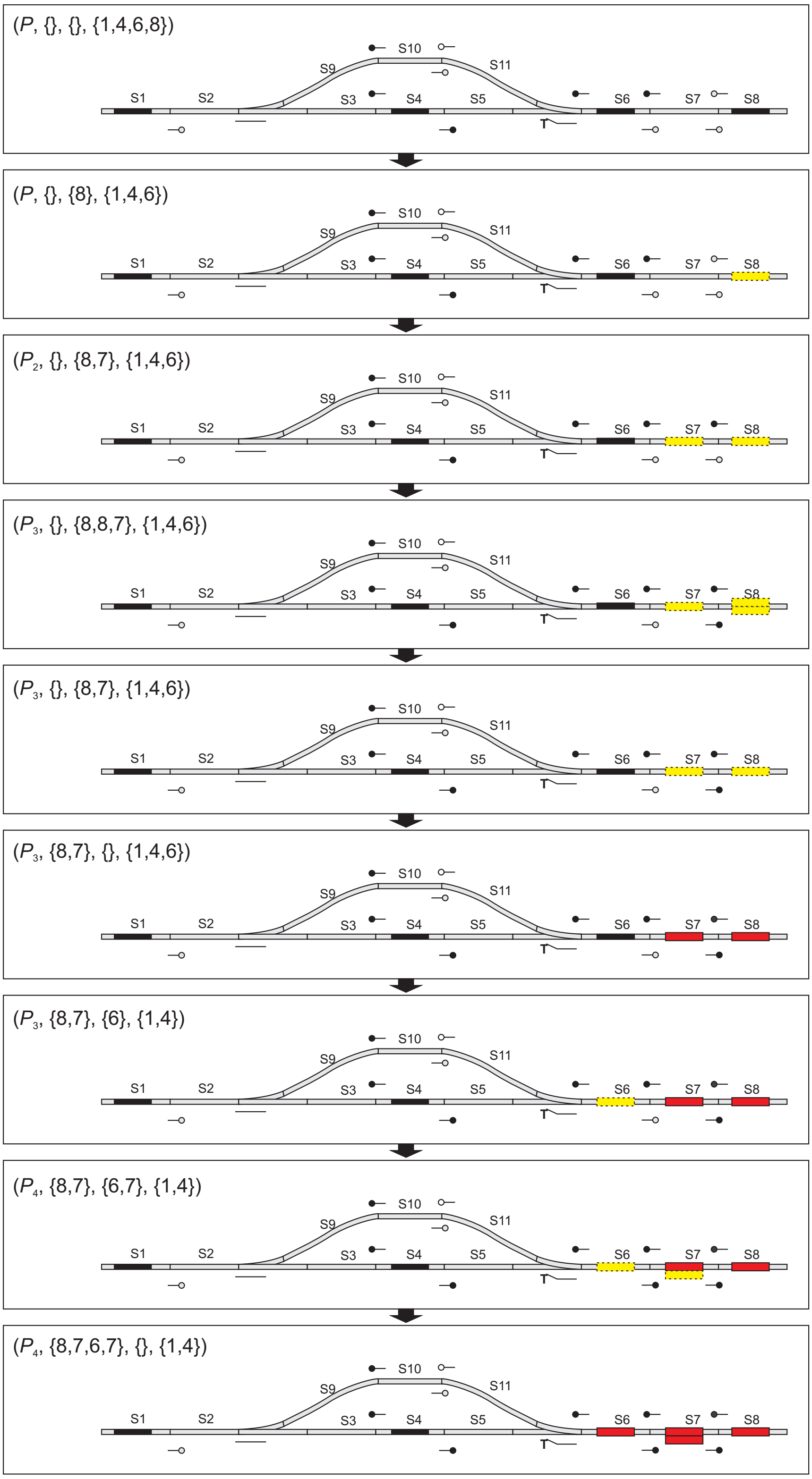

Figure 13. Intuition of the relation 'derived from' and our approach.

In this way, the extended interlocking problem $(P, \emptyset, \emptyset, \{1, 4, 6, 8\})$, or, equivalently, the simple interlocking problem $(P, \{1, 4, 6, 8\})$, is in a dangerous situation if and only if $(P_4, \{8, 7, 6, 7\}, \emptyset, \{1, 4\})$ is in a dangerous situation. The latter is indeed in a dangerous situation due to the repetition of 7 in the set {8,7,6,7}, as per Proposition 5.1.

5.4. Determining if an extended interlocking problem is in a dangerous situation

Under some conditions, we can easily determine if an extended interlocking problem is in a dangerous situation (see Proposition 5.1 and Proposition 5.2). Besides, Proposition 5.4 relates the problem of determining if an extended interlocking problem is in a dangerous situation with another one.

First, we will prove Proposition 5.1 and Proposition 5.2.

Proposition 5.1. *Let (P, S, T, Q) be an extended interlocking problem. If $S \cup Q$ contains repeated elements, (P, S, T, Q) is in a dangerous situation.*

Proof.- Let $\{u, v\} \subseteq S \cup Q$ such that $u = v$ (that is to say, u is repeated in $S \cup Q$). Let $[u]$ and $[v]$ be two lists of integers, each containing a single integer. We have that $[u]$ and $[v]$ fulfill conditions in Definition 5.3. ■

Proposition 5.2. *Let $(P, S, \emptyset, \emptyset)$ an extended interlocking problem. We have that $(P, S, \emptyset, \emptyset)$ is in a dangerous situation if and only if S contains repeated elements.*

Proof.-
⇒) Let $[u_1, \dots u_n]$ and $[v_1 \dots v_m]$ be paths fulfilling conditions in Definition 5.3. By ii) in Definition 5.3, we have that $\{u_1, v_1\} \subseteq S \cup T \cup Q = S$. Therefore, we have that u_1 and v_1 are locked sections in P. Consequently, by condition (2) and (3) in Definition 5.3, we have that $n = 1$ and $m = 1$. By (4) in Definition 5.3, we have that $u_1 = u_n = v_m = v_1$ and, therefore, S contains repeated elements.
⇐) This is an immediate consequence of Proposition 5.1. ■

Next, we will show an important proposition about the relation *'derived from'* between extended interlocking problems.

Proposition 5.3. *Let $(P_1, S_1, T_1, Q_1), (P_2, S_2, T_2, Q_2)$ be two extended interlocking problems such that (P_2, S_2, T_2, Q_2) is directly derived from (P_1, S_1, T_1, Q_1). We have that:*

(P_1, S_1, T_1, Q_1) *is in a dangerous situation* $\Leftrightarrow$ (P_2, S_2, T_2, Q_2) *is in a dangerous situation.*

Proof.- We will consider the four conditions in Definition 5.4.

Condition i). We will consider that $T_1 = \emptyset$; $T_2 = \{t\}$ where $t \in Q_1$; $S_2 = S_1$ and $Q_2 = Q_1 - \{t\}$.

⇒) Let $[u_1 \dots u_n]$ and $[v_1, \dots v_m]$ be paths fulfilling Definition 5.3 for (P_1, S_1, T_1, Q_1). We will consider the following cases:

Case $u_1 = t = v_1$**.** We will prove that $[t]$ and $[t]$ are paths fulfilling Definition 5.3 for (P_2, S_2, T_2, Q_2).

Conditions (1), (2). They hold because $P_1 = P_2$.

Condition (3). We have that $\{t, t\} \subseteq S_1 \cup T_1 \cup Q_1 = S_2 \cup T_2 \cup Q_2$. Since $T_2 = \{t\}$, we have that $\{t, t\} \subseteq S_2 \cup \{t\} \cup Q_2$. Consequently, we have that $t \in S_2 \cup Q_2$.

Conditions (4), (6), (7), (8). They are immediately proved.

Condition (5). This is immediately proved since $S_2 \cup Q_2 \subseteq S_1 \cup Q_1$.

Case $u_1 \neq t$. We have that $[u_1 \dots u_n]$ and $[v_1, \dots v_m]$ are paths fulfilling Definition 5.3 for (P_2, S_2, T_2, Q_2).

Conditions (1), (2). They hold because $P_1 = P_2$.

Condition (3). We have that $\{u_1, v_1\} \subseteq S_1 \cup T_1 \cup Q_1 = S_2 \cup T_2 \cup Q_2$. Since $u_1 \in S_2 \cup T_2 \cup Q_2 = S_2 \cup \{t\} \cup Q_2$ and $u_1 \neq t$, we have that $u_1 = S_2 \cup Q_2$.

Conditions (4), (6), (7), (8). They are immediately proved.

Condition (5). This is immediately proved since $S_2 \cup Q_2 \subseteq S_1 \cup Q_1$.

Case $v_1 \neq t$. We will prove that $[v_1, \dots v_m]$ and $[u_1 \dots u_n]$ are paths fulfilling Definition 5.3 for (P_2, S_2, T_2, Q_2).

Conditions (1), (2). They hold because $P_1 = P_2$.

Condition (3). We have that $\{u_1, v_1\} \subseteq S_1 \cup T_1 \cup Q_1 = S_2 \cup T_2 \cup Q_2$. Since $v_1 \in S_2 \cup T_2 \cup Q_2 = S_2 \cup \{t\} \cup Q_2$ and $v_1 \neq t$, we have that $v_1 = S_2 \cup Q_2$.

Conditions (4), (6), (7), (8). They are immediately proved.

Condition (5). This is immediately proved since $S_2 \cup Q_2 \subseteq S_1 \cup Q_1$.

$\Leftarrow$) Let $[u_1 \dots u_n]$ and $[v_1, \dots v_m]$ be paths fulfilling Definition 5.3 for (P_2, S_2, T_2, Q_2). We will consider the following cases:

Case. There exists i such that $u_i = t$. We have that $[u_1 \dots u_{i-1}, u_i]$ and $[t]$ are paths fulfilling Definition 5.3 for (P_1, S_1, T_1, Q_1):

Conditions (1), (2). They hold because $P_1 = P_2$.

Condition (3). We have that $\{u_1, v_1\} \subseteq S_1 \cup T_1 \cup Q_1 = S_2 \cup T_2 \cup Q_2$. We have also that $u_1 \in S_2 \cup Q_2 \subseteq S_1 \cup Q_1$.

Conditions (4), (6), (7), (8). They are immediately proved.

Condition (5). Let j be such that $1 < j < i$. We have that $u_j \in S_2 \cup Q_2 \cup \{t\}$. By condition (6) we have that $u_j \neq u_i = t$. Therefore, we have that $u_j \in S_2 \cup Q_2$.

Case. There exists i such that $v_i = t$. We have that $[t]$ and $[v_1 \dots v_{i-1}, v_i]$ are paths fulfilling Definition 5.3 for (P_1, S_1, T_1, Q_1):

Conditions (1), (2). They hold because $P_1 = P_2$.

Condition (3). We have that $\{u_1, v_1\} \subseteq S_1 \cup T_1 \cup Q_1 = S_2 \cup T_2 \cup Q_2$. We have also that $u_1 \in S_2 \cup Q_2 \subseteq S_1 \cup Q_1$.

Conditions (4), (6), (7), (8). They are immediately proved.

Condition (5). Let j be such that $1 < j < i$. By condition (5) we have that $v_j \in S_2 \cup Q_2 \cup \{t\}$. By condition (6) we have that $v_j \neq v_i = t$. Therefore, we have that $v_j \in S_2 \cup Q_2$.

Case. For all i we have that $u_i \neq t$ and $v_i \neq t$. We can prove that $[u_1 \dots u_n]$ and $[v_1, \dots v_m]$ are paths fulfilling Definition 5.3 for (P_1, S_1, T_1, Q_1):

Conditions (1), (2). They hold because $P_1 = P_2$.

Condition (3). We have that $\{u_1, v_1\} \subseteq S_1 \cup T_1 \cup Q_1 = S_2 \cup T_2 \cup Q_2$. We have also that $u_1 \in S_2 \cup Q_2 \subseteq S_1 \cup Q_1$.

Conditions (4), (6), (7), (8). They are immediately proved.

Condition (5). We have that $u_i \notin S_2 \cup Q_2$ and $v_i \notin S_2 \cup Q_2$. Since $u_i \neq t$ and $v_i \neq t$, we have that $u_i \notin S_2 \cup Q_2 \cup \{t\} = S_1 \cup Q_1$ and $v_i \notin S_2 \cup Q_2 \cup \{t\} = S_1 \cup Q_1$.

Condition ii). We will consider that $T_1 = T \cup \{t, t\}$, $T_2 = T \cup \{t\}$, $P_2 = P_1$, $S_2 = S_1$, $Q_2 = Q_1$.

It is immediate to prove that $[u_1 \dots u_n]$ and $[v_1, \dots v_m]$ are paths fulfilling Definition 5.3 for (P_1, S_1, T_1, Q_1) if and only if $[u_1 \dots u_n]$ and $[v_1, \dots v_m]$ fulfill Definition 5.3 for (P_2, S_2, T_2, Q_2).

Condition iii). We will consider that t is not repeated in T_1; all sections in T_1 are locked in P_1; $P_2 = P_1$; $T_2 = T_1 - \{t\}$; $S_2 = S_1 \cup \{t\}$; $Q_2 = Q_1$.

We have that $S_2 \cup T_2 \cup Q_2 = S_1 \cup \{t\} \cup T_2 \cup Q_1 = S_1 \cup T_1 \cup Q_1$.

⇒) Let $[u_1 \dots u_n]$ and $[v_1, \dots v_m]$ be paths fulfilling Definition 5.3 for (P_1, S_1, T_1, Q_1). We will prove that $[u_1 \dots u_n]$ and $[v_1, \dots v_m]$ fulfill Definition 5.3 for (P_2, S_2, T_2, Q_2).

Conditions (1), (2). They hold because $P_1 = P_2$.

Condition (3). We have that $\{u_1, v_1\} \subseteq S_1 \cup T_1 \cup Q_1 = S_2 \cup T_2 \cup Q_2$ and $u_1 \in S_1 \cup Q_1 \subseteq S_2 \cup Q_2$.

Conditions (4), (6), (7), (8). They are immediately proved.

Condition (5). Let i be such that $1 < i < n$. We have that $u_i \notin S_1 \cup Q_1$. Since t is a locked section in P_1, we have that $u_i \neq t$. Consequently, we have that $u_i \notin S_1 \cup Q_1 \cup \{t\} = S_2 \cup Q_2$. Consequently, for all $1 < i < n$ we have that $u_i \notin S_2 \cup Q_2$. In the same way, we can prove that for all $1 < j < m$, $v_j \notin S_2 \cup Q_2$.

⇐) Let $[u_1, \dots u_n]$ and $[v_1 \dots v_m]$ be paths fulfilling Definition 5.3 for (P_2, S_2, T_2, Q_2).

We will consider two possible cases:

Case $u_1 = t$. We will prove that $[v_1 \dots v_m]$ and $[t]$ are paths fulfilling Definition 5.3 for (P_1, S_1, T_1, Q_1).

Conditions (1), (2). They hold because $P_1 = P_2$.

Condition (5). Immediately proved since $S_1 \cup Q_1 \subset S_2 \cup Q_2$

Conditions (4), (6), (7), (8). They are immediately proved.

Condition (3). We have that $\{v_1, t\} = \{v_1, u_1\} \subseteq S_2 \cup T_2 \cup Q_2 = S_1 \cup T_1 \cup Q_1$. We will prove that $v_1 \in S_1 \cup Q_1$ by considering the following subcases:

Case $v_1 \notin T_1$. Since $v_1 \in S_1 \cup T_1 \cup Q_1$ and $v_1 \notin T_1$, we have that $v_1 \in S_1 \cup Q_1$.

Case $v_1 \in T_1$. Since T_1 are locked sections in $P_2 = P_1$ and $[t]$ and $[v_1 \dots v_m]$ are paths fulfilling Definition 5.3 for (P_2, S_2, T_2, Q_2), we have that $m = 1$, $v_1 = t$, and, by condition (3), $\{t, t\} \subseteq S_2 \cup T_2 \cup Q_2 = S_1 \cup T_1 \cup Q_1$. Since t is not repeated in T_1, we have that $v_1 = t \in S_1 \cup Q_1$.

Case $u_1 \neq t$. We will prove that $[u_1, \dots u_n]$ and $[v_1, \dots v_m]$ are paths fulfilling Definition 5.3 for (P_1, S_1, T_1, Q_1)

Conditions (1), (2). They hold because $P_1 = P_2$.

Condition (5). Immediately proved since $S_1 \cup Q_1 \subset S_2 \cup Q_2$

Conditions (4), (6), (7), (8). They are immediately proved.

Condition (3). We have that $\{u_1, v_1\} \subseteq S_2 \cup T_2 \cup Q_2 = S_1 \cup T_1 \cup Q_1$. Since $u_1 \in S_2 \cup Q_2 = S_1 \cup \{t\} \cup Q_1$ and $u_1 \neq t$, we have that $u_1 \in S_1 \cup Q_1$.

Condition iv). We will consider that $S_1 = S_2$; $Q_1 = Q_2$; $(i, j) \in P_1$; $i \in T_1$; $P_2 = P_1 - \{(i, j)\}$; $T_2 = T_1 \cup \{j\}$. We will consider the following cases:

⇒) Let $[u_1, \ldots u_n]$ and $[v_1 \ldots v_m]$ be two paths fulfilling Definition 5.3 for (P_1, S_1, T_1, Q_1).

Case 1. $\exists k$ where $1 \leq k \leq n$ such that $(u_k, u_{k+1}) = (i, j)$. We will prove that $[u_1, \ldots, u_k]$ and $[u_k]$ fulfill all conditions in Definition 5.3 for (P_2, S_2, T_2, Q_2).

Condition (1) Let $k' < k$. We have that $u_{k'} \neq u_k = i$. Consequently, $(u_{k'}, u_{k'+1}) \neq (i, j)$ and we have that $(u_{k'}, u_{k'+1}) \in P_2$.

Condition (3). We have that $u_1 \in S_1 \cup Q_1 = S_2 \cup Q_2$. We have that $u_k = i \in T_2$. Since $u_1 \neq u_k$, we have that $\{u_1, u_k\} \subseteq S_2 \cup T_2 \cup Q_2$.

Condition (5). Immediately proved since $S_1 \cup Q_1 = S_2 \cup Q_2$.

Conditions (2), (4), (6), (7), (8). They are immediately proved.

Case 2. Case 1 does not hold and $\exists k'$ where $1 \leq k' \leq m$ such that $(v_{k'}, v_{k'+1}) = (i, j)$. We will prove that $[u_1, \ldots, u_n]$ and $[v_{k'+1} \ldots v_m]$ fulfill all conditions in Definition 5.3 for (P_2, S_2, T_2, Q_2).

Condition (1). Immediately proved since Case 1 does not hold.

Condition (2). Immediately proved since $(v_{k'}, v_{k'+1}) = (i, j)$, and for every $k'' > k'$ we have that $v_{k''} \neq v_{k'} = i$, and consequently $(v_{k''}, v_{k''+1}) \in P_2$.

Condition (3). We have that $u_1 \in S_1 \cup Q_1 = S_2 \cup Q_2$. We have also that $v_{k'+1} = j \in T_2$. Since $u_1 \neq v_{k'+1}$, we have that $\{u_1, v_{k'+1}\} \cup S_2 \cup T_2 \cup Q_2$.

Condition (5). Immediately proved since $S_1 \cup Q_1 = S_2 \cup Q_2$.

Conditions (4), (6), (7), (8). They are immediately proved.

Case 3. Neither Case 1 nor Case 2 holds. We will prove that $[u_1 \ldots u_n]$ and $[v_1 \ldots v_m]$ fulfill all conditions in Definition 5.3 for (P_2, S_2, T_2, Q_2).

Conditions (1), (2). Immediately proved since neither Case 1 nor Case 2 holds.

Condition (3). We have that $u_1 \in S_1 \cup Q_1 = S_2 \cup Q_2$. We have that $v_1 \in S_1 \cup T_1 \cup Q_1 \subset S_2 \cup T_2 \cup Q_2$. Since $u_1 \neq v_1$ we have that $\{u_1, v_1\} \cup S_2 \cup T_2 \cup Q_2$.

Condition (5). Immediately proved since $S_1 \cup Q_1 = S_2 \cup Q_2$.

Conditions (4), (6), (7), (8). They are immediately proved.

⇐) Let $[u_1 \ldots u_n]$ and $[v_1 \ldots v_m]$ be paths fulfilling Definition 5.3 for (P_2, S_2, T_2, Q_2).

We have that $u_1 \in S_2 \cup Q_2 = S_1 \cup Q_1$. We will consider the following cases:

Case 1. $v_1 \notin T_2$**.** We will prove that $[u_1 \ldots u_n]$ and $[v_1 \ldots v_m]$ are paths fulfilling Definition 5.3 for (P_1, S_1, T_1, Q_1).

Conditions (1), (2). Immediately proved since $P_2 \subset P_1$.

Condition (3). We have that $u_1 \in S_2 \cup Q_2 = S_1 \cup Q_1$.
Since $v_1 \notin T_2$, we have that $v_1 \in S_2 \cup Q_2 = S_1 \cup Q_1$.
Since $u_1 \neq v_1$, we have that $\{u_1, v_1\} \subseteq S_1 \cup Q_1 \subseteq S_1 \cup T_1 \cup Q_1$.

Condition (5). Immediately proved since $S_1 \cup Q_1 = S_2 \cup Q_2$.

Conditions (4),(6),(7),(8). They are immediately proved.

Case 2. $v_1 \in T_2$ **and** $v_1 \neq j$**.** We will prove that $[u_1, \ldots, u_n]$ and $[v_1, \ldots, v_m]$ are paths fulfilling Definition 5.3 for (P_1, S_1, T_1, Q_1).

Conditions (1), (2). Immediately proved since $P_2 \subset P_1$.

Condition (3). We have that $u_1 \in S_2 \cup Q_2 = S_1 \cup Q_1$.
Since $v_1 \in T_2 = T_1 \cup \{j\}$ and $v_1 \neq j$, we have that $v_1 \in T_1 \subseteq S_1 \cup T_1 \cup Q_1$.
Since $u_1 \neq v_1$, we have that $\{u_1, v_1\} \subseteq S_1 \cup T_1 \cup Q_1$.

Condition (5). Immediately proved since $S_1 \cup Q_1 = S_2 \cup Q_2$.

Conditions (4), (6), (7), (8). They are immediately proved.

Case 3. $v_1 \in T_2$ **and** $v_1 = j$**.** We will prove that $[u_1, \ldots, u_n]$ and $[i, v_1, \ldots, v_m]$ are paths fulfilling Definition 5.3 for (P_1, S_1, T_1, Q_1).

Condition (1). Immediately proved since $P_2 \subset P_1$.

Condition (2). Immediately proved since $P_2 \subset P_1$ and $(i, j) \in P_1$.

Condition (3). We have that $u_1 \in S_2 \cup Q_2 = S_1 \cup Q_1$.
Since $i \in T_1$, we have that $i \in S_1 \cup T_1 \cup Q_1$.
Since $i \in T_1$ and $u_1 \in S_1 \cup Q_1$, we have that $\{u_1, i\} \subseteq S_1 \cup T_1 \cup Q_1$.

Condition (5). Immediately proved since $S_1 \cup Q_1 = S_2 \cup Q_2$.

Conditions (4), (6), (7), (8). They are immediately proved.

■

Proposition 5.4. *Let* $(P_1, S_1, T_1, Q_1), (P_2, S_2, T_2, Q_2)$ *be two extended interlocking problems such that* $(P_1, S_1, T_1, Q_1) \rightsquigarrow (P_2, S_2, T_2, Q_2)$*. We have that:*

$$(P_1, S_1, T_1, Q_1) \text{ is in a dangerous situation} \Leftrightarrow (P_2, S_2, T_2, Q_2) \text{ is in a dangerous situation.}$$

Proof.- This is an immediate consequence of Definition 5.5 and Proposition 5.3. ■

6. The railway interlocking problem in algebraic terms

Drawing upon the findings presented in Section 5, we could devise an algorithm to detect dangerous situations in interlocking problems. This algorithm would operate by iteratively transforming a given interlocking problem into derived extended interlocking problems until it can definitively be verified whether a dangerous situation exists. Indeed, an algorithm of linear complexity with respect to the number of sections is detailed in the appendix.

In this section we will demonstrate that the task of detecting dangerous situations can be expressed in algebraic terms. As it will be discussed in Theorem 6.3, the polynomial division algorithm can serve as the mechanism for identifying dangerous situations. Here, we will outline this polynomial-based representation and subsequently establish that the task of identifying dangerous situations is equivalent to the algebraic problem of computing the remainder of a monomial upon division by a set of polynomials.

6.1. Algebraic Preliminaries. Some introductory notes about polynomial rings

In this section, we will briefly describe some essential results about polynomial rings, which will be used to prove the validity of our approach. For a detailed introduction to the topic, see [17].

We will consider the polynomial ring $\mathbb{Z}_2[x_1 \ldots x_N]$ in the variables $x_1 \ldots x_N$. Since the coefficients of the polynomials lie in $\mathbb{Z}_2$, we have that monomials are a product of the form $x_1^{\alpha_1} \cdot x_2^{\alpha_2} \cdot \ldots x_N^{\alpha_N}$, and a polynomial is a finite sum of monomials $\sum_{\alpha=(\alpha_1,\alpha_2,\ldots,\alpha_N)} x_1^{\alpha_1} x_2^{\alpha_2} \ldots x_n^{\alpha_N}$.

For example, we have that $x_1x_2x_3$ and $x_1x_2+x_3+x_2x_3$ are, respectively, a monomial and a polynomial in $\mathbb{Z}_2[x_1, x_2, x_3]$.

If we consider the order of variables $x_1 > x_2 \ldots > x_m$, we can define a lexicographical order for the monomials on $\mathbb{Z}_2[x_1 \ldots x_N]$ (see [17], pp. 52–57). According to this lexicographical order, $x_1x_2x_4 > x_2x_3$.

Based on this lexicographical order, we define the leading term of a polynomial:

Definition 6.1. *The leading term of a polynomial $p \in \mathbb{Z}_2[x_1, \ldots, x_N]$, denoted* $\mathrm{LT}(p)$*, is its greatest monomial of the polynomial.*

For example, in $\mathbb{Z}_2[x_1, x_2, x_3]$, we have that $\mathrm{LT}(x_1^2 + x_1x_2x_3 + x_3) = x_1^2$.

This lexicographical order leads us to define a kind of division algorithm on polynomials in $\mathbb{Z}_2[x_1, \ldots, x_N]$.

Theorem 6.1. *Let $p \in \mathbb{Z}_2[x_1 \ldots x_N]$ and a finite list of polynomials on $E = [f_1, \ldots, f_m]$ (where $f_i \in \mathbb{Z}_2[x_1 \ldots x_N]$). We have that*

$$p = \alpha_1 f_1 + \ldots + \alpha_m f_m + r$$

where $\alpha_1, \ldots, \alpha_m, r \in \mathbb{Z}_2[x_1, \ldots, x_N]$ and either $r = 0$ or r is a linear combination of monomials, none of which is divisible by any of $\mathrm{LT}(f_1) \ldots \mathrm{LT}(f_m)$*.*
A proof can be found on pages 62–64 in [17].

The remainder of p on division by E is denoted by $\mathrm{NR}(p, E)$. The proof in [17] (pp. 62-64) provides an algorithm to find the polynomials $\alpha_1, \ldots \alpha_m, r \in \mathbb{Z}_2[x_1, \ldots, x_N]$ in Theorem 6.1. In each step of this algorithm, we obtain a sequence of intermediate-dividends. For our purpose, we will require only some of the first intermediate-dividends of this algorithm.

Notation 6.1. *We will use the notation $a|b$ to refer to the polynomial a divides the polynomial b. Similarly, we will say $a \nmid b$ to refer to a does not divide b.*

Definition 6.2. *Given $p \in \mathbb{Z}_2[x_1, \ldots, x_N]$ and a finite list of polynomials $E = [f_1 \ldots f_m]$, we recursively define the* first-intermediate-dividends *of p on division by E, $r_0, \ldots, r_n \in \mathbb{Z}_2[x_1, \ldots, x_N]$, as follows:*

- $r_0 = p$,
- *if $i > 0$, then*
  - *if $\exists j \in \{1, \ldots, m\}$ such that* $\mathrm{LT}(f_j)|\,\mathrm{LT}(r_i)$*, then*

$$r_{i+1} = r_i - \frac{\mathrm{LT}(r_i)}{\mathrm{LT}(f_k)} f_k$$

where k is the minimum j such that $\mathrm{LT}(f_j)|\,\mathrm{LT}(r_i)$*,*

– *if* $\forall j \in \{1,\dots,m\}, \mathrm{LT}(f_j) \nmid \mathrm{LT}(r_i)$ *then* $r_n = r_i$.

According to the proof in [17] (pp. 62-64), we have:

Proposition 6.1. *Let* $p \in \mathbb{Z}_2[x_1,\dots,x_N]$ *and let* $E = [f_1,\dots,f_m]$ *be a list of polynomials in* $\mathbb{Z}_2[x_1,\dots,x_N]$*. Let* $r_0,\dots r_n \in \mathbb{K}[x_1,\dots,x_N]$ *be the first-intermediate-dividends of* p *on division by* E*. We have that:*

$$r_n = 0 \textit{ if and only if } \mathrm{NR}(p, E) = 0$$

The Computer Algebra System, *CoCoA*, implements an algorithm for automatically calculating NR. For instance, if we want to calculate $\mathrm{NR}(x^2 + xyz, \{y^2 + z, x^2 + yz\})$ in the ring $\mathbb{Z}_2[x, y, z]$ with $x > y > z$, we only need to follow these steps:

First, we define the ring using the following syntax:

```
use ZZ/(2)[x,y,z], lex;
```

Next, input the following command:

```
NR(x^3+x^2*y*z,[x^2+y*z,y+z]);
```

The output will be:

```
x*z^2 + z^4
```

As you can see, we have:

$$x^3 + x^2yz = (x + yz)\cdot(x^2 + yz) + (xz + yz^2 + z^3)\cdot(y + z) + (xz^2 + z^4)$$

6.2. Representation of a railway station by means of a list of polynomials

We will represent a railway station by means of a list of polynomials on the following variables:

- l_{ij}: The variable $l_{i,j}$ for each pair of sections S_i and S_j where $(i, j) \in E$. That is to say, we define the variable $l_{i,j}$ if the topology of the station allows for passage from section S_i to section S_j (in some configuration of the railway station).
- q_i, t_i, s_i: The variables q_i, t_i, and s_i for each section S_i in the railway network ‡.

Once these variables are defined, we will represent the railway station by the list of polynomials $\mathcal{E}$ on

$$\mathcal{A} = \mathbb{Z}_2[l_{i,j},\dots,q_i,\dots,t_i,\dots,s_i\dots]$$

Observe that the number of variables in $\mathcal{A}$ is $3N + K$, where N is the number of sections and K is the size of E.

Definition 6.3. *We define* $\mathcal{E}$ *as the list of polynomials in* $\mathcal{A}$ *formed by:*

- $\forall(i, j) \in E$*, the polynomial:*

$$l_{ij}t_i + t_it_j$$

‡Intuitively, these variables refer to whether the section is in the multisets Q, T, or S.

- *For each section S_i, the four polynomials:*

$$t_i^2 + t_i$$

$$s_i^2$$

$$t_i + s_i$$

$$q_i + t_i$$

Observe that the number of polynomials in $\mathcal{E}$ is $4N + K$, where N is the number of sections and K is the size of E.

Example 6.1. *Let us recall the set E of the railway station of Example 4.1. Let us remember that in this case:*

$$E = \{(1,2),(2,9),(9,10),(10,11),(11,6),(2,3),(3,4),(4,5),(5,6),(6,7),(7,8)\\ (2,1),(9,2),(10,9),(11,10),(6,11),(3,2),(4,3),(5,4),(6,5),(7,6),(8,7)\}$$

Consequently, we will consider the following variables:

- *Variables:* $l_{1,2}, l_{2,9}, l_{9,10}, l_{10,11}, l_{11,6}, l_{2,3}, l_{3,4}, l_{4,5}, l_{5,6}, l_{6,7}, l_{7,8},$
  $l_{2,1}, l_{9,2}, l_{10,9}, l_{11,10}, l_{6,11}, l_{3,2}, l_{4,3}, l_{5,4}, l_{6,5}, l_{7,6}, l_{8,7}$
  and
- *Variables:* $q_1 \dots, q_{11}, t_1 \dots t_{11}, s_1, \dots, s_{11}$ *(since this railway station contains 11 sections)*

Once the variables are defined, we define the list $\mathcal{E}$ in this ring:

- $l_{1,2}t_1 + t_1t_2$ *(because $(1,2) \in E$),*
- $l_{2,9}t_2 + t_2t_9$ *(because $(2,9) \in E$),*
- $\dots$
- $t_1^2 + t_1, t_2^2 + t_2, \dots, t_{11}^2 + t_{11},$
- $s_1^2, s_2^2, \dots, s_{11}^2,$
- $t_1 + s_1, t_2 + s_2, \dots t_{11} + s_{11},$
- $q_1 + t_1, q_2 + t_2, \dots q_{11} + t_{11},$

6.3. Representation of an extended interlocking problem by means of monomials

In this section we will represent an extended interlocking problem (P, S, T, Q) by means of a monomial, $\phi(P, S, T, Q)$:

Definition 6.4. *Given the interlocking problem (P, S, T, Q) for a railway station, we define $\phi(P, S, T, Q)$ as the monomial in $\mathcal{A}$:*

$$\phi(P, S, T, Q) = \phi_P \cdot \phi_S \cdot \phi_T \cdot \phi_Q$$

where:

- *ϕ_P is a monomial representing the set P defined as:*

$$\phi_P = \prod_{(i,j)\in P} l_{ij}$$

- *ϕ_S is a monomial representing the set S defined as:*

$$\phi_S = \prod_{i \in S} s_i$$

- *ϕ_T is a monomial representing the set T defined as:*

$$\phi_T = \prod_{i \in T} t_i$$

- *ϕ_Q is a monomial representing the set Q defined as:*

$$\phi_Q = \prod_{i \in Q} q_i$$

Example 6.2. *We will consider the interlocking problem (P, Q) depicted in Figure 10.*

As we stated in Remark 5.1, an interlocking problem (P, S) can be understood as an extended interlocking problem $(P, \emptyset, \emptyset, Q)$. We will represent it by means of monomials:

The monomial ϕ_P. *Remember that P is (see Example 9):*

$$P = \{(1,2),(9,10),(10,11),(2,3),(3,4),(11,6),(6,7),(7,8),(2,1),(11,10),(5,4),(3,2),(6,11),(8,7),(5,6)\}$$

Consequently, we have that:

$$\phi_P = l_{1,2}l_{9,10}l_{10,11}l_{2,3}l_{3,4}l_{11,6}l_{6,7}l_{7,8}l_{2,1}l_{11,10}l_{5,4}l_{3,2}l_{6,11}l_{8,7}l_{5,6}$$

The monomial ϕ_S. *Since $S = \emptyset$, we have that:*

$$\phi_S = 1$$

The monomial ϕ_T. *Since $T = \emptyset$, we have that:*

$$\phi_T = 1$$

The monomial ϕ_Q. *Since $Q = \{1, 10\}$, we have that:*

$$\phi_Q = q_1 q_{10}$$

Consequently, the extended interlocking problem $(P, \emptyset, \emptyset, Q)$ depicted in Figure 10 is represented by the monomial:

$$\phi_P\phi_S\phi_T\phi_Q = \phi_P\phi_Q = l_{1,2}l_{9,10}l_{10,11}l_{11,6}l_{2,3}l_{3,4}l_{5,6}l_{6,7}l_{7,8}l_{2,1}l_{11,10}l_{6,11}l_{3,2}l_{5,4}l_{8,7} \cdot q_1 q_{10}$$

6.4. *Solving extended interlocking problems by means of the division algorithm*

In this section, we will show that we can determine if an extended interlocking problem is in a dangerous situation by means of the division algorithm.

According to the next lemma, we have that each first intermediate dividend of $\mathrm{NR}(\phi(P, S, T, Q), \mathcal{E})$ is a monomial representing an extended interlocking problem derived from (P, S, T, Q).

Lemma 6.2. *Let (P, S, T, Q) be an extended interlocking problem for a railway network E. Letting $r_0, r_1 \dots r_n$ be the first-intermediate-dividends of $\mathrm{NR}(\phi(P, S, T, Q), \mathcal{E})$, we have that:*

i) For every $i < n$ there is an extended interlocking problem (P_i, S_i, T_i, Q_i) such that $\phi(P_i, S_i, T_i, Q_i) = r_i$ and $(P, S, T, Q) \rightsquigarrow (P_i, S_i, T_i, Q_i)$.
ii) If $\mathrm{NR}(\phi(P, S, T, \emptyset), \mathcal{E}) \neq 0$, there is an extended interlocking problem $(P_n, S_n, \emptyset, \emptyset)$ such that $\phi(P_n, S_n, \emptyset, \emptyset) = r_n$, $(P, S, T, Q) \rightsquigarrow (P_n, S_n, \emptyset, \emptyset)$ and S_n does not contain repeated elements.

Proof.-

i) For $i = 0$, we have that $(P_0, S_0, T_0, Q_0) = (P, S, T, Q)$, $r_0 = \phi(P_0, S_0, T_0, Q_0) = \phi(P, S, T, Q)$, and $(P, S, T, Q) \rightsquigarrow (P, S, T, Q) = (P_0, S_0, T_0, Q_0)$.
Suppose that $0 < i < n$, $r_{i-1} = \phi(P_{i-1}, S_{i-1}, T_{i-1}, Q_{i-1})$ and $(P, S, T, Q) \rightsquigarrow (P_{i-1}, S_{i-1}, T_{i-1}, Q_{i-1})$. We will prove that $r_i = \phi(P_i, S_i, T_i, Q_i)$ for an extended interlocking problem (P_i, S_i, T_i, Q_i), such that (P_i, S_i, T_i, Q_i) is directly derived from $(P_{i-1}, S_{i-1}, T_{i-1}, Q_{i-1})$, and, consequently, we have that $(P, S, T, Q) \rightsquigarrow (P_i, S_i, T_i, Q_i)$.
According to Definition 6.2, we have that

$$r_i = r_{i-1} - \frac{\mathrm{LT}(r_{i-1})}{\mathrm{LT}(f_k)} f_k$$

where f_k is a polynomial in $\mathcal{E}$ such that $\mathrm{LT}(f_k) | \mathrm{LT}(r_{i-1})$. Since r_{i-1} is a monomial, we have that $\mathrm{LT}(r_{i-1}) = r_{i-1} = \phi(P_{i-1}, S_{i-1}, T_{i-1}, Q_{i-1}) = \phi_{P_{i-1}} \phi_{S_{i-1}} \phi_{T_{i-1}} \phi_{Q_{i-1}}$ where:

- $\phi_{P_{i-1}}$ only contains variables with the form l_{xy}.
- $\phi_{S_{i-1}}$ only contains variables with the form s_x.
- $\phi_{T_{i-1}}$ only contains variables with the form t_x.
- $\phi_{Q_{i-1}}$ only contains variables with the form q_x.

Besides, we have that $f_k \neq s_k^2$. Otherwise, we would have that f_k and r_{i-1} are monomials, and $\mathrm{LT}(f_k) = f_k$ and $\mathrm{LT}(r_{i-1}) = r_{i-1}$. Since $\mathrm{LT}(f_k) | \mathrm{LT}(r_{i-1})$ (this means in our case, $f_k | r_{i-1}$), we have that $r_i = r_{i-1} - \frac{r_{i-1}}{f_k} f_k = 0$, and therefore $i = n$ (but we have supposed that $i < n$).
Consequently, there are the following cases left:

Case $f_k = l_{xy} t_x + t_x t_y$**.** Since $\mathrm{LT}(f_k) | \mathrm{LT}(r_{i-1})$, we have that $l_{xy} t_x | \phi(P_{i-1}, S_{i-1}, T_{i-1}, Q_{i-1})$. Thus, we have that $l_{xy} | \phi_{P_{i-1}}$ and $t_x | \phi_{T_{i-1}}$. Consequently:

- $\phi_{P_{i-1}}$ is of the form $\phi_{P_{i-1}} = p \cdot l_{xy}$ where p is a monomial, and therefore, $(x, y) \in P_{i-1}$.
- $\phi_{T_{i-1}}$ is of the form $\phi_{T_{i-1}} = w \cdot t_x$ where w is a monomial, and therefore, $x \in T_{i-1}$.

We define:

- $\phi_{P_i} = p$ where $P_i = P_{i-1} - \{(x, y)\}$.
- $\phi_{T_i} = w \cdot t_x t_y$ where $T_i = T_{i-1} \cup \{y\}$.

Therefore, we have that:

$$r_i = r_{i-1} - \frac{\mathrm{LT}(r_{i-1})}{\mathrm{LT}(f_k)} f_k = p \cdot w \cdot t_x t_y \cdot \phi_{S_{i-1}} \phi_{Q_{i-1}} = \phi(P_i, S_{i-1}, T_i, Q_{i-1})$$

By iv) in Definition 5.4, $(P_i, S_{i-1}, T_i, Q_{i-1})$ is directly derived from $(P_{i-1}, S_{i-1}, T_{i-1}, Q_{i-1})$.

Case $f_k = t_x^2 + t_x$**.** Since $\mathrm{LT}(f_k) | \mathrm{LT}(r_{i-1})$ and $\mathrm{LT}(f_k) = t_x^2$, we have that $t_x^2 | \phi(P_{i-1}, S_{i-1}, T_{i-1}, Q_{i-1})$. Thus, we have that $t_x^2 | \phi_{T_{i-1}}$. Consequently:

- $\phi_{T_{i-1}}$ is of the form $\phi_{T_{i-1}} = w \cdot t_x^2$ where w is a monomial. We have also that $T_{i-1} = T \cup \{x, x\}$.

We define:

- $\phi_{T_i} = w \cdot t_x$ where $T_i = T \cup \{x\}$.

Therefore, we have that

$$r_i = r_{i-1} - \frac{\mathrm{LT}(r_{i-1})}{\mathrm{LT}(f_k)} f_k = w \cdot t_x \cdot \phi_{P_{i-1}} \phi_{S_{i-1}} \phi_{Q_{i-1}} = \phi(P_{i-1}, S_{i-1}, T_i, Q_{i-1})$$

By ii) in Definition 5.4, $(P_{i-1}, S_{i-1}, T_i, Q_{i-1})$ is directly derived from $(P_{i-1}, S_{i-1}, T_{i-1}, Q_{i-1})$.

Case $f_k = t_x + s_x$**.** Since $\mathrm{LT}(f_k) | \mathrm{LT}(r_{i-1})$ and $\mathrm{LT}(f_k) = t_x$, we have that $t_x | \phi(P_{i-1}, S_{i-1}, T_{i-1}, Q_{i-1})$. Thus, we have that $t_x | \phi_{T_{i-1}}$. Consequently:

- $\phi_{T_{i-1}}$ is of the form $\phi_{T_{i-1}} = w \cdot t_x$ where w is a monomial. We have also that $x \in T_{i-1}$.

The fact that f_k is $t_x + s_x$, instead of previous polynomials in $\mathcal{E}$ involves the following facts:

- Every section in T_{i-1} is locked in P_{i-1}. Otherwise, if $y \in T_{i-1}$ were not locked, we would have that there is a section z such that $(y, z) \in P_{i-1}$. Consequently, we have that $\mathrm{LT}(l_{y,z} t_y + t_y t_z) | r_{i-1}$. Since $l_{y,z} t_y + t_y t_z$ is in a lower position on the list $\mathcal{E}$ than the polynomial $t_x + s_x$, the polynomial f_k could not be $t_x + s_x$.
- The section x is not repeated in T_{i-1}. Otherwise, we would have that $\mathrm{LT}(t_x^2 + t_x) = t_x^2 | r_{i-1}$, and since $t_x^2 + t_x$ is in a lower position on the list $\mathcal{E}$ than the polynomial $t_x + s_x$, the polynomial f_k could not be $t_x + s_x$.

We define:

- $\phi_{T_i} = w$ where $T_i = T_{i-1} - \{x\}$.
- $\phi_{S_i} = \phi_{S_{i-1}} \cdot s_x$ where $S_i = S_{i-1} \cup \{x\}$.

Therefore, we have that:

$$r_i = r_{i-1} - \frac{\mathrm{LT}(r_{i-1})}{\mathrm{LT}(f_k)} f_k = \phi_{P_{i-1}} \phi_{S_{i-1}} \cdot s_x \cdot w \cdot \phi_{Q_{i-1}} = \phi(P_{i-1}, S_i, T_i, Q_{i-1})$$

By iii) in Definition 5.4, $(P_{i-1}, S_i, T_i, Q_{i-1})$ is directly derived from $(P_{i-1}, S_{i-1}, T_{i-1}, Q_{i-1})$.

Case $f_k = q_x + t_x$**.** Since $\mathrm{LT}(f_k) | \mathrm{LT}(r_{i-1})$, we have that $\mathrm{LT}(q_x + t_x) = q_x | \phi(P_{i-1}, S_{i-1}, T_{i-1}, Q_{i-1})$. Thus, we have that $q_x | \phi_{Q_{i-1}}$. Consequently:

- $\phi_{Q_{i-1}}$ is of the form $\phi_{Q_{i-1}} = w \cdot q_x$ where w is a monomial. We have also that $x \in Q_{i-1}$.

The fact that f_k is $q_x + t_x$ instead of previous polynomials in $\mathcal{E}$ involves the following fact:

- T_{i-1} is empty. Otherwise, if $y \in T$, we would have that $\mathrm{LT}(t_y + s_y)|r_{i-1}$, and since $t_y + s_y$ is in a lower position on the list $\mathcal{E}$ than $q_x + t_x$, f_k could not be $q_x + t_x$.

We define:

- $\phi_{T_i} = t_x$ where $T_i = \{x\}$.
- $\phi_{Q_i} = w$ where $Q_i = Q_{i-1} - \{x\}$.

Therefore, we have that:

$$r_i = r_{i-1} - \frac{\mathrm{LT}(r_{i-1})}{\mathrm{LT}(f_k)} f_k = \phi_{P_{i-1}} \phi_{S_{i-1}} \cdot x \cdot w = \phi(P_{i-1}, S_{i-1}, T_i, Q_i)$$

By i) in Definition 5.4, $(P_{i-1}, S_{i-1}, T_i, Q_i)$ is directly derived from $(P_{i-1}, S_{i-1}, T_{i-1}, Q_{i-1})$.

ii) Given $r_n \neq 0$, by applying the same line of reasoning we can establish the existence of (P_n, S_n, T_n, Q_n) such that $r_n = \phi(P_n, S_n, T_n, Q_n)$ and $(P, S, T, Q) \rightsquigarrow (P_n, S_n, T_n, Q_n)$. Besides, for every $f_k \in \mathcal{E}$, we have that $\mathrm{LT}(f_k) \not| r_n$. Consequently:

- $Q_n = \emptyset$. Otherwise, we would have that $\mathrm{LT}(q_x + t_x)|r_n$ for some section x
- $T_n = \emptyset$. Otherwise, if $y \in T_n$, we would have that $\mathrm{LT}(t_y + s_y)|r_n$.
- S_n does not have repeated elements. Otherwise, if x were repeated in S_n, we would have that $\mathrm{LT}(s_x^2)|r_n$.

■

Theorem 6.3. *Let (P, S, T, Q) be an extended interlocking problem for a railway network E. We have that:*

$$(P, S, T, Q) \textit{ is in a dangerous situation} \Leftrightarrow \mathrm{NR}(\phi(P, S, T, Q), \mathcal{E}) = 0$$

Proof.- Let $r_0, r_1 \dots r_n$ be the first intermediate remainders of $\mathrm{NR}(\phi(P, S, T, \emptyset), \mathcal{E})$.

⇒) We will prove that, if $\mathrm{NR}(\phi(P, S, T, Q), \mathcal{E}) \neq 0$, we have that (P, S, T, Q) is not in a dangerous situation.

Let us consider that $\mathrm{NR}(\phi(P, S, T, Q), \mathcal{E}) \neq 0$. By ii) in Lemma 6.2, we have that there is an extended interlocking problem $(P_n, S_n, \emptyset, \emptyset)$ such that $\phi(P_n, S_n, \emptyset, \emptyset) = r_n$, $(P, S, T, Q) \rightsquigarrow (P_n, S_n, \emptyset, \emptyset)$ and S_n has no repeated elements. Consequently, by Proposition 5.2, we have that $(P_n, S_n, \emptyset, \emptyset)$ is not in a dangerous situation. By Proposition 5.4, (P, S, T, Q) is not in a dangerous situation.

⇐) Suppose that $\mathrm{NR}(\phi(P, S, T, Q), \mathcal{E}) = 0$. According to Proposition 6.1, we have that $r_n = 0$.

Let $(P_{n-1}, S_{n-1}, T_{n-1}, Q_{n-1})$ such that $\phi(P_{n-1}, S_{n-1}, T_{n-1}, Q_{n-1}) = r_{n-1}$. According to Definition 6.2, we have that

$$r_n = r_{n-1} - \frac{\mathrm{LT}(r_{n-1})}{\mathrm{LT}(f_k)} f_k$$

where f_k is a polynomial in $\mathcal{E}$ such that $\mathrm{LT}(f_k)|\,\mathrm{LT}(r_{n-1})$.

Since r_{n-1} is a monomial and $r_n = 0$, we have that f_k must be a monomial and, consequently, f_k must be of the form s_x^2. Consequently, $s_x^2|\phi_{S_{n-1}}$, and therefore S_{n-1} contains the section x repeated. By Proposition 5.1, $(P_{n-1}, S_{n-1}, T_{n-1}, Q_{n-1})$ is in a dangerous situation. Since $(P, S, T, Q) \rightsquigarrow (P_{n-1}, S_{n-1}, T_{n-1}, Q_{n-1})$ (see i in Lemma 6.2), by Proposition 5.4 we have that (P, S, T, Q) is in a dangerous situation. ■

Corollary 6.1. *Let (P, Q) be an interlocking problem for a railway network E. We have that:*

$$(P, Q) \textit{ is in a dangerous situation} \Leftrightarrow \mathrm{NR}(\phi_P\phi_Q, \mathcal{E}) = 0$$

Proof.- An interlocking problem (P, Q) can be considered as an extended interlocking problem $(P, \emptyset, \emptyset, Q)$ (see in Remark 5.1). This corollary is an immediate consequence of the fact that $\phi(P, \emptyset, \emptyset, Q) = \phi_P\phi_Q$. ■

7. Implementation with *CoCoA*

In this section we will show how to use the Computer Algebra System *CoCoA* [18] to detect if an interlocking problem is in a dangerous situation for the railway station depicted in Figure 8.

Step 1. We will define the polynomial ring $\mathcal{A}$ and the list $\mathcal{E}$. In Example 6.1, we calculated it. In *CoCoA* syntax:

```
use ZZ/(2)[l1_2, l2_9, l9_10, l10_11, l11_6, l2_3, l3_4, l4_5, l5_6, l6_7,
l7_8, l2_1, l9_2, l10_9, l11_10, l6_11, l3_2, l4_3, l5_4, l6_5, l7_6, l8_7,
q[1..11],t[1..11],s[1..11]],
lex;

E:=[l1_2*t[1]+t[1]*t[2], l2_9*t[2]+t[2]*t[9],
l9_10*t[9]+t[9]*t[10], l10_11*t[10]+t[10]*t[11],
l11_6*t[11]+t[11]*t[6], l2_3*t[2]+t[2]*t[3],
l3_4*t[3]+t[3]*t[4], l4_5*t[4]+t[4]*t[5],
l5_6*t[5]+t[5]*t[6], l6_7*t[6]+t[6]*t[7],
l7_8*t[7]+t[7]*t[8], l2_1*t[2]+t[2]*t[1],
l9_2*t[9]+t[9]*t[2], l10_9*t[10]+t[10]*t[9],
l11_10*t[11]+t[11]*t[10], l6_11*t[6]+t[6]*t[11],
l3_2*t[3]+t[3]*t[2], l4_3*t[4]+t[4]*t[3],
l5_4*t[5]+t[5]*t[4], l6_5*t[6]+t[6]*t[5],
l7_6*t[7]+t[7]*t[6], l8_7*t[8]+t[8]*t[7],
t[1]^2+t[1], t[2]^2+t[2], t[3]^2+t[3], t[4]^2+t[4], t[5]^2+t[5], t[6]^2+t[6],
t[7]^2+t[7], t[8]^2+t[8], t[9]^2+t[9], t[10]^2+t[10], t[11]^2+t[11],
s[1]^2, s[2]^2, s[3]^2, s[4]^2, s[5]^2, s[6]^2, s[7]^2,
s[8]^2, s[9]^2, s[10]^2, s[11]^2,
t[1]+s[1], t[2]+s[2], t[3]+s[3], t[4]+s[4], t[5]+s[5], t[6]+s[6], t[7]+s[7],
t[8]+s[8], t[9]+s[9], t[10]+s[10], t[11]+s[11],
q[1]+t[1], q[2]+t[2], q[3]+t[3], q[4]+t[4], q[5]+t[5], q[6]+t[6], q[7]+t[7],
q[8]+t[8], q[9]+t[9], q[10]+t[10], q[11]+t[11]];
```

Step 2. For the particular configuration of the railway station in Figure 9, we calculate the monomial ϕ_P (see Example 6.2). In *CoCoA* syntax:

```
P:=l1_2*l9_10*l10_11*l2_3*l3_4*l11_6*l6_7*l7_8*l2_1*l11_10*l5_4*
l3_2*l6_11*l8_7*l5_6;
```

Step 3. For the particular placement of trains in the railway station in Figure 10, we calculate the monomial ϕ_Q (see Example 6.2). In *CoCoA* syntax:

```
Q:=q[1]*q[10];
```

Step 4. In order to detect if the situation is dangerous, we need to check if $\mathrm{NR}(\phi_P\phi_Q, \mathcal{E}) = 0$ (see Corollary 6.1). In *CoCoA* syntax:

```
NR(P*Q,E)=0;
```

As the output of *CoCoA* is "false", the proposed situation is safe.

Let us consider that a new train is placed in section S_8 (depicted in Figure 11). In this case, we need first to redefine the polynomial $\phi_Q = q_1q_{10}q_8$. In *CoCoA* syntax:

```
Q:=q[1]*q[10]*q[8];
NR(P*Q,E)=0;
```

As the output of *CoCoA* is "true", the interlocking problem is in a dangerous situation.

8. Experimental evaluation

In this section, we critically examine the effectiveness of the method proposed in this study, contrasting it with earlier algebraic strategies. We have performed a comparative study on the duration required to confirm the safety of a suggested scenario in large railway stations where paths may form cycles, altering the number of sections (N) and involved trains (M). We have disregarded the approach in [16] as it is not applicable to railway stations where paths can form cycles. The method we propose consistently surpasses others, attributed to its worst-case linear complexity. Table 1 displays the time necessary to evaluate the safety of various scenarios, excluding the model in [9] due to its inadequate performance in larger stations. These times reflect the average performance over ten different configurations (both safe and unsafe) of a railway station with N sections and M trains.

Notably, our innovative approach consistently takes less than one second to evaluate the safety of the proposed scenario, even with a significant number of sections, outpacing other methods. The first column corresponds to the track layout of a former Spanish railway station, with subsequent columns representing multiple concatenations of that station. The duration accounted for by our approach includes the four steps in Section 7, encompassing the computation of the polynomials for the topology, configuration, and trains.

Table 1. Comparative analysis of the time efficiency of various methods implemented for determining the safety of a proposed scenario in specific railway stations, where the paths may form cycles, involving N sections and M trains.

	N=52 M=5	N=156 M=15	N=260 M=25	N=520 M=50	N=780 M=75
Model described in [11]	15.34 s	270.132 s	> 1 h	> 1h	> 1h
Model described in [10]	35.46 s	397.58 s	> 1 h	> 1h	> 1h
Model described in [12]	3.423 s	48.342 s	674.245s	> 1h	> 1h
Model described in [14]	0.458 s	1.320 s	3.456s	> 1h	> 1h
Model described in [13]	0.050 s	0.073 s	1.456s	18.356s	> 1h
Our approach in *CoCoA*	< 0.001s	0.036	0.083 s	0.164 s	0.324s

In Table 2, we focus on railway stations where paths do not form cycles. In such instances, we can incorporate the approach in [16]. It is evident that our current approach yields rapid results, surpassing previous methods, with the exception of [16], as it is tailored for railway stations where cycle formation is not possible. However, as previously stated, this model [16] is not suitable for scenarios where cycles can occur.

Table 2. Comparative analysis of the time efficiency of various methods implemented for determining the safety of a proposed scenario in specific railway stations, where the paths may form cycles, involving N sections and M trains.

	N=52 M=5	N=156 M=15	N=260 M=25	N=520 M=50	N=780 M=75
Model described in [11]	10.23 s	250.250 s	> 1 h	> 1h	> 1h
Model described in [10]	15.356 s	380.368 s	> 1 h	> 1h	> 1h
Model described in [12]	1.124 s	38.292 s	554.180s	> 1h	> 1h
Model described in [14]	0.320 s	0.920 s	2.180s	> 1h	> 1h
Model described in [13]	0.020 s	0.047 s	1.038s	16.458s	> 1h
Model described in [16]	< 0.001s	0.015 s	0.078 s	0.156s	0.250 s
Our approach in *CoCoA*	< 0.001s	0.028 s	0.132 s	0.187s	0.223 s

9. Conclusions

In this paper, we have introduced an innovative algebraic model that identifies dangerous situations in any railway station. This model surpasses limitations of previous models, including those which have circular routes. Our model provides a mathematical resolution to the interlocking problem, achieving this in linear time. In this paper, we have defined the ‘interlocking problem’ as a safety verification for a single run-time configuration of a station, rather than a ‘command and control’ function. This is a verification issue, not a solution problem, typically handled by an interlocking system receiving movement authority requests.

The position of the trains within a railway station configuration is considered dangerous if the remainder of a specific monomial divided by a list of polynomials is zero. This finding enables the immediate implementation of the solution of the interlocking system on a computer algebra system.

Although the paper presented has a significant limitation, namely its challenging practical application in actual interlocking systems due to stringent certification requirements associated with such safety-critical applications, we believe it holds considerable theoretical interest. It bridges two seemingly disconnected fields and could prove beneficial for simulations that do not require certification credit.

We believe that our approach holds mathematical significance and can be expanded in several directions:

- Create a library that facilitates the creation of list E for any topology, defines switch changes and the aspect of the color light signals through functions, and updates train movements. The polynomials P and Q are updated through multiplication and division of monomials.
- Develop a graphical environment that enables visual design of a station and obtains the polynomials in E, P, and Q in a computer algebra system like *CoCoA*.
- Extend the interlocking problem to encompass other problems related to expert systems in railway stations, such as automatically detecting which semaphores and switches cannot be changed because they would imply a dangerous situation. Given that our approach expresses the interlocking problem as an algebraic system similar to those used for implementing expert systems, we believe that our model can be seamlessly integrated into these expert systems.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

This work has been partially funded by the University of Málaga (Spain). This paper is dedicated to the memory of Eugenio Roanes-Lozano. Eugenio's passing occurred during our discussions on the initial ideas that form the basis of this paper. He was an exceptional researcher in various fields, one of which was his extraordinary passion for trains, as evidenced by his numerous publications on the subject.

Conflict of interest

José Luis Galán-García is the Guest Editor of special issue "Computer Algebra Systems and Dynamic Geometry Systems in Mathematics Education, Engineering and Sciences" for AIMS Mathematics. José Luis Galán-García was not involved in the editorial review and the decision to publish this article. The authors declare there are no conflicts of interest.

References

1. N. Bešinović, Resilience in railway transport systems: a literature review and research agenda, *Transport Rev.*, **40** (2020), 457–478. https://doi.org/10.1080/01441647.2020.1728419

2. Y. Zhou, J. Wang, H. Yang, Resilience of transportation systems: concepts and comprehensive review, *IEEE Trans. Intell. Transp. Syst.*, **20** (2019), 4262–4276. https://doi.org/10.1109/TITS.2018.2883766

3. H. Lian, X. Wang, A. Sharma, M. A. Shah, Application and study of artificial intelligence in railway signal interlocking fault, *Informatica*, **46** (2022), 343–354. https://doi.org/10.31449/inf.v46i3.3961

4. A. Fantechi, G. Gori, A. E. Haxthausen, C. Limbrée, Compositional verification of railway interlockings: comparison of two methods, in *Reliability, Safety, and Security of Railway Systems. Modelling, Analysis, Verification, and Certification*, (2022), 3–19. https://doi.org/10.1007/978-3-031-05814-1_1

5. G. Lukács, T. Bartha, Conception of a formal model-based methodology to support railway engineers in the specification and verification of interlocking systems, in *2022 IEEE 16th International Symposium on Applied Computational Intelligence and Informatics (SACI)*, (2022), 000283–000288. https://doi.org/10.1109/SACI55618.2022.9919532

6. K. Winter, W. Johnston, P. Robinson, P. Strooper, L. van den Berg, Tool support for checking railway interlocking designs, in *10th Australian Workshop on Safety Re-lated Programmable Systems (SCS'05)*, Australian Computer Society, Sydney, **55** (2006), 101–107. Available from: https://crpit.scem.westernsydney.edu.au/confpapers/CRPITV55Winter.pdf.

7. J. Peleska, N. Krafczyk, A. E. Haxthausen, R. Pinger, Efficient data validation for geographical interlocking systems, *Formal Aspects Comput.*, **33** (2021), 925–955. https://doi.org/10.1007/s00165-021-00551-6

8. A. Fantechi, A. E. Haxthausen, M. B. R. Nielsen, Model checking geographically distributed interlocking systems using UMC, in *2017 25th Euromicro International Conference on Parallel, Distributed and Network-based Processing (PDP)*, (2017), 278–286. https://doi.org/10.1109/PDP.2017.66

9. E. Roanes-Lozano, L. M. Laita, An applicable topology-independent model for railway interlocking systems, *Math. Comput. Simul.*, **45** (1998), 175–183. https://doi.org/10.1016/S0378-4754(97)00093-1

10. E. Roanes-Lozano, E. Roanes-Macías, L. Laita, Railway interlocking systems and Gröbner bases, *Math. Comput. Simul.*, **51** (2000), 473–481. https://doi.org/10.1016/S0378-4754(99)00137-8

11. E. Roanes-Lozano, A. Hernando, J. A. Alonso, L. M. Laita, A logic approach to decision taking in a railway interlocking system using Maple, *Math. Comput. Simul.*, **82** (2011), 15–28. https://doi.org/10.1016/j.matcom.2010.05.024

12. A. Hernando, E. Roanes-Lozano, R. Maestre-Martínez, J. Tejedor, A logic-algebraic approach to decision taking in a railway interlocking system, *Ann. Math. Artif. Intell.*, **65** (2012), 317–328. https://doi.org/10.1007/s10472-012-9321-y

13. E. Roanes-Lozano, J. A. Alonso, A. Hernando, An approach from answer set programming to decision making in a railway interlocking system, *Rev. R. Acad. Cienc. Exactas, Fis. Nat. Ser. A Mat.*, **108** (2014), 973–987. https://doi.org/10.1007/s13398-013-0155-1

14. A. Hernando, R. Maestre, E. Roanes-Lozano, A new algebraic approach to decision making in a railway interlocking system based on preprocess, *Math. Probl. Eng.*, **2018** (2018), 4982974. https://doi.org/10.1155/2018/4982974
15. A. Hernando, E. Roanes-Lozano, An algebraic model for implementing expert systems based on the knowledge of different experts, *Math. Comput. Simul.*, **107** (2015), 92–107. https://doi.org/10.1016/j.matcom.2014.05.003
16. A. Hernando, A., E. Roanes-Lozano, J.L. Galán-García, G. Aguilera-Venegas, Decision making in railway interlocking systems based on calculating the remainder of dividing a polynomial by a set of polynomials, *Electron. Res. Arch.*, **31** (2023), 6160–6196. https://10.3934/era.2023313
17. D. Cox, J. Little, D. O'Shea, *Ideals, Varieties and Algorithms*, Springer, 2015. Available from: https://link.springer.com/book/10.1007/978-3-319-16721-3.
18. J. Abbott, A. M. Bigatti, CoCoALib: a C++ library for doing Computations in Commutative Algebra, 2019. Available from: http://cocoa.dima.unige.it/cocoalib.
19. D. Bjørner, *The FMERail/TRain Annotated Rail Bibliography*, 1999. Available from: http://www2.imm.dtu.dk/ db/fmerail/fmerail/.
20. M. J. Morley, *Modelling British Rail's Interlocking Logic: Geographic Data Correctness*, LFCS, Department of Computer Science, University of Edinburgh, 1991.
21. K. Nakamatsu, Y. Kiuchi, A. Suzuki, EVALPSN based railway interlocking simulator, in *Knowledge-Based Intelligent Information and Engineering Systems*, Berlin - Heidelberg, (2004), 961–967. https://doi.org/10.1007/978-3-540-30133-2_127
22. A. Janota, Using Z specification for railway interlocking safety, *Period. Polytech. Transp. Eng.*, **28** (2000), 39–53. Available from: https://pp.bme.hu/tr/article/view/1963.
23. K. M. Hansen, Formalising railway interlocking systems, in *Nordic Seminar on Dependable Computing Systems*, Lyngby, (1994), 83–94.
24. M. Montigel, *Modellierung und Gewährleistung von Abhängigkeiten in Eisenbahnsicherungsanlagen*, Ph.D. Thesis, ETH Zurich, Zurich, 1994. Available from: http://www.inf.ethz.ch/research/disstechreps/theses.
25. X. Chen, H. Huang, Y. He, Automatic generation of relay logic for interlocking system based on statecharts, in *2010 Second World Congress on Software Engineering*, (2010), 183–188. https://doi.org/10.1109/WCSE.2010.31
26. X. Chen, Y. He, H. Huang, An approach to automatic development of interlocking logic based on Statechart, *Enterp. Inf. Syst.*, **5** (2011), 273–286. https://doi.org/10.1080/17517575.2011.575475
27. X. Chen, Y. He, H. Huang, A component-based topology model for railway interlocking systems, *Math. Comput. Simul.*, **81** (2011), 1892–1900. https://doi.org/10.1016/j.matcom.2011.02.007
28. B. Luteberget, C. Johansen, Efficient verification of railway infrastructure designs against standard regulations, *Formal Methods Syst. Des.*, **52** (2018), 1–32. https://doi.org/10.1007/s10703-017-0281-z
29. M. Bosschaart, E. Quaglietta, B. Janssen, R. M. P. Goverde, Efficient formalization of railway interlocking data in RailML, *Inf. Syst.*, **49** (2015), 126–141. https://doi.org/10.1016/j.is.2014.11.007

30. E. Roanes-Lozano, L. M. Laita, E. Roanes-Macías, An application of an AI methodology to railway interlocking systems using computer algebra, in *Tasks and Methods in Applied Artificial Intelligence*, (1998), 687–696. https://doi.org/10.1007/3-540-64574-8_455
31. B. Buchberger, An algorithm for finding the basis elements of the residue class ring of a zero dimensional polynomial ideal, *J. Symb. Comput.*, **41** (2006), 475–511. https://doi.org/10.1016/j.jsc.2005.09.007
32. M. Davis, G. Logemann, D. Loveland, A machine program for theorem-proving, *Commun. ACM*, **5** (1962), 394–397. https://doi.org/10.1145/368273.368557
33. Ministerio de Transportes, Movilidad y Agenda Urbana, Estudio Informativo del Nuevo Complejo Ferroviario de la Estación de Madrid-Chamartín. Available from: https://www.mitma.gob.es/ferrocarriles/estudios-en-tramite/estudios-y-proyectos-entramite/chamartin.
34. Anonyous, Adjudicada la Modificación de Las Instalaciones de Seguridad, ERTMS, Comunicaciones y Energía de Madrid-Chamartín, *Vía Libre*, 2021. Available from: https://www.vialibre-ffe.com/noticias.asp?not=33047&cs=infr.
35. G. Aguilera-Venegas, J. L. Galán-García, E. Mérida-Casermeiro, P. Rodríguez-Cielos, An accelerated-time simulation of baggage traffic in an airport terminal, *Math. Comput. Simul.*, **104** (2014), 58–66. https://doi.org/10.1016/j.matcom.2013.12.009
36. G. Aguilera, J. L. Galán, J. M. García, E. Mérida, P. Rodríguez, An accelerated-time simulation of car traffic on a motorway using a CAS, *Math. Comput. Simul.*, **104** (2014), 21–30. https://doi.org/10.1016/j.matcom.2012.03.010
37. J. L. Galán-García, G. Aguilera-Venegas, M. Á. Galán-García, P. Rodríguez-Cielos, A new Probabilistic Extension of Dijkstra's Algorithm to simulate more realistic traffic flow in a smart city, *Appl. Math. Comput.*, **267** (2015), 780–789. https://doi.org/10.1016/j.amc.2014.11.076
38. J. L. Galán-García, G. Aguilera-Venegas, P. Rodríguez-Cielos, An accelerated-time simulation for traffic flow in a smart city, *J. Comput. Appl. Math.*, **270** (2014), 557–563. https://doi.org/10.1016/j.cam.2013.11.020
39. A. Iliasov, D. Taylor, L. Laibinis, A. B. Romanovsky, SafeCap: from formal verification of railway interlocking to its certification, 2021.
40. J. Pachl, Railway signalling principles, 2021. Available from: http://www.joernpachl.de/rsp.htm.
41. A. Borälv, Case study: formal verification of a computerized railway interlocking, *Formal Aspects Comput.*, **10** (1998), 338–360. https://doi.org/10.1007/s001650050021
42. Anonymous, Proyecto y obra del enclavamiento electrónico de la estación de Madrid-Atocha, *Proyecto Técnico*, Siemens, Madrid, 1988.
43. Anonymous, Microcomputer interlocking hilversum, Siemens, Munich, 1986.
44. Anonymous, Microcomputer interlocking rotterdam, Siemens, Munich, 1989.
45. Anonymous, Puesto de enclavamiento con microcomputadoras de la estación de Chiasso de los SBB, Siemens, Munich, 1989.
46. L. Villamandos, Sistema informático concebido por Renfe para diseñar los enclavamientos, *Vía Libre*, **348** (1993), 65.

Appendix: An algorithm with linear complexity for detecting dangerous situations in interlocking problems

Based on the results presented in Section 5, we have developed an algorithm to identify dangerous situations in interlocking problems. This algorithm operates iteratively, transforming the original interlocking problem into derived extended interlocking problems $X_1 \dots X_n$ until X_n can be solved directly. The algorithm is based on the following steps, which mirror those performed by the division algorithm of a Computer Algebra System when representing the interlocking problem in the algebraic terms of Section 6. For each section t in the railway station we consider:

Step 1. Check for repeated elements in Q. If any are found, output "It is in a dangerous situation", otherwise return "It is safe" (see Proposition 5.1).

Step 2. If $T = \emptyset$ and $Q \neq \emptyset$, apply condition i) from Definition 5.4. That is to say, consider some $t \in Q$ and update Q and T as follows:

$$Q := Q - \{t\}$$

$$T := T \cup \{t\}$$

Step 3. After applying Steps 1 and 2, we have $T \neq \emptyset$. Here, we iteratively apply condition iv) from Definition 5.4 until it can no longer be applied. That is, while there exist $t \in T$ and $(t, j) \in P$, we update P and T as follows:

$$P := P - \{(t, j)\}$$
$$\text{If } t \notin T \text{ then } T := T \cup \{j\}$$

As can be seen, before adding t to T, we check if $t \in T$. If $t \in T$, we do not need to add it again because of condition ii) in 5.4.

Step 4. After applying Step 3, it is guaranteed that all sections in T are locked. Otherwise, if $t \in T$ were not locked, we would have that there exists a j such that $(t, j) \in P$ (see Definition 5.1) and we could apply Step 3. In this way, we can iteratively apply condition iii) from Definition 5.4 until it can no longer be applied. That is, while there exists $t \in T$, we proceed as follows:

- If $t \in S$, output "It is in a dangerous situation" (see Proposition 5.1).
- Otherwise, update S and T as follows:
  $S := S \cup \{t\}$
  $T := T - \{t\}$

Next, we will introduce an algorithm in pseudocode to determine whether an interlocking problem (P, Q) is in a dangerous situation. We have assumed the following data structure to represent the sets P, S, T, Q:

- S, T, Q are represented by arrays of N integers (where N is the number of sections in the railway station), denoted as `S`, `T`, and `Q`. The value `Q[i]` indicates the number of times section i is repeated in Q.

- In addition to T for representing T, we will consider a list of integers, Ts. We have that $i \in$ Ts if and only if $i \in T$. Let us consider that $T = \{1, 1, 3\}$. We would have that T=[2,0,1,0] and Ts={1,1,3}.
- P is represented by an array of list of integers, denoted as P. The value P[i] contains a list of integers such that $j \in$ P[i] if and only if $(i, j) \in P$.

We will consider that there are two operations Push and Pop that add and remove elements from the lists. Both operations run in O(1). Additionally, we will consider that there is a function to check if the list is empty that also runs in O(1).

The presented algorithm makes use of some auxiliary variables: a list of integers, denoted as L; and some integer variables, denoted as t, i,j, and t2. We assume that N is a variable indicating the number of sections in the railway station.

```
(1) For t=1 to N
(2)   if Q[t]>1
(3)      return true  //it is in a dangerous situation
(4)   else if Q[t]=1
(5)      Q[t]:=0;
(6)      T[t]:=1; Push(Ts,t)   //Condition i) in ‘Derived from’ Relation
(7)      Push(L,t)
(8)      While L is not empty
(9)        i:=Pop(L);
(10)       While P[i] is not empty
(11)          j:=Pop(P[i]);  //Condition iv) in ‘Derived from’ Relation
(12)          if T[j]=0  //Condition ii) in ‘Derived from’ Relation
(13)             T[j]:=1; Push(Ts, j)
(14)             Push(L, j)
(15)       While Ts is not empty
(16)           t2:=Pop(Ts); Ts[t2]:=0;
(17)           if S[t2]>0
(18)              return true //it is in a dangerous situation
(19)           else
(20)              S[t2]:=1 //  Condition iii) in ‘Derived from’ Relation
(21) return false  //it is not in a dangerous situation
```

As may be seen:

Lines (2,3). They apply Step 1 described above. Initially, $S = \emptyset$.

Lines (5–6). They apply Step 2 described above.

Lines (7–14). These lines implement Step 3 as described above. A list L is utilized, which contains the sections that the algorithm has yet to examine for being locked. Lines (8-12) apply condition iv) in Definition 5.4. When P[i] is empty, it indicates that section i is locked. By the time the while loop in line (8) concludes, the list L is empty, ensuring that all sections in T are locked.

Lines (13–18). They apply Step 4 described above.

Line (21). When the `for` loop concludes, we have that $T = \emptyset$ and $Q = \emptyset$. Moreover, there are no repeated elements in S. Consequently, the interlocking problem is not in a dangerous situation (see Proposition 5.2).

Although it might seem that the algorithm is $O(N \cdot |P|)$ due to the nested while loops, upon careful examination, we can demonstrate that it operates in $O(N)$:

- Lines (5–7) are executed at most N times.
- Lines (11–14) are executed at most K times (counting all the iterations for all values of `t`). This fact is derived because the algorithm never add elements to `P` and it removes one element from `P` in line (11). Since `P` contains K elements, these lines can only be run at most K times.
- Line (9) is executed at most $N + K$ times (counting all the iterations for all values of `t`). This fact is derived because the algorithm adds one element to `L` in line (7) and line (14). It removes an element from `L` in line (9). Since line (7) is executed at most N times and line (14) is executed at most K times, line (9) can only be run at most $N + K$ times.
- Lines (16-29) are executed at most $N + K$ times (counting all the iterations for all values of `t`). This fact is derived because the algorithm adds one element to `Ts` in lines (6) and (13) and it removes one element from `Ts` in line (16). Since line (6) is executed at most N times, and line (13) is executed at most K times, lines (16–29) can only be run at most $N + K$ times.

Since K is $O(N)$ (see Remark 4.1), we have that the algorithm operates in $O(N)$.

AIMS Press