

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΗΛΕΚΤΡΟΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ ΚΑΙ ΠΛΗΡΟΦΟΡΙΚΗΣ  
ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ



ΠΑΝΕΠΙΣΤΗΜΙΟ  
ΠΑΤΡΩΝ  
UNIVERSITY OF PATRAS

ΕΞΟΡΥΞΗ ΔΙΑΔΙΚΑΣΙΩΝ ΣΕ  
ΚΥΒΕΡΝΟΦΥΣΙΚΑ ΣΥΣΤΗΜΑΤΑ: LIQUEUR  
PLANT CASE STUDY

Διπλωματική Εργασία  
του  
ΜΗΝΑΔΑΚΗ ΕΜΜΑΝΟΥΗΛ

Επιβλέπων: Θραμπουλίδης Κλεάνθης  
Καθηγητής

Συνεπιβλέπων: Μακρής Χρήστος  
Αναπληρωτής Καθηγητής

Συνεπιβλέπων: Ηλίας Αριστείδης  
Ε.ΔΙ.Π

Ιούνιος 2024





Πανεπιστήμιο Πατρών  
Πολυτεχνική Σχολή  
Τμήμα Μηχανικών Ηλεκτρονικών Υπολογιστών και Πληροφορικής

## ΕΞΟΡΥΞΗ ΔΙΑΔΙΚΑΣΙΩΝ ΣΕ ΚΥΒΕΡΝΟΦΥΣΙΚΑ ΣΥΣΤΗΜΑΤΑ: LIQUEUR PLANT CASE STUDY

Διπλωματική Εργασία  
του  
**ΜΗΝΑΔΑΚΗ ΕΜΜΑΝΟΥΗΛ**

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την .....

(Υπογραφή)

(Υπογραφή)

(Υπογραφή)

.....  
Θραμπουλίδης Κλεάνθης  
Καθηγητής

.....  
Μακρής Χρήστος  
Αναπληρωτής Καθηγητής

.....  
Ηλίας Αριστείδης  
Ε.ΔΙ.Π





Πανεπιστήμιο Πατρών  
Πολυτεχνική Σχολή  
Τμήμα Μηχανικών Ηλεκτρονικών Υπολογιστών και Πληροφορικής

Copyright ©—All rights reserved Μηναδάκης Εμμανουήλ 2024  
Με την επιφύλαξη παντός δικαιώματος.

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα.

Το περιεχόμενο αυτής της εργασίας δεν απηχεί απαραίτητα τις απόψεις του Τμήματος, του Επιβλέποντα, ή της επιτροπής που την ενέχρινε.

### Τπεύθυνη Δήλωση

Βεβαιώνω ότι είμαι συγγραφέας αυτής της διπλωματικής εργασίας, και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στην διπλωματική εργασία. Επίσης έχω αναφέρει τις όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών ή λέξεων, είτε αυτές αναφέρονται ακριβώς είτε παραφρασμένες. Επίσης, βεβαιώνω ότι αυτή η διπλωματική εργασία προετοιμάστηκε από εμένα προσωπικά ειδικά για τις απαιτήσεις του προγράμματος σπουδών του τμήματος Μηχανικών Ηλεκτρονικών Υπολογιστών και Πληροφορικής του Πανεπιστημίου Πατρών.

(Υπογραφή)

.....  
Μηναδάκης Εμμανουήλ



---

# Περίληψη

Στα ταχέως αναπτυσσόμενα πεδία του βιομηχανικού αυτοματισμού και του ψηφιακού μετασχηματισμού, τα κυβερνοφυσικά συστήματα διαδραματίζουν καθοριστικό ρόλο στην ενίσχυση της αποτελεσματικότητας και της αξιοπιστίας των συστημάτων. Αυτή η διπλωματική εργασία εξερευνά την ενσωμάτωση μιας μεθοδολογίας καθοδηγούμενης από δεδομένα που ονομάζεται εξόρυξη διαδικασιών, στον τομέα των κυβερνοφυσικών συστημάτων, εστιάζοντας πιο ειδικά σε ένα σύστημα παραγωγής λικέρ. Μέσω της case study προσέγγισης, διερευνάται η δυνατότητα της εξόρυξης διαδικασιών να αποκαλύψει, να αναλύσει και να βελτιώσει τις διαδικασίες στο συγκεκριμένο σύστημα. Η διπλωματική εργασία ξεκινά με μια ολοκληρωμένη ανασκόπηση της βιβλιογραφίας γύρω από την εξόρυξη διαδικασιών, τα συστήματα βιομηχανικού αυτοματισμού, τα κυβερνοφυσικά συστήματα και το διαδίκτυο των αντικειμένων, δημιουργώντας ένα θεωρητικό πλαίσιο για την εφαρμογή αυτής της μεθοδολογίας σε βιομηχανικά περιβάλλοντα. Αμέσως μετά, γίνεται μια πλήρης επεξήγηση θεμελιώδων εννοιών της εξόρυξης διαδικασιών και μια επισκόπηση σε τεχνικές εξόρυξης διαδικασιών, αλγόριθμους ανακάλυψης και λογισμικά. Έπειτα, περιγράφεται λεπτομερώς η προσέγγιση που υιοθετήθηκε για τη συλλογή δεδομένων, τη μοντελοποίηση της διαδικασίας, τον έλεγχο συμμόρφωσης και την περαιτέρω ανάλυση εντός του πλαισίου του συστήματος παραγωγής λικέρ. Ο σκοπός της διπλωματικής εργασίας, είναι να αναδείξει περισσότερο το πεδίο της εξόρυξης διαδικασιών σε πρακτικές εφαρμογές, κάνοντας ορατή τη χρησιμότητα της για τη βελτίωση των συστημάτων βιομηχανικού αυτοματισμού μέσα από αντικειμενική και αξιόπιστη γνώση.

**Λέξεις κλειδιά:** Βιομηχανία 4.0, Συστήματα Βιομηχανικού Αυτοματισμού, Κυβερνοφυσικά Συστήματα, Εξόρυξη Διαδικασιών



---

# Abstract

In the fast-growing fields of industrial automation and digital transformation, cyber-physical systems play a key role in enhancing system efficiency and reliability. This thesis explores the integration of a data-driven methodology called process mining, in the field of cyber-physical systems, focusing more specifically on a Liqueur Production system. Through the case study approach, the potential of process mining to reveal, analyze and improve the processes in the given system is explored. The thesis begins with a comprehensive literature review around process mining, industrial automation systems, cyber-physical systems and the Internet of Things, creating a theoretical framework for the application of this methodology in industrial settings. This is followed by a full explanation of process mining fundamental concepts and an overview of process mining techniques, discovery algorithms, and software. It then details the approach adopted for data collection, process modelling, conformance checking and further analysis within the framework of the Liqueur Production system. The purpose of this thesis is to highlight the field of process mining in practical applications, making visible its usefulness for the improvement of industrial automation systems through objective and reliable knowledge.

**Keywords:** Industry 4.0, Industrial Automation Systems, Cyber-physical systems, Process Mining



---

# Ευχαριστίες

Πρώτον, ευχαριστώ τον κ. Θραμπουλίδη για την εμπιστοσύνη και την καθοδήγηση που μου παρείχε κατά τη διάρκεια υλοποίησης της διπλωματικής εργασίας, ως επιβλέπων. Παράλληλα, ευχαριστώ όλους τους καθηγητές και δάσκαλους που έδωσαν αξία σε μένα και μέσα από τη μεταδοτικότητα τους, μου έδωσαν το κίνητρο να μαθαίνω. Τέλος, ευχαριστώ την οικογένεια μου για την υποστήριξη και την καθοριστική ύπαρξη της ως πρότυπο στην προσπάθεια να ολοκληρώσω τις σπουδές μου.

# Περιεχόμενα

<b>Περίληψη</b>	ii
<b>Abstract</b>	iv
<b>Ευχαριστίες</b>	vi
<b>Κατάλογος Σχημάτων</b>	x
<b>Κατάλογος Πινάκων</b>	xi
<b>Ακρωνύμια</b>	xii
<b>1 Εισαγωγή</b>	1
1.1 Βιομηχανία 4.0 . . . . .	2
1.1.1 Εξόρυξη διαδικασιών στη Βιομηχανία 4.0 . . . . .	4
1.2 Αντικείμενο και στόχος . . . . .	5
1.3 Δομή εργασίας . . . . .	6
<b>2 Ιστορικό</b>	8
2.1 Συστήματα βιομηχανικού αυτοματισμού . . . . .	9
2.1.1 Ανασκόπηση συστημάτων βιομηχανικού αυτοματισμού . . . . .	9
2.1.2 Προκλήσεις και στόχοι . . . . .	10
2.2 Εφαρμογές της εξόρυξης διαδικασιών . . . . .	11
2.2.1 Εφαρμογές σε συστήματα βιομηχανικού αυτοματισμού . . . . .	11
2.2.2 Άλλες εφαρμογές στη βιομηχανία . . . . .	12
<b>3 Εξόρυξη διαδικασιών</b>	14
3.1 Αρχεία καταγραφής . . . . .	15
3.2 Τεχνικές . . . . .	16
3.3 Μοντέλα διαδικασιών . . . . .	18
3.3.1 Petri net . . . . .	18

---

3.4	Αλγόριθμοι ανακάλυψης . . . . .	20
3.4.1	Αλγόριθμος-a . . . . .	21
3.4.2	Άλλοι αλγόριθμοι . . . . .	24
3.4.3	Μετρικές ποιότητας . . . . .	25
3.5	Έλεγχος συμμόρφωσης . . . . .	26
3.5.1	Επανάληψη σημείων . . . . .	27
3.5.2	Ευθυγράμμιση . . . . .	29
3.6	Εργαλεία . . . . .	32
3.6.1	Disco . . . . .	32
3.6.2	PM4PY . . . . .	32
<b>4</b>	<b>Εφαρμογή</b>	<b>33</b>
4.1	Επισκόπηση Liqueur Plant . . . . .	34
4.1.1	Περιγραφή συστήματος . . . . .	34
4.1.2	IoT εκδοχή συστήματος . . . . .	36
4.2	Εξόρυξη διαδικασιών στο Liqueur Plant . . . . .	38
4.2.1	Συλλογή και ανάλυση δεδομένων . . . . .	38
4.2.2	Ανακάλυψη διαδικασίας . . . . .	41
4.2.3	Έλεγχος συμμόρφωσης . . . . .	51
4.2.4	Σημεία συμφόρησης . . . . .	56
<b>5</b>	<b>Αποτελέσματα</b>	<b>64</b>
5.1	Αξιολόγηση ευρημάτων . . . . .	65
5.2	Στρατηγική βελτιστοποίησης . . . . .	69
<b>6</b>	<b>Συμπεράσματα και προοπτικές</b>	<b>71</b>
	<b>Παράρτημα</b>	<b>73</b>
	<b>Αναφορές</b>	<b>74</b>

# Κατάλογος Σχημάτων

1.1	Εξέλιξη της βιομηχανικής επανάστασης [26]	2
2.1	Επίπεδα κυβερνοφυσικού συστήματος [21]	10
3.1	Απλό αρχείο καταγραφής	16
3.2	Τεχνικές εξόρυξης διαδικασιών: ανακάλυψη, συμμόρφωση, εμπλουτισμός [20]	17
3.3	Μηχανή πεπερασμένων καταστάσεων	18
3.4	Petri net σε αρχική σήμανση	19
3.5	Πυροδότηση μετάβασης	20
3.6	Αποτύπωμα του L	21
3.7	Σχέσεις μεταξύ των μεταβάσεων [10]	22
3.8	Workflow net που επαληθεύει κάθε αρχείο καταγραφής με δραστηριότητες $\{a, b, \dots, h\}$ [10]	26
3.9	Token-based replay του $\sigma_1$ στο $N_1$ [10]	28
3.10	Workflow net $N_2$ [10]	30
4.1	Το φυσικό plant του συστήματος [6]	34
4.2	Αναπαράσταση του συστήματος myLiqueur [9]	36
4.3	Διακεχριμένες δραστηριότητες <code>lps_log</code>	40
4.4	Κώδικας επεξεργασίας αρχείου καταγραφής	41
4.5	Alpha miner για το <code>lps_log_A</code>	43
4.6	Inductive miner για το <code>lps_log_A</code>	44
4.7	Heuristics miner για το <code>lps_log_A</code>	45
4.8	Alpha miner για το <code>lps_log_B</code>	46
4.9	Inductive miner για το <code>lps_log_B</code>	47
4.10	Heuristics miner για το <code>lps_log_B</code>	48
4.11	Heuristic net του <code>lps_log_A</code>	49
4.12	Heuristic net του <code>lps_log_B</code>	50
4.13	Κώδικας ανακάλυψης και ελέγχου συμμόρφωσης	53

---

4.14 Κώδικας μέτρησης fitness σε αρχεία με σφάλματα . . . . .	56
4.15 Full process map (100%) διαδικασίας παραγωγής . . . . .	57
4.16 Filtered process map παραγωγής τύπου A . . . . .	59
4.17 Filtered process map παραγωγής τύπου B . . . . .	60
4.18 Περιορισμός ανάδευσης στη διαδικασία τύπου A . . . . .	62
4.19 Περιορισμός ανάδευσης στη διαδικασία τύπου B . . . . .	62
4.20 Περιορισμός μεταφοράς στη διαδικασία τύπου A . . . . .	63
4.21 Περιορισμός μεταφοράς στη διαδικασία τύπου B . . . . .	63
5.1 Activity diagram παραγωγής τύπου A [6] . . . . .	66
5.2 Activity diagram παραγωγής τύπου B . . . . .	66

# Κατάλογος Πινάκων

4.1	Αρχείο καταγραφής LPS . . . . .	39
4.2	Αρχείο καταγραφής μετά την επεξεργασία . . . . .	40
4.3	Αποτελέσματα TBR για το lps_log_A . . . . .	51
4.4	Αποτελέσματα TBR για το lps_log_B . . . . .	51
4.5	Αποτελέσματα alignments για το lps_log_A . . . . .	52
4.6	Αποτελέσματα alignments για το lps_log_B . . . . .	52
4.7	Fitness αρχείων με σφάλματα για το lps_log_A μέσω TBR . . . . .	54
4.8	Fitness αρχείων με σφάλματα για το lps_log_A μέσω alignments . . . . .	54
4.9	Fitness αρχείων με σφάλματα για το lps_log_B μέσω TBR . . . . .	55
4.10	Fitness αρχείων με σφάλματα για το lps_log_B μέσω alignments . . . . .	55
4.11	Μέσοι χρόνοι ολοκλήρωσης δραστηριοτήτων . . . . .	58
4.12	Χρονικές επιβαρύνσεις λόγω περιορισμών . . . . .	61
5.1	Χρόνοι της περίπτωσης 15 . . . . .	68

# Ακρωνύμια

**BPMN** Business Process Model Notation

**CPS** Cyber-physical Systems

**CPPS** Cyber-physical Production Systems

**IEC** International Electrotechnical Commission

**IEEE** Institute of Electrical and Electronics Engineers

**IoT** Internet of Things

**LPS** Liqueur Production System

**PLC** Programmable Logic Controller

**SCADA** Supervisory Control and Data Acquisition

**TBR** Token-based Replay

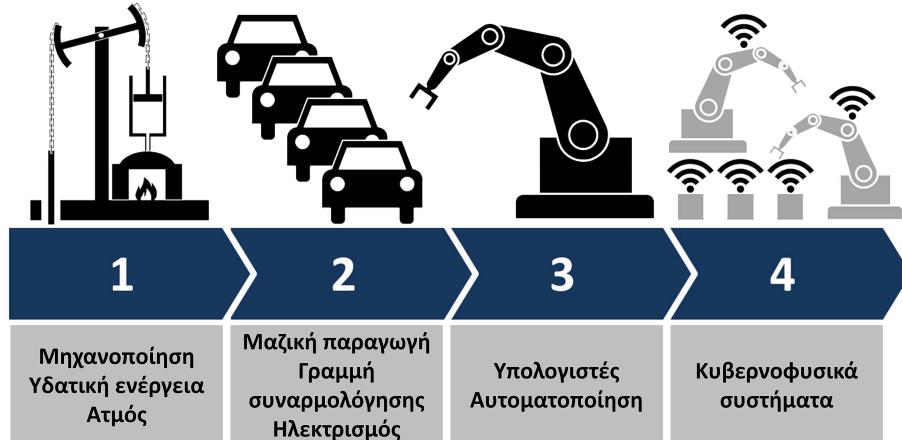
**UML** Unified Modeling Language

# Κεφάλαιο 1

## Εισαγωγή

## 1.1 Βιομηχανία 4.0

Η Βιομηχανία 4.0<sup>1</sup> αναφέρεται σε μια ριζική αλλαγή στο διεθνές βιομηχανικό προσκήνιο και σηματοδοτεί την έλευση της τέταρτης βιομηχανικής επανάστασης (σχήμα 1.1). Η εποχή μας χαρακτηρίζεται από τη χρήση ψηφιακών τεχνολογιών έχοντας καταφέρει αξιοσημείωτες αλλαγές στη βιομηχανοποίηση και παραγωγή. Η προέλευση του παραπάνω όρου ανήκει σε μια πρωτοβουλία της Γερμανικής κυβέρνησης για την ενίσχυση της υέσης της ως ηγέτη στην προηγμένη βιομηχανική παραγωγή [14]. Όμως, η ιδέα ενός «έξυπνου εργοστασίου»—μιας μονάδας βιομηχανικής παραγωγής νέας γενιάς που είναι ευέλικτη, δεν σπαταλά τους πόρους της και εφαρμόζει τις πιο πρόσφατες ψηφιακές τεχνολογίες—βρίσκεται στον πυρήνα της έννοιας της Βιομηχανίας 4.0. Αυτή η αλλαγή αντιπροσωπεύει τον συνολικό επαναπροσδιορισμό του τρόπου με τον οποίο εφευρίσκονται, αναπτύσσονται, κατασκευάζονται και παραδίδονται τα προϊόντα σε όλο τον κόσμο. Η βιομηχανική επανάσταση καθοδηγείται από τη σύγκλιση των ψηφιακών τεχνολογιών με τη συμβατική παραγωγή. Η ενσωμάτωση της μπορεί να ανεβάσει τις βιομηχανικές λειτουργίες σε επίπεδα που προηγουμένως δεν θα έχουν επιτευχθεί πάνω σε επικοινωνία, ανάλυση και αυτοματοποίηση. Πιο συγκεκριμένα οι ψηφιακές τεχνολογίες περιλαμβάνουν, προγνωστική συντήρηση, προηγμένες αναλύσεις, υπολογιστικό νέφος, διαδίκτυο των αντικειμένων και τεχνητή νοημοσύνη, ανοίγοντας το δρόμο στην πληροφορία να συμβάλλει στην αύξηση της αποτελεσματικότητας και της καινοτομίας. Τέτοιες τεχνολογίες δημιουργούν ένα ισχυρά συνδεδεμένο και ενημερωμένο περιβάλλον παραγωγής, όπου οι αποφάσεις βασίζονται στην πραγματικότητα, με δεδομένα και οι διαδικασίες ρυθμίζονται με ακρίβεια σε πραγματικό χρόνο.



Σχήμα 1.1: Εξέλιξη της βιομηχανικής επανάστασης [26]

<sup>1</sup><https://www.ibm.com/topics/industry-4-0>

## Κεφάλαιο 1. Εισαγωγή

---

Η έννοια της Βιομηχανίας 4.0 ξεπερνά τα όρια των μεμονωμένων εργοστασίων και περιλαμβάνει ολόκληρες αλυσίδες εφοδιασμού, αγορές εργασίας και οικονομικές δομές. Μέσω του ψηφιακού μετασχηματισμού ο κλάδος μπορεί να φέρει βελτίωση στον τομέα της παραγωγικότητας, της ευελιξίας και στην ικανοποίηση των χρηστών. Τα προϊόντα κατασκευάζονται κατά παραγγελία (make-to-order) για να καλύψουν τις ανάγκες των καταναλωτών. Ως εκ τούτου, η επιτάχυνση των κύκλων παραγωγής αυξάνει την ανταγωνιστικότητα των επιχειρήσεων σε όλο τον κόσμο. Επιπλέον, η Βιομηχανία 4.0 ενθαρρύνει τη βιωσιμότητα τόσο για τον άνθρωπο όσο και για το περιβάλλον, μέσω της βελτιστοποίησης των πόρων, συμπεριλαμβανομένων των ελαχιστοποιημένων αποβλήτων, προσδίδοντας ευφυΐα και αποτελεσματικότητα στον κατασκευαστικό τομέα. Με άλλα λόγια, η Βιομηχανία 4.0 είναι μια βαθύτερη έννοια που δεν αναφέρεται απλώς στο επόμενο βήμα στη βιομηχανική ανάπτυξη, αλλά σε ένα ολοκληρωμένο στρατηγικό όραμα που εντάσσει ψηφιακές τεχνολογίες στο πλαίσιο της παγκόσμιας παραγωγής. Η επανάσταση που προκαλεί φέρνει μια εικόνα για το μέλλον της παραγωγής: σε αυτήν, η ευελιξία, η βιωσιμότητα και η καινοτομία είναι οι καθοδηγητές. Με τους κορυφαίους βιομηχανικούς τομείς σε όλο τον κόσμο να υιοθετούν ήδη τις αρχές της Βιομηχανίας 4.0, ανοίγουν οι πύλες για ένα μέλλον όπου τα ψηφιακά και φυσικά συστήματα θα συμπίπτουν για να δημιουργήσουν νέες προτάσεις, σπρώχνοντας τον κόσμο σε μια εποχή τεχνολογικού δυναμισμού και οικονομικής ανάπτυξης.

Το διαδίκτυο των αντικειμένων και τα κυβερνοφυσικά συστήματα είναι δύο κορυφαίες και ακμάζουσες έννοιες της Βιομηχανίας 4.0, οι οποίες διαμορφώνουν την σημερινή εποχή γενεφυρώνοντας τα κενά μεταξύ των φυσικών αντικειμένων και του ψηφιακού σύμπαντος. Ο όρος IoT πρωτοεμφανίστηκε στα τέλη της δεκαετίας του 1990 από τον Ashton<sup>2</sup> και αφορά τη διασύνδεση φυσικών πραγμάτων ή πραγμάτων με λογισμικό, αισθητήρες, ηλεκτρονικά μέρη και δυνατότητες διαδικτύου, επιτρέποντας σε αυτά τα συνδεδεμένα πράγματα να ανταλλάσσουν δεδομένα. Η συνδεσιμότητα επιτρέπει ένα άμεσο επίπεδο ενσωμάτωσης του φυσικού κόσμου σε συστήματα που βασίζονται σε υπολογιστές, με αποτέλεσμα να αυξάνονται η απόδοση, η ακρίβεια και τα οικονομικά οφέλη [4]. Αντίθετα, το κυβερνοφυσικό σύστημα αναφέρεται σε ενσωμάτωση υπολογισμών σε φυσικές διαδικασίες. Στο CPS, ηλεκτρονικές συσκευές παρακολουθούν και ελέγχουν τις φυσικές διεργασίες, έχοντας συνήθως τους βρόχους ανάδρασης όπου οι φυσικές διεργασίες δρουν στους υπολογισμούς και αντίστοιχα οι υπολογισμοί επηρεάζουν τις φυσικές διεργασίες. Η ενοποίηση και ο συντονισμός μεταξύ φυσικών και ψηφιακών στοιχείων αποτελούν τον πυρήνα του CPS. Με άλλα λόγια, το IoT θεμελιώνει τη βάση της ικανότητας απόκτησης δεδομένων και επικοινωνίας του CPS, επιτρέποντας την ανάμειξη αυτών των φυσικών και ψηφιακών στοιχείων [27]. Τα IoT και CPS αναπτύσσουν από κοινού ένα έξυπνο περιβάλλον με συστήματα που είναι σε ετοιμότητα, προσαρμόζονται και ανταποκρίνονται απόλυτα σε μεταβαλλόμενες ανθρώπινες ανάγκες. Για παράδειγμα, σε ένα

<sup>2</sup><https://www.historyofinformation.com/detail.php?id=3411>

έξυπνο σπίτι, οι θερμοστάτες συλλέγουν δεδομένα για να εξισορροπήσουν τη θερμοκρασία, τα φώτα εξουκονομούν ενέργεια μέσω ενδοεπικοινωνίας μεταξύ τους και οι κάμερες ασφαλείας επιτηρούν το χώρο, συμβάλλουν στην αύξηση της ασφαλείας ενός σπιτιού. Έτσι, το CPS επιτρέπει έμφυτα ένα τέτοιο διασυνδεδεμένο πλαίσιο για την ανάδειξη της ψηφιακής πλευράς των καθημερινών πραγμάτων, και συνεπώς καθιστά το σύστημα έξυπνο. Συνεπώς, το CPS επωφελείται από αυτόν τον τύπο βελτιωμένης συνδεσιμότητας για παρακολούθηση και έλεγχο που προσφέρει το IoT, αυτοματοποιώντας τα φυσικά συστήματα. Μια τέτοια αμοιβαία εξαρτωμένη σχέση, ανοίγει το δρόμο για καινοτόμες εφαρμογές, μετατρέποντας τις βιομηχανίες και την καθημερινή ζωή σε μια νέα εποχή τεχνολογικής επανάστασης.

### 1.1.1 Εξόρυξη διαδικασιών στη Βιομηχανία 4.0

Η εξόρυξη διαδικασιών είναι ένας νέος κλάδος που βρίσκεται ανάμεσα στην εξόρυξη δεδομένων (data mining) και την επιστήμη διαδικασιών (process science) [3]. Η έννοια της εξόρυξης διαδικασιών αφορά ένα σύνολο από τεχνικές και αλγόριθμους για την ανακάλυψη, την παρακολούθηση και τη βελτίωση διαδικασιών που προκύπτουν μέσα από την αξιοποίηση δεδομένων που βρίσκονται σε αρχεία καταγραφής. Προσφέρει αντικειμενικές και βασισμένες σε γεγονότα γνώσεις, από πληροφορίες που υπάρχουν σε πραγματικά δεδομένα και αποσκοπεί στον έλεγχο, την ανάλυση και τη βελτίωση των υπαρχόντων διαδικασιών που ακολουθούνται από μια επιχείρηση.

Στις μέρες μας υπάρχει ολοένα και περισσότερο ενδιαφέρον για την εξόρυξη διαδικασιών, πράγμα που προκύπτει από το συνεχώς αυξανόμενο αριθμό δεδομένων που καταγράφονται σε πληροφοριακά συστήματα, προσφέροντας λεπτομέρειες για το ιστορικό των διαδικασιών και ενισχύοντας την ανάγκη να υποστηρίζουν και να βελτιώσουν της επιχειρησιακές διαδικασίες σε ανταγωνιστικά και ταχέως μεταβαλλόμενα περιβάλλοντα. Η κατάσταση αυτή έχει οδηγήσει στην δημιουργία μιας ομάδας που αποκαλείται «Ομάδα Κρούσης Εξόρυξης Διαδικασιών» του IEEE [3], με σκοπό την προώθηση της έρευνας και ανάπτυξης του κλάδου αυτού, ορίζοντας της κατευθυντήριες αρχές και απαριθμώντας σημαντικές προκλήσεις. Η εξέλιξη της τέταρτης βιομηχανικής επανάστασης θα μπορούσε να παρομοιαστεί με ένα «ταξίδι αλλαγής» προς τον ψηφιακό μετασχηματισμό και χαρακτηρίζεται από την ενσωμάτωση προηγμένων τεχνολογιών όπως το διαδίκτυο των αντικειμένων, η τεχνητή νοημοσύνη και η ανάλυση μεγάλων δεδομένων (big data) σε παραγωγικές ή επιχειρηματικές διαδικασίες. Η εξόρυξη διαδικασιών είναι μια τεχνική που θέλει να ξεκλειδώσει νέα επίπεδα λειτουργικής αποτελεσματικότητας και προσαρμοστικότητας αξιοποιώντας τον πλούτο δεδομένων που υπάρχουν στη σημερινή βιομηχανία και οφείλονται στην ενσωμάτωση των παραπάνω τεχνολογιών.

Η ενσωμάτωση της εξόρυξης διαδικασιών στην βιομηχανία της νέας γενιάς επιφυλάσσει σημαντικές δυνατότητες λειτουργικής βελτίωσης, αλλά κάνει ορατό και ένα σύνολο προκλήσεων. Η ποιότητα και η πληρότητα των δεδομένων είναι μείζονος σημασίας για

την αποτελεσματική εξόρυξη διαδικασιών, που απαιτούν ισχυρές πρακτικές διακυβέρνησης και διαχείρισης δεδομένων. Επιπλέον, η πολυπλοκότητα της ενσωμάτωσης εργαλείων εξόρυξης διαδικασιών με τις υπάρχουσες υποδομές πληροφοριακών συστημάτων θέτει τεχνικές προκλήσεις που πρέπει να αντιμετωπιστούν. Παρά αυτές τις προκλήσεις, οι ευκαιρίες που παρουσιάζονται από την εξόρυξη διαδικασιών στο πλαίσιο της Βιομηχανίας 4.0 είναι πολλές, και κυμαίνονται από ενισχυμένη διαφάνεια και αποτελεσματικότητα διαδικασιών μέχρι την απλούστευση της βελτίωσης και της εύρεσης καινοτομίας. Αξιοποιώντας την πληθώρα δεδομένων της βιομηχανίας του σήμερα, η εξόρυξη διαδικασιών παρέχει πολύτιμες πληροφορίες για την πραγματική απόδοση των διαδικασιών, επιτρέποντας στους οργανισμούς να βελτιστοποιούν τις λειτουργίες, να αυξάνουν την προσαρμοστικότητα και να οδηγούν στην καινοτομία. Η συνεχής εξέλιξη των τεχνικών εξόρυξης διαδικασιών, σε συνδυασμό με την πρόοδο των τεχνολογιών που χρησιμοποιούνται στη βιομηχανία, ξεκλειδώνει νέους ορίζοντες για λειτουργική αρτιότητα, βιωσιμότητα και ανταγωνιστικό πλεονέκτημα.

### 1.2 Αντικείμενο και στόχος

Το αντικείμενο της διπλωματικής εργασίας είναι η μελέτη και εφαρμογή της εξόρυξης διαδικασιών, καθώς και η εξερεύνηση της προοπτικής της εντός του πλαισίου της Βιομηχανίας 4.0, εστιάζοντας στα κυβερνοφυσικά συστήματα και τα συστήματα βιομηχανικού αυτοματισμού. Η εργασία ξεκινά με μια βιβλιογραφική ανασκόπηση στις εφαρμογές της εξόρυξης διαδικασιών στα συστήματα βιομηχανικού αυτοματισμού. Στη συνέχεια, αξιοποιείται η γνώση που προέκυψε εφαρμόζοντας τη μεθοδολογία αυτή σε ένα κυβερνοφυσικό σύστημα παραγωγής λικέρ, το Liqueur Plant. Σε πρώτη φάση, πραγματοποιείται μια προσομοίωση της λειτουργίας του συστήματος μέσω ενός κατάλληλου λογισμικού, με σκοπό την εξαγωγή αρχείων καταγραφής. Σε αυτό το σύστημα, επιχειρείται η ανακάλυψη των διαδικασιών παραγωγής μέσω της κατασκευής των αντίστοιχων μοντέλων. Αμέσως μετά, τα μοντέλα που προκύπτουν αξιολογούνται ως προς την ορθότητα με την οποία αναπαριστούν τα αρχεία καταγραφής του συστήματος, ενώ παράλληλα ελέγχονται και φεύγουν αρχεία καταγραφής που περιέχουν σφάλματα επάνω σε αυτά. Ακόμη, επιχειρείται ο εντοπισμός πιθανών σημείων συμφόρησης μέσα από την εστίαση σε συγκεκριμένα τμήματα της διαδικασίας, τα οποία σχετίζονται με περιορισμούς που επιβάλλει το σύστημα. Οι πληροφορίες που αντλούνται μέσα από την παραπάνω διαδικασία, δημιουργούν ένα πλαίσιο ανάδειξης προτάσεων για τη βελτίωση των διαδικασιών παραγωγής του συστήματος και οργανώνεται μια στρατηγική βελτιστοποίησης ως προς αυτές. Ο κύριος στόχος της διπλωματικής εργασίας, είναι να αναδείξει τη χρησιμότητα της εξόρυξης διαδικασιών στον κλάδο των βιομηχανικών συστημάτων παραγωγής και χρησιμοποιώντας το σύστημα ως test-bed, να ορίσει κατευθυντήριες γραμμές ως προς την βελτίωση συστημάτων παραγωγής, χρησιμοποιώντας την παραπάνω μεθοδολογία. Πιο συγκεκριμένα,

επιδιώκεται η ενίσχυση της αποτελεσματικότητας, της ασφάλειας και της αποδοτικότητας του συστήματος, μέσω της αξιοποίησης της γνώσης που προκύπτει από την ενσωμάτωση των τεχνικών εξόρυξης διαδικασιών στο σύστημα. Παράλληλα, αυτή η ενσωμάτωση προσπαθεί να συνδράμει ύστεικά ως προς την εξέλιξη του σχεδιασμού των κυβερνοφυσικών συστημάτων και την υλοποίηση των συστημάτων βιομηχανικού αυτοματισμού, στη μελλοντική εποχή της βιομηχανίας.

### 1.3 Δομή εργασίας

Στο κεφάλαιο 1, γίνεται η εισαγωγή στην έννοια της Βιομηχανίας 4.0 και την συνύπαρξη της με κυβερνοφυσικά συστήματα. Στη συνέχεια το κεφάλαιο μετατοπίζεται στην εξόρυξη διαδικασιών και τη χρήση της εντός του παραπάνω πλαισίου, παραθέτοντας τις προκλήσεις και τους στόχους που έχει να αντιμετωπίσει. Οι δύο αυτές έννοιες θέτουν τη βάση της διπλωματικής εργασίας, παρέχοντας μια επισκόπηση των τεχνολογιών εξελίξεων και τον κεντρικό ρόλο της εξόρυξης διαδικασιών στην αξιοποίηση τους για τη βελτίωση των συστημάτων.

Στο κεφάλαιο 2, γίνεται μια ιστορική ανασκόπηση των συστημάτων βιομηχανικού αυτοματισμού εξετάζοντας την εξέλιξή τους και την ενοποίηση με τα κυβερνοφυσικά συστήματα και το διαδίκτυο των αντικειμένων. Στη συνέχεια, ερευνώνται στη βιβλιογραφία οι εφαρμογές της εξόρυξης διαδικασιών εντός του πλαισίου αυτού, τονίζοντας τον ρόλο της στη βελτίωση των συστημάτων βιομηχανικού αυτοματισμού και αναφέροντας τις βιομηχανικές εφαρμογές της μέσα από την προϋπάρχουσα έρευνα.

Στο κεφάλαιο 3, γίνεται μια λεπτομερής επεξήγηση θεμελιωδών εννοιών που χρησιμοποιούνται στην εξόρυξη διαδικασιών. Σκοπός του κεφαλαίου είναι να συνδέσει το θεωρητικό υπόβαθρο που αναλύθηκε στα προηγούμενα κεφάλαια με τη μαθηματική πλευρά, εστιάζοντας στην περιγραφή των μοντέλων, των αλγόριθμων και των τεχνικών που πλαισιώνουν την εξόρυξη διαδικασιών.

Στο κεφάλαιο 4, ξεκινά η πρακτική εφαρμογή των τεχνικών της εξόρυξης διαδικασιών στο Liqueur Plant, αφού πρώτα γίνεται επισκόπηση και περιγραφή του συστήματος. Έπειτα προχωρά στις βασικές στρατηγικές υλοποίησης, χρησιμοποιώντας δεδομένα από έναν προσομοιωτή του συστήματος για την ανακάλυψη της διαδικασίας παραγωγής, καλύπτοντας τη φάση προ-επεξεργασίας των δεδομένων, τον έλεγχο συμμόρφωσης και την ανάλυση των σημείων συμφόρησης του συστήματος.

Στο κεφάλαιο 5, σχολιάζονται τα αποτελέσματα που προέκυψαν από την εξόρυξη διαδικασιών στο Liqueur Plant. Επίσης, γίνεται μια λεπτομερής αξιολόγηση της πληροφορίας που προέκυψε μέσα κάθε τεχνική και βγαίνουν συμπεράσματα που σχετίζονται με την παραγωγή κάθε τύπου. Στη συνέχεια, αναζητούνται οι τρόποι με τους οποίους αυτή η γνώση μπορεί να βοηθήσει στην βελτίωση της παράλληλης διαδικασίας παραγωγής.

Στο κεφάλαιο 6, παρουσιάζονται τα συμπεράσματα που προέκυψαν από την υλοποίη-

## Κεφάλαιο 1. Εισαγωγή

ση της διπλωματικής εργασίας και προτείνονται ορισμένες μελλοντικές κατευθύνσεις ως προς την περαιτέρω ανάπτυξης του συστήματος, καθώς επίσης αναφέρονται ορισμένοι περιορισμοί και εναλλακτικοί τρόποι προσέγγισης της μελέτης του συστήματος σε σχέση με την εξόρυξη διαδικασιών.

---

Κεφάλαιο 2

Ιστορικό

## 2.1 Συστήματα βιομηχανικού αυτοματισμού

### 2.1.1 Ανασκόπηση συστημάτων βιομηχανικού αυτοματισμού

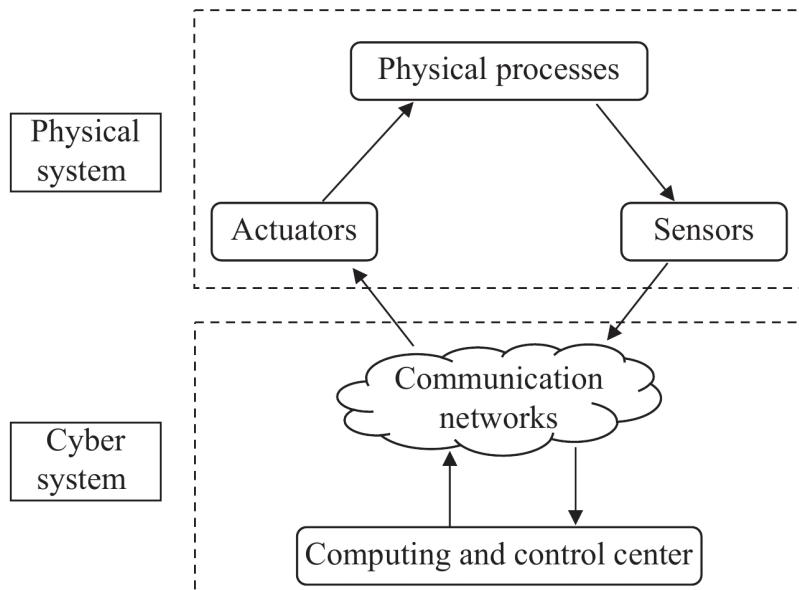
Η ανάπτυξη των συστημάτων βιομηχανικού αυτοματισμού από ηλεκτρομηχανολογικά συστήματα, σε περίπλοκες ενσωματώσεις κυβερνοφυσικών στοιχείων φανερώνει την τάση ανάπτυξης των τεχνολογιών αυτοματισμού κατά την περασμένο αιώνα. Αρχικά, τα συστήματα βιομηχανικού αυτοματισμού έτειναν προς μια απλουστευμένη φύση που επικεντρώνονταν σε μηχανισμούς άμεσου ελέγχου. Την αλλαγή σε αυτό έφερε ο ερχομός των προγραμματιζόμενων λογικών ελεγκτών<sup>1</sup> ή PLC, οι οποίοι πρωτοεμφανίστηκαν στα τέλη της δεκαετίας του 1960, επιτρέποντας την εκτέλεση πολύπλοκων διαδικασιών αυτοματισμού. Αυτή η ανακάλυψη, άνοιξε το δρόμο για ψηφιοποίηση των συστημάτων βιομηχανικού ελέγχου προκαλώντας αυξημένη πολυπλοκότητα αλλά παράλληλα παρέχοντας περισσότερες δυνατότητες [8]. Η ψηφιακή εποχή έχει επιταχύνει το μετασχηματισμό των συστημάτων βιομηχανικού αυτοματισμού ώστε να συμπεριλαμβάνει προηγμένες τεχνολογίες πληροφορικής. Αυτή η ενσωμάτωση είναι κρίσιμη για πέρασμα του βιομηχανικού τομέα στην εποχή της Βιομηχανίας 4.0, όπου η ενσωμάτωση των φυσικών λειτουργιών με την ψηφιακή νοημοσύνη είναι απρόσκοπτη. Το επόμενο στάδιο προόδου, το οποίο αντιπροσωπεύεται από το IoT και τα CPS, υποστηρίζει ένα περαιτέρω επίπεδο αποτελεσματικότητας, αξιοπιστίας και ευφυΐας. Στο παρόν στάδιο του βιομηχανικού αυτοματισμού, οι φυσικές διαδικασίες δρουν με ψηφιακή επίβλεψη και ψηφιακούς μηχανισμούς ελέγχου [7]. Η σημερινή εποχή είναι αυτή, στην οποία τα κυβερνοφυσικά συστήματα και το διαδίκτυο των πραγμάτων συναντιούνται με τα συστήματα βιομηχανικού αυτοματισμού. Η ενοποίηση αυτή συμβαίνει στα πλαίσια της Βιομηχανίας 4.0, αλλάζοντας τα πρότυπα τους από παραδοσιακά και απομονωμένα συστήματα, προς έξυπνα και διασυνδεδεμένα, ενισχύοντας την αποδοτικότητα, την επεκτασιμότητα και την παραγωγικότητα των βιομηχανικών διαδικασιών.

Τα CPS είναι μία γέφυρα μεταξύ του φυσικού και ψηφιακού κόσμου, χρησιμεύοντας ως υλικό και λογισμικό που μπορούν να ελέγχουν κάποια φυσική διαδικασία (σχήμα 2.1). Με το σύστημα βιομηχανικού αυτοματισμού, η άμεση παρακολούθηση και ο έλεγχος των φυσικών διαδικασιών βασίζεται σε ψηφιακά σήματα με τέτοιο τρόπο, ώστε οι αποφάσεις και οι παρεμβάσεις να πραγματοποιούνται σε πραγματικό χρόνο μέσω αυτοματισμού. Ως αποτέλεσμα αυτής της ολοκλήρωσης, μπορεί να επιτευχθεί μεγαλύτερη ευελιξία και ακρίβεια στον έλεγχο της διαδικασίας, επιφέροντας αποτελεσματική ενίσχυση της αποτελεσματικότητας και παραγωγικότητας. Η ομαλή αλληλεπίδραση των φυσικών λειτουργιών και των συστημάτων ψηφιακού αυτοματισμού, επιτρέπει στις βιομηχανίες να αποκομίσουν οφέλη από σύγχρονους υπολογιστικούς αλγόριθμους για τη βελτιστο-

<sup>1</sup><https://www.unitronicsplc.com/what-is-plc-programmable-logic-controller/>

ποίηση των διαδικασιών, την προγνωστική συντήρηση και τον εντοπισμό ανωμαλιών, ελαχιστοποιώντας το downtime και τα λειτουργικά έξοδα [21].

Το IoT επεκτείνει τα συστημάτων βιομηχανικού αυτοματισμού ώστε να φιλοξενούν μια πλατφόρμα, όπου ένας τεράστιος αριθμός συνδεδεμένων συσκευών και αισθητήρων εξυπηρετούνται στο βιομηχανικό περιβάλλον. Αυτή η συνδεσιμότητα επιτρέπει τη συλλογή δεδομένων, τη μετάδοση και την ανάλυση όλων τα στοιχεία του για την παροχή πληροφοριών, οι οποίες μέχρι τώρα ήταν αδύνατες. Η ενσωμάτωση με το IoT παρέχει την ικανότητα στο σύστημα βιομηχανικού αυτοματισμού να έχει πρόσβαση σε πολλά δεδομένα σε πραγματικό χρόνο, μέσα από τα οποία ο χειριστής θα είναι σε θέση να λαμβάνει τεκμηριωμένες αποφάσεις, να βελτιστοποιεί τις διαδικασίες και να προβλέπει τις απαιτήσεις συντήρησης. Αυτό ενισχύει περαιτέρω την λειτουργική ευελιξία του, λαμβάνοντας υπόψη ότι θα ήταν αδύνατη η απόκριση σε πραγματικό χρόνο σε μεταβαλλόμενες συνθήκες ή σε απαιτήσεις μέσω συστημάτων απομακρυσμένης παραχολούθησης και ελέγχου [23].



Σχήμα 2.1: Επίπεδα κυβερνοφυσικού συστήματος [21]

## 2.1.2 Προκλήσεις και στόχοι

Αφενός η ενοποίηση των CPS και IoT με συστήματα βιομηχανικού αυτοματισμού συγκεντρώνει πολλά πλεονεκτήματα, αλλά αφετέρου δημιουργεί κάποιες προκλήσεις. Πρώτο και κύριο, αποτελεί η έκθεση του συστήματος βιομηχανικού αυτοματισμού σε κυβερνοεπιθέσεις και για το λόγο αυτό, είναι σημαντική η αναζήτηση μεθόδων που οχυρώνουν το σύστημα από πιθανές απειλές. Ένα άλλο ρίσκο είναι, ότι με την εξάρτηση

των συστημάτων από όλο και περισσότερα ψηφιακά μέσα, επιφυλάσσονται κίνδυνοι που σχετίζονται με αστοχίες και δυσλειτουργίες του συστήματος, που οφείλονται στα κυβερνοφυσικά αυτά στοιχεία. Η διασφάλιση της αξιοπιστίας των συστημάτων και της ακεραιότητας των συσκευών CPS και IoT είναι ζωτικής σημασίας για την πρόληψη ατυχημάτων και τη εγγύηση της συνεχούς λειτουργίας των βιομηχανικών διαδικασιών. Μια τελευταία πρόκληση που εισάγουν τα CPS και IoT στα συστήματα βιομηχανικού αυτοματισμού είναι η επεκτασιμότητα, δηλαδή το κατά πόσο το σύστημα θα μπορεί να κλιμακωθεί αποτελεσματικά. Αυτή η πρόκληση αφορά και την ικανότητα να μπορέσει να ελέγξει το σύστημα ο φυσικός χειριστής του, διατηρώντας στο μεταξύ την απόδοση και αξιοπιστία του. Αυτοί οι λόγοι οδήγησαν στην αναζήτηση μεθόδων που θα ελέγχουν τις διαδικασίες, και καθιστούν τα συστήματα ανεκτικά σε σφάλματα.

## 2.2 Εφαρμογές της εξόρυξης διαδικασιών

### 2.2.1 Εφαρμογές σε συστήματα βιομηχανικού αυτοματισμού

Η εξόρυξη διαδικασιών έχει αναδειχθεί τα τελευταία χρόνια ως μια καινοτόμα προσέγγιση στην ενίσχυση της αποτελεσματικότητας και ασφάλειας σε συστήματα βιομηχανικού αυτοματισμού. Οι Xavier, Dubinin, Patil, Vyatkin [24] κάνουν γνωστούς μέσω της έρευνας τους, τους τρόπους που συμβάλλει η εξόρυξη διαδικασιών στα συστήματα βιομηχανικού ελέγχου δηλώνοντας ότι «χρησιμοποιούνται για την εξαγωγή μοντέλων διαδικασιών σε διαφορετική σημειογραφία σε σχέση με τα καταγεγραμμένα ίχνη συμπεριφοράς του συστήματος, για εφαρμογές όπως ανίχνευση ανωμαλιών, ανίχνευση επιθέσεων και ανάλυση συναγερμών». Η εφαρμογή τους, ένα σύστημα με μια λαβή και έναν ιμάντα μεταφοράς, δείχνει το πως η εξόρυξη διαδικασιών μπορεί να αξιοποιήσει τα αρχεία καταγραφής για τη κατασκευή μοντέλων, την επαλήθευση και τη βελτίωση των κυβερνοφυσικών συστημάτων που με αυτό τον τρόπο συμβάλλει στην ασφάλεια του συνολικού συστήματος και στην αποδοτικότητα του. Πιο αναλυτικά, τα παραδείγματα εφαρμογής εξόρυξης διαδικασιών σε συστήματα βιομηχανικού ελέγχου του πραγματικού κόσμου που αναλύονται, περιστρέφονται γύρω από:

- **Βελτίωση μοντέλων:** Οι τεχνικές εξόρυξης διαδικασιών χρησιμοποιούνται για τη βελτίωση των μοντέλων στον αυτοματισμό εργοστασίων, επιτρέποντας την πλήρη συμμόρφωση με τις επίσημες προδιαγραφές.
- **Ανίχνευση ανωμαλιών:** Η εξόρυξη διαδικασιών χρησιμοποιείται για τον έλεγχο συμμόρφωσης και τον εντοπισμό ανωμαλιών σε συστήματα βιομηχανικού ελέγχου. Βοηθά στον εντοπισμό αποκλίσεων συγχρίνοντας τα αρχεία καταγραφής

συμβάντων με μοντέλα αναφοράς, επιτρέποντας τον εντοπισμό επιθέσεων στον κυβερνοχώρο και της ανώμαλης συμπεριφοράς του συστήματος.

- **Ανάλυση συναγερμών:** Οι τεχνικές εξόρυξης διαδικασιών μπορούν να εφαρμοστούν για ανάλυση συναγερμών της βάσης δεδομένων καταγραφής συμβάντων στις βιομηχανικές εγκαταστάσεις. Αυτό περιλαμβάνει την εξαγωγή μοντέλων διεργασιών από αρχεία καταγραφής συμβάντων και τη χρήση εργαλείων εξόρυξης διαδικασιών για να αναλυθούν τα μοτίβα συναγερμών και η συμπεριφορά όλου του συστήματος.
- **Online εκτίμηση παραμέτρων για CPPS:** Η εξόρυξη διαδικασιών χρησιμοποιείται για την online εκτίμηση παραμέτρων στα CPPS και επιτρέπει την προσαρμογή του λογισμικού παραγωγής και αυτοματισμού, βασιζόμενο σε καταγεγραμμένα ίχνη συμπεριφοράς.
- **Μοντελοποίηση λογικής προγραμματισμού PLC:** Η εξόρυξη διαδικασιών χρησιμοποιείται για τη μοντελοποίηση της λογικής προγραμματισμού PLC αναλύοντας αρχεία καταγραφής συμβάντων με αλγόριθμους όπως ο split miner και ο DREAM-NAP για την πρόβλεψη των επόμενων δραστηριοτήτων. Αυτή η προσέγγιση βοηθά στην κατανόηση και την εξαγωγή κρυφής συμπεριφοράς σε βιομηχανικά συστήματα.

### 2.2.2 Άλλες εφαρμογές στη βιομηχανία

Την προοπτική γύρω από την ασφάλεια και την αποτροπή κυβερνοεπιθέσεων σε συστήματα βιομηχανικού ελέγχου, μέσα από χρήση της εξόρυξης διαδικασιών εξερευνούν οι Myers, Suriadi, Radke, Foo [13] οι οποίοι προτείνουν μια εφαρμογή για τον εντοπισμό μη αναμενόμενης συμπεριφοράς σε ένα σύστημα που αποτελείται από διάφορες συσκευές και συστήματα βιομηχανικού ελέγχου μεταξύ αυτών και SCADA<sup>2</sup>. Το σύστημα αποτελείται από έναν ιμάντα μεταφοράς, ένα σύστημα άντλησης νερού και ένα δοχείο πίεσης που ελέγχονται μέσω PLC. Η εφαρμογή εξόρυξης διαδικασιών που υλοποιεί, καταφέρνει τον εντοπισμό συγκεκριμένων ανωμαλιών εκτελώντας έλεγχο συμμόρφωσης σε αρχεία καταγραφής SCADA, που παράγονται από το σύστημα. Βασικός στόχος αυτής της εφαρμογής, είναι να υπερπηδήσει με τη βοήθεια της εξόρυξης διαδικασιών τα εμπόδια που θέτουν τα παραδοσιακά συστήματα προστασίας των συστημάτων βιομηχανικού ελέγχου, δηλαδή τα συστήματα ανίχνευσης εισβολής.

Οι πρακτικές εφαρμογές της εξόρυξης διαδικασιών στη βιομηχανία επεκτείνονται περισσότερο με τους Abonyi, Dorgo [15] να παρουσιάζουν ένα εύστοχο case study σε βιομηχανικά συστήματα παραγωγής, διερευνώντας τις δυνατότητες των λογισμικών

<sup>2</sup><https://en.wikipedia.org/wiki/SCADA>

εξόρυξης διαδικασιών για την ανάλυση βάσεων δεδομένων βιομηχανικών συναγερμών και καταδεικνύουν την αποτελεσματικότητα της στην «παρακολούθηση δραστηριοτήτων σε διάφορα επίπεδα ιεραρχίας, για την ανίχνευση του spillover effect και τον εντοπισμό αλλαγών στις συνθήκες λειτουργίας». Έτσι αναδεικνύεται η ευελιξία της εξόρυξης διαδικασιών στην αντιμετώπιση σύνθετων προκλήσεων σε βιομηχανικά περιβάλλοντα, δείχνοντας και την ικανότητα της να παρέχει αξιόπιστες πληροφορίες από αρχεία καταγραφής βάσεων δεδομένων συναγερμών.

Στην αυτοκινητοβιομηχανία, η εφαρμογή της εξόρυξης διαδικασιών δείχνει πολλά υποσχόμενη για τη βελτιστοποίηση των διαδικασιών παραγωγής σύμφωνα με τους Siek, Mukti [17]. Χρησιμοποιώντας αλγόριθμους εξόρυξης διαδικασιών «δημιουργούν αυτόματα διαδικασίες κατασκευής αυτοκινήτων, ελέγχοντας τη συμμόρφωση μεταξύ πραγματικών και υποθετικών διαδικασιών». Αυτό κάνει πιο εμφανή τη χρησιμότητα της εξόρυξης διαδικασιών στον εντοπισμό και την επίλυση προβλημάτων παραγωγής, καθώς επίσης είναι ένα χαρακτηριστικό παράδειγμα του ρόλου της στη βελτίωση της αποτελεσματικότητας και της ποιότητας των διαδικασιών παραγωγής αυτοκινήτων.

Η εξερεύνηση της εξόρυξης διαδικασιών στη βιομηχανική παραγωγή, κάνει πιο έντονη την επιρροή της στην ενίσχυση της ασφάλειας, της αποτελεσματικότητας και των λειτουργικών δυνατοτήτων βιομηχανικών συστημάτων. Κοιτώντας μέσα από το πρίσμα των μελετών που αναφέρθηκαν πιο πάνω, γίνεται προφανές ότι η εξόρυξη διαδικασιών είναι ένα σημαντικό εργαλείο για την εξαγωγή πολύτιμων πληροφοριών από τα αρχεία καταγραφής του συστήματος και διευκολύνει τον εντοπισμό ανωμαλιών, τη βελτιστοποίηση διαδικασίας και τον έλεγχο προδιαγραφών. Η πληθώρα εφαρμογών της εξόρυξης διαδικασιών, που κυμαίνονται από συστήματα βιομηχανικού ελέγχου έως την κατασκευή αυτοκινήτων, δίνει έμφαση στην ευελιξία και τις δυνατότητές της εξόρυξης διαδικασιών να φέρει επανάσταση στις παραδοσιακές βιομηχανικές διαδικασίες.

---

## Κεφάλαιο 3

### Εξόρυξη διαδικασιών

### 3.1 Αρχεία καταγραφής

Η εξόρυξη διαδικασιών αξιοποιεί δεδομένα συμβάντων ή αλλιώς αρχεία καταγραφής (event logs) για να εντοπίσει πληροφορίες σχετικές με συμβάντα [10]. Το αρχείο καταγραφής είναι μία λίστα από δεδομένα, στην οποία κάθε γραμμή αντιστοιχεί σε ένα συμβάν (event) και κάθε στήλη σε μια ιδιότητα (attribute). Για να γίνει εφικτό να αξιοποιηθεί ένα αρχείο καταγραφής με τεχνικές της εξόρυξη διαδικασιών, είναι αναγκαίο σε όλα τα συμβάντα του να περιέχεται τουλάχιστον μια στήλη για την περίπτωση και μια στήλη που για τη δραστηριότητα, ενώ επίσης σημαντικό αλλά όχι απαραίτητο στοιχείο είναι η χρονοσήμανση κάθε συμβάντος. Στο σχήμα 3.1 φαίνεται ένα αρχείο καταγραφής με τις εξής ιδιότητες:

- Η περίπτωση (case) λειτουργεί σαν ένα περίβλημα για τα αντίστοιχα επιμέρους συμβάντα. Όλες οι περιπτώσεις συνιστούν μια διαδικασία και ξεχωρίζουν με ένα μοναδικό αναγνωριστικό εντός του αρχείο καταγραφών. Με άλλα λόγια, σηματοδοτεί μια διαδικασία από την αρχή μέχρι το τέλος της.
- Η δραστηριότητα (activity) περιγράφει το είδος του συμβάντος. Ουσιαστικά, δίνει πληροφορία σχετικά με τη φύση του συμβάντος και αποτελεί θεμελιώδες στοιχείο στην ανάλυση της διαδικασίας και την αληθούσχια των συμβάντων.
- Η χρονοσήμανση (timestamp) καθορίζει την ακριβή ώρα που έλαβε χώρα ένα συγκεκριμένο συμβάν κατά τη διάρκεια της διαδικασίας. Είναι σημαντικό στοιχείο για την κατασκευή της χρονικής σειράς με την οποία έλαβαν χώρα τα συμβάντα της διαδικασίας, ενώ παράλληλα δείχνει την ώρα που ξεκίνησε και το πόσο διήρκησε ένα συμβάν.

Όλες οι υπόλοιπες στήλες στο αρχείο καταγραφής, αποτελούν επιπρόσθετες ιδιότητες και μπορεί να περιέχουν βιοηθητικά δεδομένα (resources), όπως για παράδειγμα το όνομα του χρήστη που σχετίζεται με τη δραστηριότητα, το κόστος κ.λπ. Τα βιοηθητικά δεδομένα μπορούν να χρησιμοποιηθούν για συγκεκριμένες τεχνικές της εξόρυξης διαδικασιών που σχετίζονται με φιλτράρισμα δεδομένων, όπως ανίχνευση κοινοτήτων [22]. Οι περισσότεροι αλγόριθμοι που εφαρμόζονται στα αρχεία καταγραφής για την ανακάλυψη του μοντέλου της διαδικασίας αρκούνται στην ύπαρξη των τριών πρώτων ιδιοτήτων, ωστόσο σήμερα υπάρχουν αλγόριθμοι που μπορούν να το κάνουν αυτό χωρίς δεδομένα χρονοσήμανσης ή περίπτωσης [11].

case id (here an order)	activity	timestamp	resource	costs	customer
...	...	...	...	...	...
2019-88201	create purchase requisition	25-07-2019:09.15	John	€20.20	9950
2019-88201	create purchase order	25-07-2019:09.35	Mary	€48.30	9950
2019-88201	approve purchase order	25-07-2019:09.55	Sue	€30.70	9950
2019-88202	create purchase requisition	25-07-2019:10.15	John	€28.20	9955
2019-88202	create purchase order	25-07-2019:10.25	Mary	€29.30	9955
2019-88202	approve purchase order	25-07-2019:10.40	Sue	€37.60	9955
2019-88201	receive order confirmation	25-07-2019:11.50	Mary	€42.10	9950
2019-88201	receive goods	27-07-2019:09.35	Peter	€50.20	9950
2019-88202	receive order confirmation	27-07-2019:09.45	Mary	€42.30	9955
2019-88202	receive invoice	28-07-2019:10.10	Sue	€44.90	9955
2019-88201	receive invoice	28-07-2019:10.20	Sue	€30.80	9950
2019-88201	pay invoice	29-07-2019:11.05	Sue	€30.70	9950
2019-88202	receive goods	29-07-2019:11.35	Peter	€51.30	9955
2019-88202	pay invoice	29-07-2019:12.15	Sue	€29.20	9955
...	...	...	...	...	...

Σχήμα 3.1: Απλό αρχείο καταγραφής

## 3.2 Τεχνικές

Η βασική ιδέα της εξόρυξης διαδικασιών, είναι η εδραιώση μιας ισχυρής συσχέτισης μεταξύ των αρχείων καταγραφής, δηλαδή του «τι γίνεται στην πραγματικότητα» και των μοντέλων διαδικασιών. Όπως αναφέριθηκε στον ορισμό, η εξόρυξη διαδικασιών αφορά ένα σύνολο από αλγόριθμους και τεχνικές για την ανακάλυψη διαδικασιών. Αυτές οι τεχνικές διαφοροποιούνται ανάλογα με το πως αξιοποιούν τα αρχεία καταγραφής και τα μοντέλα διαδικασιών.

Για την υλοποίηση των τεχνικών της εξόρυξης διαδικασιών, χρησιμοποιούνται οι παρακάτω μέθοδοι [10].

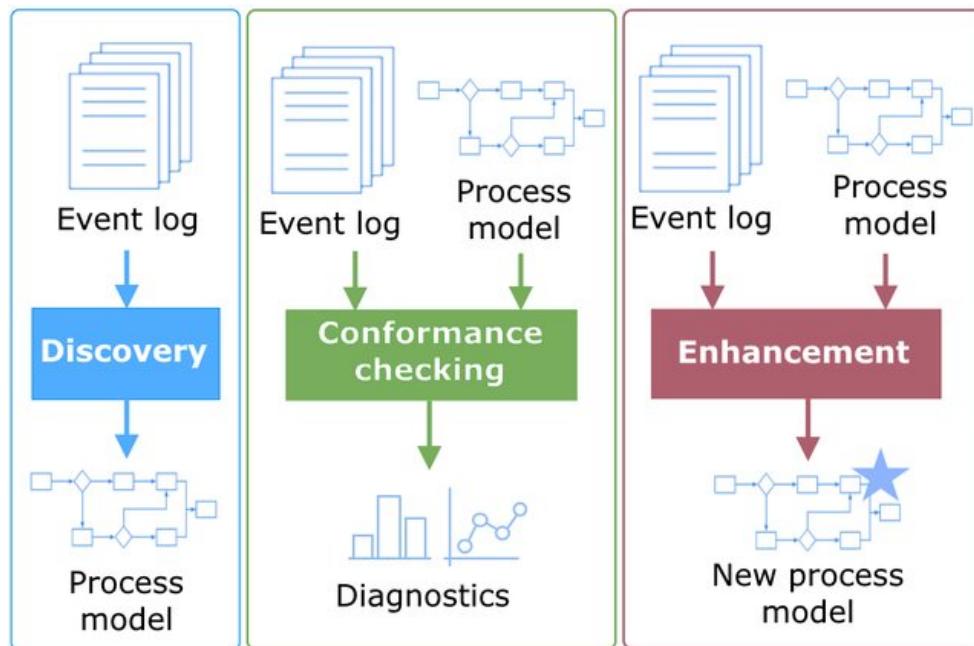
- Η μέθοδος play-out αφορά την κλασσική χρήση των μοντέλων διαδικασιών ως είσοδο (petri nets) με σκοπό να εξαγχθεί η συμπεριφορά πειραματικά, για να αναπαρασταθεί η διαδικασία.
- Η μέθοδος play-in, ξεκινάει από τη μελέτη της συμπεριφοράς ως είσοδο με σκοπό να κατασκευαστεί το μοντέλο της διαδικασίας. Οι τεχνικές του τύπου play-in προϋποθέτουν την ύπαρξη ενός αρχείου καταγραφής.
- Η μέθοδος replay αφορά την αξιοποίηση και των δύο ως εισόδους, δηλαδή αρχείου καταγραφής και μοντέλων διαδικασιών, όπου το αρχείο καταγραφής επαναλαμβάνε-

### Κεφάλαιο 3. Εξόρυξη διαδικασιών

ται σε αντιπαράθεση με το μοντέλο της διαδικασίας. Με αυτό τον τρόπο, γίνεται εφικτό να βρεθούν αποκλίσεις μεταξύ αρχείου καταγραφής και μοντέλου.

Την τελευταία μέθοδο υλοποιεί η τεχνική του ελέγχου συμμόρφωσης (conformance checking) που αποτελεί ένα από τους τύπους της εξόρυξης διαδικασιών. Σκοπός του ελέγχου συμμόρφωσης είναι να βρεθεί το κατά πόσο το μοντέλο σχετίζεται με τα δεδομένα. Μέσα από τη συμμόρφωση, αντλούνται διάφορες πληροφορίες σχετικές με το χρόνο και το πόσο συχνά επαναλαμβάνεται ένα μέρος της διαδικασίας. Επιπλέον, οι πληροφορίες βοηθούν στο να κατασκευαστεί προγνωστικό μοντέλο βασισμένο στα συμβάντα (μετά την ενεργοποίηση ενός συμβάντος μπορεί να γίνει η πρόβλεψη ότι η υπόλοιπη διαδικασία θα τρέζει για συγκεκριμένο χρονικό διάστημα).

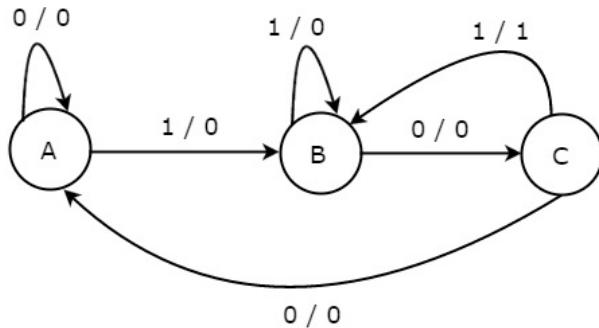
Η τεχνική της ανακάλυψης διαδικασιών (process discovery) χρησιμοποιεί τη μέθοδο του play-in, στην οποία δεν προϋπάρχει ένα μοντέλο διαδικασιών αλλά αυτό ανακαλύπτεται μέσα από τα δεδομένα. Γύραρχουν διάφοροι αλγόριθμοι που εφαρμόζονται στα δεδομένα για να κατασκευαστεί το μοντέλο. Η επόμενη τεχνική της εξόρυξης διαδικασιών, που απαιτεί ένα προϋπάρχον μοντέλο διαδικασιών είναι ο εμπλουτισμός διαδικασιών (process enhancement). Σκοπεύει στην επέκταση ή βελτίωση του μοντέλου διαδικασιών χρησιμοποιώντας τις πρόσθετες πληροφορίες σχετικά με την πραγματική διαδικασία που παρουσιάζεται στο αρχείο καταγραφής [10]. Στο σχήμα 3.2 συνοψίζονται οι παραπάνω βασικές τεχνικές στην εξόρυξη διαδικασιών.



Σχήμα 3.2: Τεχνικές εξόρυξης διαδικασιών: ανακάλυψη, συμμόρφωση, εμπλουτισμός [20]

### 3.3 Μοντέλα διαδικασιών

Τα μοντέλα διαδικασιών χαρακτηρίζονται ως μια οπτική αναπαράσταση του τρόπου με τον οποίο κυλάει κάποια διαδικασία. Είναι ένα θεμελιώδες κομμάτι της εξόρυξης διαδικασιών καθώς οι αλγόριθμοι που εφαρμόζονται στα αρχεία καταγραφής παράγουν τέτοιου είδους μοντέλα και έπειτα αυτά χρησιμοποιούνται σε τεχνικές της εξόρυξης διαδικασιών. Η πιο απλή σημειογραφία ενός μοντέλου διαδικασίας είναι το σύστημα μεταβάσεων ή μηχανή πεπερασμένων καταστάσεων (finite-state machine) το οποίο αποτελείται από καταστάσεις και μεταβάσεις, όπως φαίνεται στο σχήμα 3.3.



**Σχήμα 3.3:** Μηχανή πεπερασμένων καταστάσεων

Πλέον έχουν αναπτυχθεί κι άλλες σημειογραφίες μοντέλων οι οποίες εφαρμόζονται συγκεκριμένα πλαίσια, όπως BPMN για περιγραφή επιχειρησιακών διαδικασιών ή UML για περιγραφή λογισμικού, ωστόσο στην εξόρυξη διαδικασιών δίνεται ιδιαίτερη έμφαση στα petri nets. Ως petri net ή δίκτυο τόπου-μετάβασης, χαρακτηρίζεται ένα μαθηματικό μοντέλο που χρησιμοποιείται για να περιγράψει κάποιο σύστημα του οποίου οι πιθανές καταστάσεις ανήκουν σε ένα πεπερασμένο σύνολο. Έχει πάρει το όνομα του προς τιμήν του Γερμανού μαθηματικού Carl Adam Petri<sup>1</sup>, που επινόησε την έννοια το 1939 για να περιγράψει χημικές διαδικασίες. Παράλληλα, είναι από τις παλιότερες και πιο διαδεδομένες γλώσσες μοντελοποίησης διαδικασιών και σκοπός του είναι να δείξει ποιες δραστηριότητες πρέπει να εκτελεστούν και σε ποια σειρά [10].

#### 3.3.1 Petri net

Το petri net είναι ένα κατευθυνόμενο γράφημα το οποίο αποτελείται από τόπους (places) και μεταβάσεις (transitions), ενώ μέσα σε αυτό μπορούν να μετακινούνται σημεία (tokens). Η κατάσταση του petri net ονομάζεται σήμανση (marking) και χαρακτηρίζεται

<sup>1</sup>[https://en.wikipedia.org/wiki/Carl\\_Adam\\_Petri](https://en.wikipedia.org/wiki/Carl_Adam_Petri)

από την κατανομή των σημείων στο δίκτυο. Οι τόποι συμβολίζονται με κύκλους, ενώ οι μεταβάσεις με ορθογώνια.

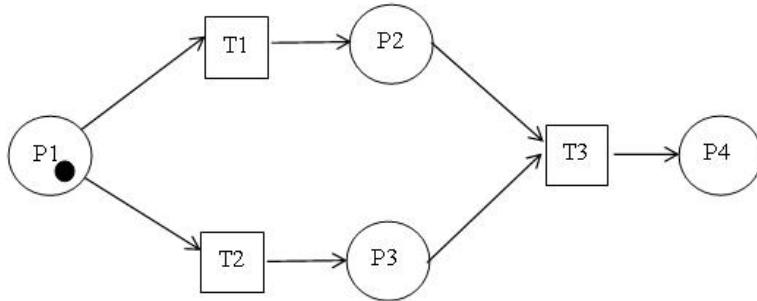
**Ορισμός 1.** Γράφημα petri net έιναι μια τριάδα  $N = (P, T, F)$  όπου

- $P$  ένα πεπερασμένο σύνολο τόπων
- $T$  ένα πεπερασμένο σύνολο μεταβάσεων
- $P \cap T = \emptyset$
- $F \subseteq (P \times T) \cup (T \times P)$  ένα σύνολο κατευθυνόμενων ακμών

**Ορισμός 2.** Σήμανση petri net είναι ένα ζεύγος  $(N, M)$  όπου

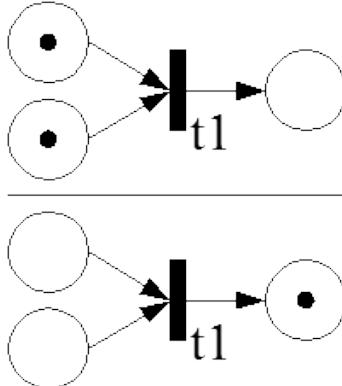
- $N = (P, T, F)$  ένα γράφημα petri net
- $M \in \mathbb{B}(P)$  ένα πολυσύνολο των τόπων  $P$

Από τον ορισμό του γραφήματος petri net γίνεται εμφανές ότι οι τόποι δεν μπορούν να είναι ταυτόχρονα και μεταβάσεις, ενώ επίσης οι ακμές δε μπορούν να συνδέουν δύο τόπους ή δύο μεταβάσεις μεταξύ τους. Αντίστοιχα, από την ορισμό της σήμανσης petri net συμπεραίνεται ότι σε κάθε τόπο μπορεί να υπάρχουν παραπάνω από ένα σημεία. Για το δίκτυο του σχήματος 3.4 με βάση τους παραπάνω ορισμούς έχουμε:  $P = \{P1, P2, P3, P4\}$ ,  $T = \{T1, T2, T3\}$ ,  $F = \{(P1, T1), (P1, T2), (T1, P2), (T2, P3), (P3, T3), (P2, T3), (T3, P4)\}$  και  $M = [P1]$ .



**Σχήμα 3.4:** Petri net σε αρχική σήμανση

Η ροή των σημείων μέσα στο δίκτυο πραγματοποιείται από την πυροδότηση (firing) που προκαλείται σε μια ενεργοποιημένη μετάβαση, δηλαδή σε μια μετάβαση που σε κάθε τόπο πριν από αυτή υπάρχει ένα σημείο. Στο σχήμα 3.5 φαίνεται ότι η πυροδότηση της μετάβασης προκαλεί την εξαφάνιση ενός σημείου από κάθε τόπο πριν τη μετάβαση, και την εμφάνιση ενός σημείου σε κάθε τόπο μετά τη μετάβαση.



**Σχήμα 3.5:** Πυροδότηση μετάβασης

## 3.4 Αλγόριθμοι ανακάλυψης

Ως αλγόριθμος ανακάλυψης χαρακτηρίζεται η μεθοδολογία με την οποία τα αρχεία καταγραφής μετατρέπονται σε μοντέλα. Η μελέτη των αλγόριθμων ανακάλυψης αποτελεί το σημαντικότερο κεφάλαιο και μια από τις πιο απαιτητικές εργασίες της εξόρυξης διαδικασιών. Για το σκοπό αυτό έχουν αναπτυχθεί μια πληθώρα αλγόριθμων οι οποίοι αντιμετωπίζουν συγκεκριμένες προκλήσεις που παρουσιάζονται λόγω της ποικιλομορφίας των αρχείων καταγραφής. Ένας γενικός ορισμός που θεμελιώνει ο van der Aalst [10] για την έννοια του αλγόριθμου ανακάλυψης είναι ο παρακάτω:

**Ορισμός 3.** Δοθέντος ενός αρχείου καταγραφής  $L$  όπως ορίζεται παραπάνω ή σύμφωνα με το πρότυπο XES<sup>2</sup>, αλγόριθμος ανακάλυψης είναι μια συνάρτηση που αντιστοιχίζει το  $L$  σε ένα μοντέλο διαδικασίας, τέτοιο ώστε το μοντέλο διαδικασίας να είναι αντιπροσωπευτικό της συμπεριφοράς που περιγράφεται στο αρχείο καταγραφής.

<sup>2</sup><https://xes-standard.org/>

### 3.4.1 Αλγόριθμος-a

Ο πρώτος αλγόριθμος που αναπτύχθηκε για να ξεπεράσει τα εμπόδια που υπήρχαν στην ανακάλυψη διαδικασιών και ο πρώτος που αντιμετώπισε επαρκώς το συγχρονισμό και την επαναληψιμότητα, είναι ο αλγόριθμος-a ή alpha miner. Αν και δεν είναι ο πιο προσιτός αλγόριθμος ανακάλυψης σε εφαρμογές εξόρυξης διαδικασιών της σημερινής εποχής, προσφέρει ένα ικανοποιητικό υπόβαθρο για την κατανόηση και την ανάλυση των πιο προηγμένων αλγόριθμων [10]. Ο αλγόριθμος-a δίνει σημασία μόνο στη ροή της διαδικασίας, αγνοώντας τις βοηθητικές στήλες του αρχείου καταγραφής και λαμβάνοντας υπόψιν τη σειρά με την οποία εκτελούνται οι δραστηριότητες σε κάθε περίπτωση. Αυτό σημαίνει ότι επιχειρεί να ανιχνεύσει τη σχέση μεταξύ των δραστηριοτήτων, δίχως να υπολογίζει τη συχνότητα εμφάνισης τους σε κάθε περίπτωση. Ο ορισμός που θεμελιώνει ο van der Aalst [10] για τη συσχέτιση των δραστηριοτήτων είναι ο παρακάτω:

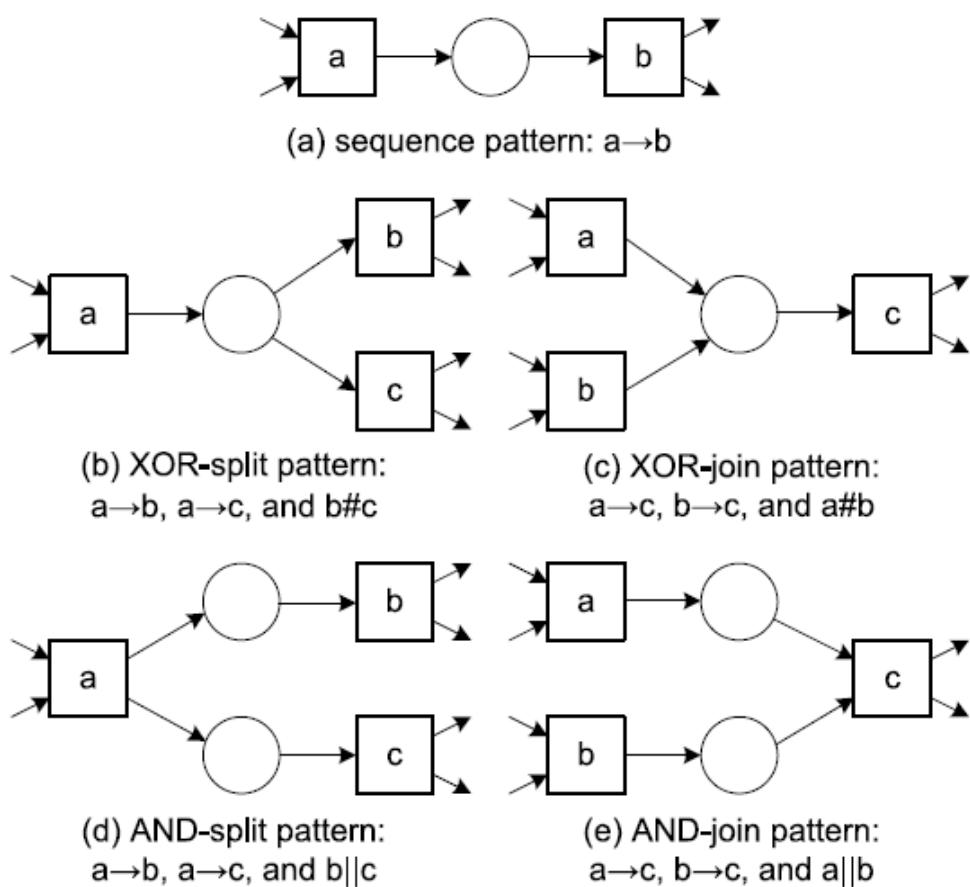
**Ορισμός 4.** Δούλευτος ενός αρχείου καταγραφής  $L$  πάνω στο  $\mathcal{A}$ , π.χ.  $L \in \mathbb{B}(\mathcal{A}^*)$ . Εστω  $a, b \in \mathcal{A}$ .

- $a >_L b$  αν και μόνο αν υπάρχει ένα ίχνος  $s = \langle t_1, t_2, t_3, \dots, t_n \rangle$  και  $i \in \{1, \dots, n-1\}$  τέτοιο ώστε  $s \in L$  και  $t_i = a$  και  $t_{i+1} = b$ ,
- $a \rightarrow_L b$  αν και μόνο αν  $a >_L b$  και  $b \not>_L a$ ,
- $a \#_L b$  αν και μόνο αν  $a \not>_L b$  και  $b \not>_L a$ ,
- $a \parallel_L b$  αν και μόνο αν  $a >_L b$  και  $b >_L a$ .

Για να γίνει κατανοητός ο ορισμός, αν υποθέσουμε ότι  $L = [\langle a, b, c, d \rangle^3, \langle a, c, b, d \rangle^4, \langle a, b, c, e, f, b, c, d \rangle^2, \langle a, b, c, e, f, c, b, d \rangle, \langle a, c, b, e, f, b, c, d \rangle^2, \langle a, c, b, e, f, b, c, e, f, c, b, d \rangle]$  το αποτύπωμα του αρχείου καταγραφής φαίνεται στο σχήμα 3.6. Το αποτύπωμα παρουσιάζεται ως ένας πίνακας και έχει μέγεθος  $n * n$ , όπου  $n$  το πλήθος των διακεχριμένων δραστηριοτήτων στο  $L$ . Το σχήμα 3.7 δείχνει τη γραφική αναπαράσταση της σχέσης για κάθε σύμβολο συσχέτισης του πίνακα.

<i>a</i>	<i>b</i>	<i>c</i>	<i>d</i>	<i>e</i>	<i>f</i>
<i>a</i>	#	→	→	#	#
<i>b</i>	←	#		→	→
<i>c</i>	←		#	→	←
<i>d</i>	#	←	←	#	#
<i>e</i>	#	←	←	#	→
<i>f</i>	#	→	→	#	←

**Σχήμα 3.6:** Αποτύπωμα του  $L$



**Σχήμα 3.7:** Σχέσεις μεταξύ των μεταβάσεων [10]

Με βάση την παραπάνω ανάλυση, ο ορισμός που θεμελιώνει ο van der Aalst [10] για τον αλγόριθμο-α είναι ο εξής:

**Ορισμός 5.** Έστω  $L$  ένα αρχείο καταγραφής στο  $T \subseteq \mathcal{A}$ . Τότε το  $a(L)$  ορίζεται ακολούθως:

1.  $T_L = \{t \in T \mid \exists \sigma_t \in \sigma\},$
2.  $T_I = \{t \in T \mid \exists \sigma_t = first(\sigma)\},$
3.  $T_O = \{t \in T \mid \exists \sigma_t = last(\sigma)\},$
4.  $X_L = \{(A, B) \mid A \subseteq T_L \wedge A \neq \emptyset \wedge B \subseteq T_L \wedge B \neq \emptyset \wedge \forall a \in A \ \forall b \in B (a \rightarrow_L b \wedge \forall a_1, a_2 \in A (a_1 \neq a_2 \Rightarrow \forall b_1, b_2 \in B (b_1 \neq b_2))),$
5.  $Y_L = \{(A, B) \in X_L \mid (\forall a', b' \in X_L A \subseteq A' \wedge B \subseteq B' \Rightarrow (A, B) = (A', B'))\},$
6.  $P_L = \{(p_{A,B}) \mid (A, B) \in Y_L\} \cup \{(i_L, o_L)\},$
7.  $F_L = \{(a, p_{A,B}) \mid (A, B) \in Y_L \wedge a \in A\} \cup \{(p_{A,B}, b) \mid (A, B) \in Y_L \wedge b \in B\} \cup \{(i_L, t) \mid t \in T_L\} \cup \{(t, o_L) \mid t \in T_O\},$
8.  $\alpha(L) = (P_L, T_L, F_L).$

Κάθε βήμα της εφαρμογής του αλγόριθμου-α σε ένα αρχείο καταγραφής ερμηνεύεται ως εξής:

1. Εντοπίζει όλες τις διακεχριμένες δραστηριότητες.
2. Εντοπίζει τις αρχικές δραστηριότητες.
3. Εντοπίζει τις τελικές δραστηριότητες.
4. Κατασκευάζει τον πίνακα συσχετίσεων μεταξύ των δραστηριοτήτων, ανακαλύπτοντας τα μοτίβα. Για την ανακάλυψη των μοτίβων κατασκευάζονται δύο σύνολα  $A$  και  $B$ , όπου  $A$  είναι η μετάβαση εισόδου και  $B$  η μετάβαση εξόδου.
5. Διαγράφει όλα τα μη μέγιστα (non-maximum) σύνολα που βρέθηκαν στο προηγούμενο βήμα.
6. Δημιουργεί ένα σύνολο τόπων για όλα σύνολα που προέκυψαν και έπειτα κατασκευάζει τον αρχικό και τον τελικό τόπο.
7. Καθορίζει την κατεύθυνση των ακμών.
8. Κατασκευάζει το τελικό petri net.

Πρακτικά, ο αλγόριθμος-α φτιάχνει το αποτύπωμα του αρχείου καταγραφής, δηλαδή ένα πίνακα από άμεσες, ακολουθιώκες, παράλληλες και επιλεκτικές συσχετίσεις μεταξύ των δραστηριοτήτων του, τις οποίες χρησιμοποιεί για να κατασκευάσει το τελικό petri net.

Ωστόσο, ο αλγόριθμος-α παρ' ότι είναι ένας αξιόπιστος και αποδοτικός αλγόριθμος ανακάλυψης, παρουσιάζει κάποια σοβαρά μειονεκτήματα. Το κύριο μειονέκτημα, όπως αναφέρθηκε και στον ορισμό είναι ότι δεν λαμβάνει υπόψιν τη συχνότητα με αποτέλεσμα να οδηγεί σε πολύ πυκνά μοντέλα από τα οποία είναι δύσκολο και χρονοβόρο να εξαχθούν συμπεράσματα. Ένα ακόμα μειονέκτημα του είναι η προβληματική συμπεριφορά σε ανίχνευση βρόγχων μήκους 2 και κάτω και για το λόγο αυτό ο αλγόριθμος έχει αποκτήσει κάποιες παραλλαγές όπως τον alpha+ miner ο οποίος μπορεί τους αντικείμενους επαρκώς. Τέλος, ο αλγόριθμος-α δεν εγγυάται την εγγυάται την ορθότητα (soundness) του μοντέλου [10].

### 3.4.2 Άλλοι αλγόριθμοι

Στην πρακτική εφαρμογή της ανακάλυψης διαδικασιών σε ένα αρχείο καταγραφής που αποτελείται από χιλιάδες δραστηριότητες και αλληλουχίες μεταξύ τους, δεν ενδιαφερόμαστε τόσο για κάθιε πιθανή συμπεριφορά που παρουσιάζει το αρχείο, παρά μόνο στην πιο συνηθισμένη. Χρειάζεται να απομονώσουμε το θόρυβο (τη σπάνια συμπεριφορά) και εξαιτίας αυτού ο αλγόριθμος-α, ο οποίος ανακαλύπτει την πλήρη συμπεριφορά, αλλά χωρίς να υπολογίζει τη συχνότητα, δεν είναι ιδανικός. Παρότι ένας από τους σκοπούς της εξόρυξης διαδικασιών είναι και η ανακάλυψη τέτοιας συμπεριφοράς, κάποιες φορές θέλουμε να έχουμε μια πιο συνολική εικόνα της διαδικασίας. Για να συλλάβουμε τη συνολική εικόνα της διαδικασίας, μπορούμε να χρησιμοποιήσουμε ορισμένους αλγόριθμους που το πετυχαίνουν καλύτερα και συνοψίζονται παρακάτω.

Ο heuristics miner είναι ένας αλγόριθμος ανακάλυψης που λαμβάνει υπόψιν το τη σειρά αλλά και τη συχνότητα των διαδικασιών, για την ανακάλυψη του μοντέλου. Με αυτό τον τρόπο καταφέρνει να απομονώσει το θόρυβο και να παράγει ένα πιο κατανοητό μοντέλο, που εστιάζει στη συνολική συμπεριφορά του αρχείου [10]. Αυτή η προσαρμοστικότητα του αλγόριθμου στα δεδομένα, προκύπτει από τη χρήση μετρικών όπως η καταλληλότητα (fitness), η ακρίβεια (precision), η γενίκευση (generalization) και η απλότητα (simplicity). Εντοπίζοντας τη συχνότητα που εμφανίζονται οι δραστηριότητες, ο αλγόριθμος μπορεί να αναπαρασταθεί ως directly-follows graph ή heuristic net, το οποίο κάνει ορατή την συχνότητα των δραστηριοτήτων οδηγώντας σε ένα πιο εύρωστο αλλά λιγότερο ακριβές μοντέλο.

Ο inductive miner είναι ένας αλγόριθμος ανακάλυψης που λειτουργεί κόβωντας το αρχείο σε σε υπο-αρχεία. Αυτό που επιχειρεί μέσα από τις περικοπές σε μικρότερα αρχεία, είναι να ανακαλύψει τη συσχέτιση αλλά και την αλληλουχία των δραστηριοτήτων. Μετά την περικοπή των αρχείων, εντοπίζει τις συσχετίσεις και τις ομαδοποιεί σε 4 κατηγορίες, sequence, exclusive or, parallel και redo loop [5]. Είναι αρκετά αποδοτικός

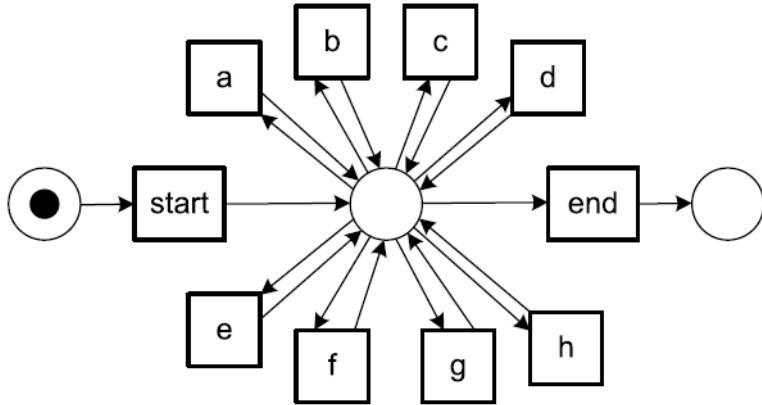
στο να βρίσκει το είδος συσχέτισης μεταξύ των δραστηριοτήτων.

Στο πλαίσιο αυτό, υπάρχουν και δεκάδες άλλοι αλγόριθμοι ανακάλυψης που χρησιμοποιούνται σε διαφορετικά είδη αρχείων καταγραφής και οι οποίοι θεμελιώνονται από τους αλγόριθμους που αναλύθηκαν προηγουμένως. Ενδεικτικά αναφέρουμε τους fuzzy miner, genetic miner, region-based miner, split miner, correlation miner και ILP miner [10]. Οι αλγόριθμοι που χρησιμοποιούνται για να ανακαλύψουν τη διαδικασία, δεν περιορίζουν το είδος των μοντέλων στο οποίο αυτή μπορεί να αναπαρασταθεί. Αυτό σημαίνει ότι, όλοι οι παραπάνω αλγόριθμοι ανακάλυψης μπορούν να οδηγήσουν σε μοντέλα όπως petri nets, process trees ή causal nets.

### 3.4.3 Μετρικές ποιότητας

Ο προσδιορισμός του αποτελέσματος ενός αλγόριθμου ανακάλυψης είναι ένα δύσκολο και πολυδιάστατο έργο. Για να μπορέσει να μετρηθεί το κατά πόσο το μοντέλο που ανακάλυψε ο αλγόριθμος αναπαριστά καλά τη διαδικασία, υπάρχουν 4 μετρικές ποιότητας: καταλληλότητα (fitness), ακρίβεια (precision), γενίκευση (generalization) και απλότητα (simplicity) [10]. Η καταλληλότητα μετράει το βαθμό στον οποίο το μοντέλο της διαδικασίας επαληθεύει την παρατηρούμενη συμπεριφορά. Ένας υψηλός δείκτης, σημαίνει ότι το μοντέλο μπορεί να μιμηθεί με ακρίβεια το αρχείο καταγραφής και αυτό είναι κρίσιμο για την κατανόηση και βελτίωση της διαδικασίας μέσα από το μοντέλο. Αντίθετα, η ακρίβεια εκτιμά το πόση πρόσθετη συμπεριφορά (πέραν αυτής που παρατηρείται στο αρχείο καταγραφής) επιτρέπει το μοντέλο. Η μεγάλη ακρίβεια υποδηλώνει ότι το μοντέλο δεν είναι γενικό, δηλαδή δεν επιτρέπει πολλή πρόσθετη συμπεριφορά, κάτι που φανερώνει μια στενή σχέση μοντέλου και αρχείου καταγραφής. Αν ένα μοντέλο επιτρέπει πρόσθετη συμπεριφορά, τότε υπάρχουν μονοπάτια δραστηριοτήτων τα οποία δεν υπάρχουν στο αρχείο καταγραφής αλλά είναι εφικτά στο μοντέλο. Η γενίκευση είναι η ικανότητα του μοντέλου να μπορεί να αναπαριστά μελλοντική συμπεριφορά της διαδικασίας. Ένα μοντέλο με μεγάλη γενίκευση, αποφεύγει την προσκόλληση στο αρχείο καταγραφής και προσαρμόζεται σε παραλλαγές της διαδικασίας με το πέρασμα του χρόνου, κάτι που είναι χρήσιμο για ένα μοντέλο που πρέπει να συνεχίσει να ισχύει όσο η διαδικασία εξελίσσεται. Η τελευταία μετρική σχετίζεται με το πόσο σύνθετο είναι το μοντέλο, δηλαδή το πλήθος των ακμών και τον κόμβων στην αναπαράσταση του.

Η ύπαρξη των παραπάνω μετρικών δίνει μια επιπλέον εξήγηση στο γιατί χρησιμοποιούνται διαφορετικοί τύποι αλγόριθμων ανακάλυψης σε κάθε αρχείο καταγραφής. Οι αλγόριθμοι ανακάλυψης, πρέπει να οδηγούν σε μοντέλα με υψηλές μετρικές αλλά παράλληλα πρέπει και να διατηρούν την ισορροπία μεταξύ τους. Για παράδειγμα, ένα overfitting και under-precise μοντέλο (σχήμα 3.8) επαληθεύει τέλεια το αρχείο καταγραφής, αλλά στο μεταξύ επιτρέπει και οποιαδήποτε άλλη συμπεριφορά που δεν είναι καταγεγραμμένη, με αποτέλεσμα να μην είναι ιδανικό για την αναπαράσταση της διαδικασίας.



**Σχήμα 3.8:** Workflow net που επαληθεύει κάθε αρχείο καταγραφής με δραστηριότητες  $\{a, b, \dots, h\}$  [10]

### 3.5 Έλεγχος συμμόρφωσης

Ος έλεγχος συμμόρφωσης χαρακτηρίζεται η αντιπαράθεση της παρατηρούμενης συμπεριφοράς ή δεδομένων (αρχεία καταγραφής) με τη μοντελοποιημένη συμπεριφορά (μοντέλα διαδικασιών). Αυτή η αντιπαράθεση είναι ζωτικής σημασίας, διότι μέσα από αυτή γίνεται κατανοητός ο συνολικός τρόπος συμπεριφοράς της διαδικασίας, αλλά είναι και επιτακτική για την ευθυγράμμιση με το στόχο της. Από αυτή την οπτική, το νόημα του ελέγχου συμμόρφωσης είναι η εύρεση των ομοιοτήτων και διαφορών των μοντέλων διαδικασιών με τα αρχεία καταγραφής, κάτι που είναι βοηθά στον εντοπισμό αποκλίσεων από τον πραγματικό σκοπό της. Αυτές οι αποκλίσεις μπορεί να υποδεικνύουν αναποτελεσματικότητα στη διαδικασία, αλλά μπορεί να υποδεικνύουν και αδυναμίες στη λειτουργία των αλγόριθμων ανακάλυψης να μοντελοποιήσουν ορθώς τη συμπεριφορά. Οι δυνατότητες που παρέχονται μέσω του ελέγχου συμμόρφωσης, συμπεριλαμβάνουν την εκτίμηση του κανονιστικού μοντέλου μέσω μετρικών ποιότητας, την επιδιόρθωση τους μέσω της γνώσης από τα δεδομένα συμβάντων και την ανίχνευση προβλημάτων στη ροή της διαδικασίας [10].

Ο έλεγχος συμμόρφωσης χωρίζεται σε δύο βασικές κατηγορίες, τις καθολικές μετρήσεις συμμόρφωσης (global conformance measures) και τα τοπικά διαγνωστικά (local diagnostics). Η πρώτη κατηγορία μετράει την καταλληλότητα του μοντέλου, δηλαδή σε τι βαθμό η παρατηρούμενη συμπεριφορά επαληθεύει το μοντέλο. Η δεύτερη κατηγορία αφορά λεπτομέρειες που σχετίζονται με συγκεκριμένες δραστηριότητες του μοντέλου, στις οποίες οι διαφορές μεταξύ μοντέλου και αρχείου καταγραφής γίνονται ορατές. Αυτό μπορεί να σημαίνει ότι μια δραστηριότητα στην πραγματικότητα εκτελέστηκε παραπάνω φορές από ότι επιτρέπεται. Η ύπαρξη της συμμόρφωσης, είναι σχετικά υποκειμενική και

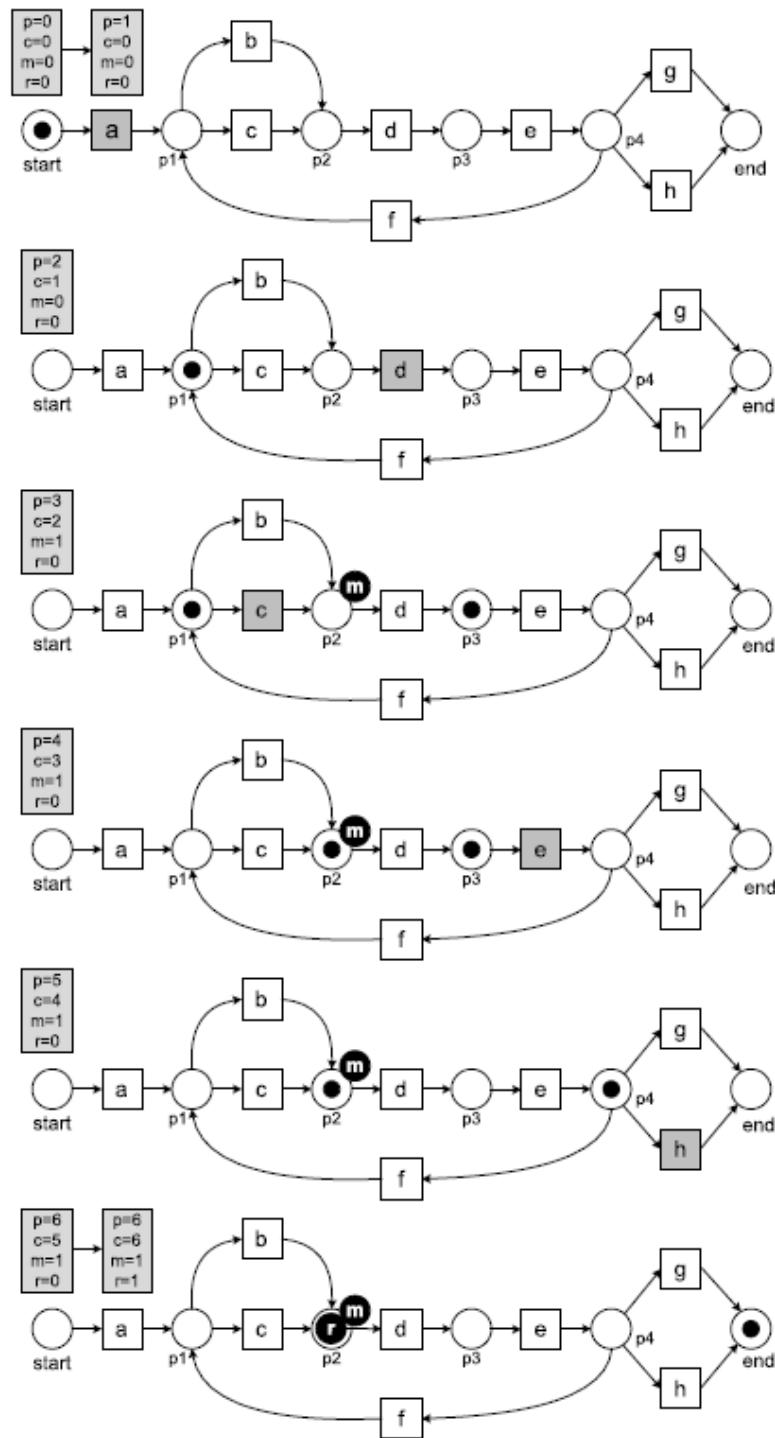
διαφοροποιείται ανάλογα με την οπτική της συμμόρφωσης, δηλαδή αν το μοντέλο είναι λάθος και δεν αναπαριστά την πραγματικότητα ή αν τα δεδομένα αποκλίνουν από το μοντέλο. Η καταλληλότητα του μοντέλου χυμαίνεται από το 0 (το αρχείο καταγραφής δεν σχετίζεται καθόλου με το μοντέλο) έως το 1 (το αρχείο καταγραφής σχετίζεται τέλεια με το μοντέλο). Για τον έλεγχο συμμόρφωσης υπάρχουν πολλές τεχνικές, ωστόσο οι δύο πιο γνωστές τεχνικές είναι η ευθυγράμμιση (alignments) και η επανάληψη σημείων (token-based replay).

### 3.5.1 Επανάληψη σημείων

Η επανάληψη σημείων (token-based replay) είναι μια θεμελιώδης τεχνική της εξόρυξης διαδικασιών, που ανήκει στην πρώτη κατηγορία τεχνικών ελέγχου συμμόρφωσης. Αυτή η τεχνική βασίζεται σε στην επαναληπτική προσομοίωση της ροής σημείων (tokens) πάνω σε ένα workflow net, με σκοπό να εντοπιστεί αν τα δεδομένα το αντιπροσωπεύουν. Το workflow net είναι πρακτικά μια παραλλαγή του petri net, στο οποίο ορίζεται ένας ένας αρχικός (start) και ένας τελικός (end) τόπος. Για το σκοπό αυτό, χρησιμοποιούνται 4 μεταβλητές σε κάθε στάδιο της επανάληψης:

- παραχθέντα σημεία (produced tokens) -  $p$
- καταναλωμένα σημεία (consumed tokens) -  $c$
- υπολειπόμενα σημεία (missing tokens) -  $m$
- εναπομείναντα σημεία (remaining tokens) -  $r$

Αρχικά, οι τόποι (places) στο workflow net είναι άδειοι και οι μεταβλητές ισούνται με 0. Τα σημεία εμφανίζονται (ή εξαφανίζονται) με την καταγραφή ενός συμβάντος στο αρχείο. Το πρώτο σημείο εμφανίζεται στον αρχικό τόπο του workflow net και το  $p$  αυξάνεται κατά 1. Μετά από την πυροδότηση της μετάβασης, το  $p$  μηδενίζεται και αυξάνεται το  $c$  ανάλογα τοποθετώντας ένα σημείο σε κάθε τόπο μετά τη μετάβαση αυτή. Αν κάποιο σημείο λείπει από έναν τόπο εισόδου και μια μετάβαση δεν μπορεί αν πυροδοτηθεί, τότε αυτό εισάγεται στο workflow net και αυξάνεται το  $m$  ανάλογα. Το σημείο που εισάγεται εξωτερικά στο workflow net μένει στη θέση του και μετά την πυροδότηση της μετάβασης. Αν στην τελική κατάσταση του workflow net υπάρχουν υπολειπόμενα σημεία αυτά προστίθενται στο  $r$ , ενώ αν μετά το πέρας της διαδικασίας ισχύει  $m = r = 0$ , τότε το ίχνος επαναλήφθηκε άφογα πάνω στο μοντέλο. Αντίστοιχα, όσο μεγαλύτερα τα  $m, r$  τόσο χαμηλότερο το fitness του ίχνους. Στο σχήμα 3.9 φαίνεται ένα παράδειγμα εφαρμογής του token-based replay αλγόριθμου στο ίχνος  $\sigma_1 = \langle a, d, c, e, h \rangle$  για το παρακάτω workflow net  $N_1$ .



**Σχήμα 3.9:** Token-based replay του  $\sigma_1$  στο  $N_1$  [10]

Στο παραπάνω σχήμα βλέπουμε αναλυτικά την επανάληψη του ίχνους στο μοντέλο, καθώς και τις τιμές των μεταβλητών μετά το πέρας της επανάληψης. Η καταλληλότητα της παραπάνω περίπτωσης με ίχνος  $\sigma_1 = \langle a, d, c, e, h \rangle$  δίνεται με αντικατάσταση από τη σχέση 3.1.

$$fitness(\sigma_1, N_1) = \frac{1}{2} \left( 1 - \frac{m}{c} \right) + \frac{1}{2} \left( 1 - \frac{r}{p} \right) = \frac{1}{2} \left( 1 - \frac{1}{6} \right) + \frac{1}{2} \left( 1 - \frac{1}{6} \right) = \frac{10}{12} \simeq 0.83 \quad (3.1)$$

Η συνολική καταλληλότητα του αρχείου καταγραφής δίνεται από τη σχέση 3.2.

$$fitness(L, N) = \frac{1}{2} \left( 1 - \frac{\sum_{\sigma \in L} L(\sigma) \times m_{N,\sigma}}{\sum_{\sigma \in L} L(\sigma) \times c_{N,\sigma}} \right) + \frac{1}{2} \left( 1 - \frac{\sum_{\sigma \in L} L(\sigma) \times r_{N,\sigma}}{\sum_{\sigma \in L} L(\sigma) \times p_{N,\sigma}} \right) \quad (3.2)$$

Η επανάληψη σημείων (token-based replay) είναι ένας απλός και αποδοτικός αλγόριθμος ελέγχου συμμόρφωσης, που επιτρέπει μια αρχική διάγνωση του αρχείου καταγραφής, ωστόσο σε αρχεία καταγραφής με πολλές παραλλαγές της διαδικασίας μπορεί να προκαλέσει υπερχείλιση tokens με αποτέλεσμα να επιτρέπει οποιαδήποτε συμπεριφορά. Ένα άλλο μειονέκτημα, είναι ότι εφαρμόζεται μόνο σε petri nets ή petri net based μοντέλα (όπως το workflow net) [10].

### 3.5.2 Ευθυγράμμιση

Η ευθυγράμμιση (alignments) είναι μια τεχνική που στοχεύει στο να παρακάμψει τα εμπόδια της τεχνικής token-based replay, επιχειρώντας μια αντιστοίχηση της συμπεριφοράς του αρχείου καταγραφής στο μοντέλο της διαδικασίας [10]. Η τεχνική αυτή επιτρέπει τον εντοπισμό αποκλίσεων σε δραστηριότητες που δεν έχουν μοντελοποιηθεί ή μοντελοποιούνται διαφορετικά, δημιουργώντας μια ακολουθία κινήσεων (moves) που αντιπροσωπεύει την προσαρμογή του αρχείου καταγραφής στο μοντέλο. Σε σύγχριση με τον token-based replay αλγόριθμο, η ευθυγράμμιση εφαρμόζεται σε όλες τις σημειογραφίες μοντέλων διαδικασιών που περιέχουν αρχική και τελική κατάσταση, ενώ παράλληλα δίνει πιο ακριβή και κατανοητά διαγνωστικά. Ας υποθέσουμε ένα ίχνος  $\sigma_2 = \langle a, d, b, e, h \rangle$  του οποίου η ευθυγράμμιση φαίνεται στον πίνακα 3.3.

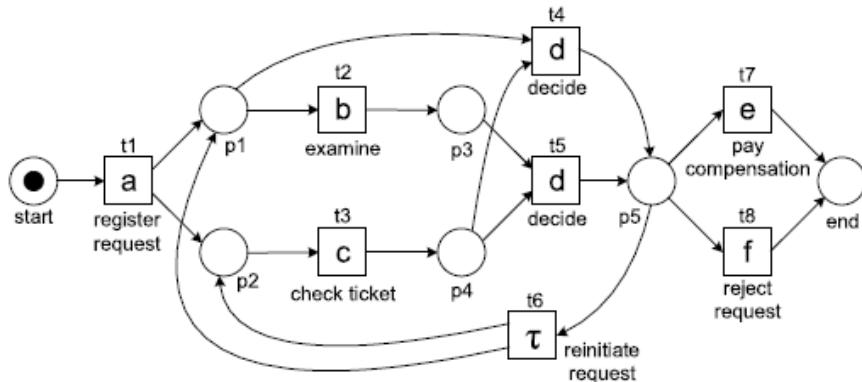
$$\gamma_{21} = \begin{array}{|c c c c c|} \hline & a & d & b & e & h \\ \hline a & & d & b & e & h \\ \hline \end{array} \quad (3.3)$$

Η πάνω γραμμή αναφέρεται στο ίχνος, ενώ η κάτω γραμμή στο μοντέλο. Όταν αυτές ταυτίζονται έχουμε μια ιδανική ευθυγράμμιση, ενώ όταν δεν ταυτίζονται τότε προκύπτει ένας πίνακας όπως ο 3.4.

$$\gamma_{22} = \begin{array}{|c c c c c|} \hline & a & \gg & d & b & e & h \\ \hline a & b & d & \gg & e & h \\ \hline \end{array} \quad (3.4)$$

Το σύμβολο  $\gg$  (no move), αναφέρεται στην κακή ευθυγράμμιση (misalignment) και σημαίνει ότι η κίνηση δεν ακολουθείται από το αρχείο ή το μοντέλο, ανάλογα με το αν το σύμβολο είναι στην πάνω ή στην κάτω γραμμή. Αν είναι πάνω, σημαίνει ότι την κίνηση στο μοντέλο δε μπορεί να τη μιμηθεί το αρχείο καταγραφής και αντίστοιχα αν είναι κάτω, σημαίνει ότι την κίνηση στο αρχείο καταγραφής δε μπορεί να τη μιμηθεί το μοντέλο. Συγκεκριμένα στον παραπάνω πίνακα, βλέπουμε ότι το μοντέλο κάνει μια  $b$  κίνηση πριν τη  $d$  που δεν υπάρχει στο ίχνος, ενώ πιο μετά το ίχνος κάνει μια  $b$  κίνηση που δεν είναι εφικτή στο μοντέλο.

Η τεχνική αυτή έρχεται επίσης να λύσει το πρόβλημα των όμοιων (duplicate) και αόρατων (silent) μεταβάσεων, βάζοντας επιγραφές (labels) σε κάθε μετάβαση. Πιο συγκεκριμένα, με τις επιγραφές καταφέρνει να αναγνωρίζει μεταβάσεις οι οποίες στο μοντέλο παρουσιάζονται ως ξεχωριστές αλλά έχουν την ίδια ακριβώς λειτουργία ή και μεταβάσεις οι οποίες απλώς γυρνούν την διαδικασία σε μια προηγούμενη κατάσταση. Για παράδειγμα ας υποθέσουμε το workflow net  $N_2$  στο σχήμα 3.10 και το ίχνος  $\sigma_3 = \langle a, c, d, b, c, d, c, d, c, b, d, f \rangle$ .



Σχήμα 3.10: Workflow net  $N_2$  [10]

Στο  $N_2$ , υπάρχουν δύο μεταβάσεις με επιγραφή  $\text{decide}$ , οι  $t_4, t_5$  όπου έχουν την ίδια λειτουργία και οι οποίες δεν συνδέονται με κάποια συγκεκριμένη μετάβαση (δηλαδή το σημείο μετά μπορεί να καταλήξει στο  $e$  ή το  $f$ ). Επίσης υπάρχει μια αόρατη μετάβαση, δηλαδή μια μετάβαση η οποία ξαναρχίζει τη διαδικασία από ένα προηγούμενο τόπο. Στον πίνακα 3.5, βλέπουμε ότι παρόλο που η συμπεριφορά του αρχείου καταγραφής δεν ταυτίζεται απόλυτα με το μοντέλο, η ευθυγράμμιση θεωρείται ιδανική αφού τα  $\gg$  moves που υπάρχουν, αφορούν αόρατες μεταβάσεις και μεταβάσεις απόφασης όπου λογίζονται ως ταυτόσημες, επειδή οδηγούν στον ίδιο τόπο.

$\gamma_3 =$	<table border="1"> <tbody> <tr> <td><math>a</math></td><td><math>c</math></td><td><math>d</math></td><td><math>\gg</math></td><td><math>b</math></td><td><math>c</math></td><td><math>d</math></td><td><math>\gg</math></td><td><math>c</math></td><td><math>d</math></td><td><math>\gg</math></td><td><math>c</math></td><td><math>b</math></td><td><math>d</math></td><td><math>f</math></td></tr> <tr> <td><math>a</math></td><td><math>c</math></td><td><math>d</math></td><td><math>\tau</math></td><td><math>b</math></td><td><math>c</math></td><td><math>d</math></td><td><math>\tau</math></td><td><math>c</math></td><td><math>d</math></td><td><math>\tau</math></td><td><math>c</math></td><td><math>b</math></td><td><math>d</math></td><td><math>f</math></td></tr> <tr> <td><math>t_1</math></td><td><math>t_3</math></td><td><math>t_4</math></td><td><math>t_6</math></td><td><math>t_2</math></td><td><math>t_3</math></td><td><math>t_5</math></td><td><math>t_6</math></td><td><math>t_3</math></td><td><math>t_4</math></td><td><math>t_6</math></td><td><math>t_3</math></td><td><math>t_2</math></td><td><math>t_5</math></td><td><math>t_8</math></td></tr> </tbody> </table>	$a$	$c$	$d$	$\gg$	$b$	$c$	$d$	$\gg$	$c$	$d$	$\gg$	$c$	$b$	$d$	$f$	$a$	$c$	$d$	$\tau$	$b$	$c$	$d$	$\tau$	$c$	$d$	$\tau$	$c$	$b$	$d$	$f$	$t_1$	$t_3$	$t_4$	$t_6$	$t_2$	$t_3$	$t_5$	$t_6$	$t_3$	$t_4$	$t_6$	$t_3$	$t_2$	$t_5$	$t_8$	(3.5)
$a$	$c$	$d$	$\gg$	$b$	$c$	$d$	$\gg$	$c$	$d$	$\gg$	$c$	$b$	$d$	$f$																																	
$a$	$c$	$d$	$\tau$	$b$	$c$	$d$	$\tau$	$c$	$d$	$\tau$	$c$	$b$	$d$	$f$																																	
$t_1$	$t_3$	$t_4$	$t_6$	$t_2$	$t_3$	$t_5$	$t_6$	$t_3$	$t_4$	$t_6$	$t_3$	$t_2$	$t_5$	$t_8$																																	

Με βάση τα παραπάνω, η ευθυγράμμιση ορίζει ότι η κίνηση  $(x, (y, t))$  είναι μια έγκυρη κίνηση όταν ισχύει ένα από αυτά [10]:

- συγχρονισμένη κίνηση  $x = y$  όπου το  $y$  είναι μια ορατή επιγραφή της μετάβασης  $t$ ,
- ορατή κίνηση μοντέλου  $x \ggg$  και το  $y$  είναι μια ορατή επιγραφή της μετάβασης  $t$ ,
- αόρατη κίνηση μοντέλου  $x \ggg$  και το  $y = t$  όπου  $t$  αόρατη μετάβαση,
- κίνηση αρχείου καταγραφής  $x \neq \ggg$  και  $(y, t) = \ggg$

Η καταλληλότητα του αρχείου καταγραφής δίνεται από τον τύπο 3.6.

$$fitness(\sigma, N) = 1 - \frac{\delta(\lambda_{opt}^N(\sigma))}{\delta(\lambda_{worst}^N(\sigma))} \quad (3.6)$$

Στον παραπάνω τύπο, το  $\delta$  αφορά τη συνάρτηση κόστους, στην οποία η ευθυγράμμιση είναι ιδανική (optimal) αν δεν υπάρχει κάποια ευθυγράμμιση με μικρότερο κόστος, ενώ είναι χείριστη (worst) αν δεν υπάρχουν συγχρονισμένες ή αόρατες κινήσεις, παρά μόνο κινήσεις στο μοντέλο ή το αρχείο. Για παράδειγμα το ίχνος  $\sigma_4 = \langle a, b, d, f \rangle$  στο  $N_2$  έχει τη χείριστη ευθυγράμμιση του πίνακα 3.7.

$$\lambda_{worst}^{N_2}(\sigma_4) = \begin{array}{ccccccccc} a & b & d & f & \gg & \gg & \gg & \gg \\ \gg & \gg & \gg & \gg & a & c & d & f \\ & & & & t1 & t3 & t4 & t8 \end{array} \quad (3.7)$$

Ταυτόχρονα, η ιδανική ευθυγράμμιση φαίνεται στον πίνακα 3.8.

$$\lambda_{opt}^{N_2}(\sigma_4) = \begin{array}{ccccc} a & \gg & b & d & f \\ a & c & b & d & f \\ t1 & t3 & t2 & t5 & t8 \end{array} \quad (3.8)$$

Εδώ, το κόστος της ιδανικής ευθυγράμμισης ισούται με 1 επειδή περιέχει μόνο 1 κίνηση στο αρχείο, ενώ το κόστος της χείριστης ευθυγράμμισης ισούται με 8 επειδή περιέχει 8 κινήσεις αρχείου και μοντέλου. Κατά συνέπεια, ο υπολογισμός της καταλληλότητας του ίχνους προκύπτει από τη σχέση 3.9.

$$fitness(\sigma_4, N_2) = 1 - \frac{\delta(\lambda_{opt}^{N_2}(\sigma_4))}{\delta(\lambda_{worst}^{N_2}(\sigma_4))} = 1 - \frac{1}{8} = 0.875 \quad (3.9)$$

## 3.6 Εργαλεία

### 3.6.1 Disco

Το Disco<sup>3</sup> είναι ένα εμπορικό λογισμικό που δημιουργήθηκε το 2009 από μέλη της ομάδας εξόρυξης διαδικασιών του Technical University of Eindhoven και αναπτύσσεται έως σήμερα από τη Fluxicon. Το εργαλείο αυτό διατίθεται δωρεάν για ακαδημαϊκή χρήση σε περίπου 700 πανεπιστήμια μέσω της «Ακαδημαϊκής Πρωτοβουλίας Fluxicon». Είναι ένα εργαλείο που δημιουργήθηκε βασισμένο σε γνώμες επαγγελματιών του χώρου, που επιθυμούσαν ένα λογισμικό φιλικό, εύκολο και γρήγορο και παρακάμπτει αρκετά εμπόδια που αντιμετωπίζουν αρχάριοι της εξόρυξης διαδικασιών [1]. Το κύριο πλεονέκτημα του είναι η αυτοματοποιημένη ανακάλυψη διαδικασιών, χρησιμοποιώντας παράλληλα μια κατανοητή μορφή απεικόνισης χάρτη, ενώ ο αλγόριθμος ανακάλυψης που υλοποιεί είναι ο disco miner (μια παραλλαγή του fuzzy miner). Διαθέτει μια πληθύρα από φίλτρα που δίνουν τη δυνατότητα στην εστίαση σε συγκεκριμένα κομμάτια της διαδικασίας, καθώς και προβολή στατιστικών στοιχείων. Τέλος, επιτρέπει την ανάλυση της χρονικής ροής των δεδομένων και την απεικόνιση της διαδικασίας σε πραγματικό χρόνο.

### 3.6.2 PM4PY

Το PM4PY<sup>4</sup> είναι μια βιβλιοθήκη της Python που παρέχει μια μεγάλη συλλογή από εργαλεία που σχετίζονται με ανακάλυψη διαδικασιών και έλεγχο συμμόρφωσης. Πρόκειται για ένα λογισμικό ανοιχτού κώδικα το οποίο ενθαρρύνει την συμμετοχή της κοινότητας στην ανάπτυξη του, καλιεργώντας μια διαδραστική σχέση στο πεδίο της εξόρυξης διαδικασιών. Από το 2019 μέχρι σήμερα, πάνω από 1 εκατομμύριο χρήστες έχουν κατεβάσει τη βιβλιοθήκη, ενώ στο GitHub έχει πάνω από 500 stars και 235 forks, πράγμα που φανερώνει την ευρεία αποδοχή της κοινότητας [25]. Το εργαλείο αύτο καταφέρνει να διορίζεται ορισμένα μειονεκτήματα που τα προϋπάρχοντα λογισμικά εμφάνιζαν, αφού είναι εύκολα επεκτάσιμο, επιτρέπει την τροποποίηση των αλγορίθμων και συνίσταται για μεγάλης κλίμακας πειράματα [16]. Το PM4PY έχει ήδη χρησιμοποιηθεί σε εφαρμογές του πραγματικού κόσμου που σχετίζονται με τον τομέα υγείας, την οικονομία και την ασφάλεια υπολογιστικών συστημάτων. Το κύριο πλεονέκτημα του, είναι ότι υποστηρίζει ένα τεράστιο πλήθος αλγόριθμων ανακάλυψης όπως alpha, heuristics, inductive miner αλλά και ελέγχου συμμόρφωσης όπως token-based replay, alignments, ενώ έχει πλήρη συμβατότητα με κάθε τύπο αρχείων καταγραφής και μοντέλων, εκτελώντας παράλληλα και τις μετατροπές από το ένα στο άλλο. Τέλος, η συμβολή του στην ακαδημαϊκή έρευνα είναι σημαντική, αφού αποτελεί μια βάση για την εφεύρεση και βελτίωση των αλγόριθμων.

<sup>3</sup><https://fluxicon.com/disco/>

<sup>4</sup><https://pm4py.fit.fraunhofer.de/>

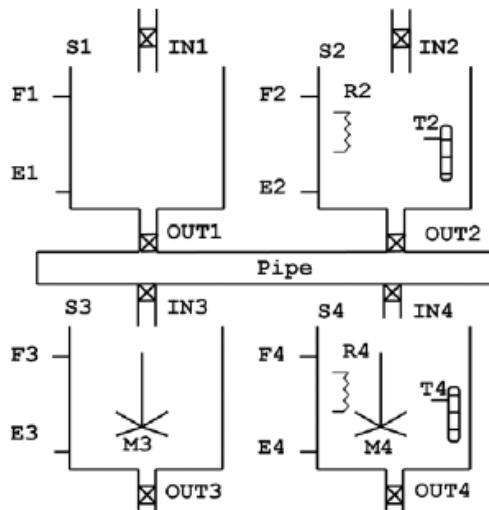
## Κεφάλαιο 4

### Εφαρμογή

## 4.1 Επισκόπηση Liqueur Plant

### 4.1.1 Περιγραφή συστήματος

Το Liqueur Plant<sup>1</sup> ή LPS είναι ένα σύστημα παραγωγής λικέρ, το οποίο περιστρέφεται γύρω από την ενοποίηση κυβερνοφυσικών στοιχείων με φυσικά στοιχεία για τη βελτιστοποίηση των διαδικασιών παραγωγής. Ο σχεδιασμός της κυβερνοφυσικής του αρχιτεκτονικής είναι βασισμένος στο πρότυπο IEC 61131 και έχει σχεδιαστεί για να μειώνει την πολυπλοκότητα και να αντιμετωπίζει τις ποικίλες προκλήσεις που υπάρχουν στην παραγωγή λικέρ, διασφαλίζοντας έτσι αποτελεσματικές και ελεγχόμενες λειτουργίες. Το σύστημα (σχήμα 4.1) παράγει δύο είδη λικέρ, τύπου A και τύπου B και απαρτίζεται από 4 σιλό τα οποία είναι συνδεδεμένα σε ένα κοινό αγωγό: τα σιλό s1 και s4 που εκτελούν την παραγωγή του λικέρ τύπου A και τα σιλό s2 και s3 που εκτελούν την παραγωγή του λικέρ τύπου B. Σε κάθε σιλό υπάρχουν δύο βαλβίδες και δύο αισθητήρες, ένας στο πάνω και ένας στο κάτω μέρος που ελέγχουν τη στάθμη του υγρού. Σε κάποια από τα σιλό υπάρχουν επίσης στοιχεία ανάδευσης, θέρμανσης καθώς και αισθητήρες θερμοκρασίας.



**Σχήμα 4.1:** Το φυσικό plant του συστήματος [6]

Η διαδικασία παραγωγής ξεκινά με τη θέρμανση ακατέργαστου υγρού στο σιλό s2 σε μια συγκεκριμένη θερμοκρασία και τη μεταφορά του στο σιλό s3, όπου υφίσταται ανάμειξη για ένα προκαθορισμένο χρόνο. Αντίστοιχα, η διαδικασία παραγωγής ξεκινά με την προ-επεξεργασία ακατέργαστου υγρού στο σιλό s1 και την μεταφορά του στο σιλό s4 για την ανάμειξη και θέρμανση του έως μια συγκεκριμένη θερμοκρασία. Αυτές οι

<sup>1</sup><https://sites.google.com/site/uml4iot/liqueur-plant-case-study>

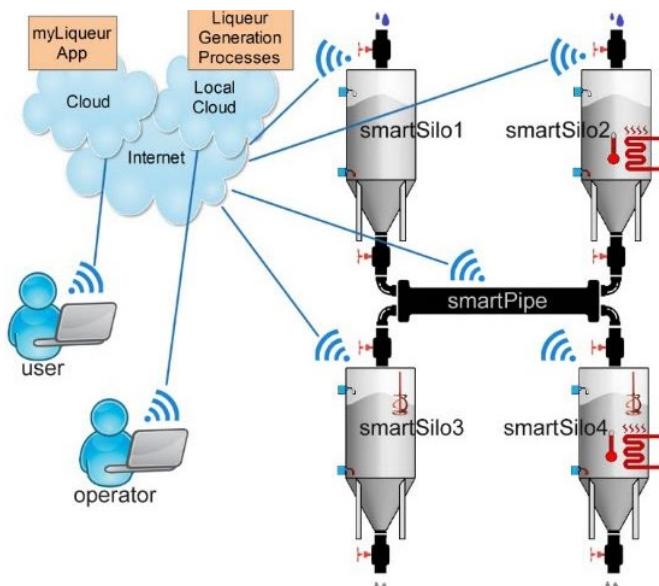
## Κεφάλαιο 4. Εφαρμογή

διαδικασίες παρότι είναι ανεξάρτητες, λειτουργούν παράλληλα και απαιτούν προσεκτικό συγχρονισμό λόγω του κοινόχρηστου αγωγού μεταφοράς του υγρού, κάνοντας έτσι εμφανή τον περίπλοκο σχεδιασμό του συστήματος ως προς την αποδοτική διαχείριση των ταυτόχρονων λειτουργιών. Σαν ένα πρόσθετο επίπεδο πολυπλοκότητας εισάγεται ο περιορισμός κατανάλωσης ενέργειας του συστήματος μέσω της απαγόρευσης ταυτόχρονης ανάδευσης στα σιλό s3 και s4, που απαιτεί έναν μηχανισμό ελέγχου για την αποτελεσματική διαχείριση της αποδοτικής χρήσης ενέργειας.

Τα στοιχεία εντός του συστήματος χωρίζονται σε κυβερνοφυσικά στοιχεία και φυσικά στοιχεία (σιλό, αγωγούς, αναδευτήρες και αισθητήρες), με την σχέση μεταξύ τους να είναι καλώς ορισμένη για να καλύψει το συνολικό φάσμα της λειτουργικότητας του. Οι λεπτομερείς προδιαγραφές αυτών των στοιχείων και οι αλληλεπιδράσεις τους ορίζουν την κατεύθυνση του συστήματος προς τη συνεργιστική ολοκλήρωση, στην ανάπτυξη βασισμένη σε οντότητες και του προσανατολισμού σε αντικείμενα. Με αυτό τον τρόπο αντικετωπίζονται οι άμεσες ανάγκες του συστήματος παραγωγής λικέρ και δημιουργείται ένα υπόβαθρο για τα σύγχρονα συστήματα βιομηχανικού αυτοματισμού, το οποίο αξιοποιεί τις σύγχρονες τάσεις στη μηχατρονική και τη μηχανική λογισμικού για να ανταποκριθεί στις προκλήσεις της πολυπλοκότητας του συστήματος και της ζήτησης για ανώτερης ποιότητας προϊόντα.

### 4.1.2 IoT εκδοχή συστήματος

Το myLiqueur [9] είναι μια IoT εφαρμογή που αξιοποιεί την τεχνολογία των κυβερνοφυσικών συστημάτων εστιάζοντας στην ενσωμάτωση έξυπνων εξαρτημάτων για την παραγωγή λικέρ. Το myLiqueur αποτελείται από κυβερνοφυσικά στοιχεία, συμπεριλαμβανομένων των έξυπνων σιλό (smartSilos) και του έξυπνου αγωγού (smartPipe). Τα εξαρτήματα αυτά εκτελούν συγκεκριμένες λειτουργίες όπως γέμισμα, άδειασμα, ανάμειξη και θέρμανση. Κάθε ένα από τα smartSilos είναι εξοπλισμένο με αισθητήρες και μηχανισμούς ελέγχου, ενώ το smartPipe ευθύνεται για τη μεταφορά των υγρών μεταξύ των smartSilos. Ο σχεδιασμός του συστήματος όπως φαίνεται στο σχήμα 4.2 ενσωματώνει τα εξης χαρακτηριστικά σε κάθε ένα από τα κυβερνοφυσικά στοιχεία: βαλβίδες εισόδου, βαλβίδες εξόδου, στοιχεία ανάδευσης και στοιχεία θέρμανσης.



**Σχήμα 4.2:** Αναπαράσταση του συστήματος myLiqueur [9]

Μια πιο αναλυτική περιγραφή της παραγωγή λικέρ τύπου A:

1. Γέμισμα του smartSilo1 με υγρό
2. Μίξη του υγρού με ύλες στο smartSilo1
3. Άδειασμα του smartSilo1 και γέμισμα του smartSilo4 μέσω του smartPipe
4. Θέρμανση έως μια προκαθορισμένη θερμοκρασία στο smartSilo4
5. Ανάδευση για έναν προκαθορισμένο χρόνο στο smartSilo4
6. Άδειασμα του smartSilo4

## Κεφάλαιο 4. Εφαρμογή

ενώ για την παραγωγή λικέρ τύπου B:

1. Γέμισμα του smartSilo2 με υγρό
2. Θέρμανση έως μια προκαθορισμένη θερμοκρασία στο smartSilo2
3. Άδειασμα του smartSilo2 και γέμισμα του smartSilo3 μέσω του smartPipe
4. Ανάδευση για ένα προκαθορισμένο χρόνο στο smartSilo3
5. Άδειασμα του smartSilo3

Μετά το άδειασμα των τελευταίων σιλό, το υγρό μεταφέρεται σε άλλο σημείο παραγωγής. Κάθε βήμα της διαδικασίας προβλέπεται να εκτελείται για συγκεκριμένο χρονικό διάστημα και σε συμμόρφωση με την επιτυχή ολοκλήρωση των προηγούμενων βημάτων. Οι χρήστες επικοινωνούν με το σύστημα μέσω της εφαρμογής myLiqueur, όπου μπορούν να καθορίσουν τον επιθυμητό τύπο λικέρ μέσω παραμέτρων. Το σύστημα δέχεται τις εισόδους του χρήστη για να προσαρμόσει τις παραμέτρους παραγωγής ανάλογα, και να εγγυηθεί ότι κάθε τύπος λικέρ ταιριάζει με τις καθορισμένες απαιτήσεις του χρήστη.

Το σύστημα παραγωγής myLiqueur ακολουθεί τους δύο βασικούς περιορισμούς όπως ορίζονται στο Liqueur Plant: την κοινή χρήση ενός κοινού αγωγού για τη μεταφορά υγρών μεταξύ των σιλό και τον περιορισμό της κατανάλωσης ενέργειας, ο οποίος απαγορεύει την ταυτόχρονη ανάμειξη στα σιλό. Αυτοί οι περιορισμοί απαιτούν μια άριστα συντονισμένη λειτουργία, που επιτυγχάνεται μέσω του έξυπνου σχεδιασμού του συστήματος και της ενσωμάτωσης τεχνολογιών IoT. Ένα πρωτότυπο του συστήματος έχει αναπτυχθεί και δοκιμαστεί με επιτυχία αποδεικνύοντας την αποτελεσματικότητα του παραπάνω σχεδιασμού. Το πρωτότυπο χρησιμοποιεί ένα Raspberry Pi για κάθε κυβερνοφυσικό στοιχείο, δείχνοντας την δυνατότητα του να ενσωματώνει κλασικά εξαρτήματα βιομηχανικού αυτοματισμού με σύγχρονες τεχνολογίες IoT.

## 4.2 Εξόρυξη διαδικασιών στο Liqueur Plant

Η εξόρυξη διαδικασιών στο LPS κινείται γύρω από τρεις βασικούς άξονες: την ανακάλυψη της διαδικασίας, τον έλεγχο συμμόρφωσης και τον εντοπισμό των σημείων συμφόρησης. Μέσα από τις παραπάνω τεχνικές, ωστε προσπαθήσουμε να αντλήσουμε αξιόπιστες πληροφορίες για το σύστημα, να ελέγξουμε αν το σύστημα ανταποκρίνεται βάσει της σχεδίασης του, να υπολογίσουμε την επιβάρυνση των περιορισμών του συστήματος στα σημεία συμφόρησης και να συγκρίνουμε το μοντέλο που προκύπτει με το προϋπάρχον μοντέλο του LPS. Για τους σκοπούς αυτούς, ωστε χρησιμοποιήσουμε τη γλώσσα Python (3.10.4) και πιο συγκεκριμένα τη βιβλιοθήκη pandas<sup>2</sup> (1.4.2) για την μορφοποίηση των αρχείων καταγραφής, καθώς και τη βιβλιοθήκη PM4PY (2.8.7.4) για την ανακάλυψη της διαδικασίας και τον έλεγχο συμμόρφωσης. Τέλος, ωστε χρειαστούμε το λογισμικό Disco (3.6.7) για να εστιάσουμε σε τμήματα της διαδικασίας και να εντοπίσουμε τα σημεία συμφόρησης.

### 4.2.1 Συλλογή και ανάλυση δεδομένων

Το αρχείο καταγραφής που ωστε χρησιμοποιήσουμε στο πρώτο βήμα της εφαρμογής προέρχεται από έναν προσομοιωτή του, το LiqueurPlant2024<sup>3</sup>. Το πρόγραμμα αυτό έχει δημιουργηθεί για τις ανάγκες της παρούσας εργασίας και επιτρέπει την εκτέλεση της διαδικασίας παραγωγής λικέρ ως ο χειριστής του συστήματος, ο οποίος έχει τη δυνατότητα να εκτελέσει τη διαδικασία για την παραγωγή τύπου A και B παράλληλα. Επίσης, ελέγχει ο ίδιος τους χρόνους ανάδευσης, θέρμανσης, γεμίσματος και αδειάσματος κάθε σιλό, εισάγοντας τα ως παραμέτρους στο αρχείο εισόδου του προγράμματος. Ο προσομοιωτής έχει σχεδιαστεί με βάσει τους κανόνες που ορίστηκαν παραπάνω και προσφέρει μια έμπιστη απεικόνιση της λειτουργίας του συστήματος. Σκοπός του προσομοιωτή είναι να δημιουργεί αρχεία παρόμοια με αυτά που ωστε υπήρχαν σε λογισμικό που ελέγχει το κυβερνοφυσικό σύστημα, περιέχοντας συμβάντα από το φυσικό και το κυβερνητικό επίπεδο. Πρέπει να αναφερθεί ότι, το πρόγραμμα αυτό δημιουργεί αρχεία καταγραφής τα οποία προσομοιώνουν μια απρόσκοπη λειτουργία του συστήματος, στην οποία τα βήματα της διαδικασίας τηρούνται με την αναμενόμενη σειρά κάτι που πιθανώς να διαφέρει σε μια πραγματική υλοποίηση του συστήματος όπου ωστε μπορούσε να εντοπιστεί μη αναμενόμενη συμπεριφορά. Για το λόγο αυτό, δεν ωστε χρειαστεί να ελέγξουμε το αν το σύστημα τηρεί τους περιορισμούς και λειτουργεί βάσει της θεωρητικής περιγραφής του, καθώς γνωρίζουμε εξ ορισμού ότι ο προσομοιωτής δεν παρουσιάζει σφάλματα και προσομοιώνει την λειτουργία του συστήματος όπως σχεδιάστηκε.

Αρχικά, για την δημιουργία του αρχείου καταγραφής χρειάστηκε να εκτελεστεί ο

<sup>2</sup><https://pandas.pydata.org/>

<sup>3</sup><https://github.com/ThramboulidisKleanthis/LiqueurPlant2024>

## Κεφάλαιο 4. Εφαρμογή

---

προσομοιωτής για 15 περιπτώσεις. Σε κάθε μια από τις περιπτώσεις, το λικέρ παράχθηκε και από τις δύο γραμμές παράλληλα επιλέγοντας ποικίλους χρόνους ώστε να εντοπιστούν όσο περισσότερες διαφορετικές παραλλαγές της διαδικασίας, ενώ η διαδικασία εκτελέστηκε ακριβώς δύο φορές για κάθε γραμμή. Μετά το πέρας της προσομοίωσης τα δεδομένα μεταφέρθηκαν σε ένα txt αρχείο με 721 γραμμές.

case	timestamp	source	info
1	03-04-24 14:46:40	s1	StartFilling
1	03-04-24 14:46:40	s2	StartFilling
1	03-04-24 14:46:46	s2	FillingCompleted
1	03-04-24 14:46:46	s1	FillingCompleted
1	03-04-24 14:46:46	s2	StartHeating
1	03-04-24 14:46:46	lgpA	StartTransfer
1	03-04-24 14:46:50	s2	HeatingCompleted
1	03-04-24 14:47:02	s1	PouringCompleted
1	03-04-24 14:47:02	lgpB	StartTransfer
1	03-04-24 14:47:02	s4	FillingCompleted
1	03-04-24 14:47:02	lgpA	TransferCompleted
1	03-04-24 14:47:02	s1	StartFilling
1	03-04-24 14:47:02	s4	StartHeating

**Πίνακας 4.1:** Αρχείο καταγραφής LPS

Στον πίνακα 4.1 βλέπουμε ένα στιγμιότυπο από αυτό το αρχείο καταγραφής, το οποίο περιέχει 4 στήλες: case, timestamp, source και info. Η πρώτη στήλη δίνει πληροφορίες σχετικά με την περίπτωση της διαδικασίας παραγωγής στην οποία αντιστοιχίζεται το συμβάν. Η δεύτερη στήλη καθορίζει τη στιγμή που πραγματοποιήθηκε το συμβάν, ενώ η στήλη source αναφέρεται στο στοιχείο που αφορά το συμβάν και η στήλη info την κατάσταση του. Στο πραγματικό σύστημα, οι τιμές αυτές αφορούν το χυμονητικό επίπεδο και αντιπροσωπεύουν αυτές που στέλνει το σύστημα ελέγχου στους ενεργοποιητές (actuators) του κάθε στοιχείου. Πρέπει να αναφερθεί ότι κάθε βήμα της διαδικασίας όπως περιγράφηκε παραπάνω, δεν αντιστοιχεί σε μία δραστηριότητα του παρόντος αρχείου καταγραφής αλλά σε περισσότερες. Το φαινόμενο αυτό ονομάζεται data granularity [19] και σημαίνει ότι η θεωρητική περιγραφή της διαδικασίας είναι σε υψηλότερο επίπεδο από αυτό που παρατηρείται μέσα στο αρχείο καταγραφής. Για παράδειγμα, οι δραστηριότητες lgpA-StartTransfer, s1-PouringCompleted, s4-FillingCompleted και lgpA-TransferCompleted του αρχείου καταγραφής, αποτελούν το βήμα της διαδικασίας που αφορά τη μεταφορά του υγρού στο κάτω σιλό. Τέλος, παρατηρούμε ότι πολλά συμβάντα είναι καταγεγραμμένα στο ίδιο δευτερόλεπτο.

Για να προχωρήσουμε στην ανακάλυψη της διαδικασίας θα πρέπει να φορτώσουμε

το αρχείο καταγραφής ως dataframe, να αντιστοιχίσουμε τις στήλες που έχουμε στη διάθεση μας με αυτές που είναι αναγκαίες για την εφαρμογή του αλγόριθμου ανακάλυψης και έπειτα να το μετατρέψουμε σε event log. Μετά την εκτέλεση του κώδικα τα δεδομένα έχουν αλλάξει μορφή, αφού συγχωνεύτηκαν όλες οι στήλες που σχετίζονται με μια δραστηριότητα ώστε να εξαχθούν όσο το δυνατόν πιο απλά μοντέλα. Ένα στιγμιότυπο από τα δεδομένα που πρόκειται να εισάγουμε στους αλγόριθμους ανακάλυψης, φαίνεται στον πίνακα 4.2.

case:concept:name	time:timestamp	concept:name
7	2024-03-04 15:02:34.00+00:00	s2-StartFilling
7	2024-03-04 15:02:34.00+00:00	s1-StartFilling
7	2024-03-04 15:02:37.00+00:00	s2-FillingCompleted
7	2024-03-04 15:02:37.00+00:00	s1-FillingCompleted
7	2024-03-04 15:02:37.00+00:00	s2-StartHeating

**Πίνακας 4.2:** Αρχείο καταγραφής μετά την επεξεργασία

Για να εντοπίσουμε όλες τις διακεκριμένες δραστηριότητες που περιέχει το αρχείο καταγραφής θα χρησιμοποιήσουμε τη συνάρτηση `get_event_attribute_values`. Είναι σημαντικό να αντιστοιχίσουμε την παράμετρο `attributes` με τη στήλη που περιέχει τις δραστηριότητες του αρχείου. Το αποτέλεσμα είναι ένα dictionary με τα ονόματα των δραστηριοτήτων ως key και το πλήθος εμφάνισης τους στο αρχείο καταγραφής ως value, όπως φαίνεται στο σχήμα 4.3.

```
1 { "s1-StartFilling": 30, "s2-StartFilling": 30, "s2-FillingCompleted": 30, "s1-FillingCompleted": 30, "s2-StartHeating": 30, "lgpA-StartTransfer": 30, "s2-HeatingCompleted": 30, "s1-PouringCompleted": 30, "lgpB-StartTransfer": 30, "s4-FillingCompleted": 30, "lgpA-TransferCompleted": 30, "s4-StartHeating": 30, "s4-HeatingCompleted": 30, "s4-StartMixing": 30, "s2-PouringCompleted": 30, "s3-FillingCompleted": 30, "lgpB-TransferCompleted": 30, "s4-MixingCompleted": 30, "s3-StartMixing": 30, "s4-PouringStarted": 30, "s3-MixingCompleted": 30, "s3-PouringStarted": 30, "s4-PouringCompleted": 30, "s3-PouringCompleted": 30}
```

**Σχήμα 4.3:** Διακεκριμένες δραστηριότητες `lps_log`

Έτσι επιβεβαιώνουμε ότι το αρχείο καταγραφής περιέχει μόνο τα κυβερνητικά συμβάντα που σχετίζονται με την παράλληλη διαδικασία παραγωγής και μας επιτρέπει να κάνουμε μια λεπτομερή μελέτη της διαδικασίας χωρίζοντας την κύρια διαδικασία σε δύο διαδικασίες, την παραγωγή τύπου A και τύπου B. Δίπλα στο όνομα κάθε δραστηριότητας είναι το πλήθος εμφάνισης της στο αρχείο, έχοντας 2 εμφανίσεις σε κάθε μια από τις τις 15 περιπτώσεις. Χωρίζοντας το αρχείο σε δύο επιμέρους, διατηρούμε τη σειρά των δραστηριοτήτων ως έχει και παράλληλα απομονώνουμε τη συμπεριφορά που σχετίζεται με την παραγωγή των δύο λικέρ, σε ξεχωριστά μοντέλα. Η παράμετρος `retain`

της συνάρτησης `filter_event_attribute_values`, καθορίζει το αν οι δραστηριότητες αφαιρούνται ή παραμένουν στο αρχείο καταγραφής. Μετά την εκτέλεση του κώδικα, έχουμε το `lps_log_A` το οποίο περιέχει 12 δραστηριότητες σχετικές με την παραγωγή τύπου A και το `lps_log_B` το οποίο περιέχει 12 δραστηριότητες σχετικές με την παραγωγή τύπου B. Θα πρέπει να σημειωθεί ότι, αν επιλέγαμε να εστιάσουμε στο ενιαίο αρχείο, θα είχαμε μια αρκετά πολύπλοκη διαδικασία (spaghetti process) από την οποία θα ήταν αδύνατο να συγκρίνουμε τις επί μέρους διαδικασίες και να εξάγουμε ωφέλιμα συμπεράσματα. Ωστόσο, το ενιαίο αρχείο καταγραφής θα είναι χρήσιμο για να ελέγξουμε άλλες πτυχές της διαδικασίας παραγωγής αργότερα. Ο κώδικας επεξεργασίας του αρχείου καταγραφής φαίνεται στο σχήμα 4.4.

```

1 import pandas as pd
2 import pm4py
3
4 # DATA PRE-PROCESSING
5 df = pd.read_csv("LiqueurPlant2024_LOG.txt", sep=",")
6
7 df["activity"] = df["source"] + " - " + df["info"]
8 df = df.drop(["source", "info"], axis=1)
9 df = df.rename(columns={"case": "case:concept:name", "activity": "concept:name", "timestamp": "time:timestamp"})
10
11 # MAPPING COLUMNS
12 df = pm4py.format_dataframe(df, case_id="case:concept:name", activity_key="concept:name", timestamp_key="time:timestamp")
13 event_log = pm4py.convert_to_event_log(df)
14
15 # SHOWING ACTIVITIES
16 print(pm4py.get_event_attribute_values(event_log, attribute="concept:name"))
17
18 A_ACTIVITIES = {"s1-StartFilling", "s1-FillingCompleted", "lgpA-StartTransfer", "s1-PouringCompleted", "lgpA-TransferCompleted", "s4-FillingCompleted", "s4-PouringStarted", "s4-PouringCompleted", "s4-StartMixing", "s4-MixingCompleted", "s4-StartHeating", "s4-HeatingCompleted"}
19
20 # SPLITTING LOGS
21 lps_log_A = pm4py.filter_event_attribute_values(event_log, attribute_key="concept:name", values=A_ACTIVITIES, level="event", retain=True)
22 lps_log_B = pm4py.filter_event_attribute_values(event_log, attribute_key="concept:name", values=A_ACTIVITIES, level="event", retain=False)

```

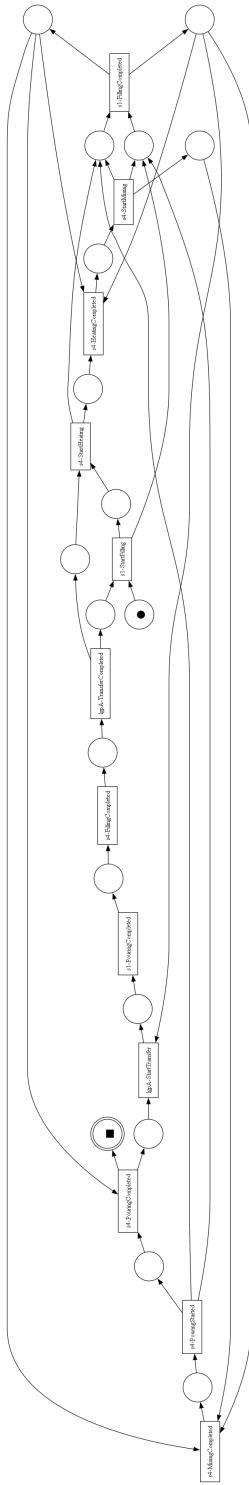
**Σχήμα 4.4:** Κώδικας επεξεργασίας αρχείου καταγραφής

#### 4.2.2 Ανακάλυψη διαδικασίας

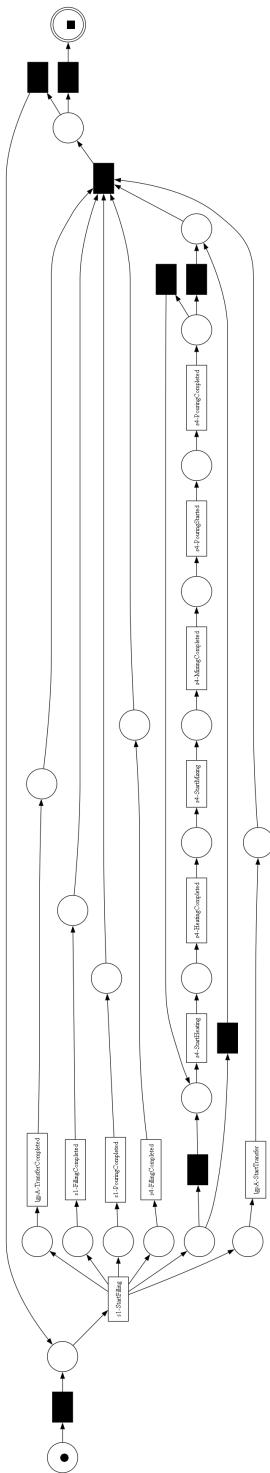
Το επόμενο βήμα της εφαρμογής, είναι να χρησιμοποιήσουμε αλγόριθμους ανακάλυψης για να ανακαλύψουμε το μοντέλο της διαδικασίας. Η ανακάλυψη της διαδικασίας θα μας δώσει μια σαφή εικόνα για το τι συμβαίνει πραγματικά στο σύστημα. Θα χρησιμοποιήσουμε τους αλγόριθμους alpha miner, inductive miner και heuristics miner για

να ανακαλύψουμε τα αντίστοιχα petri nets και έπειτα όταν μετατρέψουμε τα petri nets σε λιγότερο σύνθετα μοντέλα διαδικασιών, για να εντοπίσουμε τη ροή της κάθε διαδικασίας παραγωγής. Για αρχή ας εστιάσουμε στην παραγωγή τύπου A χρησιμοποιώντας τους παραπάνω αλγόριθμους ανακάλυψης. Το petri net που προκύπτει από τον alpha miner (σχήμα 4.5) είναι σχετικά απλό μοντέλο, ενώ τα petri nets που προκύπτουν από inductive miner και heuristics miner είναι πιο σύνθετα, περιέχοντας και αόρατες μεταβάσεις (σχήματα 4.6, 4.7). Το ίδιο ισχύει και για petri net τη διαδικασία τύπου B (σχήματα 4.8, 4.9, 4.10) τα οποία φαίνεται να είναι ελαφρώς πιο περίπλοκα, πράγμα που μπορεί να οφείλεται στην ύπαρξη στοιχείων θέρμανσης και ανάδευσης σε 2 σιλό. Επιπλέον, γίνεται ήδη εμφανές ότι και στις δύο διαδικασίες ο inductive miner γενικεύει υπερβολικά τη συμπεριφορά του αρχείου, έχοντας δύο μεταβάσεις τις s1-StartFilling και s2-StartFilling να ενεργοποιούν πολλούς τόπους μαζί. Αυτό όπως αναφέρθηκε ήδη, είναι και ένα από τα χαρακτηριστικά του inductive miner, ο οποίος εστιάζει στο να ανακαλύψει συσχετίσεις μεταξύ δραστηριοτήτων και δεν ασχολείται με το πόσο συχνά ακολουθεί η μια την άλλη.

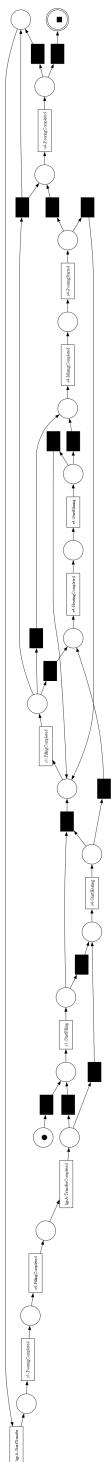
Είναι γνωστό ότι, τα petri nets δεν είναι η ιδανική αναπαράσταση μοντέλων διαδικασιών τα οποία αποτελούνται από δραστηριότητες σε μη αυστηρά καθορισμένη σειρά και οι οποίες εκτελούνται συγχρονισμένα. Στο δικό μας σύστημα, μας θεωρείται αναμενόμενο μια δραστηριότητα να ακολουθείται από ένα σύνολο δραστηριοτήτων και όχι μόνο από μια συγκεκριμένη κάτι που μπορεί να συμβαίνει με διαφορετικές συχνότητες ανά περίπτωση. Για το λόγο αυτό, όταν μετατρέψουμε το petri net που εξάγεται από τον heuristics miner σε μια πιο απλουστευμένη μορφή αναπαράστασης, όπως το heuristic net. Το μοντέλο που προκύπτει (σχήμα 4.11) είναι πιο εύκολο να το χρησιμοποιήσουμε για να συγχρίνουμε με το προϋπάρχον activity diagram της διαδικασίας, αλλά και να δούμε ποια μονοπάτια ακολουθούνται περισσότερο συχνά σε σχέση με άλλα. Αντίστοιχα, μπορούμε να ανακαλύψουμε το heuristic net μοντέλο της διαδικασίας τύπου B (σχήμα 4.12). Σε κάθε heuristic net, τα παραλληλόγραμμα συμβολίζουν τις δραστηριότητες, ενώ ο πράσινος και πορτοκαλί κόμβος την αρχική και τελική δραστηριότητα της διαδικασίας. Πάνω στις ακμές, αναφέρεται η συχνότητα με την οποία ακολουθείται μια δραστηριότητα και το γράφημα παρουσιάζει ολόκληρη τη συμπεριφορά του αρχείου (όλες οι ακμές εμφανίζονται στο γράφημα). Τέλος, για να υπολογίσουμε το βαθμό που τα μοντέλα αναπαριστούν τη διαδικασία παραγωγής όταν πρέπει να εκτελέσουμε την τεχνική του ελέγχου συμμόρφωσης. Η μελέτη των μετρικών ποιότητας που θα προκύψουν μέσα από τον έλεγχο συμμόρφωσης, όταν μας δώσει μια πιο εγγυημένη άποψη για το ποιο από τα παραπάνω petri net αναπαριστά καλύτερα τη διαδικασία παραγωγής κάθε τύπου.



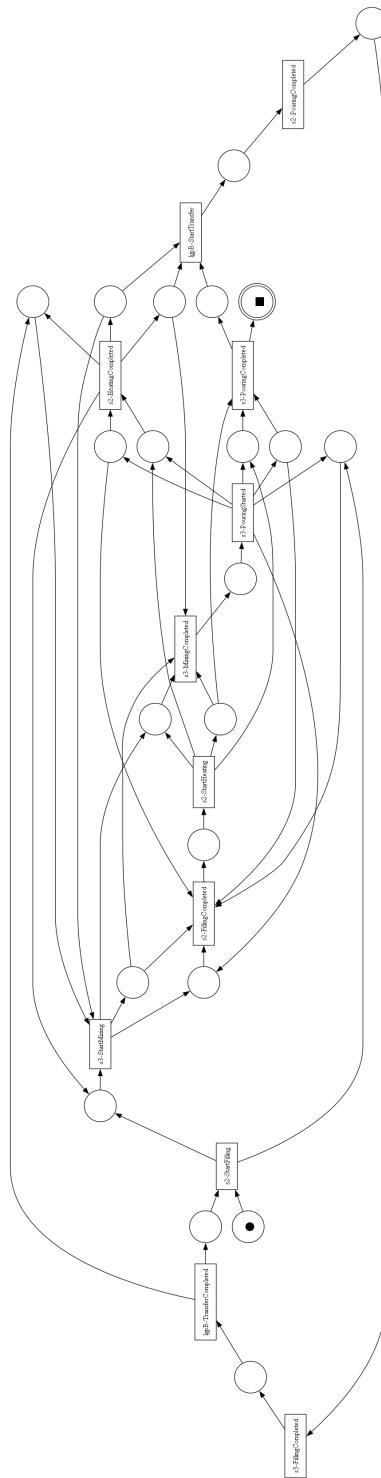
**Σχήμα 4.5:** Alpha miner για το lps\_log\_A



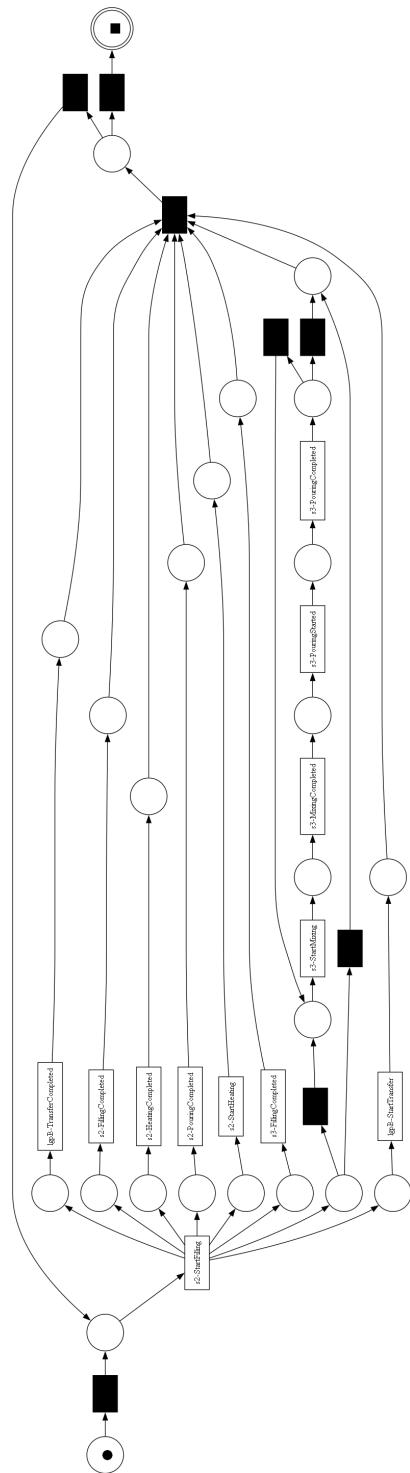
Σχήμα 4.6: Inductive miner για το lps\_log\_A



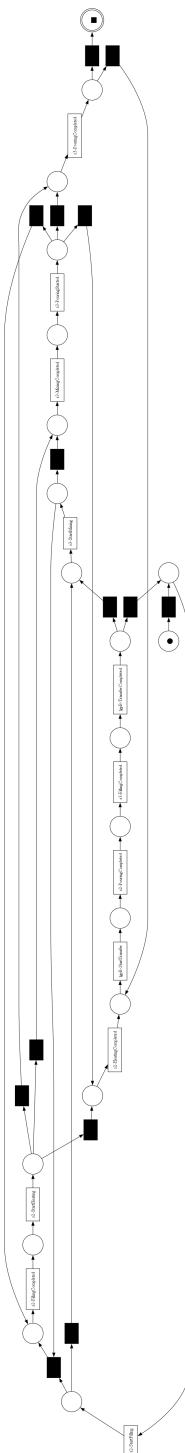
**Σχήμα 4.7:** Heuristics miner για το lps\_log\_A



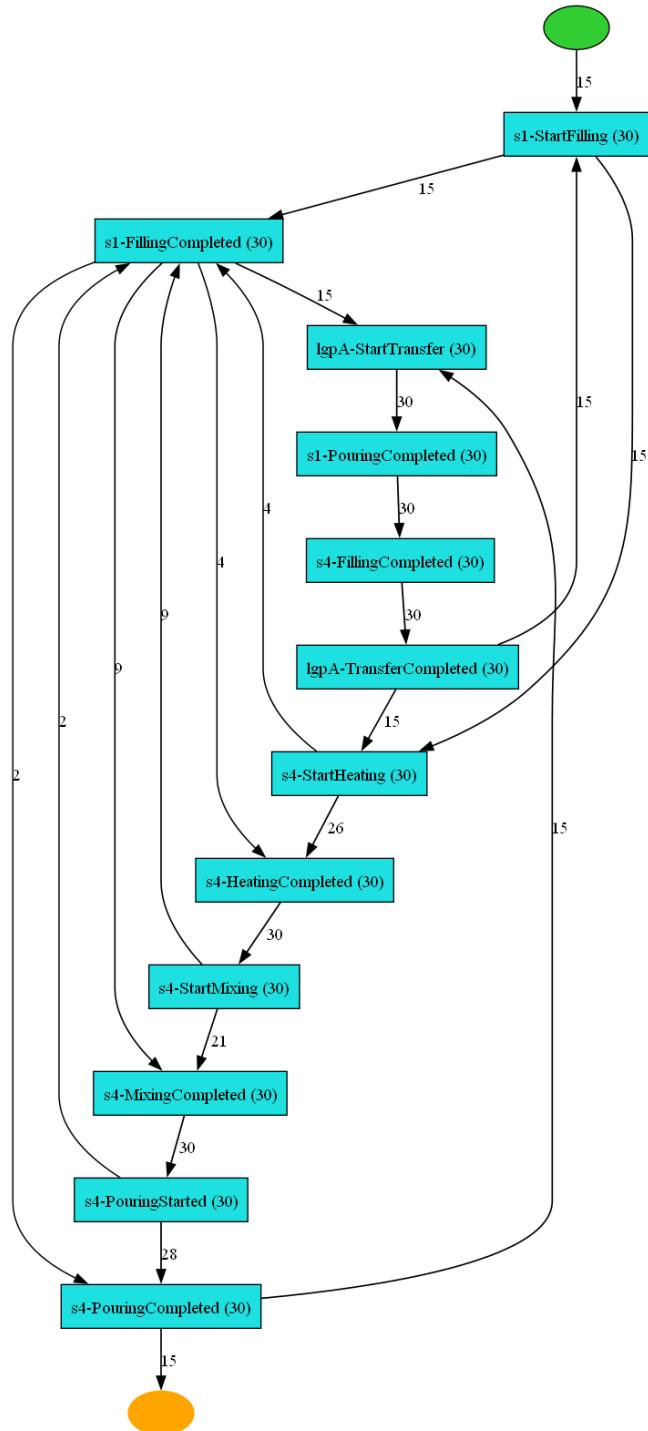
Σχήμα 4.8: Alpha miner για το `lps_log_B`



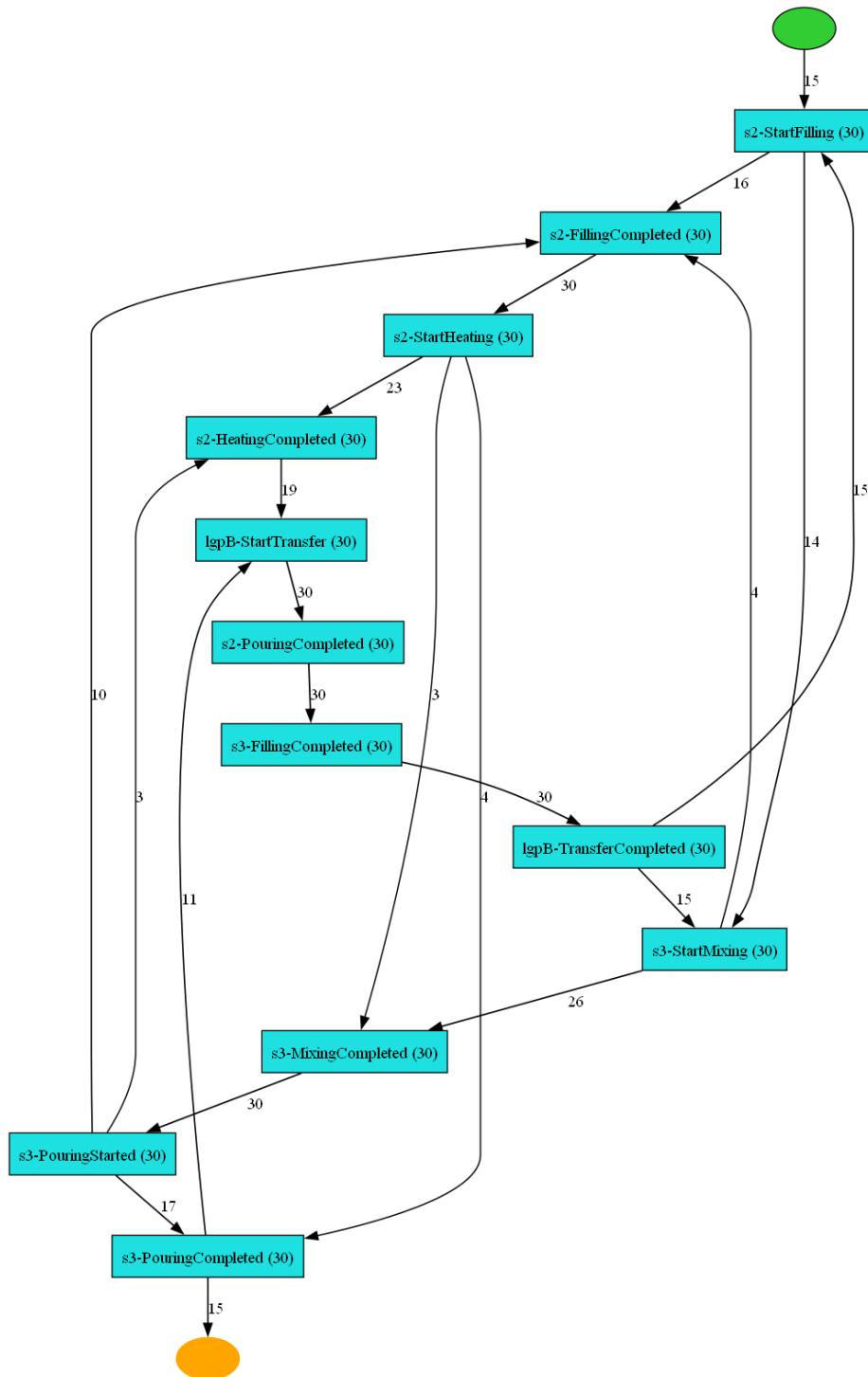
**Σχήμα 4.9:** Inductive miner για το lps\_log\_B



**Σχήμα 4.10:** Heuristics miner για το lps\_log\_B



$\Sigma\chi\eta\mu\alpha$  4.11: Heuristic net του `lps_log_A`



$\Sigma\chi\eta\mu\alpha$  4.12: Heuristic net του lps\_log\_B

#### 4.2.3 Έλεγχος συμμόρφωσης

Η τεχνική του ελέγχου συμμόρφωσης ωστε πραγματοποιηθεί με δύο τρόπους: token-based replay και alignments. Αρχικά, ωστε επιχειρήσουμε να εφαρμόσουμε token-based replay για κάθε petri net που προέκυψε από τους αλγόριθμους ανακάλυψης, με σκοπό να αξιολογήσουμε το πόσο καλά αναπαριστούν την κάθε διαδικασία παραγωγής. Τα αποτελέσματα εκτέλεσης του κώδικα συνοψίζονται στους πίνακες 4.3 και 4.4. Η πρώτη στήλη του πίνακα αφορά το ποσοστό των περιπτώσεων στο αρχείο καταγραφής που επαληθεύουν το μοντέλο. Η δεύτερη στήλη αφορά τη μέση τιμή fitness κάθε περίπτωσης, η τρίτη στήλη το συνολικό fitness του αρχείου καταγραφής ενώ η τελευταία το precision του μοντέλου. Παρατηρούμε, ότι και στους δύο τύπους παραγωγής ο inductive miner δίνει το καλύτερο fitness ενώ παράλληλα το χειρότερο precision. Το μοντέλο που κατασκευάζει, επαληθεύει τέλεια όλες τις περιπτώσεις στο αρχείο καταγραφής όμως δεν είναι τόσο ακριβές αφού επιτρέπει μεγάλο ποσοστό πρόσθετης συμπεριφοράς, η οποία δεν παρατηρείται στο αρχείο καταγραφής. Από αυτό συμπεραίνουμε, ότι το μοντέλο που κατασκευάζει ο inductive miner είναι αρκετά γενικό. Ο alpha miner δίνει το καλύτερο precision σε κάθε τύπο παραγωγής, κάτι που είναι αναμενόμενο αφού γνωρίζουμε ότι ο αλγόριθμος αυτός προσαρμόζεται στο αρχείο καταγραφής και δεν επιτρέπει πρόσθετη συμπεριφορά. Παράλληλα, δίνει το χειρότερο fitness και στους δύο τύπους, το οποίο όμως δεν είναι τόσο κακό για να χαρακτηρίσει το μοντέλο under-fitting. Τέλος, ο heuristics miner καταφέρνει να διατηρεί την καλύτερη ισορροπία μεταξύ των μετρικών και στους δύο τύπους παραγωγής. Πιο συγκεκριμένα, πετυχαίνει precision 0.5 και στους δύο τύπους, ενώ fitness 0.96 και 0.94 αντίστοιχα. Το μοντέλο που κατασκευάζει είναι λιγότερο γενικό από αυτό του inductive miner, χάνοντας ελάχιστα σε fitness.

	perc_fit_traces	average_trace_fitness	log_fitness	precision
Alpha	0.0	0.72	0.72	1.0
Inductive	100.0	1.0	1.0	0.31
Heuristics	0.0	0.96	0.96	0.5

Πίνακας 4.3: Αποτελέσματα TBR για το lps\_log\_A

	perc_fit_traces	average_trace_fitness	log_fitness	precision
Alpha	0.0	0.67	0.67	1.0
Inductive	100.0	1.0	1.0	0.25
Heuristics	0.0	0.94	0.94	0.5

Πίνακας 4.4: Αποτελέσματα TBR για το lps\_log\_B

Για να επαληθεύσουμε τις τιμές των μετρικών που προέκυψαν από την τεχνική token-based replay, θα επιχειρήσουμε να εφαρμόσουμε και alignments σε κάθε αρχείο καταγραφής. Τα αποτελέσματα εκτέλεσης του κώδικα για κάθε αρχείο καταγραφής, συνοψίζονται στους πίνακες 4.5 και 4.6. Στους πίνακες αυτούς παρατηρούμε ότι ο έλεγχος συμμόρφωσης με alignments δεν εκτελέστηκε στα petri nets που ανακάλυψε ο alpha miner όμως εκτελέστηκε για τους άλλους, κάτι που οφείλεται στο ότι ο alpha miner δεν εγγυάται την ορθότητα (soundness) του petri net. Η έλλειψη ορθότητας μπορεί να σημαίνει αρκετά πράγματα, ανάμεσα στα οποία είναι ότι το petri net μπορεί να περιέχει αδιέξοδα, να μην έχει σωστό τερματισμό ή να μην είναι εφικτό να φτάσει στον τερματισμό του [10]. Στη δική μας περίπτωση, τα petri nets που κατασκεύασε ο alpha miner δεν έχουν ορθότητα, κάτι που πιστωνώς να οφείλεται στην περίπλοκη και αδιάρθρωτη διαδικασία, δηλαδή ότι οι δραστηριότητες μπορούν να ακολουθούνται από ένα ευρύ σύνολο δραστηριοτήτων. Επίσης, παρατηρούμε ότι οι τιμές που δίνει η τεχνική alignments για το lps\_log\_B είναι ίδιες με αυτές που έδωσε η token-based replay, όσον αφορά τον inductive miner. Ωστόσο, δεν ισχύει το ίδιο για τον heuristics miner του lps\_log\_A για τον οποίο φαίνεται να δίνει χειρότερο fitness, κάτι που αποτελεί και μειονέκτημα της τεχνικής token-based replay, η οποία σε αρχεία καταγραφής με πολλές παραλλαγές επιτρέπει περισσότερη συμπεριφορά οδηγώντας σε καλύτερο fitness από αυτό που υπάρχει πραγματικά [10]. Ο κώδικας που χρησιμοποιήθηκε για την ανακάλυψη των μοντέλων διαδικασιών και του ελέγχου συμμόρφωσης φαίνεται στο σχήμα 4.13.

	perc_fit_traces	average_trace_fitness	log_fitness	precision
Alpha	-	-	-	1.0
Inductive	100.0	1.0	1.0	0.31
Heuristics	0.0	0.79	0.79	0.5

Πίνακας 4.5: Αποτελέσματα alignments για το lps\_log\_A

	perc_fit_traces	average_trace_fitness	log_fitness	precision
Alpha	-	-	-	1.0
Inductive	100.0	1.0	1.0	0.25
Heuristics	0.0	0.83	0.83	0.5

Πίνακας 4.6: Αποτελέσματα alignments για το lps\_log\_B

## Κεφάλαιο 4. Εφαρμογή

---

```
1 from pm4py.algo.discovery.heuristics import algorithm as heuristic_miner
2 from pm4py.visualization.petri_net import visualizer as pn_visualizer
3 from pm4py.visualization.heuristics_net import visualizer as hn_visualizer
4
5 # TOKEN-BASED REPLAY DIAGNOSTICS
6 def tbr_lps(logFile, net, im, fm):
7     prec = pm4py.precision_token_based_replay(logFile, net, im, fm)
8     print(prec)
9     fit = pm4py.fitness_token_based_replay(logFile, net, im, fm)
10    print(fit)
11
12 # ALIGNMENTS DIAGNOSTICS
13 def align_lps(logFile, net, im, fm):
14     try:
15         prec = pm4py.precision_alignments(logFile, net, im, fm)
16         print(prec)
17         fit = pm4py.fitness_alignments(logFile, net, im, fm)
18         print(fit)
19     except Exception as e:
20         print(e)
21
22 # PROCESS DISCOVERY AND CONFORMANCE CHECKING AT LPS LOGS
23 for logFile in [lps_log_A, lps_log_B]:
24
25     # ALPHA MINER
26     net, im, fm = pm4py.discover_petri_net_alpha(logFile)
27     gviz = pn_visualizer.apply(net, im, fm)
28     pn_visualizer.view(gviz)
29     tbr_lps(logFile, net, im, fm)
30     align_lps(logFile, net, im, fm)
31
32     # INDUCTIVE MINER
33     net, im, fm = pm4py.discover_petri_net_inductive(logFile)
34     gviz = pn_visualizer.apply(net, im, fm)
35     pn_visualizer.view(gviz)
36     tbr_lps(logFile, net, im, fm)
37     align_lps(logFile, net, im, fm)
38
39     # HEURISTICS MINER
40     net, im, fm = pm4py.discover_petri_net_heuristics(logFile)
41     gviz = pn_visualizer.apply(net, im, fm)
42     pn_visualizer.view(gviz)
43     tbr_lps(logFile, net, im, fm)
44     align_lps(logFile, net, im, fm)
45
46     # HEURISTIC NET
47     heu_net = heuristic_miner.apply_heu(logFile)
48     gviz = hn_visualizer.apply(heu_net)
49     hn_visualizer.view(gviz)
50
```

**Σχήμα 4.13:** Κώδικας ανακάλυψης και ελέγχου συμμόρφωσης

Για να γίνει περισσότερο εμφανής η χρησιμότητα του ελέγχου συμμόρφωσης για εντοπισμό μη αναμενόμενης συμπεριφοράς, φτιάχτηκαν χειροκίνητα κάποια ψεύτικα αρχεία καταγραφής (`error_log1`, `error_log2`, `error_log3`) ξεκινώντας από τα δεδομένα του προσομοιωτή και τα οποία σε πραγματικές συνθήκες θα μπορούσαν να εμφανιστούν ως αρχεία με σφάλματα. Κάθε ένα από αυτά περιέχει ένα διαφορετικό σφάλμα, για παράδειγμα το πρώτο αρχείο έχει τη διαδικασία με δραστηριότητες να λείπουν (δεν εκτελείται heating ή mixing), το δεύτερο αρχείο έχει τη διαδικασία με δραστηριότητες σε λάθος σειρά (πρώτα mixing και μετά heating) ενώ το τελευταίο έχει δραστηριότητες να εκτελούνται παραπάνω φορές (δύο φορές heating ή mixing). Ο σκοπός αυτής της ενέργειας, είναι να ελεγχθούν τα δεδομένα που περιέχουν σφάλματα πάνω στα μοντέλα που ανακαλύφθηκαν προηγουμένως με βάσει το αληθινό αρχείο καταγραφής του προσομοιωτή και τα οποία αναπαριστούν σωστά τη διαδικασία παραγωγής. Τα αποτελέσματα του fitness μετά την εκτέλεση του ελέγχου συμμόρφωσης με token-based replay και alignments στα τρία νέα αρχεία πάνω στα τρία μοντέλα της παραγωγής τύπου A, φαίνονται στους πίνακες 4.7, 4.8. Παρατηρούμε ότι, ο έλεγχος συμμόρφωσης για την διαδικασία τύπου A μέσω token-based replay δίνει οριακά χειρότερο fitness σε σχέση με αυτό του αληθινού αρχείου καταγραφής, ενώ με την τεχνική alignments οι διαφορές είναι μεγαλύτερες.

	Alpha	Inductive	Heuristics
lps_log_A	0.72	1.0	0.96
error_log1	0.65	0.94	0.88
error_log2	0.69	0.97	0.94
error_log3	0.66	0.97	0.96

Πίνακας 4.7: Fitness αρχείων με σφάλματα για το `lps_log_A` μέσω TBR

	Alpha	Inductive	Heuristics
lps_log_A	-	1.0	0.79
error_log1	-	0.84	0.62
error_log2	-	0.86	0.67
error_log3	-	0.88	0.71

Πίνακας 4.8: Fitness αρχείων με σφάλματα για το `lps_log_A` μέσω alignments

## Κεφάλαιο 4. Εφαρμογή

---

Αντίστοιχα, για την παραγωγή τύπου B τα αποτελέσματα φαίνονται στους πίνακες 4.9, 4.10. Στη διαδικασία τύπου B η token-based replay δεν καταφέρνει να κάνει ορατά όλα τα σφάλματα σε κανένα petri net, αφού δεν φαίνονται σημαντικές διαφορές στα fitness σε σχέση με το lps\_log\_B, ενώ η alignments δίνει σχετικά χειρότερα fitness.

	Alpha	Inductive	Heuristics
lps_log_B	0.67	1.0	0.94
error_log1	0.60	0.85	0.86
error_log2	0.67	1.0	0.95
error_log3	0.67	0.92	0.89

**Πίνακας 4.9:** Fitness αρχείων με σφάλματα για το lps\_log\_B μέσω TBR

	Alpha	Inductive	Heuristics
lps_log_B	-	1.0	0.83
error_log1	-	0.85	0.59
error_log2	-	1.0	0.79
error_log3	-	0.92	0.67

**Πίνακας 4.10:** Fitness αρχείων με σφάλματα για το lps\_log\_B μέσω alignments

Με βάση τα παραπάνω, η τεχνική που θεωρούμε καλύτερη στο να κάνει εμφανή τα σφάλματα και στους δύο τύπους παραγωγής είναι η alignments, αφού η token-based replay δεν εμφανίζει σημαντικές απώλειες στο fitness σε σχέση με το αληθινό αρχείο, για καμία διαδικασία παραγωγής. Αν θέλαμε να επιλέξουμε έναν αλγόριθμο για να κατασκευάσουμε το μοντέλο petri net της διαδικασίας παραγωγής κάθισ τύπου, θα επιλέγαμε τον heuristics miner, ο οποίος φέρεται να χρατάει περισσότερο ακριβή τη διαδικασία σε σχέση με τον inductive miner, εντοπίζοντας όλα τα σφάλματα που ορίσαμε. Αυτή η έλλειψη ακριβειας στον inductive miner γίνεται περισσότερο ορατή, αφού δεν καταφέρνει να εντοπίσει τα σφάλματα του error\_log2 δίνοντας ακριβώς το ίδιο fitness με το αληθινό αρχείο και στις δύο τεχνικές. Ο κώδικας που χρησιμοποιήθηκε για τη μέτρηση του fitness στα ψεύτικα αρχεία καταγραφής φαίνεται στο σχήμα 4.14.

```

1 # CONFORMANCE CHECKING AT TEST LOGS
2 for logFile in [lps_log_A, lps_log_B]:
3
4     # ALPHA MINER
5     net, im, fm = pm4py.discover_petri_net_alpha(logFile)
6     gviz = pn_visualizer.apply(net, im, fm)
7     for errorLog in [error_log1, error_log2, error_log3]:
8         tbr_lps(errorLog, net, im, fm)
9         align_lps(errorLog, net, im, fm)
10
11    # INDUCTIVE MINER
12    net, im, fm = pm4py.discover_petri_net_inductive(logFile)
13    gviz = pn_visualizer.apply(net, im, fm)
14    for errorLog in [error_log1, error_log2, error_log3]:
15        tbr_lps(errorLog, net, im, fm)
16        align_lps(errorLog, net, im, fm)
17
18    # HEURISTICS MINER
19    net, im, fm = pm4py.discover_petri_net_heuristics(logFile)
20    gviz = pn_visualizer.apply(net, im, fm)
21    for errorLog in [error_log1, error_log2, error_log3]:
22        tbr_lps(errorLog, net, im, fm)
23        align_lps(errorLog, net, im, fm)
24

```

**Σχήμα 4.14:** Κώδικας μέτρησης fitness σε αρχεία με σφάλματα

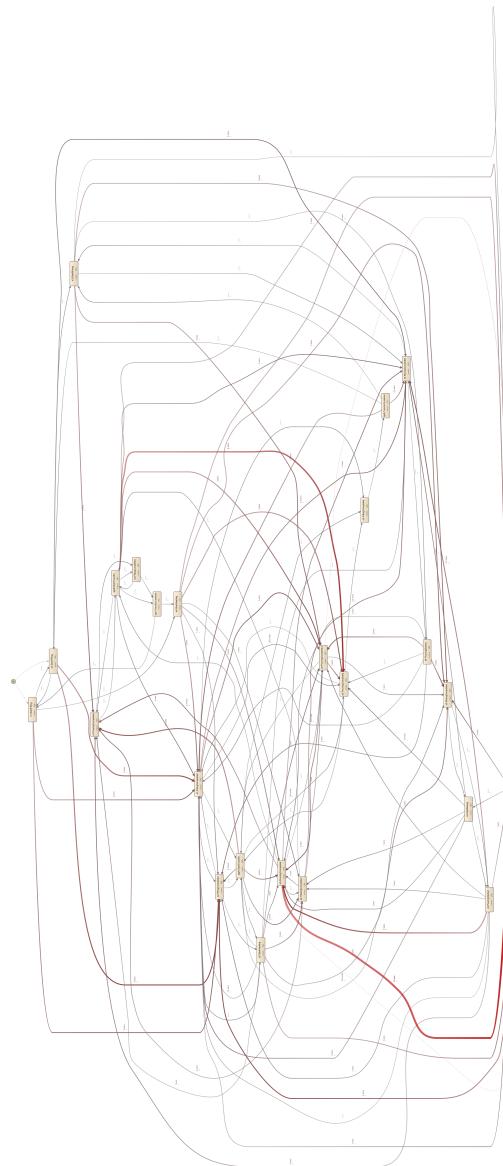
#### 4.2.4 Σημεία συμφόρησης

Η εξόρυξη διαδικασιών στο LPS, δεν περιορίζεται μόνο στην ανακάλυψη μοντέλων και τον έλεγχο συμμόρφωσης. Για να έχουμε μια ολοκληρωμένη εικόνα της λειτουργίας του, όταν πρέπει να εστιάσουμε σε συγκεκριμένα τμήματα της διαδικασίας παραγωγής. Θα πρέπει να βεβαιωθούμε ότι το σύστημα δεν παρουσιάζει σημεία συμφόρησης και δεν σπαταλά χρόνο στις λειτουργίες του. Εξετάζοντας αυτές τις πτυχές, όταν επιδιώξουμε να ανακαλύψουμε γνώσεις σχετικά με την αποτελεσματικότητα και την απόδοση του. Σε αυτό όταν μας βοηθήσει το λογισμικό Disco, το οποίο δεν επικεντρώνεται σε αλγόριθμους ανακάλυψης και έλεγχο συμμόρφωσης όπως το PM4PY, αλλά προσφέρει μια ξεκάθαρη και κατανοητή απεικόνιση της διαδικασίας παραγωγής, ενώ παράλληλα διαθέτει φίλτρα που όταν βοηθήσουν στο να εστιάσουμε στα πιο σημαντικά τμήματα της. Η ενιαία διαδικασία παραγωγής στο Disco παρουσιάζεται σαν ένα process map το οποίο έχει ανακαλυφθεί με χρήση του fuzzy miner και φαίνεται στο σχήμα 4.15. Σε αυτό το μοντέλο της διαδικασίας, βλέπουμε την παράλληλη διαδικασία με αρχικές δραστηριότητες την εκκίνηση του γεμίσματος στα s1 και s2 και τελικές το πέρας του αδειάσματος στα s3 και s4. Πάνω στις ακμές, συμβολίζεται ο μέσος χρόνος διάρκειας (median duration), η συχνότητα (absolute frequency) με την οποία ακολουθείται η κάθιε δραστηριότητα και το γράφημα αποτυπώνει ολόκληρη την καταγεγραμμένη συμπεριφορά, δημιουργώντας έτσι ένα πυκνό process map.

## Κεφάλαιο 4. Εφαρμογή

---

Ως σημεία συμφόρησης (bottlenecks) στην εξόρυξη διαδικασιών, χαρακτηρίζονται δραστηριότητες ή τμήματα της διαδικασίας τα οποία καθυστερούν την κύρια διαδικασία. Η απόδοση μιας διαδικασίας μπορεί να αυξηθεί αν εξαλειφθούν τα σημεία συμφόρησης, καθώς αυτά μπορεί να προκαλέσουν καθυστερήσεις, να σπαταλήσουν πόρους ή να επηρεάσουν την παραγωγικότητα του συστήματος [18]. Τέλος, τα σημεία συμφόρησης μπορεί να επηρεάσουν σημαντικά το χρόνο ολοκλήρωσης της διαδικασίας παραγωγής.

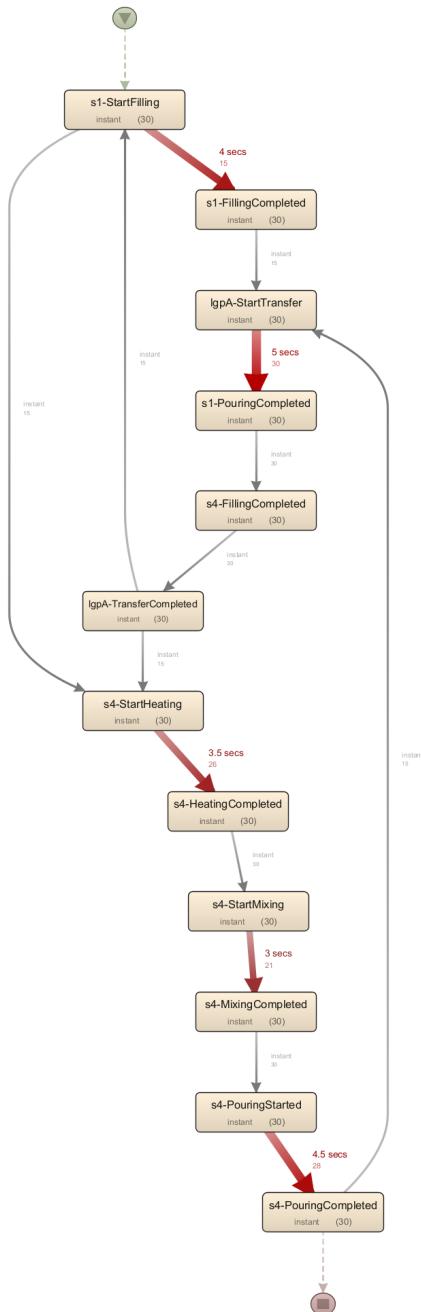


**Σχήμα 4.15:** Full process map (100%) διαδικασίας παραγωγής

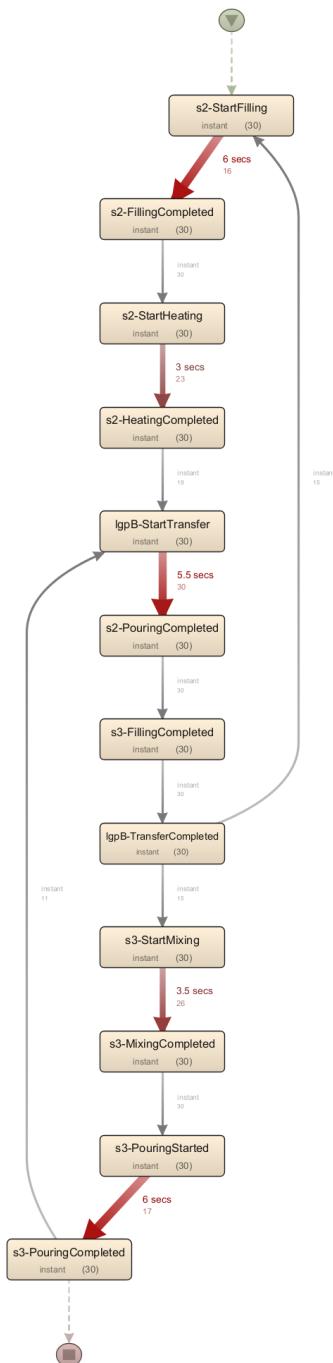
Για να εντοπίσουμε τα σημεία συμφόρησης στο LPS, θα πρέπει να δώσουμε σημασία στη χρονοσήμανση (timestamp) των συμβάντων. Θα χωρίσουμε τη διαδικασία σε παραγωγή τύπου A και B όπως προηγουμένως και θα σχεδιάσουμε τα process maps, χρησιμοποιώντας το φίλτρο attributes και επιλέγοντας τις επιμέρους δραστηριότητες της κάθε διαδικασίας. Με αυτό τον τρόπο, το process map θα μας δώσει πληροφορίες για το μέσο χρόνο που απαιτείται για την ολοκλήρωση των βημάτων της διαδικασίας παραγωγής, αλλά και τη συχνότητα με την οποία ακολουθείται κάθε μονοπάτι δραστηριοτήτων. Στα σχήματα 4.16 και 4.17 βλέπουμε το process map για κάθε τύπο παραγωγής. Κάθε παραλληλόγραμμο συμβολίζει μια δραστηριότητα, ενώ ο πράσινος και ο κόκκινος κόμβος αντιστοιχίζουν στην αρχική και τελική δραστηριότητα κάθε διαδικασίας. Είναι φανερό ότι τα process maps των διαδικασιών τύπου A και B, έχουν μεγάλη ομοιότητα με τα heuristic nets που ανακαλύφθηκαν προηγουμένως με τη διαφορά ότι λείπουν κάποιες ακμές, δηλαδή δεν αποτυπώνουν ολόκληρη τη συμπεριφορά παρά μόνο την πιο σύνηθη. Οι κόκκινες ακμές στο γράφημα, συμβολίζουν δραστηριότητες οι οποίες διαρκούν περισσότερο χρόνο σε σχέση με άλλες. Χρησιμοποιώντας την καρτέλα cases, ανάμεσα στις 15 φορές που εκτελέστηκε κάθε γραμμή παραγωγής, εντοπίζουμε μόνο 3 διαφορετικές παραλλαγές της διαδικασίας τύπου A και 5 για τη διαδικασία τύπου B. Εστιάζοντας σε ζεύγη δραστηριοτήτων, μπορούμε εύκολα να υπολογίσουμε το μέσο χρόνο που απαιτεί το σύστημα για να γεμίσει ή να αδειάσει τα σιλό και το χρόνο που θερμαίνει ή αναμιγνύει το υγρό. Στον πίνακα 4.11, συγκεντρώνεται όλη η σχετική με το χρόνο πληροφορία που συλλέγεται από τα process map κάθε διαδικασίας παραγωγής. Παράλληλα, εφόσον γνωρίζουμε ότι σε κάθε επιμέρους διαδικασία οι δραστηριότητες εκτελούνται ακριβώς μια φορά, μπορούμε να υπολογίσουμε και το μέσο χρόνο ολοκλήρωσης της διαδικασίας παραγωγής μέσω των process maps, αθροίζοντας τις τιμές όλων των ακμών τους. Για την διαδικασία τύπου A, ο μέσος χρόνος ολοκλήρωσης είναι 20 sec ενώ για την διαδικασία τύπου B είναι 24 sec. Παρατηρούμε ακόμη, ότι όλες δραστηριότητες στα process maps εκτός από αυτές που αναφέρονται στον πίνακα εκτελούνται αμέσως (instant), δηλαδή με χρονική απόσταση κάτω του 1 sec μεταξύ τους καθώς έχουμε επιλέξει ακρίβεια δευτερολέπτου. Στην πραγματικότητα, μεσολαβούν κάποια ms μεταξύ των συμβάντων του κυβερνοφυσικού συστήματος.

	s1	s4	s2	s3
Filling	4 sec	5 sec	6 sec	5.5 sec
Pouring	5 sec	4.5 sec	5.5 sec	6 sec
Mixing	-	3 sec	-	3.5 sec
Heating	-	3.5 sec	3 sec	-

Πίνακας 4.11: Μέσοι χρόνοι ολοκλήρωσης δραστηριοτήτων



**Σχήμα 4.16:** Filtered process map παραγωγής τύπου A



**Σχήμα 4.17:** Filtered process map παραγωγής τύπου B

## Κεφάλαιο 4. Εφαρμογή

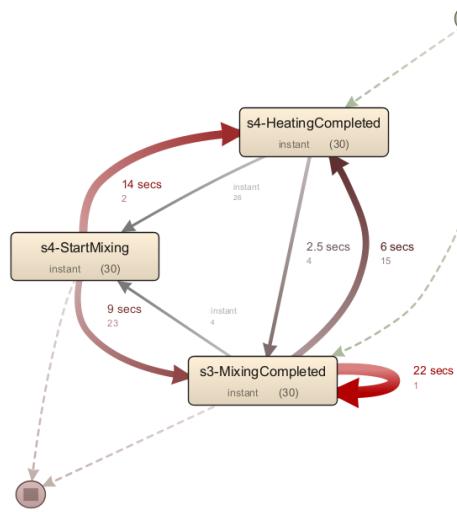
---

Σε ότι αφορά την παράλληλη διαδικασία, οι περιορισμοί ταυτόχρονης ανάδευσης και μεταφοράς υποδεικνύουν πιθανές χρονικές επιβαρύνσεις στο χρόνο ολοκλήρωσης των τμημάτων της κάθε διαδικασίας παραγωγής. Αυτή τη φορά, όταν χρησιμοποιήσουμε το ενιαίο process map και όταν ελέγξουμε αρχικά την επίπτωση της ανάδευσης στη διαδικασία τύπου A, επιλέγοντας ως attributes τις δραστηριότητες s4-HeatingCompleted, s4-StartMixing και s3-MixingCompleted. Παρατηρούμε ότι, η διαδικασία χρειάστηκε να περιμένει μόνο σε 4/30 φορές προκαλώντας καθυστέρηση 2.5 sec, ενώ 26/30 φορές εκτελέστηκε άμεσα. Ακολουθώντας την ίδια διαδικασία για την επίπτωση του περιορισμού ανάδευσης στη διαδικασία παραγωγής B, εντοπίζονται 8/30 φορές στις οποίες χρειάστηκε να περιμένει η διαδικασία προκαλώντας μέση καθυστέρηση 2.5 sec, ενώ 22/30 φορές εκτελέστηκε άμεσα. Στα σχήματα 4.18 και 4.19, φαίνονται τα process maps των παραπάνω τμημάτων στις αντίστοιχες διαδικασίες παραγωγής.

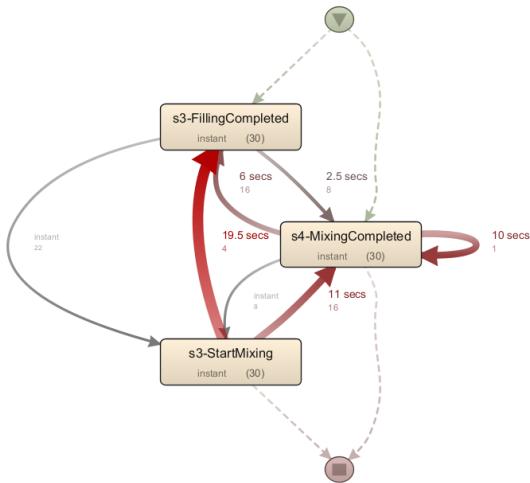
Για τον περιορισμό ταυτόχρονης μεταφοράς η κατάσταση είναι λίγο πιο περίπλοκη, αφού πρέπει να εξασφαλίσουμε ότι το σιλό δεν καθυστερεί τη μεταφορά επειδή δεν έχει αδειάσει το κάτω σιλό, αλλά επειδή χρησιμοποιείται ο αγωγός από την άλλη διαδικασία. Για τη διαδικασία τύπου A όταν ορίσουμε στο φίλτρο attributes τις δραστηριότητες s1-FillingCompleted, lgpA-StartTransfer, s4-PouringCompleted, s2-PouringCompleted και lgpB-TransferCompleted. Θα εστιάσουμε σε περιπτώσεις όπου το s1 έχει γεμίσει και περιμένει να αδειάσει το s2, για να ξεκινήσει η μεταφορά. Παρατηρούμε ότι 10/30 φορές η διαδικασία τύπου A, χρειάζεται να περιμένει την χρήση του αγωγού από την άλλη διαδικασία προκαλώντας μέση καθυστέρηση 1.5 sec. Αντίστοιχα, για την γραμμή παραγωγής B όταν ακολουθήσουμε την ίδια διαδικασία ψάχνοντας για περιπτώσεις στις οποίες το s2 έχει γεμίσει και δεν ξεκινά αμέσως η μεταφορά του υγρού, αλλά περιμένει να τελειώσει το άδειασμα του s1. Στα σχήματα 4.20 και 4.21, φαίνονται τα process map των παραπάνω τμημάτων στις αντίστοιχες διαδικασίες παραγωγής. Παρατηρούμε ότι εντοπίζονται 12/30 φορές στις οποίες το s2 περιμένει να τελειώσει η χρήση του αγωγού από τη γραμμή παραγωγής A, προκαλώντας μέση καθυστέρηση 2 sec στη διαδικασία τύπου B. Αντίστοιχα, στις διαδικασίες κάθε τύπου εντοπίζονται και περιπτώσεις στις οποίες η μεταφορά δεν ξεκινά αμέσως, περιμένοντας το άδειασμα του κάτω σιλό. Στον πίνακα 4.12, συνοψίζονται οι μέσες χρονικές καθυστέρησεις που προκαλούν οι περιορισμοί του συστήματος μαζί με τη συχνότητα τους, στις διαδικασίες παραγωγής κάθε τύπου.

	Περιορισμός μεταφοράς	Περιορισμός ανάδευσης
Τύπος A	2.5 sec (13%)	1.5 sec (26%)
Τύπος B	2.5 sec (33%)	2 sec (40%)

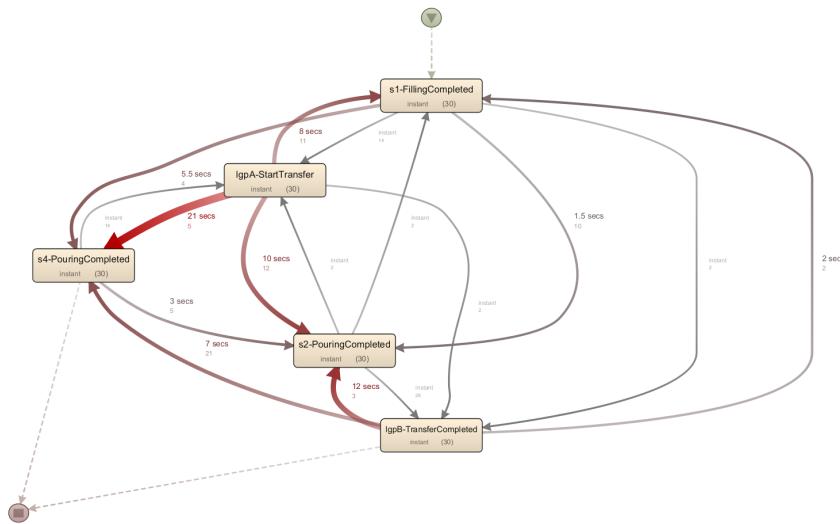
**Πίνακας 4.12:** Χρονικές επιβαρύνσεις λόγω περιορισμών



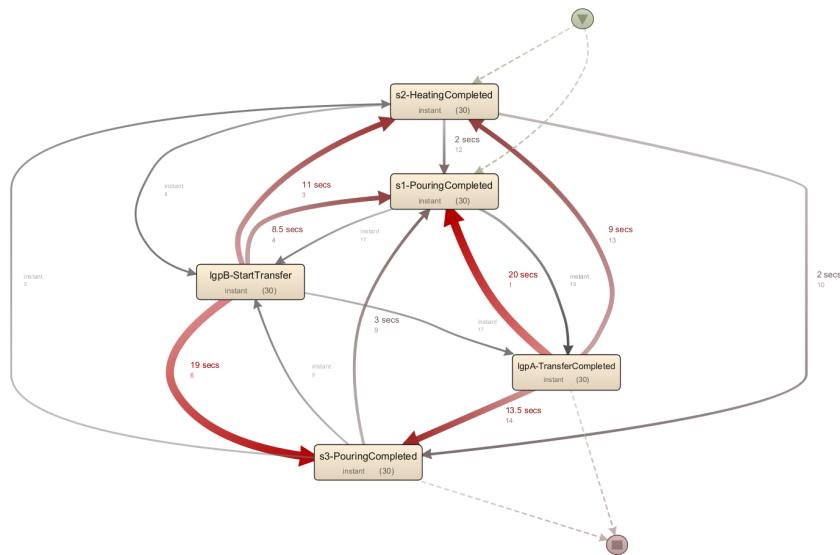
Σχήμα 4.18: Περιορισμός ανάδευσης στη διαδικασία τύπου Α



Σχήμα 4.19: Περιορισμός ανάδευσης στη διαδικασία τύπου Β



Σχήμα 4.20: Περιορισμός μεταφοράς στη διαδικασία τύπου Α



Σχήμα 4.21: Περιορισμός μεταφοράς στη διαδικασία τύπου Β

---

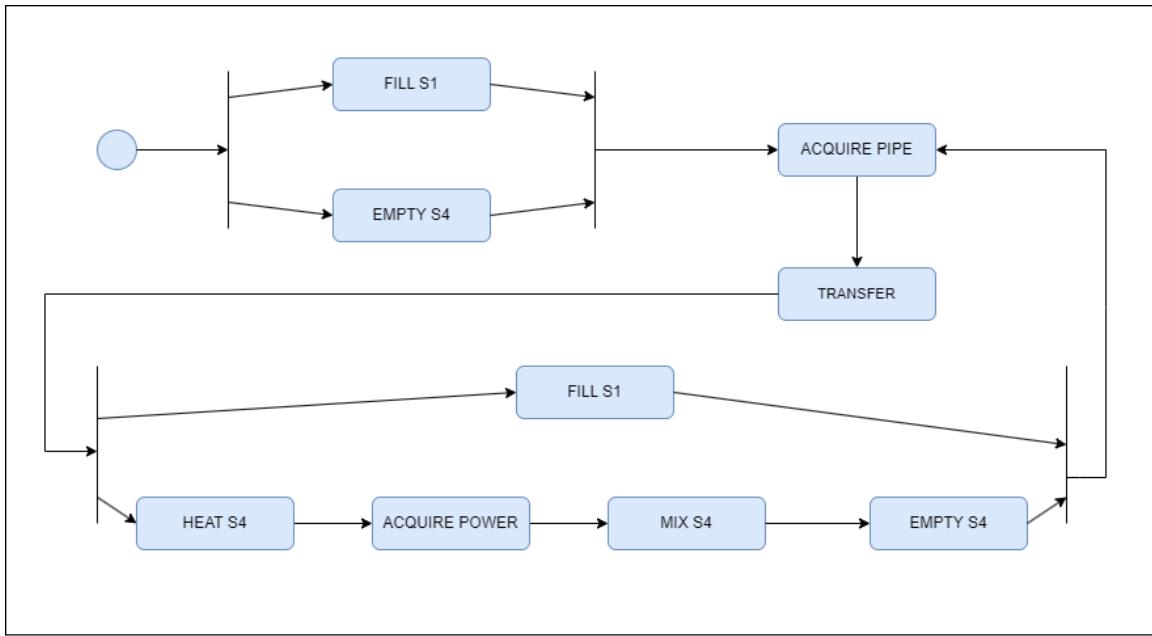
## Κεφάλαιο 5

### Αποτελέσματα

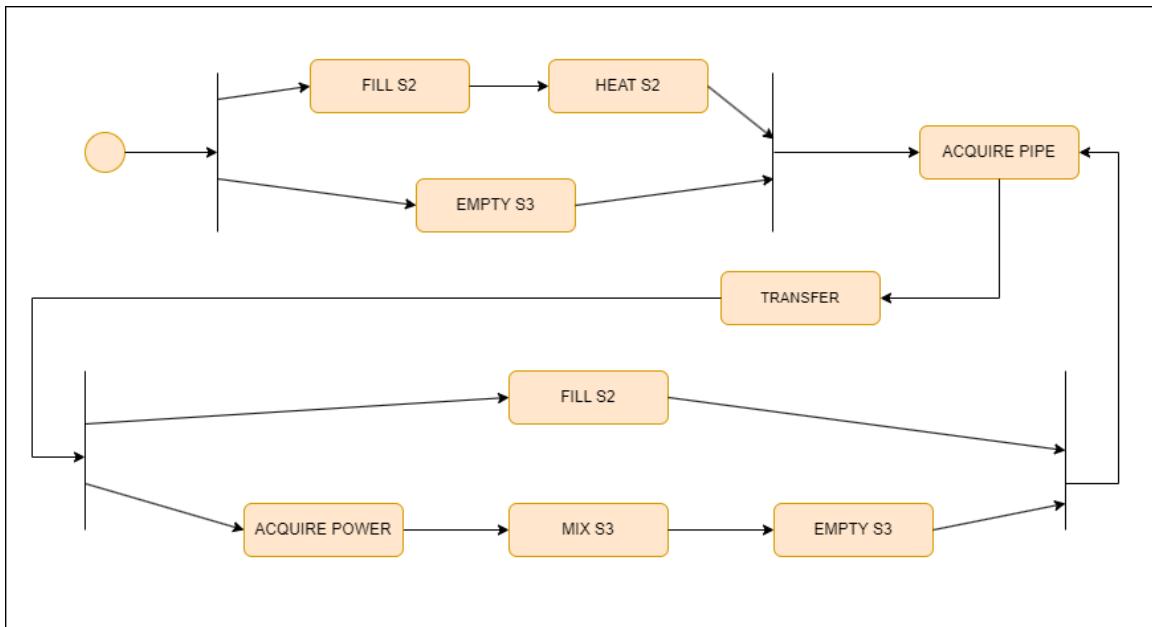
## 5.1 Αξιολόγηση ευρημάτων

### Ανακάλυψη διαδικασίας

Η ανακάλυψη της διαδικασίας και η κατασκευή των μοντέλων διαδικασιών μας δίνει μια πρώτη εικόνα του τι συμβαίνει πραγματικά στο σύστημα και μας δίνει τη δυνατότητα να συγχρίνουμε τα προϋπάρχοντα διαγράμματα με αυτά. Ελέγχοντας το heuristic net της παραγωγής τύπου A, επιβεβαιώνουμε ότι η διαδικασία είναι πανομοιότυπη με αυτή που περιγράφεται θεωρητικά και αποτυπώνεται στο σχήμα 5.1, δηλαδή οι δραστηριότητες ακολουθούν την αναμενόμενη σειρά ενώ τα αρχικά και τελικά συμβάντα (πράσινος και πορτοκαλί κόμβος) είναι αυτά που αναμέναμε. Μέσω του μοντέλου αυτού είναι εφικτό να εντοπίσουμε το πως οι δραστηριότητες ακολουθούνται μεταξύ τους, όταν η γραμμή παραγωγής δεν εκτελείται μόνο για μια φορά αλλά συνεχόμενα και επίσης να επιβεβαιώσουμε ότι το λογισμικό ενεργοποιεί τις διεργασίες που αναμένουμε στο φυσικό επίπεδο. Βλέπουμε ότι ο τερματισμός της χρήσης του αγωγού (lgpA-TransferCompleted) ενεργοποιεί το γέμισμα του s1 (s1-StartFilling), ενώ το πέρας του αδειάσματος του s4 (s4-PouringCompleted) ενεργοποιεί την εκκίνηση της μεταφοράς μέσω του αγωγού (lgpA-StartTransfer). Με τον ίδιο τρόπο, ελέγχοντας το heuristic net της παραγωγής τύπου B, δεν βλέπουμε σημαντικές αποκλίσεις από τη θεωρητική διαδικασία και το προϋπάρχον διάγραμμα (σχήμα 5.2). Πάλι επιβεβαιώνουμε ότι οι δραστηριότητες ακολουθούν την αναμενόμενη σειρά και εντοπίζουμε τα αρχικά και τελικά συμβάντα της διαδικασίας. Αυτό που γίνεται εμφανές σε σχέση με την προηγούμενη διαδικασία, είναι ότι ο διαχωρισμός της ανάδευσης και θέρμανσης σε 2 στοιχεία διευρύνει τους τρόπους με τους οποίους ακολουθούνται οι δραστηριότητες όταν η γραμμή παραγωγής εκτελείται συνεχόμενα. Επίσης ισχύει ότι και στην διαδικασία τύπου A, σχετικά τις δραστηριότητες που ακολουθούν ο τερματισμός χρήσης του αγωγού και το πέρας αδειάσματος του s3, επιβεβαιώνοντας ότι οι γραμμές παραγωγής λειτουργούν ανεξάρτητα αλλά με κοινό σχεδιασμό.



**Σχήμα 5.1:** Activity diagram παραγωγής τύπου A [6]



**Σχήμα 5.2:** Activity diagram παραγωγής τύπου B

## Έλεγχος συμμόρφωσης

Η εύρεση του μοντέλου που αναπαριστά καλύτερα τη διαδικασία παραγωγής, πραγματοποιείται μέσα από τον έλεγχο συμμόρφωσης και δείχνει αν το αρχείο καταγραφής επαληθεύει ορθά το μοντέλο και αν ο αλγόριθμος κατάφερε να την αποτυπώσει με ακρίβεια. Βρήκαμε ότι και οι τρεις αλγόριθμοι που εφαρμόσαμε, κατάφεραν να κατασκευάσουν μοντέλα με ικανοποιητικό fitness, κάτι που συνεπάγεται ότι αυτά είναι αντιπροσωπευτικά της παρατηρούμενης συμπεριφοράς και κατά συνέπεια της διαδικασίας παραγωγής κάθε τύπου. Σε ότι αφορά την ακρίβεια (precision) του κάθε μοντέλου είδαμε ότι inductive miner έχει πολύ χαμηλή, με αποτέλεσμα να γενικεύει υπερβολικά τη συμπεριφορά επιτρέποντας αρκετή πρόσθετη, κάτι που φαίνεται και από τον έλεγχο συμμόρφωσης στα ψεύτικα αρχεία καταγραφής τα οποία δεν κατάφερε να διαφοροποιήσει επαρκώς από το αληθινό. Ο heuristic miner έκανε μια πιο ισορροπημένη δουλειά ως προς τις δύο αυτές μετρικές, δίνοντας μεγαλύτερο precision σε σχέση με τον προηγούμενο αλλά χαμηλότερο fitness σε σχέση με τον alpha miner, εντοπίζοντας παράλληλα όλα τα σφάλματα των αρχείων καταγραφής μέσω alignments. Ο alpha miner αν και έδωσε το απόλυτο precision (100%) κατασκεύασε ένα μοντέλο χωρίς ορθότητα, που σημαίνει το μοντέλο παρουσιάζει κάποια δομικά προβλήματα τα οποία αποθαρρύνουν την περαιτέρω χρησιμοποίηση του. Ωστόσο, η επιλογή του μοντέλου που αναπαριστά καλύτερα τη διαδικασία παραγωγής, εξαρτάται κυρίως από τους στόχους μας ως προς το σύστημα. Για παράδειγμα, αν θέλουμε ένα μοντέλο για να επιβλέπουμε την γραμμή παραγωγής του συστήματος για ανίχνευση ανωμαλιών, τότε θα επιλέξουμε έναν αλγόριθμο που οδηγεί σε ένα μοντέλο με υψηλό precision. Με την ίδια λογική, αν θέλουμε ένα μοντέλο για να αναπαραστήσουμε όσο το δυνατόν καλύτερα την ήδη υπάρχουσα συμπεριφορά της γραμμής παραγωγής του συστήματος, τότε θα επιλέξουμε έναν αλγόριθμο που οδηγεί σε ένα μοντέλο με υψηλό fitness.

## Σημεία συμφόρησης

Ο εντοπισμός σημείων των συμφόρησης, δίνει πολύτιμες πληροφορίες για την απόδοση πραγματικού χρόνου της διαδικασίας παραγωγής. Η αξιολόγηση της γνώσης που προκύπτει από τους χρόνους ολοκλήρωσης των δραστηριοτήτων μπορεί να γίνει με τη χρήση μετρικών απόδοσης. Μια τέτοια μετρική στα πλαίσια βιομηχανικών συστημάτων και συστημάτων παραγωγής, είναι ο ρυθμός παραγωγής (production rate)<sup>1</sup>, ο οποίος αναφέρεται στα αγαθά που παράγει ένα σύστημα ανά μονάδα χρόνου. Σε ένα σύστημα παραγωγής λικέρ, η τελευταία μετρική αφορά την παραγωγή σε λίτρα ανά ώρα και είναι ανάλογη με το χρόνο που χρειάζεται το σύστημα για να ολοκληρώσει τις επιμέρους διαδικασίες παραγωγής και να παράγει λικέρ. Το σύστημα ορίζει συγκεκριμένες δραστηριότητες στη διαδικασία παραγωγής στις οποίες δεν υπάρχει εναλλακτική, πράγμα

<sup>1</sup><https://www.investopedia.com/terms/p/production-rate.asp>

που τις καθιστά κρίσιμες για τη λειτουργία του. Με βάση τις παραπάνω παρατηρήσεις και τα δεδομένα που έχουμε στη διάθεση μας, στο σύστημα γίνονται εμφανή 4 τμήματα της κάθε διαδικασίας τα οποία καταλαμβάνουν τον περισσότερο χρόνο, τα Filling, Pouring, Heating και Mixing ενώ όλες οι υπόλοιπες δραστηριότητες εκτελούνται άμεσα. Και τα 4 αυτά τμήματα, αποτελούν αναπόσπαστα κομμάτια της διαδικασίας, καθώς αυτή δεν μπορεί να προχωρήσει χωρίς να έχουν ολοκληρωθεί. Πιο συγκεκριμένα, για τη διαδικασία τύπου A παρατηρούμε ότι η μεταφορά του υγρού είναι αυτή που δημιουργεί την μεγαλύτερη καθυστέρηση και ευθύνεται για το 25% του συνολικού χρόνου ενώ για τη διαδικασία τύπου B είναι το γέμισμα και το άδειασμα των s2 και s3 που καταλαμβάνει έκαστη το 25% του χρόνου της. Επιπλέον, βρήκαμε ότι οι περιορισμοί του συστήματος στην παράλληλη διαδικασία παραγωγής προκαλούν μη αμελητέα χρονική καθυστέρηση στην ολοκλήρωση της κάθε διαδικασίας και επιβαρύνουν περαιτέρω τους παραπάνω χρόνους. Πιο συγκεκριμένα, για την διαδικασία τύπου A, ο περιορισμός μεταφοράς και ανάδευσης εμφανίζεται πιο σπάνια σε σχέση με τη διαδικασία τύπου B προκαλώντας μέση καθυστέρηση 2.5 sec και 1.5 sec αντίστοιχα. Με τον ίδιο τρόπο, βρήκαμε ότι για τη διαδικασία τύπου B προκαλείται μέση καθυστέρηση 2.5 sec και 2 sec. Αυτό που αξίζει να σημειωθεί είναι ότι, αυτή η καθυστέρηση εμφανίζεται τουλάχιστον στο 1/3 των περιπτώσεων που αφορούν τη διαδικασία τύπου B ενώ στη διαδικασία τύπου A εμφανίζεται πιο σπάνια, προκαλώντας όμως παρόμοια χρονική καθυστέρηση. Εστιάζοντας περισσότερο στις περιπτώσεις αυτές, έγινε εφικτό να εντοπίσουμε μια περίπτωση (case) όπου οι δύο επιμέρους διαδικασίες λειτουργησαν παράλληλα με τις μικρότερες δυνατές χρονικές επιβαρύνσεις λόγω των 2 περιορισμών, της οποίας οι χρόνοι φαίνονται στον πίνακα 5.1. Η μελέτη των χρόνων της συγκεκριμένης περίπτωσης και η ανακάλυψη συσχετισμών μεταξύ αυτών, μας δίνει σημαντική γνώση για καλύτερη παραμετροποίηση της γραμμής παραγωγής με σκοπό την πιο αποδοτική λειτουργία του συστήματος.

	s1	s4	s2	s3
Filling	4 sec	2 sec	3 sec	5 sec
Pouring	2 sec	4 sec	5 sec	3 sec
Mixing	-	2.4 sec	-	1.6 sec
Heating	-	1.8 sec	2.8 sec	-

Πίνακας 5.1: Χρόνοι της περίπτωσης 15

Εστιάζοντας στην περίπτωση 15, παρατηρούμε ότι η διαδικασία τύπου A ολοκληρώνει το γέμισμα και την μεταφορά σε 6 sec και η διαδικασία τύπου B ολοκληρώνει το γέμισμα και τη θέρμανση σε 5.8 sec, περιμένοντας 0.2 sec. Αντίστοιχα, η διαδικασία τύπου A ολοκληρώνει τη θέρμανση και ανάδευση σε 4.2 sec, ενώ η διαδικασία τύπου B ολοκληρώνει την μεταφορά σε 5 sec, δηλαδή δεν χρειάζεται να περιμένει τον τερματισμό της ανάδευσης στο A. Αυτές οι δύο παρατηρήσεις κάνουν ορατές τις χρονικές

συσχετίσεις μεταξύ των των δύο διαδικασιών, υποδεικνύοντας ένα σενάριο στο οποίο παρατηρείται η μικρότερη δυνατή συμφόρηση στην παράλληλη διαδικασία.

## 5.2 Στρατηγική βελτιστοποίησης

Τα παραπάνω ευρήματα που προκύπτουν μέσω της εξόρυξης διαδικασιών μας δίνουν σημαντική γνώση για το σύστημα η οποία προηγουμένως δεν ήταν ορατή. Αρχικά, έχουμε στη διάθεση μας μοντέλα που κατασκευάστηκαν μέσα από αξιόπιστους αλγόριθμους ανακάλυψης και αποτυπώνουν την πραγματική συμπεριφορά του συστήματος, σε σχέση με θεωρητικές περιγραφές και ανθρώπινα μοντέλα. Επίσης, ο έλεγχος συμμόρφωσης προσφέρει έναν έγκυρο τρόπο να ελέγχουμε τη διαδικασία παραγωγής μέσα από τα αρχεία καταγραφής που παράγει το σύστημα. Τέλος, ο έλεγχος των περιορισμών και ο εντοπισμός των σημείων συμφόρησης μας υποδεικνύουν τμήματα της διαδικασίας που είναι κρίσιμα για τη λειτουργία του συστήματος κάνοντας ορατό ένα πλαίσιο βελτιώσεων.

Αυτές οι αξιόπιστες πληροφορίες, κάνουν δυνατή την θεμελίωση προτάσεων για την βελτίωση της διαδικασίας παραγωγής και παρέχουν μια εναλλακτική οπτική ως προς την υλοποίηση του συστήματος. Όπως αναφέρθηκε στο κεφάλαιο 3.2, ο τρίτος τύπος της εξόρυξης διαδικασιών αφορά τον εμπλουτισμό (enhancement), ο οποίος βασίζεται στην ανάλυση που προκύπτει από τις δύο προηγούμενες τεχνικές, την ανακάλυψη διαδικασιών και τον έλεγχο συμμόρφωσης. Ωστόσο, δεν υπάρχουν αντίστοιχοι αλγόριθμοι ή αυτοματοποιημένες μέθοδοι που να εκτελούν εμπλουτισμό διαδικασιών όπως στις άλλες τεχνικές, καθώς αυτό εξαρτάται σε μεγάλο βαθμό από πράγματα που διαφέρουν για κάθε διαδικασία [12]. Για το λόγο αυτό, η στρατηγική βελτιστοποίησης θα βασιστεί σε διαισθητικές παρατηρήσεις που προκύπτουν από τα αποτελέσματα της εφαρμογής, αλλά και μέσα από τη γνώση που προέκυψε από την εξόρυξη διαδικασιών να ορίσουμε τις παρακάτω κατευθύνσεις ως προς τη βελτιστοποίηση του συστήματος.

1. Η πρώτη κατεύθυνση οδηγεί προς τον επανασχεδιασμό της συνολικής αρχιτεκτονικής του συστήματος, με γνώμονα το συντονισμό των δύο διαδικασιών παραγωγής μέσω αξιοποίησης των δεδομένων και ελέγχου των παραμέτρων του συστήματος σε πραγματικό χρόνο. Αυτός ο συντονισμός θα στοχεύει στην ελάττωση της εμφάνισης των περιορισμών ανάδευσης και μεταφοράς στις δύο διαδικασίες παραγωγής, με σκοπό την μείωση της χρονικής επιβάρυνσης που αυτοί προκαλούν στην παράλληλη διαδικασία και κατά συνέπεια την μείωση της συμφόρησης στο σύστημα. Κάτι τέτοιο κάνει ορατές τις προοπτικές εξερεύνησης νέων προτύπων σχεδιασμού για το σύστημα, είτε επέκτασης του ήδη υπάρχοντος πλαισίου με βάση το IEC 61131.
2. Η δεύτερη κατεύθυνση οδηγεί προς την τροποποίηση των φυσικών στοιχείων στο plant. Κάτι τέτοιο προϋποθέτει τον επαναπροσδιορισμό του τρόπου με τον οποίο ακολουθούνται οι διαδικασίες παραγωγής και καθιστά μια πιο ριζική αλλαγή στο σύστημα. Αυτό σημαίνει ότι, θα πρέπει να εξεταστούν εναλλακτικές διατάξεις των στοιχείων ψέρμανσης και ανάδευσης εντός των σιλό ή και προσθήκη περισσότερων τέτοιων στοιχείων στο σύστημα, με σκοπό να βρεθεί η ιδανική διάταξη στην οποία θα αποφεύγεται η συχνή εμφάνιση περιορισμών και θα μεγιστοποιείται η απόδοση του συστήματος μέσα από ταχύτερη εκτέλεση των διαδικασιών. Το τελευταίο μπορεί να επιτευχθεί από την αναβάθμιση των φυσικών στοιχείων με γνώμονα την αυξημένη ισχύ, χωρητικότητα και ταχύτητας μεταφοράς.
3. Η τρίτη κατεύθυνση οδηγεί προς την αντιμετώπιση των πιθανών προβλημάτων που ανακύπτουν εξαιτίας των περιορισμών που επιβάλλει το σύστημα. Αυτό δημιουργεί κάποια ερωτήματα σχετικά με το αν πρέπει να αλλάξει η ροή της διαδικασίας ή αν πρέπει να παραμένει ίδια ανεξάρτητα με τι είδους περιορισμούς συναντά. Για παράδειγμα, ένας περιορισμός που αναστέλλει τη διαδικασία παραγωγής ενός τύπου για αρκετά μεγάλο χρονικό διάστημα, μπορεί να απαιτεί την επανεκτέλεση κάποιας κύριας διαδικασίας (ανάδευσης ή ψέρμανσης) ή την απόρριψη του υγρού. Κάτι τέτοιο, απαιτεί τον αναθεώρηση της λογικής με την οποία το κυβερνοφυσικό σύστημα ενεργοποιεί ή απενεργοποιεί την ροή λειτουργίας της διαδικασίας κάθε τύπου και τον τρόπο με τον οποίο ακολουθεί η κάθε διαδικασία ακολουθεί τις δραστηριότητες που εμπεριέχονται σε αυτή.

## Κεφάλαιο 6

# Συμπεράσματα και προοπτικές

Η παρούσα εργασία ανέδειξε τον κλάδο της εξόρυξης διαδικασιών ως μια μεθοδολογία που μπορεί να συμβάλλει θετικά στο πλαίσιο της Βιομηχανίας 4.0 και πιο συγκεκριμένα στα κυβερνοφυσικά συστήματα. Μελέτησε τα θεμέλια της επιστήμης της εξόρυξης διαδικασιών και των κυβερνοφυσικών συστημάτων και έχανε μια βιβλιογραφική ανασκόπηση σε εφαρμογές της σε συστήματα βιομηχανικού αυτοματισμού. Επιχείρησε να εφαρμόσει τη γνώση που παρήγαγε, μέσω ενός case study πάνω σε ένα αρχείο καταγραφής ενός τέτοιου συστήματος. Αυτό το case study κατάφερε να κάνει περισσότερο εμφανείς τις προοπτικές της εξόρυξης διαδικασιών στο πλαίσιο των κυβερνοφυσικών συστημάτων. Χρησιμοποιώντας το σύστημα σαν test-bed, έγινε δυνατό να εφαρμοστεί η μεθοδολογία σε ένα τέτοιο σύστημα και να εξερευνηθούν οι τρόποι με τους οποίους μπορεί να συμβάλλει στην βελτίωση του. Η εξόρυξη διαδικασιών κατάφερε να αποτυπώσει τη διαδικασία παραγωγής όπως είναι (as is) σε αντίθεση με θεωρητικές περιγραφές ή μοντέλα κατασκευασμένα από ανθρώπους, όπως διαγράμματα ροής (as designed). Πρότεινε μια αξιόπιστη τεχνική ελέγχου του συστήματος σε ανώμαλη συμπεριφορά και εντοπισμό σφαλμάτων. Έκανε ορατό ένα πλαίσιο, μέσω του οποίου μπορούν να ελεγχθούν τα στοιχεία του συστήματος ως προς την ορθή λειτουργία τους. Κατέγραψε με ακρίβεια τις χρονικές επιπτώσεις που προκαλούν οι περιορισμοί του συστήματος δίνοντας έμφαση στα σημεία συμφόρησης. Τέλος, αξιοποιώντας αυτή τη γνώση έδωσε κατευθύνσεις γύρω από τη μελλοντική ανάπτυξη και τους εναλλακτικούς τρόπους σχεδίασης της αρχιτεκτονικής του, με σκοπό την πιο αποδοτική λειτουργία του.

Σε συνέχεια της έρευνας γύρω από την εξόρυξη διαδικασιών στο σύστημα, προτείνονται ορισμένες μελλοντικές κατευθύνσεις. Αρχικά, καθώς το case study υλοποιήθηκε σε ένα αρχείο καταγραφής του προσωμοιωτή του συστήματος δεν έγινε εμφανής η πλήρης δυναμική της εξόρυξης διαδικασιών, που περιλαμβάνει και τον έλεγχο της τήρησης των απαιτήσεων. Αυτό σημαίνει ότι, σε μια πραγματική υλοποίηση του συστήματος θα μπορούσε να ελεγχθεί η διαδικασία στο αν πληρεί τις προδιαγραφές στις παραμέτρους που επιβάλλει ο χρήστης, όπως για παράδειγμα την επιλογή του χρόνου θέρμανσης και α-

---

νάδευσης, καθώς και αν τηρεί τους περιορισμούς μεταφοράς και ανάδευσης σε κάθε διαδικασία παραγωγής. Κάτι τέτοιο δεν θα είχε νόημα να συμβεί στα πλαίσια ενός προσομοιωτή και σε ένα σχετικά μικρό δείγμα της συμπεριφοράς του συστήματος, ο οποίος είναι δεδομένο ότι λειτουργεί απρόσκοπτα. Παράλληλα, μέσα από το αρχείο καταγραφής μια πραγματικής υλοποίησης θα μπορούσε να δοθεί έμφαση και στα συμβάντα του φυσικού επιπέδου εκτός του κυβερνητικού. Ακόμη, θα είχε ενδιαφέρον να συγκριθούν τα αποτελέσματα από μια τέτοια υλοποίηση στα αποτελέσματα από το case study της παρούσας εργασίας, με σκοπό να αξιολογηθεί το κατά πόσο η προσομοίωση έχει κοινά με πραγματικές συνθήκες.

Τέλος, μια άλλη μελλοντική κατεύθυνση ως προς την υλοποίηση του case study θα μπορούσε να αφορά την εναλλακτική προσέγγιση όσον αφορά τις τεχνικές ανακάλυψης διαδικασιών και ελέγχου συμμόρφωσης. Η εξόρυξη διαδικασιών ως ένας ολοένα πιο εξελισσόμενος κλάδος κάνει εφικτό να εξεταστούν και εναλλακτικοί ή νέοι αλγόριθμοι ανακάλυψης που κατασκευάζονται, όπως genetic miner, correlation miner και split miner, νέοι τρόποι ελέγχου συμμόρφωσης όπως footprints και temporal profile, καθώς και άλλα είδη εξόρυξης διαδικασιών όπως semantic-based [2], social network analysis [15] και άλλα.

## Παράρτημα

Τα scripts που χρησιμοποιήθηκαν για την υλοποίηση του case study, όλα τα αρχεία καταγραφής, καθώς και τα αρχεία με τα αναλυτικά αποτελέσματα της εκτέλεσης τους βρίσκονται στον παρακάτω σύνδεσμο.

[www.github.com/manosmin/lps-casestudy](https://www.github.com/manosmin/lps-casestudy)

## Αναφορές

- [1] C. Günther and A. A. Rozinat, «Disco: discover your processes», *International Conference on Business Process Management*, pp. 40–44, Jan. 2012. [Online]. Available: <http://ceur-ws.org/Vol-940/paper8.pdf>.
- [2] J. E. Ingvaldsen and J. A. Gulla, «Industrial application of semantic process mining», *Enterprise information systems*, vol. 6, no. 2, pp. 139–163, May 2012. DOI: 10.1080/17517575.2011.593103. [Online]. Available: <https://doi.org/10.1080/17517575.2011.593103>.
- [3] W. M. P. Van Der Aalst, A. A. Adriansyah, A. K. A. De Medeiros, *et al.*, *Process Mining manifesto*. Jan. 2012, pp. 169–194. DOI: 10.1007/978-3-642-28108-2\{\_}19. [Online]. Available: [https://doi.org/10.1007/978-3-642-28108-2\\_19](https://doi.org/10.1007/978-3-642-28108-2_19).
- [4] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, «Internet of Things (IoT): A vision, architectural elements, and future directions», *Future Generation Computer Systems*, vol. 29, no. 7, pp. 1645–1660, Sep. 2013. DOI: 10.1016/j.future.2013.01.010. [Online]. Available: <https://doi.org/10.1016/j.future.2013.01.010>.
- [5] S. J. J. Leemans, D. Fahland, and W. M. P. Van Der Aalst, *Discovering Block-Structured Process Models from Event Logs - A Constructive Approach*. Jan. 2013, pp. 311–329. DOI: 10.1007/978-3-642-38697-8\{\_}17. [Online]. Available: [https://doi.org/10.1007/978-3-642-38697-8\\_17](https://doi.org/10.1007/978-3-642-38697-8_17).
- [6] K. Thramboulidis, «A framework for the implementation of industrial automation systems based on PLCs.», *arXiv (Cornell University)*, Feb. 2014. [Online]. Available: <https://arxiv.org/pdf/1402.3920.pdf>.
- [7] J. Lee, B. Bagheri, and H.-A. Kao, «A Cyber-Physical Systems architecture for Industry 4.0-based manufacturing systems», *Manufacturing Letters*, vol. 3, pp. 18–23, Jan. 2015. DOI: 10.1016/j.mfglet.2014.12.001. [Online]. Available: <https://doi.org/10.1016/j.mfglet.2014.12.001>.

- [8] K. Stouffer, J. A. Falco, and K. Scarfone, «Guide to Industrial Control Systems (ICS) Security», Tech. Rep., Jun. 2015. DOI: 10.6028/nist.sp.800-82r2. [Online]. Available: <https://doi.org/10.6028/nist.sp.800-82r2>.
- [9] K. Thramboulidis and F. Christoulakis, «UML4IoT—A UML-based approach to exploit IoT in cyber-physical manufacturing systems», *Computers in Industry*, vol. 82, pp. 259–272, Oct. 2016. DOI: 10.1016/j.compind.2016.05.010. [Online]. Available: <https://doi.org/10.1016/j.compind.2016.05.010>.
- [10] W. Van Der Aalst, *Process Mining: Data Science in action*. Apr. 2016. [Online]. Available: [https://vipa.wiwi.uni-saarland.de/wordpress/wp-content/uploads/2014/11/14WS\\_BWinfo\\_Schlueko\\_ProcessMining.pdf](https://vipa.wiwi.uni-saarland.de/wordpress/wp-content/uploads/2014/11/14WS_BWinfo_Schlueko_ProcessMining.pdf).
- [11] S. S. Pourmirza, R. Dijkman, and P. P. Grefen, «Correlation Miner: Mining business process models and event correlations without case identifiers», *International Journal of Cooperative Information Systems*, vol. 26, no. 02, p. 1742002, May 2017. DOI: 10.1142/s0218843017420023. [Online]. Available: <https://doi.org/10.1142/s0218843017420023>.
- [12] M. Halaška and R. Šperka, «Process Mining – the Enhancement of Elements Industry 4.0», *International Conference on Computer and Information Sciences (ICCOINS)*, Aug. 2018. DOI: 10.1109/iccoins.2018.8510578. [Online]. Available: <https://doi.org/10.1109/iccoins.2018.8510578>.
- [13] D. B. Myers, S. Suriadi, K. Radke, and E. Foo, «Anomaly detection for industrial control systems using process mining», *Computers & Security*, vol. 78, pp. 103–125, Sep. 2018. DOI: 10.1016/j.cose.2018.06.002. [Online]. Available: <https://doi.org/10.1016/j.cose.2018.06.002>.
- [14] S. Tay, T. Lee, N. Hamid, and A. Ahmad, «An Overview of Industry 4.0: Definition, Components, and Government Initiatives», *Journal of Advanced Research in Dynamical and Control Systems*, vol. 10, pp. 1379–1387, Dec. 2018. [Online]. Available: <https://www.jardcs.org/backissues/abstract.php?archiveid=6328>.
- [15] J. Abonyi and G. Dorgo, «Process mining in production systems», *2019 IEEE 23rd International Conference on Intelligent Engineering Systems (INES)*, Apr. 2019. DOI: 10.1109/ines46365.2019.9109537. [Online]. Available: <https://doi.org/10.1109/ines46365.2019.9109537>.
- [16] A. Berti, S. J. Van Zelst, and W. M. P. Van Der Aalst, «Process Mining for Python (PM4PY): bridging the gap between process- and data science», *arXiv (Cornell University)*, May 2019. [Online]. Available: <https://arxiv.org/pdf/1905.06169.pdf>.

- [17] M. Siek and R. M. G. Mukti, «Process Mining with Applications to Automotive Industry», *IOP Conference Series: Materials Science and Engineering*, vol. 924, no. 1, p. 012033, Sep. 2020. DOI: 10.1088/1757-899x/924/1/012033. [Online]. Available: <https://doi.org/10.1088/1757-899x/924/1/012033>.
- [18] R. Bemthuis, N. Van Slooten, J. J. Arachchige, J. P. S. Piest, and F. A. Bukhsh, «A Classification of Process Mining Bottleneck Analysis Techniques for Operational Support», *18th International Conference on e-Business*, Jan. 2021. DOI: 10.5220/0010578601270135. [Online]. Available: <https://doi.org/10.5220/0010578601270135>.
- [19] F. Zerbato, R. Seiger, G. Di Federico, A. Burattin, and B. Weber, «Granularity in Process Mining: Can We Fix It?», *PROBLEMS 2021: BPM Problems to Solve Before We Die 2021*, pp. 40–44, Sep. 2021. [Online]. Available: <http://ceur-ws.org/Vol-2938/paper-PROBLEMS-40.pdf>.
- [20] E. Batista, «Contributions to Context-Aware Smart Healthcare: A security and privacy perspective», *arXiv (Cornell University)*, Jun. 2022. DOI: 10.48550/arxiv.2206.14567. [Online]. Available: <https://arxiv.org/abs/2206.14567>.
- [21] W. Duo, M. Zhou, and A. Abusorrah, «A survey of Cyber attacks on Cyber Physical Systems: Recent advances and challenges», *IEEE/CAA Journal of Automatica Sinica*, vol. 9, no. 5, pp. 784–800, May 2022. DOI: 10.1109/jas.2022.105548. [Online]. Available: <https://doi.org/10.1109/jas.2022.105548>.
- [22] B. F. W. M. Fazle Rabbi, «Analysis of patient pathways with contextual process mining», *The International Health Data Workshop (HEDA-2022)*, 2022. [Online]. Available: [https://ceur-ws.org/Vol-3264/HEDA22\\_paper\\_1.pdf](https://ceur-ws.org/Vol-3264/HEDA22_paper_1.pdf).
- [23] S. A. Varghese, A. D. Ghadim, A. Balador, Z. Alimadadi, and P. Papadimitratos, «Digital twin-based intrusion detection for industrial control systems», *2022 IEEE International Conference on Pervasive Computing and Communications Workshops and other Affiliated Events (PerCom Workshops)*, Mar. 2022. DOI: 10.1109/percomworkshops53856.2022.9767492. [Online]. Available: <https://doi.org/10.1109/percomworkshops53856.2022.9767492>.
- [24] M. Xavier, V. Dubinin, S. Patil, and V. Vyatkin, «Process mining in industrial control systems», *2022 IEEE 20th International Conference on Industrial Informatics (INDIN)*, Jul. 2022. DOI: 10.1109/indin51773.2022.9976111. [Online]. Available: <https://doi.org/10.1109/indin51773.2022.9976111>.

- [25] A. Berti, S. Van Zelst, and D. Schuster, «PM4Py: A process mining library for Python», *Software Impacts*, vol. 17, p. 100556, Sep. 2023. DOI: 10.1016/j.simpa.2023.100556. [Online]. Available: <https://doi.org/10.1016/j.simpa.2023.100556>.
- [26] W. Commons, *File:industry\_4.0.png — wikimedia commons, the free media repository*, [Online; accessed 4-March-2024], 2023. [Online]. Available: [https://commons.wikimedia.org/w/index.php?title=File:Industry\\_4.0.png&oldid=820198126](https://commons.wikimedia.org/w/index.php?title=File:Industry_4.0.png&oldid=820198126).
- [27] M. Ryalat, H. ElMoaqet, and M. Alfaouri, «Design of a Smart Factory Based on Cyber-Physical Systems and Internet of Things towards Industry 4.0», *Applied sciences*, vol. 13, no. 4, p. 2156, Feb. 2023. DOI: 10.3390/app13042156. [Online]. Available: <https://doi.org/10.3390/app13042156>.