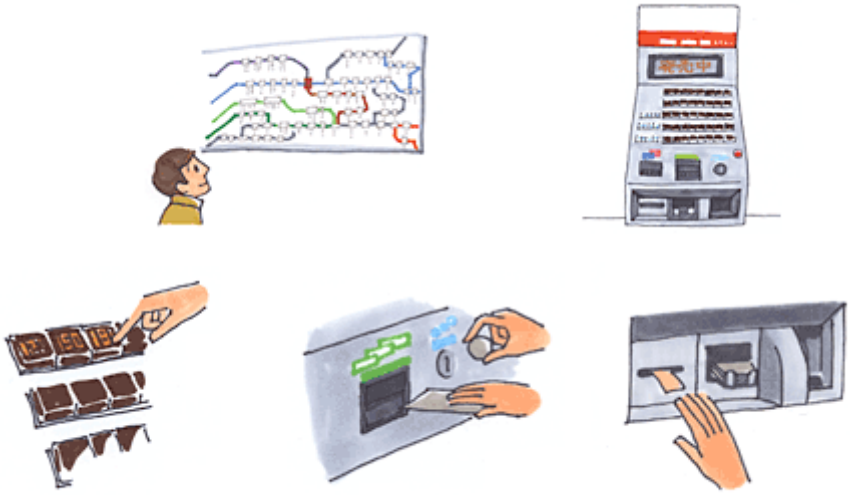
Πολυτεχνείο Κρήτης
Τμήμα: Μηχανικών Παραγωγής & Διοίκησης
Τομέας: Συστήματα Παραγωγής

Συνθετική Μεταπτυχιακή Διατριβή

Μοντελοποίηση και προσομοίωση των λειτουργιών ενός μηχανήματος αυτόματης πώλησης εισιτηρίων με έμφαση στην πρόληψη ανθρώπινων λαθών



Καλλιόπη Μπουράκη
Επιβλέπων Καθηγητής: Θωμάς Κοντογιάννης

Χανιά
Νοέμβριος, 2010

Η παρούσα διατριβή της Καλλιόπης Μπουράκη εγκρίνεται από τους κ.κ.

Επιστημονικός Υπεύθυνος Εργασίας

Κοντογιάννης Θωμάς ~ Αναπληρωτής Καθηγητής

Μέλη Επιτροπής

Μουστάκης Βασίλειος ~ Καθηγητής

Μπιλάλης Νικόλαος ~ Καθηγητής

Ευχαριστίες

Φθάνοντας στο τέλος της περιπλάνησης της μεταπτυχιακής ειδίκευσης, νιώθω την ανάγκη να ευχαριστήσω όσους με στήριξαν, αλλά και όσους μου στάθηκαν εμπόδιο, κάνοντας με πιο επίμονη, για το ρόλο τους στο ταξίδι αυτό.

Πρώτο από όλους θα ήθελα να ευχαριστήσω τον «παράξενο» φίλο μου και συνταξιδιώτη Ηλία Κουρτούδη, τον οποίο εκτιμώ και αγαπώ. Δίχως τη δική του ηθική και υλική υποστήριξη, ίσως να είχα αφήσει ημιτελή την περιπλάνηση αυτή. Αμέριστη ήταν και η συμπαράσταση της ξαδέρφης μου Κατερίνας Τζιγκουνάκη, η οποία εκτός των άλλων βοήθησε στη συντακτική επιμέλεια της παρούσας διεργασίας.

Μεγάλο κεφάλαιο στη ζωή μου, είναι οι φίλες μου Μαρία Καλαϊτζάκη, Φλώρα Γιαμάκη, Γεωργία Σερδενέ, Μαρία Κουλεντάκη, Δήμητρα Τριανταφύλλου και Γεωργία Μούγιου, η ξαδέρφη μου Κατερίνα Καναρά, αλλά και ο φίλος μου Γιάννης Αυγουλέας, με τους οποίους μοιράστηκα τα άγχη και όλες τις ανασφάλειες μου και οι οποίοι μου έδιναν ώθηση και πνοή να συνεχίσω καθετί που έκανα.

Επίσης, ουσιαστική ήταν η συμβολή του επιστημονικού μου υπεύθυνου, αναπληρωτή καθηγητή Θωμά Κοντογιάννη, που με τις πολύτιμες υποδείξεις του συνέβαλε ενεργά στην πραγμάτωση της παρούσας διατριβής. Ευχαριστώ επίσης θερμά τα μέλη της επιτροπή μου κ. Νικόλαο Μπιλάλη και κ. Μουστάκη Βασίλειο.

Φυσικά, όσο και αν φαίνεται τυπικό, δεν ήταν δυνατό να μην ευχαριστήσω την οικογένεια μου και την συντέκνισσα μου Κατερίνα Χονδράκη, που με βοήθησαν και με βοηθούν σε όλα τα στάδια της ζωής μου.

Τελειώνοντας, θα ήθελα να αφιερώσω την παρούσα μεταπτυχιακή διατριβή, στον παππού και στη γιαγιά μου Μάρκο και Μαρία Κωστάκη, που κατά την εκπόνηση της διατριβής έχασα. Καλό ταξίδι!

Περιεχόμενα

1. Εισαγωγή……..1

1.1 Προσδιορισμός θέματος μεταπτυχιακής διατριβής……..2

1.2 Στόχος της διατριβής……..……3

1.3 Δομή της διατριβής……..……3

2. Θεωρητικό Υπόβαθρο……6

2.1 Διαχείριση ροής εργασιών (Workflow management)……7

2.2 Μεθοδολογία SADT……..……..8

2.3 Μεθοδολογία IDEF0……..……13

2.4 Έγχρωμα δίκτυα Petri – CPN……..……...17

2.5 Υλοποίηση με έγχρωμα δίκτυα Petri……26

3. Περιγραφή Μοντελοποίησης μιας Μηχανής Αυτόματης Πώλησης Εισιτηρίων *(TVM)*……28

3.1 Περιγραφή αρχικού μοντέλου TVM σε IDEF0……..……29

3.2 Καταγραφή, πρόληψη & αντιμετώπιση ανθρώπινων λαθών...36

3.3 Μοντελοποίηση της μηχανής έκδοσης εισιτηρίων *(TVM)* με χρήση CPN-Tools……....38

3.3.1 Περιγραφή μοντέλου μηχανής γρήγορης χρέωσης *(QFM)*...43

3.3.2 Περιγραφή μοντέλου μηχανής πολλαπλής χρέωσης *(MFM)*..54

4. Προσομοίωση……65

4.1 Ανάλυση απόδοσης συστημάτων & προσομοίωση……....66

4.2 Προσομοίωση μηχανών QFM & MFM……68

5. Πειράματα……72

5.1 Περιγραφή πειραμάτων……...73

5.2 Σύγκριση QFM & MFM για 32 δυνατές επιλογές……76

5.3 Σύγκριση QFM & MFM για 48 δυνατές επιλογές……78

5.4 Σύγκριση QFM & MFM για 80 δυνατές επιλογές……..……80

5.5 Σύγκριση QFM & MFM για 144 δυνατές επιλογές……...82

5.6 Σύγκριση QFM & MFM για 192 δυνατές επιλογές……...84

5.7 Σύγκριση QFM & MFM για 240 δυνατές επιλογές……86

6. Συμπεράσματα & Μελλοντική Εργασία……..88

6.1 Συμπεράσματα & τελική αξιολόγηση……....89

6.2 Μελλοντικές κατευθύνσεις έρευνας……..…94

Βιβλιογραφία……..95

Παράρτημα Α……97

1 Εισαγωγή

1.1 Προσδιορισμός θέματος μεταπτυχιακής διατριβής

Η παρούσα διατριβή πραγματεύεται την μοντελοποίηση και την προσομοίωση των λειτουργιών ενός μηχανήματος αυτόματης πώλησης εισιτηρίων (Ticket Vending Machine[1]). Για την αρχική μοντελοποίηση θα χρησιμοποιηθεί η τεχνική Integration Definition for Function Modeling[2] *(IDEF0)*, που θα αποτελέσει βοηθητικό εργαλείο για τη μελέτη της ροής των πληροφοριών μέσα στο σύστημα. Στη συνέχεια, η προσομοίωση και η ανάλυση των ιδιοτήτων του συστήματος θα υλοποιηθούν με τη χρήση των έγχρωμων δικτύων Petri (Colored Petri Nets[3]).

Η IDEF0 ανήκει στην οικογένεια των μεθόδων IDEF και χαρακτηρίζεται ως μια από τις σημαντικότερες γλώσσες μοντελοποίησης και βασίζεται στο γενικότερο πλαίσιο της τεχνικής Structured Analysis and Design Technique[4]. Αποτελείται από μια σειρά διαγραμμάτων ροής πληροφοριών, που συνθέτονται από βέλη και πλαίσια, με ιεραρχική ανάλυση. Η ιεραρχική σχεδίαση του μοντέλου υλοποιείται με τον τμηματικό προγραμματισμό. Ξεκινώντας από ένα γενικότερο, ευρύτερο μοντέλο χωρίς περιττές πληροφορίες και καταλήγουμε στα ειδικότερα μέρη του προβλήματος με κατεύθυνση “από πάνω προς τα κάτω” (top-down approach) έχοντας ως σκοπό τη μείωση της πολυπλοκότητας του προβλήματος και την βαθύτερη και αναλυτικότερη περιγραφή του συστήματος.

Τα δίκτυα Petri αποτελούν ένα μαθηματικό και γραφικό εργαλείο μοντελοποίησης, του οποίου τα κυριότερα χαρακτηριστικά είναι η μοντελοποίηση εργασιών και πληροφοριών με γραφικά, οι δομές δεδομένων και η χρήση κάποιας γλώσσας προγραμματισμού (πχ Standard ML), η προσομοίωση της ροής των εργασιών και η χρήση στοχαστικών μεταβλητών, και η μαθηματική ανάλυση των ιδιοτήτων του μοντέλου

[1] Στο εξής TVM
[2] Στο εξής IDEF0
[3] Στο εξής CPN
[4] Στο εξής SADT

με δένδρα προσεγγισιμότητας και γραμμική άλγεβρα (πίνακες αναλλοίωτων). Για την παρούσα υλοποίηση θα χρησιμοποιηθεί το λογισμικό CPN Tools.

1.2 Στόχος της διατριβής

Ο στόχος της διατριβής είναι η μοντελοποίηση των λειτουργιών ενός μηχανήματος αυτόματης πώλησης εισιτηρίων και των πιθανών ανθρώπινων λαθών κατά τη χρήση του. Γίνεται μελέτη ποικίλων συνεντεύξεων με χρήστες, ανασκόπηση λαθών και παρενεργειών, με σκοπό την καταγραφή πιθανών λαθών και ανεπιθύμητων καταστάσεων του συστήματος.

Προσομοιώνονται δύο τύποι μηχανών αυτόματης πώλησης εισιτηρίων, Quick Fare Machine[5] και Multi Fare Machine[6]. Στον πρώτο τύπο *(QFM)*, ο χρήστης για να κατορθώσει το στόχο του, να παραλάβει δηλαδή το επιθυμητό εισιτήριο, αρκεί η επιλογή δυο μόνο κουμπιών, ενώ στον δεύτερο τύπο για την επίτευξη του στόχου του, θα πρέπει να πατήσει πολλαπλά κουμπιά. Επιπρόσθετα, έμφαση δίνεται στα σχεδιαστικά χαρακτηριστικά της μηχανής που θα μπορούσαν να μειώσουν δυνητικά τα λάθη είτε να επιτρέπουν την έγκαιρη διόρθωσή τους.

Τέλος, πραγματοποιούνται διάφορα πειράματα μεταβολής του αριθμού των επιλογών των χρηστών σε καθένα τύπο μηχανής και μελετάται κατά πόσο οι αλλαγές αυτές επηρεάζουν το χρόνο έκδοσης του επιθυμητού εισιτηρίου καθώς και τα σφάλματα στην έκδοση του εισιτηρίου.

1.3 Δομή της διατριβής

Η παρούσα μεταπτυχιακή διατριβή δομείται στις παρακάτω θεματικές ενότητες:

[5] Στο εξής QFM
[6] Στο εξής MFM

- **Κεφάλαιο 1ο – Εισαγωγή:** Πραγματοποιείται προσδιορισμός του θέματος της μεταπτυχιακής διατριβής καθώς και σύντομη περιγραφή των τεχνικών σχεδίασης και μοντελοποίησης που πρόκειται να χρησιμοποιηθούν. Στη συνέχεια αναφέρεται ο στόχος της εργασίας και ο τρόπος με τον οποίο σκοπεύουμε να προσεγγίσουμε το στόχο αυτό. Τέλος, παρατίθεται η οργανωτική διάρθρωση της εργασίας.
- **Κεφάλαιο 2ο – Θεωρητικό Υπόβαθρο:** Παρέχεται το θεωρητικό υπόβαθρο βάση του οποίου πραγματοποιείται η όλη μελέτη. Αρχικά γίνεται μια μικρή περιγραφή του όρου «ροή εργασιών» καθώς και του τρόπου διαχείρισής τους. Στη συνέχεια επεξηγείται η μεθοδολογία SADT, η οποία αποτελεί τη βάση της τεχνικής μοντελοποίησης IDEF. Παρουσιάζονται τα δομικά στοιχεία και βασικότερα πλεονεκτήματα της τεχνικής IDEF0 που χρησιμοποιείται για την αρχική μοντελοποίηση. Επιπλέον, γίνεται αναλυτική παρουσία των έγχρωμων δικτύων Petri με τα οποία υλοποιείται η τελική μοντελοποίηση και γίνονται οι προσομοιώσεις, ενώ αναφέρονται τα πλεονεκτήματα εκείνα που μας οδήγησαν στην επιλογή της υλοποίησης με τα έγχρωμα δίκτυα Petri.
- **Κεφάλαιο 3ο – Περιγραφή Μοντελοποίησης μιας Μηχανής Αυτόματης Πώλησης Εισιτηρίων *(TVM)*:** Γίνεται η μοντελοποίηση μιας μηχανής αυτόματης πώλησης εισιτηρίων με χρήση της τεχνικής IDEF0, χωρίς να λαμβάνονται υπόψη τα ανθρώπινα λάθη. Γίνεται μελέτη ποικίλων συνεντεύξεων με χρήστες, με σκοπό την καταγραφή των πιθανών λαθών και ανεπιθύμητων καταστάσεων του συστήματος. Μοντελοποιούνται και περιγράφονται με τη βοήθεια του λογισμικού CPN-Tools,δύο τύποι μηχανών γρήγορης χρέωσης εισιτηρίων *(QFM)* και πολλαπλής χρέωσης *(MFM)*.
- **Κεφάλαιο 4ο – Προσομοίωση:** Περιγράφεται η προσθήκη καταγραφέων *(monitors)* για τους δύο τύπους μηχανών, για την παρακολούθηση συγκεκριμένων μεταβλητών και τη διεξαγωγή χρήσιμων συμπερασμάτων. Επίσης, παρουσιάζεται εκτενέστατα η προσομοίωση της λειτουργίας των δύο μηχανών, έτσι ώστε να

καταστεί πιο κατανοητός ο τρόπος λειτουργίας του συστήματος και οι μηχανισμοί αποφυγής ανθρώπινων λαθών.

- **Κεφάλαιο 5ο – Πειράματα:** Περιγράφονται τα πειράματα αξιολόγησης των μηχανών, παρατίθενται τα αποτελέσματά τους, καθώς και πίνακες με τις τιμές των μέτρων απόδοσης για την κάθε μηχανή ξεχωριστά, προκειμένου να γίνει η τελική αξιολόγηση των μηχανών.
- **Κεφάλαιο 6ο – Συμπεράσματα & Μελλοντική Έρευνα:** Εκτίθενται συγκριτικοί πίνακες, αλλά και συγκεντρωτικές γραφικές παραστάσεις με σκοπό την ανάπτυξη των συμπερασμάτων που προκύπτουν από την παρούσα διατριβή, καθώς και την τελική αξιολόγηση των μηχανών QFM και MFM. Τέλος, γίνεται αναφορά σε ορισμένα θέματα που αποτελούν μελλοντικές κατευθύνσεις έρευνας.

Θεωρητικό Υπόβαθρο

2.1 Διαχείριση ροής εργασιών (Workflow management)

Μια ροή εργασίας *(workflow management)*, αποτελείται από μια σειρά συνδεδεμένων βημάτων. Είναι μια απεικόνιση της ακολουθίας των πράξεων ενός ατόμου, μιας ομάδας προσώπων, του προσωπικού ενός οργανισμού, ενός ή περισσότερων απλών ή πολύπλοκων μηχανισμών. Μια ροή εργασίας είναι ένα πρότυπο, που αντιπροσωπεύει μια πραγματική εργασία για περαιτέρω εκτίμηση, όπως η περιγραφή μιας αξιόπιστης επαναλαμβανόμενης αλληλουχίας ενεργειών. Γενικότερα, μια ροή εργασίας είναι ένα μοτίβο δραστηριότητας, που ενεργοποιείται από τη συστηματική οργάνωση των πόρων, από καθορισμένους ρόλους και μάζες, από την ενέργεια και τις ροές πληροφοριών, σε ένα σύστημα παραγωγής[7].

Μια διαδικασία είναι μια πιο συγκεκριμένη έννοια από ότι η ροή εργασιών και μπορεί να εφαρμόζεται για παράδειγμα, σε φυσικές ή βιολογικές διεργασίες. Στο πλαίσιο αντίληψης γύρω από την εργασία, μια διαδικασία που μπορεί να διακριθεί από μια ροή εργασίας, είναι το γεγονός ότι έχει σαφώς καθορισμένες εισροές, εκροές και σκοπούς. Απεναντίας, η έννοια της ροής εργασιών μπορεί να εφαρμοστεί γενικά, σε κάθε συστηματικό τρόπο διεξαγωγής των δραστηριοτήτων.

Ο έλεγχος της ροής εργασιών απομακρύνεται από στατικές έννοιες ελέγχου που ασκούνται στα αποθέματα, που διαχειρίζονται απλώς τις αποθήκες *(buffer)* υλικού ή παραγγελίες και αναφέρεται σε μια δυναμική έννοια του ελέγχου, η οποία διαχειρίζεται τη ροή της ταχύτητας και των κινουμένων ποσοτήτων στη διαδικασία. Ο εν λόγω προσανατολισμός προς τις δυναμικές πτυχές είναι το βασικό θεμέλιο για την προετοιμασία σε πιο προηγμένες εργασίες ελέγχου.

Τα δίκτυα Petri, είναι ένα καθιερωμένο εργαλείο για τη μοντελοποίηση και την ανάλυση των διεργασιών. Μάλιστα συγκεντρώνουν όλες τις απαραίτητες προϋποθέσεις για να μπορέσουν να χρησιμοποιηθούν ως εργαλείο για την εφαρμογή της διαχείρισης

[7] http://en.wikipedia.org/wiki/Workflow

ροής εργασιών. Ακόμα, μπορούν να χρησιμοποιηθούν ως μια γλώσσα σχεδίασης για την ανάλυση των προδιαγραφών σύνθετων ροών εργασιών. Η θεωρία πάνω στην οποία είναι βασισμένα τα δίκτυα Petri, μας δίνει τη δυνατότητα χρησιμοποίησης ισχυρών τεχνικών ανάλυσης, οι οποίες μπορούν να χρησιμοποιηθούν για την επαλήθευση της ορθότητας του μοντέλου ροής των εργασιών. Υπάρχουν τουλάχιστον τρεις σημαντικοί λόγοι για να επιλέξουμε τα δίκτυα Petri [8] :

- Μπορούν να είναι κατανοητά, ευκολόχρηστα αλλά και να περιγράφουν με πληρότητα το σύστημα επειδή έχουν ισχυρό θεωρητικό υπόβαθρο.
- Βασίζονται στις καταστάσεις παρά στα γεγονότα. Έτσι υπάρχει ξεκάθαρος διαχωρισμός, ανάμεσα στην διαθεσιμότητα μιας κατάστασης και την εκτέλεσή της. Ακόμα μπορεί να γίνει η μοντελοποίηση ταυτόχρονα διαθέσιμων (ανταγωνιστικών) διεργασιών.
- Παρέχουν πληρότητα τεχνικών ανάλυσης. Η χρήση προσομοίωσης επιτρέπει την ανάλυση σύνθετων ροών εργασιών και την πιστοποίηση της έλλειψης αδιεξόδων ή της απουσίας μη ενεργών μερών.

2.2 Μεθοδολογία SADT

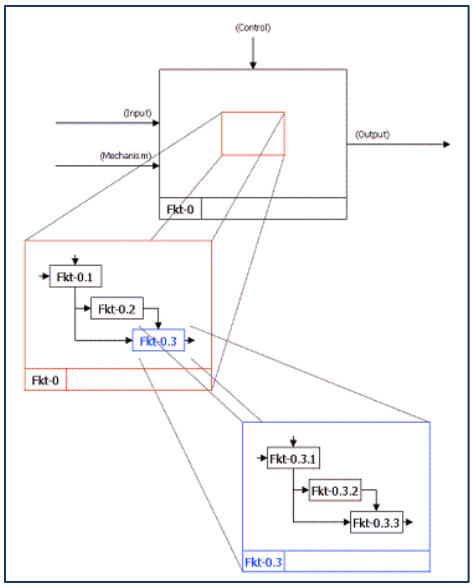
Τη δεκαετία του ‘60 οι εμπορικότερες γλώσσες προγραμματισμού ήταν η Cobol και η Fortran και στη συνέχεια η C και η Basic. Οι απαιτήσεις στο σχεδιασμό και στις προγραμματιστικές τεχνικές ήταν μικρές, όμως με την πάροδο του χρόνου τα συστήματα έγιναν πολυπλοκότερα και τα πληροφοριακά συστήματα όλο και πιο δύσκολα. Για να βοηθήσουν στη διαχείριση του πολύπλοκου λογισμικού, στη μείωση των λαθών, στην εξασφάλιση της κατανόησης των προγραμμάτων αλλά και στη διόρθωση αυτών, στα τέλη της δεκαετίας του ‘60 προέκυψαν ποικίλες μέθοδοι δόμησης. Η μεθοδολογία Δομημένης Ανάλυσης και Σχεδίου *(Structured Analysis and Design Technique)* είναι, μία διαδεδομένη μέθοδος διαγραμματικής περιγραφής και κατανόησης δομημένων

[8] Aalst W.M.P., ‘The Application of Petri Nets to Workflow Management’ The Journal of Circuits - Systems and Computers, 8, (1998), 21-66

πληροφοριακών συστημάτων. Αναπτύχθηκε κατά την περίοδο 1969 – 1973 από τον Douglas T. Ross και την πολυεθνική εταιρία λογισμικού SofTech[9]. Η μεθοδολογία αρχικά εφαρμόστηκε στο project *APT (Automatic Programming Tool)* του *MIT (Massachusetts Institute of Technology),* χρησιμοποιήθηκε δε εκτενέστερα το 1973 από την πολεμική αεροπορία των ΗΠΑ στο πρόγραμμα *"Integrated Computer Aided Manufacturing".* Αργότερα υιοθετήθηκε σε εφαρμογές διαφόρων κλάδων και επιστημονικών πεδίων όπως στη διαχείριση εφοδιαστικής αλυσίδας, στο σχεδιασμό τεχνολογιών, στις τηλεφωνικές επικοινωνίες στα δίκτυα παραγωγής κ.α.[10]

Η τεχνική SADT περιλαμβάνει ένα μεθοδολογικό πλαίσιο για τη μοντελοποίηση των λειτουργιών ενός συστήματος. Η μοντελοποίηση αυτή αποσκοπεί στην περιγραφή των διαφόρων δραστηριοτήτων, των μεταξύ τους σχέσεων αλλά και των πληροφοριακών δεδομένων που απορρέουν από αυτές τις συσχετίσεις, ενώ μεγάλη σπουδαιότητα έχει η περιγραφή της ροής των δεδομένων στο σύστημα. Κατά τη διαδικασία ανάπτυξης του διαγράμματος ροής δεδομένων ο αναλυτής δεν πρέπει να εμπλέκεται σε λεπτομέρειες που αφορούν την ακολουθία της επεξεργασίας. Δεν αποτελεί σκοπό της, η ακριβής περιγραφή του τρόπου με τον οποίο εκτελείται η κάθε διαδικασία.

Ένα από τα κυριότερα πλεονεκτήματα της SADT είναι ότι παρέχεται η δυνατότητα ανάπτυξης ιεραρχικών μοντέλων από το ευρύτερο στα ειδικότερα μέρη του προβλήματος με κατεύθυνση "από πάνω προς τα κάτω" *(top-down approach),* έχοντας ως σκοπό τη μείωση της πολυπλοκότητας του προβλήματος και την βαθύτερη και αναλυτικότερη περιγραφή του συστήματος (εικόνα

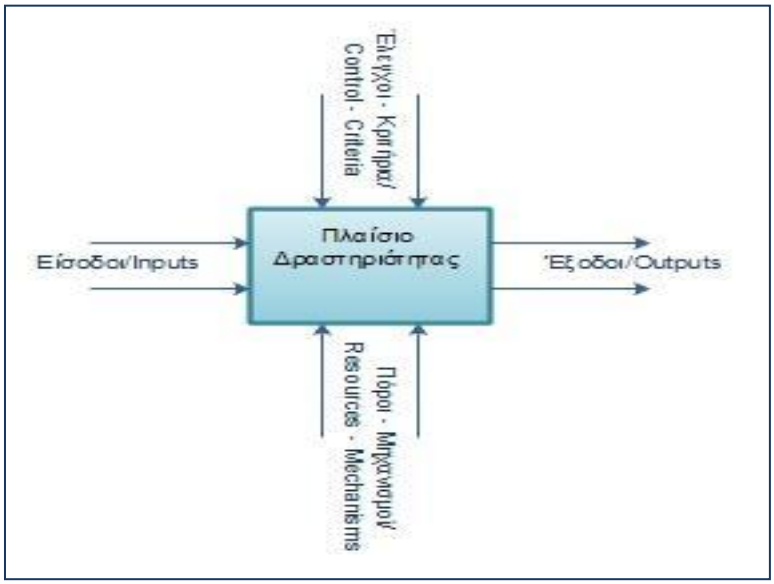


Εικόνα 2.1 Ιεραρχική περιγραφή διαδικασιών

[9] http://en.wikipedia.org/wiki/SADT
[10] Salimifard K., 'Petri net-based modeling of workflow systems: An overview' European Journal of Operations Research, 134, (2001), 664–676

2.1). Τα χρησιμοποιούμενα σύμβολα είναι μόνο δύο, τα πλαίσια *(activity boxes)* που χρησιμοποιούνται για την περιγραφή των δραστηριοτήτων και τα βέλη *(arrows)* με τα οποία συνδέονται τα πλαίσια και υποδηλώνουν την ροή των δεδομένων, γεγονός που καθιστά τη SADT αρκετά απλή σημασιολογικά. Στα μοντέλα δεν περιλαμβάνεται το χρονικό στοιχείο, δηλαδή δεν εξετάζεται η αλληλουχία των δραστηριοτήτων. Αντικείμενο της μοντελοποίησης αποτελεί η ορθή σύνδεση των δραστηριοτήτων και η διατήρηση της συνέχειας στη ροή της πληροφορίας, προκειμένου να επιτευχθεί ο τελικός στόχος.

Στην εικόνα 2.2 φαίνεται ένα βασικό δομικό στοιχείο της SADT. Αποτελείται από ένα ορθογώνιο που ονομάζεται πλαίσιο δραστηριότητας *(activity box),* το οποίο παράγει αποτελέσματα με χρήση των στοιχείων εισόδου, δεδομένου ότι υπάρχουν οι κατάλληλοι πόροι και ικανοποιούνται κάποια συγκεκριμένα κριτήρια εργασίας. Τα πλαίσια δραστηριότητας αναπαριστούν τις λειτουργίες του συστήματος και η ονομασία τους, σχετίζεται με την περιγραφή της λειτουργίας που αναπαρίσταται.



Εικόνα 2.2 Βασικό δομικό στοιχείο της SADT

Μέσω των βελών επιτυγχάνεται η σύνδεση των διαφόρων πλαισίων και κατά συνέπεια, η ανταλλαγή πληροφοριών καθώς και η ροή των εργασιών μέσα στο σύστημα. Τα στοιχεία εισόδου της κάθε διεργασίας προέρχονται από τα στοιχεία εξόδου

προηγουμένων δραστηριοτήτων. Αντίστοιχα οι έξοδοι της κάθε διεργασίας χρησιμεύουν είτε σαν είσοδοι σε άλλες είτε σαν έξοδοι του συστήματος.

Η σύνδεση των πλαισίων χρησιμεύει στην συνεχόμενη ροή και στην περιγραφή της πληροφορίας που διακινείται στο σύστημα. Οι είσοδοι του πλαισίου χωρίζονται σε τρεις κατηγορίες: τα στοιχεία εισόδου *(inputs),* τους πόρους *(resources)* και τους ελέγχους/κριτήρια *(controls/criteria).* Συγκεκριμένα, η κάθε κατηγορία αντιπροσωπεύει τα εξής:

- **Στοιχεία Εισόδου (Inputs):** Τα στοιχεία αυτά τοποθετούνται αριστερά του πλαισίου με σκοπό να τροφοδοτούν την δραστηριότητα και είτε μετασχηματίζονται είτε αναλώνονται κατά τη διάρκεια της παραγωγικής διαδικασίας. Συνήθως αντιπροσωπεύουν πρώτες ύλες, αναλώσιμα κ.α.
- **Πόροι (Resources):** Τοποθετούνται στην κάτω πλευρά του πλαισίου και αντιπροσωπεύουν τα μέσα τα οποία χρησιμεύουν στον μετασχηματισμό που γίνεται από τα στοιχεία εισόδου στα στοιχεία εξόδου. Οι πόροι δεν μεταβάλλονται ούτε καταναλώνονται κατά τη διάρκεια της δραστηριότητας και συνήθως αντιπροσωπεύουν το μηχανολογικό εξοπλισμό, το ανθρώπινο δυναμικό κ.α.
- **Έλεγχοι/Κριτήρια (Controls/Criteria):** Τοποθετούνται στην πάνω πλευρά του πλαισίου και χρησιμεύουν ως πρότυπα για τον έλεγχο της επιτυχούς ολοκλήρωσης της διεργασίας. Τα κριτήρια συνήθως αντιπροσωπεύουν νομοθετικές διατάξεις, πρότυπα ασφάλειας, πρότυπα ποιότητας κ.α.

Οι **έξοδοι (outputs)** του πλαισίου δραστηριότητας τοποθετούνται στη δεξιά πλευρά του πλαισίου και αποτελούν το παραδοτέο της διαδικασίας που εκτελείται από το συγκεκριμένο πλαίσιο, μεταφέροντας επίσης την πληροφορία προς άλλα πλαίσια. Τέτοιου είδους πληροφορίες μπορεί να είναι ανεπιθύμητα συμβάντα κατά τη διάρκεια της διαδικασίας ή προβλήματα στην παραγωγή του επιθυμητού αποτελέσματος.

Ένα στοιχείο εξόδου μπορεί να συνδεθεί:

- με ένα στοιχείο εισόδου ενός διαφορετικού πλαισίου,
- με τον έλεγχο/κριτήρια ενός διαφορετικού πλαισίου,
- με τους πόρους ενός διαφορετικού πλαισίου,
- με ανάδραση σε ένα στοιχείο εισόδου του ίδιου πλαισίου και
- με ανάδραση στον έλεγχο/κριτήρια του ίδιου πλαισίου.

Ως μέτρο για την αντιμετώπιση της πολυπλοκότητας στη σχεδίαση των προγραμμάτων, χρησιμοποιείται η ανάλυση κάποιων δραστηριοτήτων σε μεγαλύτερο βάθος, ακόμα και ο κατακερματισμός τους σε επιμέρους δραστηριότητες. Ο δομημένος προγραμματισμός ενθαρρύνει την ανάλυση του προγράμματος σε επί μέρους τμήματα. Με τη μέθοδο SADT υπάρχει η δυνατότητα υποστήριξης ιεραρχικών διαγραμμάτων. Σκοπός της ιεραρχικής σχεδίασης είναι η διάσπαση του προβλήματος σε μια σειρά από απλούστερα θέματα, τα οποία να είναι εύκολο να επιλυθούν, ενώ παράλληλα δίνεται η δυνατότητα στον αναλυτή να διαχειριστεί καλυτέρα ένα μεγαλύτερο όγκο πληροφοριών. Η ιεραρχική σχεδίαση προγράμματος υλοποιείται με τον τμηματικό προγραμματισμό. Μετά την ανάλυση του προβλήματος σε αντίστοιχα θέματα, καθένα από αυτά αποτελεί ανεξάρτητη ενότητα *(module)* που γράφεται ξεχωριστά από τα υπόλοιπα τμήματα προγράμματος. Ο τμηματικός προγραμματισμός διευκολύνει τη δημιουργία προγράμματος, μειώνει τα λάθη και επιτρέπει ευκολότερη παρακολούθηση και συντήρηση του προγράμματος.

Το αρχικό διάγραμμα αποτελείται από ένα μόνο πλαίσιο ενώ τα υπόλοιπα είθισται να αποτελούνται από τρία έως έξι. Τα διαγράμματα του υψηλού επιπέδου θεωρούνται ως γονείς/γεννήτορες *(parent)* των διαγραμμάτων χαμηλότερου επιπέδου τα οποία ονομάζονται και απόγονοι/παιδιά *(child)*. Κάθε διάγραμμα μπορεί να έχει έως έξι διαγράμματα απογόνους. Η αρχική δραστηριότητα στο ανώτερο διάγραμμα αποκαλείται “δραστηριότητα 0” και συμβολίζεται ως A0. Σκοπός της είναι να αναπαραστήσει τη γενικότερη ιδέα του συστήματος και την οπτική της μοντελοποίησης. Το διάγραμμα αυτό ονομάζεται “διάγραμμα περιεχομένου” ή A-0 και αποτελεί γονέα όλων των παρακάτω

διαγραμμάτων. Η δραστηριότητα Α0 περιλαμβάνει τις υποδραστηριότητες Α1, Α2, Α3 κοκ. Αν αυτές έχουν διάγραμμα απόγονο, οι υποδραστηριότητες τους ονομάζονται με το όνομα του γονέα, προσθέτοντας στο τέλος ένα επιπλέον αριθμό, οποίος σχετίζεται με τον τρόπο διάδοσης της πληροφορίας (πχ οι υποδραστηριότητες της Α1 ονομάζονται Α11, Α12, Α13, της Α2 ονομάζονται Α21, Α22, Α23 κοκ). Σε κάθε επίπεδο της ιεραρχίας προστίθεται, δηλαδή, ένα επιπλέον ψηφίο στο τέλος του ονόματος της κάθε δραστηριότητας. Κατά αυτόν τον τρόπο, κάθε δραστηριότητα αναγνωρίζεται με τρόπο μοναδικό.

Βασική αρχή που πρέπει να ακολουθείται από κάθε δραστηριότητα που αναλύεται σε απογόνους είναι η διατήρηση της συνέχειας στη ροή της πληροφορίας. Η είσοδος και η έξοδος σε κάθε εκλέπτυνση πρέπει να παραμένει η ίδια, ενώ κάθε γονέας μεταβιβάζει στον απόγονό του τα γενικά χαρακτηριστικά του. Οι σχέσεις οι οποίες προέρχονται από τη δραστηριότητα γονεά ονομάζονται διασυνοριακές/εξωτερικές σχέσεις *(boundary arrows)*, ενώ οι σχέσεις οι οποίες αναπτύσσονται μεταξύ δραστηριοτήτων του ίδιου διαγράμματος ονομάζονται εσωτερικές σχέσεις *(internal arrows)*.

Συμπερασματικά, η μέθοδος SADT αποτελεί ένα εύχρηστο και σχετικά απλό συστηματικό τρόπο αναπαράστασης διεργασιών. Αυτό συμβαίνει γιατί χρησιμοποιεί ένα μικρό αριθμό εννοιών και συμβόλων με αποτέλεσμα να είναι εύκολα κατανοητή. Αντιθέτως, αποτελεί μια στατική μέθοδο, καθώς παρουσιάζει μόνο τις δραστηριότητες και τις αλληλεπιδράσεις τους χωρίς να λαμβάνει υπόψη λεπτομέρειες, όπως ο χρόνος που απαιτείται για την περαίωση κάθε εργασίας ή το κόστος της κάθε δραστηριότητας.

2.3 Μεθοδολογία IDEF0

Η IDEF0 αποτελεί ένα μεθοδολογικό πλαίσιο μοντελοποίησης δραστηριοτήτων που περιγράφουν τις λειτουργίες ενός συστήματος και τις αλληλοσυσχετίσεις μεταξύ τους, με απώτερο σκοπό την ανάπτυξη, αναδιοργάνωση και ενοποίηση πληροφοριακών

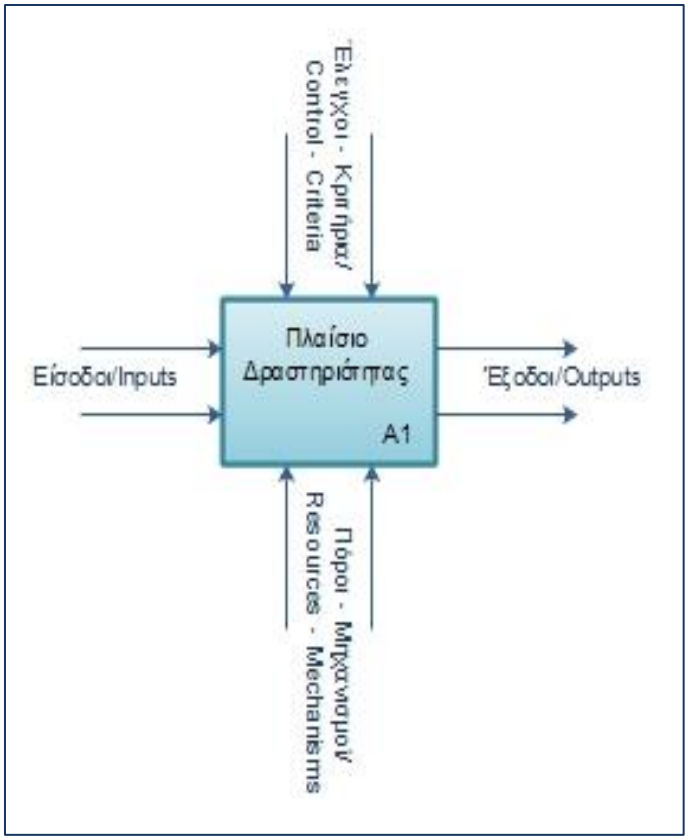
συστημάτων και επιχειρηματικών διαδικασιών. Αποτελεί μέρος της ευρύτερης οικογένειας μεθόδων IDEF που είναι βασισμένη στη γλώσσα μοντελοποίησης SADT που περιγράφηκε παραπάνω[11]. Πρωτοεμφανίστηκε κατά τη διάρκεια της δεκαετίας του '70, όταν η Πολεμική Αεροπορία των ΗΠΑ ανέπτυξε το πρόγραμμα *"Integrated Computer Aided Manufacturing" (ICAM),* επιδιώκοντας να αυξήσει την κατασκευαστική παραγωγικότητα μέσω της συστηματικής εφαρμογής της τεχνολογίας των υπολογιστών. Το πρόγραμμα ICAM προσδιόρισε την ανάγκη για καλύτερη ανάλυση και αποδοτικότερες τεχνικές επικοινωνίας μεταξύ των αναλυτών, όσον αφορά στη βελτίωση της παραγωγικότητας. Ως αποτέλεσμα, το 1981 αναπτύχθηκε μια σειρά από τεχνικές που είναι γνωστή ως τεχνικές IDEF *(ICAM Definition),* οι οποίες στοχεύουν στα ακόλουθα[12]:

- **IDEF0**, χρησιμοποιείται για την παραγωγή λειτουργικών μοντέλων, δηλαδή δομημένων απεικονίσεων λειτουργιών, δραστηριοτήτων και διεργασιών ενός συστήματος ή μιας θεματικής περιοχής.
- **IDEF1**, χρησιμοποιείται για την παραγωγή πληροφοριακών μοντέλων, τα οποία αντιπροσωπεύουν τη δομή και τη σημασιολογία των πληροφοριών ενός συστήματος ή μιας θεματικής περιοχής.
- **IDEF2**, χρησιμοποιείται για την παραγωγή δυναμικών μοντέλων, τα οποία αντιπροσωπεύουν τα χρονικώς μεταβαλλόμενα χαρακτηριστικά της συμπεριφοράς συστήματος ή μιας θεματικής περιοχής.

Η οικογένεια IDEF παραμένει ακόμα και σήμερα ίσως η δημοφιλέστερη τεχνική μοντελοποίησης σε ένα ευρύ φάσμα επιχειρήσεων και χρησιμοποιείται κατά κόρον σε κυβερνητικά αναπτυξιακά προγράμματα καθώς και από το βιομηχανικό αλλά και τον εμπορικό κλάδο.

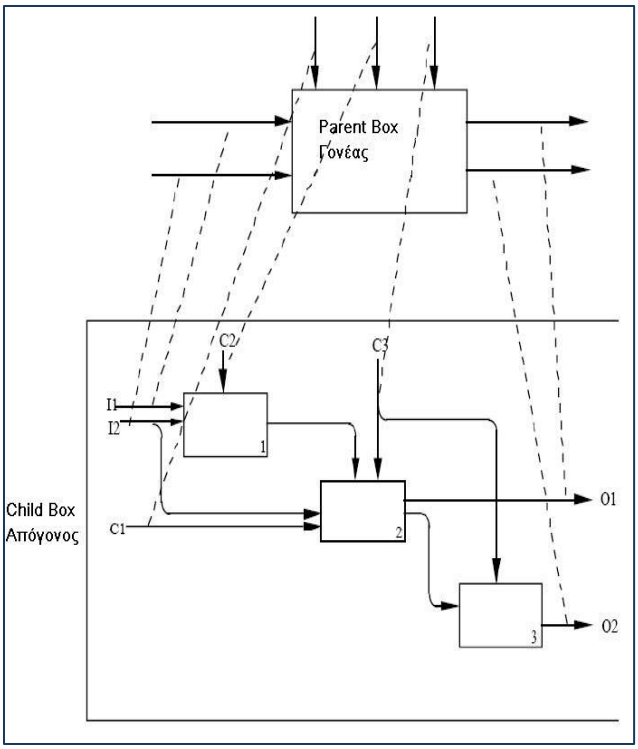
[11] Aalst W.M.P., 'The Application of Petri Nets to Workflow Management' The Journal of Circuits - Systems and Computers, 8, (1998), 21-66
[12] http://en.wikipedia.org/wiki/IDEF

Το βασικό δομικό στοιχείο στην IDEF0 είναι ένα πλαίσιο εισόδων, εξόδων, ελέγχων και μηχανισμών *(Inputs, Controls, Outputs, Mechanisms - ICOM)* όμοιο με εκείνο που περιγράψαμε στη μέθοδο SADT (εικόνα 2.3). Οι είσοδοι είναι δεδομένα και στοιχεία τα οποία, μέσω της περιγραφόμενης διαδικασίας, μετατρέπονται σε εξόδους. Τα στοιχεία ελέγχου παρουσιάζουν τις συνθήκες κάτω από τις οποίες είναι εφικτό να παραχθούν οι ζητούμενες έξοδοι. Οι μηχανισμοί μας δείχνουν τους πόρους που θα χρησιμοποιηθούν για την περάτωση της διαδικασίας και τέλος οι έξοδοι περιγράφουν δεδομένα και στοιχεία τα οποία παράγονται μέσω της περιγραφόμενης διαδικασίας. Όλες οι σχέσεις δεν είναι υποχρεωτικές για κάθε διαδικασία. Συγκεκριμένα, κάθε οντότητα θα πρέπει να έχει τουλάχιστον έναν μηχανισμό ελέγχου ώστε να διασφαλίζεται ότι είναι γνωστός ο τρόπος εκτέλεσης της δραστηριότητας και μία έξοδο για να υπάρχει διασύνδεση με άλλες διαδικασίες, ενώ οι είσοδοι και οι μηχανισμοί είναι σχέσεις προαιρετικές. Επίσης, παρέχεται και η δυνατότητα κάθε οντότητα να έχει περισσότερες από μία σχέσεις σε κάθε κατηγορία οι οποίες διαχωρίζονται έχοντας η κάθε μια τη δική της ονομασία.



Εικόνα 2.3 Βασικό δομικό στοιχείο της IDEF0

Το κυριότερο πλεονέκτημα της μεθόδου IDEF0 έγκειται στο ότι τα μοντέλα που αναπτύσσονται μπορούν να ακολουθήσουν μια ιεραρχική δομή, η οποία αναπτύσσεται από πάνω προς τα κάτω, από γενικότερα χαρακτηριστικά, δηλαδή, προς πιο εξειδικευμένα *(top-down approach)*. Οι κανόνες που διέπουν την ιεραρχία σε ένα σύστημα που περιγράφεται μέσω της IDEF0 είναι ίδιοι με εκείνους που ισχύουν και στην περίπτωση της SADT, πράγμα που οφείλεται στη στενή συγγένεια των δύο μεθόδων. Κατά συνέπεια, τα διαγράμματα της IDEF0 χαρακτηρίζονται από το έντονο στοιχείο της κληρονομικότητας. Κάθε διαδικασία που αναλύεται, μεταβιβάζει στον απόγονό της τα γενικά της χαρακτηριστικά. Έτσι, οι είσοδοι του πλαισίου γονέα αποτελούν και τις εισόδους στο διάγραμμα των απογόνων του (εικόνα 2.4)



Εικόνα 2.4 Ιεραρχική δομή της IDEF0

Μερικά ακόμα ουσιαστικά χαρακτηριστικά της IDEF0 συνοψίζονται παρακάτω:

- Στην IDEF0 υπάρχει απουσία της συνισταμένης του χρόνου, δηλαδή δεν μελετάται η αλληλουχία των δραστηριοτήτων. Αντικείμενο της μοντελοποίησης είναι ο τρόπος επικοινωνίας και σύνδεσης των δραστηριοτήτων μεταξύ τους, η ροή της πληροφορίας μέσα στο σύστημα, καθώς και η επίτευξη των συνολικών στόχων.
- Η γλώσσα IDEF0 χαρακτηρίζεται από απλότητα ως προς τα σημασιολογικά στοιχεία και τα σύμβολα που χρησιμοποιεί. Αναλυτικότερα χρησιμοποιούνται δύο μόνο γραφικά στοιχεία, τα πλαίσια που απεικονίζουν τις λειτουργίες του

συστήματος και τα βέλη που στοχεύουν στην απόδοση του τρόπου των συνδέσεων των λειτουργιών.

- Παρέχεται η δυνατότητα λεκτικής τεκμηρίωσης των περιγραφόμενων από το μοντέλο λειτουργιών (documentation, glossary).
- Τα μοντέλα της IDEF0 αποσκοπούν στην αποδοτικότερη επικοινωνία μεταξύ των αναλυτών, των σχεδιαστών συστημάτων και των χρηστών, δίνοντας έμφαση στη σταδιακή αποκάλυψη της λεπτομέρειας.

Ένα από τα κυριότερα μειονεκτήματα της IDEF0 εντοπίζεται στην έλλειψη του χρονικού στοιχείου στη ροή των διαδικασιών και των πληροφοριών. Το χαρακτηριστικό αυτό όμως, μπορεί να θεωρηθεί και ως ένα από τα σημαντικότερα πλεονεκτήματά της, αφού επιτρέπει στον αναλυτή να επικεντρωθεί στη αλληλουχία των διαδικασιών και να μελετήσει τις αλληλεξαρτήσεις σε λογικό και οργανωτικό επίπεδο, απαλλαγμένος από τους περιορισμούς που θέτει η χρονική διάταξη των εργασιών και ο πραγματικός τρόπος με το οποίο εκτελούνται. Γενικά ένα μοντέλο IDEF0 δίνει απάντηση στο ερώτημα “τι κάνει” το πληροφοριακό σύστημα (λειτουργίες - operations), δηλαδή με ποιόν τρόπο επιδιώκεται η επίτευξη των στόχων και όχι στο “πως ακριβώς εκτελείται” η κάθε δραστηριότητα του συστήματος. Επομένως, η μοντελοποίηση με την IDEF0 αποσκοπεί στην περιγραφή των διαφόρων λειτουργιών και των μεταξύ τους σχέσεων, καθώς και στην κατανόηση και βελτίωση του συστήματος.

2.4 Έγχρωμα δίκτυα Petri - CPN

Τα δίκτυα Petri ή αλλιώς γνωστά ως δίκτυα θέσεων/μεταβάσεων (place/transition - PT Nets) δημιουργήθηκαν το 1962 κατά την εκπόνηση της διδακτορικής διατριβής του Carl Adam Petri και είναι μια γραφική γλώσσα μοντελοποίησης που συνδυάζει γραφικά και μαθηματικά εργαλεία[13]. Στοχεύει στην μοντελοποίηση και στην ανάλυση συστημάτων, τα οποία χαρακτηρίζονται από την ταυτόχρονη εκτέλεση διαφόρων υπολογιστικών

[13] http://en.wikipedia.org/wiki/Petri_net

διεργασιών με δυνατότητα αλληλεπίδρασης μεταξύ τους (concurrent systems), καθώς και συστημάτων τα οποία περιγράφονται ως ασύγχρονα, κατανεμημένα, παράλληλα, στοχαστικά ή στα οποία παρουσιάζονται συγκρούσεις σχετικά με την κατανομή των πηγών[14]. Είναι χαρακτηριστικό ότι τις τελευταίες δεκαετίες έχει αυξηθεί η χρήση και η αποδοχή των δικτύων Petri ως βασικού μοντέλου για ασύγχρονα συστήματα συντρέχοντος υπολογισμού. Τα δίκτυα αυτά διευκολύνουν την κατανόηση του τρόπου λειτουργίας των συστημάτων, ενώ χρησιμοποιούνται σαν ένα είδος μηχανικής γλώσσας για ανάπτυξη, εξομοίωση και εφαρμογή. Ένα δίκτυο Petri είναι ένα περιληπτικό και τυπικό μοντέλο ροής πληροφοριών. Οι ιδιότητες, οι έννοιες και οι τεχνικές των δικτύων Petri αναπτύσσονται στην προσπάθεια εύρεσης απλών, φυσικών και ισχυρών μεθόδων περιγραφής και ανάλυσης της ροής πληροφοριών και του ελέγχου, κυρίως σε συστήματα τα οποία μπορούν να εκτελέσουν συντρέχουσες και ταυτόχρονες ενέργειες.

Ορίζοντας επίσημα ένα δίκτυο Petri μπορούμε να πούμε ότι είναι σύνολο (tuple) πέντε σημείων με την μορφή $PN= (P, T, A, W, M_0)$ όπου:

- $P= \{ p_1, p_2 \ldots p_n \}$ ένα πεπερασμένο και μη κενό σύνολο θέσεων.
- $T= \{ t_1, t_2, \ldots, t_n \}$ ένα πεπερασμένο και μη κενό σύνολο μεταβάσεων.
- $A \subseteq (P \times T) \cup (T \times P)$ ένα πεπερασμένο σύνολο βελών/διανυσμάτων.
- $W: A \rightarrow \{1, 2, 3, \ldots\}$ η συνάρτηση βάρους για κάθε βέλος.
- $M_0: P \rightarrow \{0, 1, 2, 3, \ldots\}$ η αρχική σήμανση του δικτύου.

Σημειώστε ότι $P \cap T = \varnothing$

Ένα δίκτυο Petri χωρίς σήμανση συμβολίζεται ως $N= (P, T, A, W)$[15].

Μια από τις δημοφιλέστερες εκδόσεις του υψηλού επιπέδου των δικτύων Petri είναι τα Έγχρωμα Δίκτυα Πέτρι *(CPN)* με τα οποία θα ασχοληθούμε στη συνέχεια. Βασικό χαρακτηριστικό τους αποτελεί ο συνδυασμός της γραφικής μεθόδου μοντελοποίησης των

[14] Murata T., 'Petri Nets: Properties, Analysis and Applications' IEEE, 77, (1989), 541-580

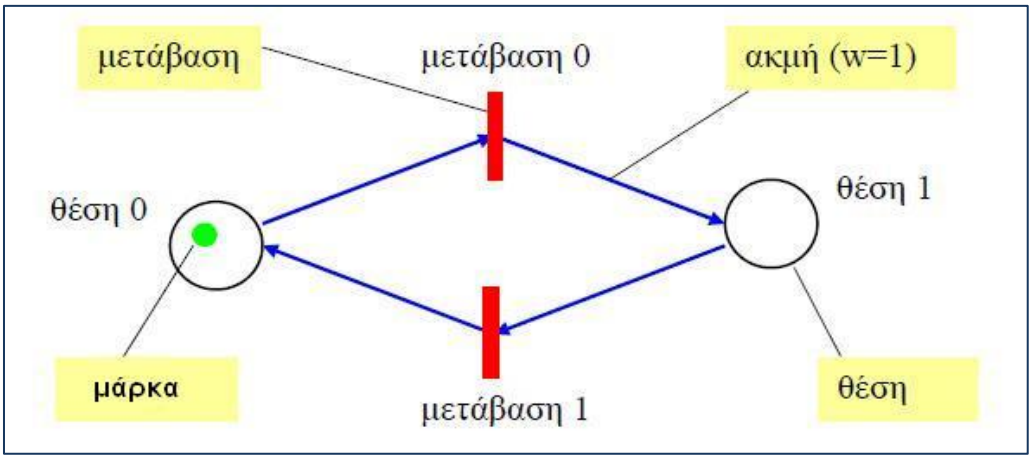
[15] Proth J.M. and Xie X., 'Petri Nets: A tool for design and management of manufacturing systems', Wiley, Chichester UK 1996

κοινών δικτύων Petri, η οποία παρέχει τα βασικά συστατικά για την διαδικασία αλληλεπίδρασης (process interaction), μαζί με τη γλώσσα προγραμματισμού CPN-ML η οποία απορρέει από την Τυπική Γλώσσα Μοντελοποίησης (Standard Modeling Language – SML), παρέχοντας τις βασικές έννοιες για τον ορισμό των τύπων δεδομένων και τη διαχείριση των τιμών και των δεδομένων. Στον επιστημονικό τομέα τα δίκτυα Petri χρησιμοποιούνται σε πληθώρα εφαρμογών στον κλάδο σχεδιασμού υλικού (VLSI chips), λογισμικού ηλεκτρονικών υπολογιστών, τηλεπικοινωνιών, αυτόματων συστημάτων παραγωγής κ.α.

Βασικό χαρακτηριστικό των έγχρωμων δικτύων Petri είναι η σαφώς καθορισμένη συντακτικά αλλά και σημασιολογικά, τυπική μαθηματική αναπαράσταση. Αυτή η αναπαράσταση αποτελεί τη βάση για τον ορισμό των διαφόρων χαρακτηριστικών ιδιοτήτων και των μεθόδων ανάλυσης. Χωρίς αυτή θα ήταν αδύνατο να αναπτυχθεί μια ισχυρή γλώσσα CPN. Ωστόσο, για την πρακτική χρήση των έγχρωμων δικτύων Petri και των εργαλείων τους, αρκεί μια διαισθητική κατανόηση της σύνταξης και της σημασιολογίας. Αυτό είναι ανάλογο με τις γλώσσες προγραμματισμού, οι οποίες εφαρμόζονται με επιτυχία από χρήστες που δεν είναι εξοικειωμένοι με τους τυπικούς, μαθηματικούς ορισμούς των γλωσσών.

Τα μοντέλα των έγχρωμων δικτύων Petri ακολουθούν την αρχή της ιεραρχικής προσέγγισης, αποτελούνται από ένα σύνολο συσχετιζόμενων αλλά αυτοτελών ομάδων *(modules/pages)* που το καθένα περιέχει ένα δίκτυο, κάτι ιδιαίτερα χρήσιμο σε μεγάλα και πολύπλοκα συστήματα. Οι ομάδες αυτές, αλληλεπιδρούν μεταξύ τους μέσα από ένα αυστηρά καθορισμένο σύνολο διασυνδέσεων, περιγράφοντας τις καταστάσεις στις οποίες μπορεί να βρίσκεται ένα σύστημα, καθώς και τις μεταβάσεις μεταξύ τους. Ο έλεγχος της συμπεριφοράς του εκάστοτε συστήματος γίνεται μέσω της προσομοίωσης με κατάλληλο εργαλείο λογισμικού (στην παρούσα εργασία χρησιμοποιήθηκε το πακέτο λογισμικού CPN Tools). Παράλληλα, παρέχεται η δυνατότητα χρησιμοποίησης της διάστασης του χρόνου στα μοντέλα, πράγμα που ενισχύει τη ρεαλιστικότητα και την ευελιξία τους, ενώ βοηθάει και στην αξιολόγηση της απόδοσης τους. Τα μοντέλα που

κάνουν χρήση αυτής της δυνατότητας ονομάζονται χρονισμένα. Ένα στοιχειώδες έγχρωμο δίκτυο Petri φαίνεται στην παρακάτω εικόνα 2.5.



Εικόνα 2.5 Στοιχειώδες έγχρωμο δίκτυο Petri

Βασικά δομικά στοιχεία ενός δικτύου Petri και κατά συνέπεια ενός έγχρωμου δικτύου Petri είναι οι θέσεις, οι μεταβάσεις, τα τόξα/βέλη και οι μάρκες.

- Οι **θέσεις (places)** που συμβολίζονται με το σχήμα της έλλειψης ή του κύκλου περιγράφουν τις καταστάσεις, ενώ παράλληλα αποτελούν και τις αποθήκες του συστήματος οι οποίες μπορούν να περιέχουν υλικά (πχ προϊόντα, πρώτες ύλες) ή πληροφορίες για την κατάσταση και τη λειτουργία του συστήματος.
- Οι **μεταβάσεις (transitions)** που συμβολίζονται με το σχήμα του παραλληλόγραμμου περιγράφουν τις λειτουργίες/ενέργειες που συμβαίνουν στο σύστημα, όπως μια διαδικασία, μια δραστηριότητα, μια επεξεργασία, το υπολογιστικό βήμα ενός αλγορίθμου, μια λογική πρόταση ή ένα επεξεργαστή σήματος.
- Τα **τόξα/βέλη (arcs)** που συμβολίζονται με το σχήμα του βέλους συνδέουν τις θέσεις με τις μεταβάσεις και αντίστροφα. Αναπαριστούν τη φυσική ή λογική σύνδεση, τα δικαιώματα πρόσβασης, την απαίτηση πόρων και την εξαγωγή συμπερασμάτων, ενώ καθορίζουν και τη σειρά εκτέλεσης ανταγωνιζόμενων ή μη εργασιών.

- Οι **μάρκες (tokens)** που συμβολίζονται με κουκίδες μπορεί να απεικονίζουν έναν αριθμό πηγών, να υποδεικνύουν την αλήθεια μιας συνθήκης ή να επιβεβαιώνουν την συνέχιση μιας λειτουργίας ανάλογα με το τι δηλώνει η θέση στην οποία βρίσκονται. Η διανομή των μαρκών στις θέσεις (σήμανση) καθορίζει την δυναμική κατάσταση του δικτύου, ενώ παράλληλα δίνει την δυνατότητα στον ερευνητή να προβεί στον έλεγχο των ιδιοτήτων, της συμπεριφοράς και της ροής πληροφοριών μέσα στο σύστημα.

Η ροή των μαρκών μέσα στο σύστημα ισοδυναμεί με την τήρηση του **κανόνα πυροδότησης ή εκπυρσοκρότησης** - γνωστός και ως token game - ο οποίος εξομοιώνει τη δυναμική συμπεριφορά ενός συστήματος. Απαραίτητη προϋπόθεση για να εκπυρσοκροτήσει μια μετάβαση *(firing),* είναι να έχει ενεργοποιηθεί, πράγμα το οποίο συμβαίνει όταν κάθε μια από τις θέσεις εισόδου της μετάβασης περιέχει τουλάχιστον τόσες μάρκες όσες και το βάρος του τόξου που συνδέει τη θέση με την μετάβαση *(enabling)*. Μια ενεργή μετάβαση μπορεί να πυροδοτηθεί, μπορεί και όχι. Όταν συμβεί μια εκπυρσοκρότηση, καταναλώνεται ή παράγεται ίσος αριθμός μαρκών με αυτόν του βάρους του τόξου στις θέσεις εισόδου και θέσεις εξόδου αντίστοιχα.

Οι ονομασίες των θέσεων και των μεταβάσεων επιλέγονται με τέτοιο τρόπο ώστε να περιγράφουν κατάλληλα τη λειτουργία ή την κατάσταση που αντιπροσωπεύουν. Οι θέσεις επίσης, χαρακτηρίζονται από τον τύπο των δεδομένων που μπορεί να περιέχουν. Ο τύπος αυτός δύναται να είναι απλός, όπως για παράδειγμα, ένας ακέραιος *(integer)* ή ένα αλφαριθμητικό *(string)* είτε πιο σύνθετος, όπως μία εγγραφή *(record)* με πολλά πεδία ή μια λίστα *(list)*.

Η κατάσταση ενός δικτύου καθορίζεται από την κατανομή των μαρκών στις θέσεις. Αρχικά τοποθετούνται σε συγκριμένες θέσεις δίνοντας έτσι την αρχική κατάσταση του δικτύου (αρχική σήμανση/initial marking), ενώ στη συνέχεια μεταφέρονται μέσα σε αυτό παρέχοντας έτσι τις διάφορες καταστάσεις του δικτύου. Οι μάρκες που αντιστοιχούν σε κάθε θέση έχουν τον ίδιο τύπο δεδομένων με τη θέση αυτή.

Τα τόξα χαρακτηρίζονται από τις εκφράσεις τους *(arc expressions)* που καθορίζουν τον τύπο των μαρκών που μπορούν να μεταφέρουν, ενώ ταυτόχρονα μπορούν να φέρουν και εντολές, οι οποίες ελέγχουν το πότε το τόξο θα μεταφέρει δεδομένα και το είδος αυτών. Αυτό επιτυγχάνεται είτε με χρήση συναρτήσεων *(functions),* είτε με απευθείας χρήση εντολών πάνω στα τόξα. Ανάμεσα σε μια θέση και μια μετάβαση μπορούν να υπάρχουν μέχρι δύο τόξα διαφορετικής κατεύθυνσης. Όταν αυτού του είδους τα τόξα έχουν ίδιες εκφράσεις, μπορούν να θεωρηθούν ως ένα διπλό τόξο *(double arc)*.

Μία μετάβαση απορροφά τις μάρκες από τα τόξα εισόδου και παράγει μάρκες στα τόξα εξόδου. Τα τόξα που κατευθύνονται από μια θέση προς μια μετάβαση ονομάζονται εισερχόμενα τόξα *(input arcs),* ενώ εκείνα που κατευθύνονται από μια μετάβαση σε μια θέση ονομάζονται εξερχόμενα τόξα *(output arcs)*. Επίσης, οι μεταβάσεις περιέχουν εντολές ενέργειας και ελέγχου που υλοποιούνται με τη χρήση της γλώσσας CPN-ML σε τρία συγκεκριμένα τμήματα.

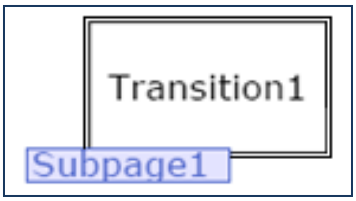
- Στο **τμήμα ελέγχου (guard)**, όπου ορίζεται μία λογική (boolean) συνθήκη η οποία ελέγχει την πυροδότηση ή μη της μετάβασης, εκμεταλλευόμενη τις μεταβλητές των τόξων εισόδου της μετάβασης.
- Στο **τμήμα κώδικα (code segment)**, όπου παρέχεται η δυνατότητα της επεξεργασίας των δεδομένων εισόδου και η παραγωγή διαφορετικής μορφής δεδομένων εξόδου.
- Στο **τμήμα χρόνου (timer),** όπου ορίζεται μία χρονική καθυστέρηση στην ενεργοποίηση και περάτωση της εργασίας της μετάβασης προσθέτοντας περισσότερο το στοιχείο του ρεαλισμού μέσα στο σύστημα.

Όπως ήδη αναφέραμε παραπάνω, τα έγχρωμα δίκτυα Petri υποστηρίζουν την λειτουργία της ιεραρχίας, ιδιότητα που τους δίνει τη δυνατότητα να περιγράφουν με ακρίβεια αρκετά πολύπλοκα μοντέλα, αναλύοντας σε υποεπίπεδα (pages) τις διάφορες λειτουργίες του συστήματος. Μία μετάβαση αναλύεται σε ένα ξεχωριστό υποδίκτυο, παρέχοντας έτσι μια πιο ακριβή περιγραφή για το ποια ακριβώς λειτουργία εκτελεί. Με

τον τρόπο αυτό επιτυγχάνεται η διατήρηση μιας πιο απλής και γενικής μορφής του μοντέλου σε ανώτερα επίπεδα, χωρίς να μας απασχολεί το πώς λαμβάνουμε τα κατάλληλα αποτελέσματα. Στα κατώτερα επίπεδα περιγράφεται αναλυτικότερα η λειτουργικότητα συγκεκριμένων στοιχείων του δικτύου.

Στη συνέχεια παρουσιάζονται τα δομικά στοιχεία με τη χρήση των οποίων επιτυγχάνεται η ιεραρχική προσέγγιση των μοντέλων των έγχρωμων δικτύων Petri. Αναλυτικότερα χρησιμοποιούνται τα εξής:

- **Μεταβάσεις αντικατάστασης (Substitution transitions):** Η μετάβαση αντικατάστασης είναι μέρος ενός δικτύου και αποτελεί μία γενική απεικόνιση ενός λεπτομερέστερου δικτύου. Ξεχωρίζει από τις απλές μεταβάσεις λόγω της ετικέτας αντικατάστασης που διακρίνεται στο κάτω μέρος της μετάβασης (εικόνα 2.6).

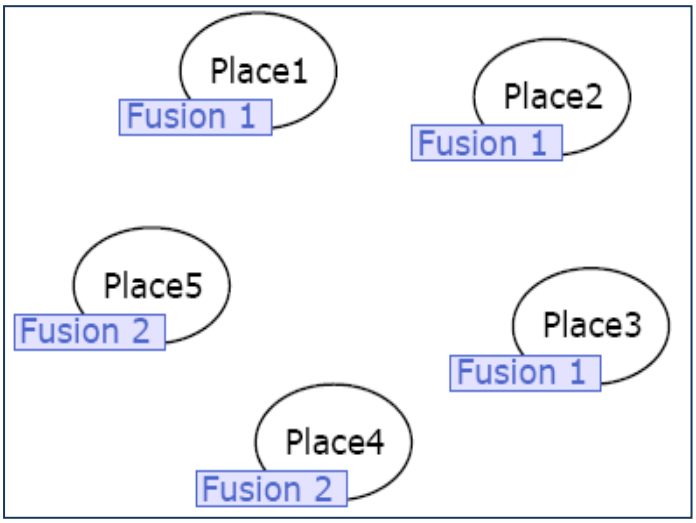


Εικόνα 2.6 Μετάβαση αντικατάστασης

- **Υποσελίδες (Subpages):** Κάθε μετάβαση αντικατάστασης κρύβει μια υποσελίδα, η οποία είναι ένα χαμηλότερου επιπέδου και συνάμα πιο εξειδικευμένο έγχρωμο δίκτυο Petri που περιγράφει την ακριβή λειτουργία της αντίστοιχης μετάβασης αντικατάστασης.
- **Θέσεις πύλης (Ports)**: Για τη διεκπεραίωση της επικοινωνίας μεταξύ του συστήματος και της κάθε υποσελίδας τα έγχρωμα δίκτυα Petri διαθέτουν ένα αριθμό εξειδικευμένων θέσεων. Οι θέσεις αυτές μπορεί να είναι θέσεις εισόδου (In-tags), θέσεις εξόδου (Out-tags), είτε θέσεις εισόδου/εξόδου (I/O tags) όπως φαίνεται και στην εικόνα 2.7. Αυτές οι θέσεις ονομάζονται θέσεις πύλης. Μια θέση πύλης εισόδου σε μια υποσελίδα μπορεί να δεχτεί μάρκες, μια θέση πύλης εξόδου μπορεί να παραδώσει μάρκες, ενώ μια θέση εισόδου/εξόδου μπορεί να κάνει και τα δύο.

Εικόνα 2.7 Θέσεις πύλης

- **Θέσεις υποδοχής (Sockets)**: Οι θέσεις πύλης που βρίσκονται σε υποσελίδες έχουν κάποιες αντίστοιχες θέσεις σε άλλες σελίδες στο σύστημα με τις οποίες συνδέονται και ανταλλάσουν μάρκες. Αυτές οι θέσεις ονομάζονται θέσεις υποδοχής και μπορούν να είναι επίσης θέσεις εισόδου, θέσεις εξόδου και θέσεις εισόδου/εξόδου. Η διαδικασία κατά την οποία συνδέεται μια θέση πύλης με μια υποδοχής ονομάζεται ανάθεση πύλης (port assignment). Όταν γίνεται αυτή η ανάθεση οι θέσεις γίνονται στην ουσία μία και οι μάρκες τους είναι πάντα οι ίδιες. Με αυτό τον τρόπο επιτυγχάνεται η παραλαβή και αποστολή μαρκών από και προς μια υποσελίδα.
- **Θέσεις συγχώνευσης (Fusion Places)**: Είναι μία ομάδα από θέσεις (εικόνα 2.8) που λειτουργούν ως μία, παρόλο που είναι ξεχωριστές. Όταν μία μάρκα προστεθεί ή αφαιρεθεί σε μία από αυτές τις θέσεις η ίδια μάρκα θα προστεθεί ή θα αφαιρεθεί αντίστοιχα από όλες τις υπόλοιπες θέσεις που ανήκουν στην ίδια ομάδα συγχώνευσης. Θέσεις συγχώνευσης μπορούν να υπάρχουν όχι μόνο στην ίδια σελίδα αλλά και σε πολλές σελίδες διαφορετικού επιπέδου.



Εικόνα 2.8 Θέσεις συγχώνευσης

Τα χρονισμένα δίκτυα Petri βασίζονται στην ύπαρξη ενός καθολικού ρολογιού *(global timer),* οι τιμές του οποίου είναι ακέραιες ή πραγματικές και αντιπροσωπεύουν τον χρόνο μοντελοποίησης. Το ρολόι είναι εικονικό, οι τιμές του δηλαδή μπορούν να συγκριθούν με τον πραγματικό χρόνο και δεν έχει καμία έννοια εκτός του μοντέλου. Οι μάρκες του συστήματος εκτός από τις τιμές τους, εμπεριέχουν και τις χρονικές σφραγίδες *(time stamps),* που εκφράζουν τον ελάχιστο χρόνο προσομοίωσης μετά την παρέλευση του οποίου μπορεί να χρησιμοποιηθεί η μάρκα. Επιπρόσθετα, παρέχεται στο δίκτυο η χρήση της χρονικής επιγραφής *(time inscription),* η οποία προσδιορίζεται με το σύμβολο "@" και βρίσκεται πάνω στο αντίστοιχο τμήμα κάθε μετάβασης. Η χρονική επιγραφή καθορίζει χρονική καθυστέρηση της πυροδότησης ή της ολοκλήρωσης της συγκεκριμένης μετάβασης.

Συνοψίζοντας παρουσιάζουμε τα βασικότερα πλεονεκτήματα των έγχρωμων δικτύων Petri:

- Μπορούν να αναπαρασταθούν γραφικά, γεγονός το οποίο τα κάνει περισσότερο προσιτά και ελκυστικά στους χρήστες.
- Είναι πολύ γενικά και μπορούν να χρησιμοποιηθούν για να περιγράψουν ένα μεγάλο αριθμό διαφορετικών συστημάτων, ενώ παράλληλα αποτελούν το επαρκέστερο πρότυπο για την περιγραφή και την ανάλυση του συγχρονισμού, της επικοινωνίας και το διαμερισμό πόρων μεταξύ παράλληλων διαδικασιών.
- Προσφέρουν μία λεπτομερή και εκτενή περιγραφή των καταστάσεων του συστήματος αλλά και των ενεργειών του. Αυτό δίνει τη δυνατότητα στους χρήστες να κατανοήσουν καλύτερα το σύστημα και τον τρόπο λειτουργίας του.
- Υποστηρίζουν ιεραρχικές περιγραφές και εξομοιώσεις αλληλεπίδρασης όπου τα αποτελέσματα παρουσιάζονται κατευθείαν πάνω στο διάγραμμα των έγχρωμων δικτύων Petri. Με αυτό τον τρόπο παρουσιάζεται βήμα προς βήμα η εξέλιξη του δικτύου, παρέχοντας τη δυνατότητα της μελετής των διάφορων καταστάσεων του, του ελέγχου της ορθής λειτουργίας και των αναμενόμενων αποτελεσμάτων.

- Τέλος, παρέχεται το χρονικό στοιχείο, γεγονός που φέρνει το μοντέλο πιο κοντά στην πραγματικότητα και βοηθάει στην εξαγωγή συμπερασμάτων καθώς και στον έλεγχο της απόδοσης του δικτύου.

2.5 Υλοποίηση με έγχρωμα δίκτυα Petri

Στην παρούσα εργασία επιλέχθηκε να γίνει η υλοποίηση του μοντέλου της μηχανής αυτόματης πώλησης εισιτηρίων *(TVM)* με τη βοήθεια των έγχρωμων δικτύων Petri. Η παραπάνω μέθοδος αποτελεί μια από τις ευρύτατα διαδεδομένες μεθόδους μοντελοποίησης και περιγραφής συστημάτων, η οποία υπερισχύει σε διάφορους τομείς έναντι άλλων μεθόδων μοντελοποίησης.

Η ουσιαστικότερη διαφορά των έγχρωμων δικτύων Petri έγκειται στην ικανότητα τους να προσθέτουν εξεζητημένη λειτουργικότητα και πολυπλοκότητα στο σύστημα με τη χρήση της γλώσσας προγραμματισμού SML *(Standard Modeling Language)*. Ο σχεδιαστής/αναλυτής αποκτά τη δυνατότητα εισαγωγής διάφορων τμημάτων κώδικα στο σύστημα, γεγονός που τον διευκολύνει στο να προσδώσει τα επιθυμητά χαρακτηριστικά στο μοντέλο του, καθώς επίσης και να προχωρήσει σε προσομοιώσεις ποικίλων σεναρίων για την επαλήθευση της ορθής λειτουργίας του.

Οι προσομοιώσεις είναι αρκετά ρεαλιστικές, αφού μπορούν να εμπεριέχουν και στοιχεία χρονισμού και διαδραστικές, ενώ σημαντική είναι και η προσφορά τους στο τμήμα της αποσφαλμάτωσης του συστήματος. Τα αποτελέσματα τους παρουσιάζονται απευθείας στο διάγραμμα του CPN, ενώ είναι δυνατόν να ελεγχθούν οι τιμές των δεδομένων των κινούμενων μαρκών.

Με τη βοήθεια των τμημάτων κώδικα και των προσομοιώσεων τα έγχρωμα δίκτυα Petri επιτρέπουν την παρακολούθηση και αποθήκευση των τιμών των δεδομένων συγκεκριμένων μεταβλητών, σε ειδικά αρχεία με απώτερο σκοπό την περαιτέρω επεξεργασία τους και την διεξαγωγή χρήσιμων συμπερασμάτων.

Ένα ακόμα από τα βασικότερα πλεονεκτήματα είναι η ενσωμάτωση της περιγραφής ελέγχου και χρονισμού με την περιγραφή της διαχείρισης των δεδομένων. Αυτό σημαίνει πως στην ίδια ακριβώς περιοχή μπορεί ο σχεδιαστής/αναλυτής να παρατηρεί όλο το περιβάλλον, τις ενεργοποιημένες συνθήκες και τα αποτελέσματα των λειτουργιών, γεγονός που κάνει το μοντέλο πολύ πιο ελκυστικό και προσιτό προς αυτόν.

Τέλος, σημαντική επίσης είναι η υποστήριξη της ιεραρχικής προσέγγισης που επιτρέπει την μοντελοποίηση μεγάλων συστημάτων καταφέρνοντας να μειώσει την πολυπλοκότητα τους και να διευκολύνει τη διαχείρισή τους.

3

Περιγραφή Μοντελοποίησης μιας Μηχανής Αυτόματης Πώλησης Εισητηρίων *(TVM)*

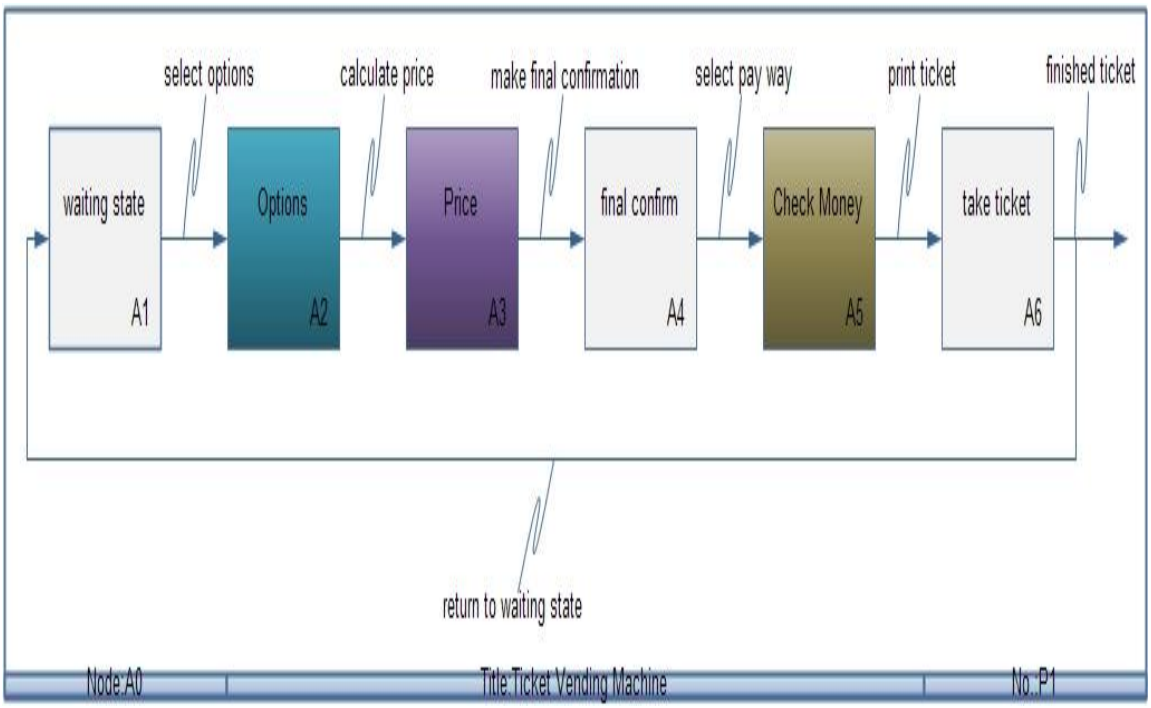
3.1 Περιγραφή αρχικού μοντέλου TVM σε IDEF0

Ξεκινώντας, θα περιγράψουμε τη μοντελοποίηση μιας μηχανής αυτόματης πώλησης εισιτηρίων, χωρίς όμως να λάβουμε υπόψη μας τα πιθανά ανθρώπινα λάθη ώστε να μπορέσουμε να κατανοήσουμε ευκρινώς τον τρόπο λειτουργίας της, καθώς και την διαδρομή της ροής των πληροφοριών μέσα στο σύστημα. Στην αρχική αυτή μοντελοποίηση χρησιμοποιήθηκε η μεθοδολογία μοντελοποίησης Integrated Definition for Function Modeling (IDEF0). Η IDEF0 βασίζεται στο γενικότερο πλαίσιο της τεχνικής Structured Analysis and Design Technique (SADT), η οποία παρέχει ένα συστηματικό τρόπο περιγραφής και αναπαράστασης.

Η περιγραφή του μοντέλου βασίζεται σε μια σειρά από διαγράμματα IDEF0, γεγονός που διευκολύνει την παρακολούθηση και τη μελέτη των διεργασιών που λαμβάνουν χώρα μέσα στο σύστημα. Η IDEF0 είναι μια από τις δημοφιλέστερες τεχνικές μοντελοποίησης που χρησιμοποιείται ευρέως για την περιγραφή επιχειρησιακών διαδικασιών. Τα διαγράμματά της υπόκεινται στους κανόνες τις ιεραρχικής ανάλυσης. Αυτό συνεπάγεται ότι το μοντέλο που προκύπτει, ξεκινάει από ένα γενικό διάγραμμα πρώτου επιπέδου, που περιγράφει χωρίς λεπτομέρειες τις βασικές διεργασίες του συστήματος, καθώς και τις εισόδους και τις εξόδους του μοντέλου. Σταδιακά, ορισμένες από τις αρχικές διεργασίες αναλύονται λεπτομερώς σε άλλα υποεπίπεδα χαμηλότερου επιπέδου και εκείνα με τη σειρά τους μπορεί να αναλυθούν περαιτέρω. Το αρχικό αυτό μοντέλο της IDEF0 θα αποτελέσει τη βάση της μοντελοποίησης με χρήση των έγχρωμων δικτύων Petri, για την ανάλυση του τρόπου λειτουργίας της μηχανής αυτόματης πώλησης εισιτηρίων καθώς και την υλοποίηση των προσομοιώσεων.

Στην εικόνα 3.1 φαίνεται το διάγραμμα επιπέδου A0 με ονομασία Ticket Vending Machine της IDEF0, στο οποίο αναπαρίστανται οι γενικότερες διεργασίες που εκτελούνται κατά τη λειτουργία της μηχανής αυτόματης πώλησης εισιτηρίων, από τη στιγμή που ένας χρήστης έρχεται στη μηχανή και επιθυμεί να εκδώσει ένα εισιτήριο. Η

αρχή της μοντελοποίησης βρίσκει την μηχανή στην κατάσταση αναμονής, waiting state – A1, για τον επόμενο χρήστη. Όταν ο χρήστης καταφθάσει στη μηχανή, καλείται να έχει ήδη λάβει τις αποφάσεις που αφορούν το εισιτήριο που επιθυμεί, καθώς και να εκτελέσει μια σειρά επιλογών που θα τον οδηγήσουν σταδιακά στην παραλαβή του επιθυμητού εισιτηρίου. Όλη αυτή η διαδικασία αναπαρίσταται με τη διεργασία Options – A2. Αφού γίνουν οι επιθυμητές επιλογές, η μηχανή ευθύνεται για τον υπολογισμό της τιμής του συγκεκριμένου εισιτηρίου, διαδικασία που περιγράφεται στη διεργασία Price – A3. Στη συνέχεια, παρουσιάζεται στο χρήστη η τιμή του εισιτηρίου και καλείται να κάνει την τελική επιβεβαίωση, final confirm – A4, για να συνεχίσει η λειτουργία της μηχανής. Έπειτα, ο χρήστης επιλέγει τον τρόπο με τον οποίο επιθυμεί να πληρώσει το εισιτήριο και η συνολική διαδικασία της πληρωμής εκτελείται στη διεργασία Check Money – A5. Όταν ολοκληρωθεί η διαδικασία αυτή, η μηχανή τυπώνει το εισιτήριο, το οποίο παραλαμβάνεται από το χρήστη, take ticket – A6, ενώ η μηχανή επιστρέφει στην κατάσταση αναμονής, waiting state – A1, του επόμενου χρήστη.

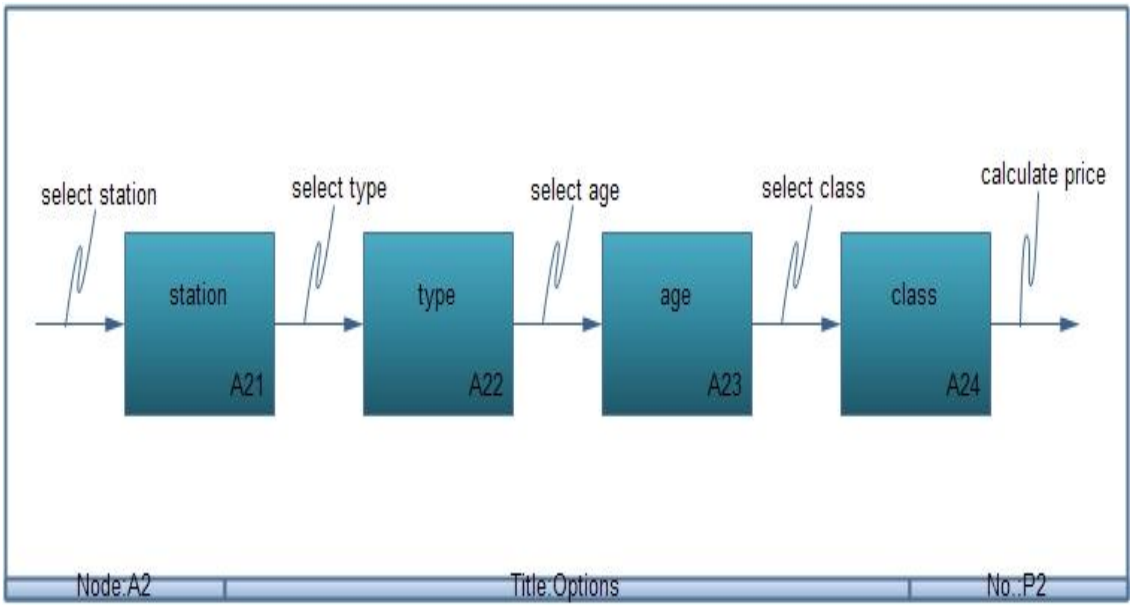


Εικόνα 3.1 Διάγραμμα IDEF0, επιπέδου A0 – για την μοντελοποίηση της μηχανής αυτόματης πώλησης εισιτηρίων *(TVM)*

Οι τρεις από τις παραπάνω διεργασίες που η ονομασία τους έχει γίνει με το πρώτο γράμμα κεφαλαίο και στο διάγραμμα A0 Ticket Vending Machine, της IDEF0 εμφανίζονται έγχρωμες είναι εκείνες που αναλύονται σε υποεπίπεδα, για να παρουσιαστεί με μεγαλύτερη λεπτομέρεια ο τρόπος με τον οποίο εκτελούνται. Συγκεκριμένα αυτές είναι η Options – A2, η Price – A3 και η Check Money – A5. Αναλυτικότερα, στην Options πραγματοποιούνται μια σειρά επιλογών που αφορούν το επιθυμητό εισιτήριο. Στο υποεπίπεδο της, A2 με ονομασία Options, όλες οι διεργασίες απόγονοι ονομάζονται βάση του ονόματος της γονικής διεργασίας προσθέτοντας στο τέλος του ονόματός της ένα επιπλέον ψηφίο το οποίο σχετίζεται με τη σειρά της ροής της πληροφορίας μέσα στη συγκεκριμένη διεργασία δηλαδή A21, A22, A23 κοκ. Στην Price γίνεται ο υπολογισμός της τιμής του εισιτηρίου λαμβάνοντας υπόψη τις προηγούμενες επιλογές και στο υποεπίπεδο της A3 με ονομασία Price, οι διεργασίες απόγονοι ονομάζονται σύμφωνα με την παραπάνω τακτική και παίρνουν ονόματα A31, A32, A33, κοκ. Τέλος στην Check Money γίνεται η επιλογή του τρόπου πληρωμής καθώς και η όλη διαδικασία παραλαβής/ελέγχου του χρηματικού ποσού από το χρήστη, παρομοίως στο υποεπίπεδο της A5 με ονομασία Check Money, οι διεργασίες απόγονοι ονομάζονται A51, A52, A53, κοκ.

Στο υποεπίπεδο A2 *(Options)* που φαίνεται στην εικόνα 3.2, όλες οι διεργασίες απόγονοι έχουν το ίδιο χρώμα *(πετρόλ)* με τη γονική διεργασία A2. Αρχικά, ο χρήστης στη διεργασία A21 *(station),* πρέπει να επιλέξει μέσα από ένα σύνολο σταθμών/προορισμών εκείνον για τον οποίο εκζητά να εκδώσει το τρέχων εισιτήριο. Στη συνέχεια στη διεργασία A22 *(type)*, καλείται να διαλέξει τον τύπο του εισιτηρίου του, εισιτήριο δηλαδή είτε με απλή μετάβαση προς τον προορισμό είτε με επιστροφή. Στη διεργασία A23 *(age),* ο χρήστης προχωρά στην τρίτη επιλογή, η οποία αφορά την ηλικία του επιβάτη για τον οποίο προορίζεται το εισιτήριο, εισιτήριο δηλαδή για ενήλικο επιβάτη, έφηβο ή φοιτητή, ή ακόμα και παιδικό εισιτήριο. Η τέταρτη και τελευταία επιλογή που γίνεται από το χρήστη στη διεργασία Options, είναι η A24 *(class),* η οποία αφορά στην κλάση του προτιμώμενου εισιτηρίου, δηλαδή είτε εισιτήριο πρώτης θέσης είτε

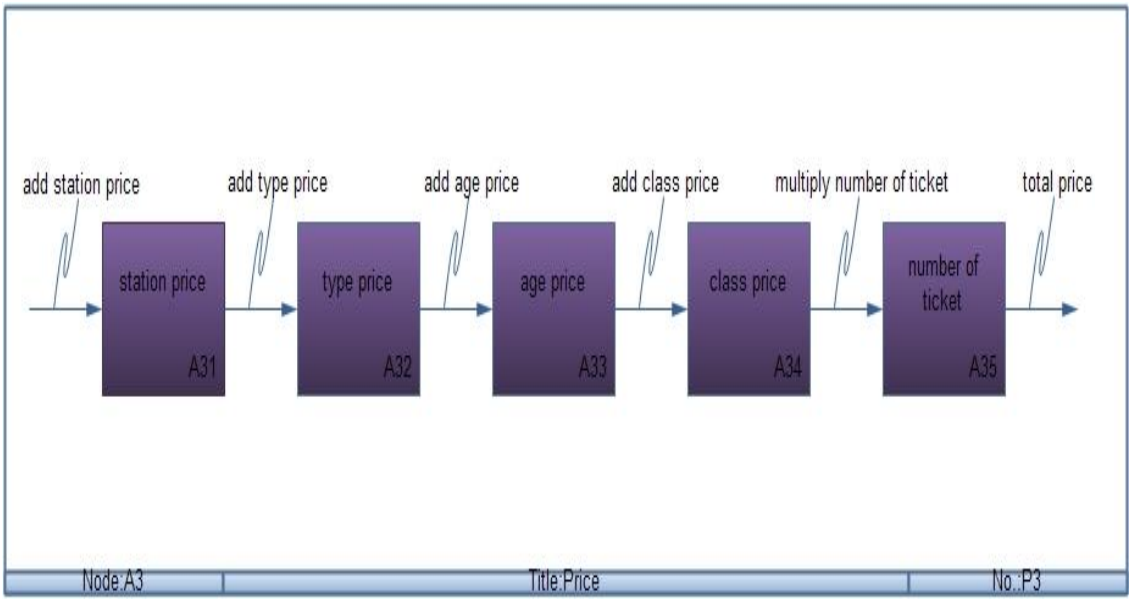
εισιτήριο οικονομικής θέσης. Αφού περατωθεί η σειρά αυτή των επιλογών, το μοντέλο επιστρέφει στο διάγραμμα του επιπέδου Α0 και συγκεκριμένα στο τέλος της διεργασίας Options.



Εικόνα 3.2 Υποεπίπεδο Α2 – Επιλογές *(Options)*

Στο υποεπίπεδο Α3 *(Price)* που φαίνεται στην εικόνα 3.3, όλες οι διεργασίες απόγονοι έχουν το ίδιο χρώμα *(μοβ)* με τη γονική διεργασία Α3. Η μηχανή αυτόματης πώλησης εισιτηρίων έχοντας ήδη αποθηκεύσει τις επιλογές του χρήστη που προηγήθηκαν στη διεργασία Options, αρχίζει να υπολογίζει την τιμή του τρέχοντος εισιτηρίου. Η τιμή υπολογίζεται ως εξής: ανάλογα την επιλογή του χρήστη προστίθεται στην τιμή του εισιτηρίου ένα ποσό, έτσι ώστε, συνυπολογίζοντας την τιμή όλων των επιλογών, να προκύπτει η τελική τιμή του εισιτηρίου. Συγκεκριμένα, στη διεργασία Α31 *(station price),* προστίθεται η τιμή ανάλογα με τον προορισμό που έχει επιλέξει ο χρήστης και στη διεργασία Α32 *(type price),* προστίθεται η τιμή με βάση τον προεπιλεγμένο τύπο του εισιτηρίου. Στη διεργασία Α33 *(age price),* προστίθεται η τιμή σύμφωνα με την ηλικία του επιβάτη ο οποίος πρόκειται να χρησιμοποιήσει το εισιτήριο αυτό και στη διεργασία Α34 *(class price),* προστίθεται η τιμή της κλάσης που διάλεξε ο χρήστης. Τέλος, στη διεργασία Α35 *(number of ticket),* η τιμή που έχει προκύψει μέχρι τώρα

πολλαπλασιάζεται με τον αριθμό των εισιτηρίων που προσδοκεί ο χρήστης και έτσι καταλήγει ο υπολογισμός της τελικής τιμής *(total price)* του εισιτηρίου.



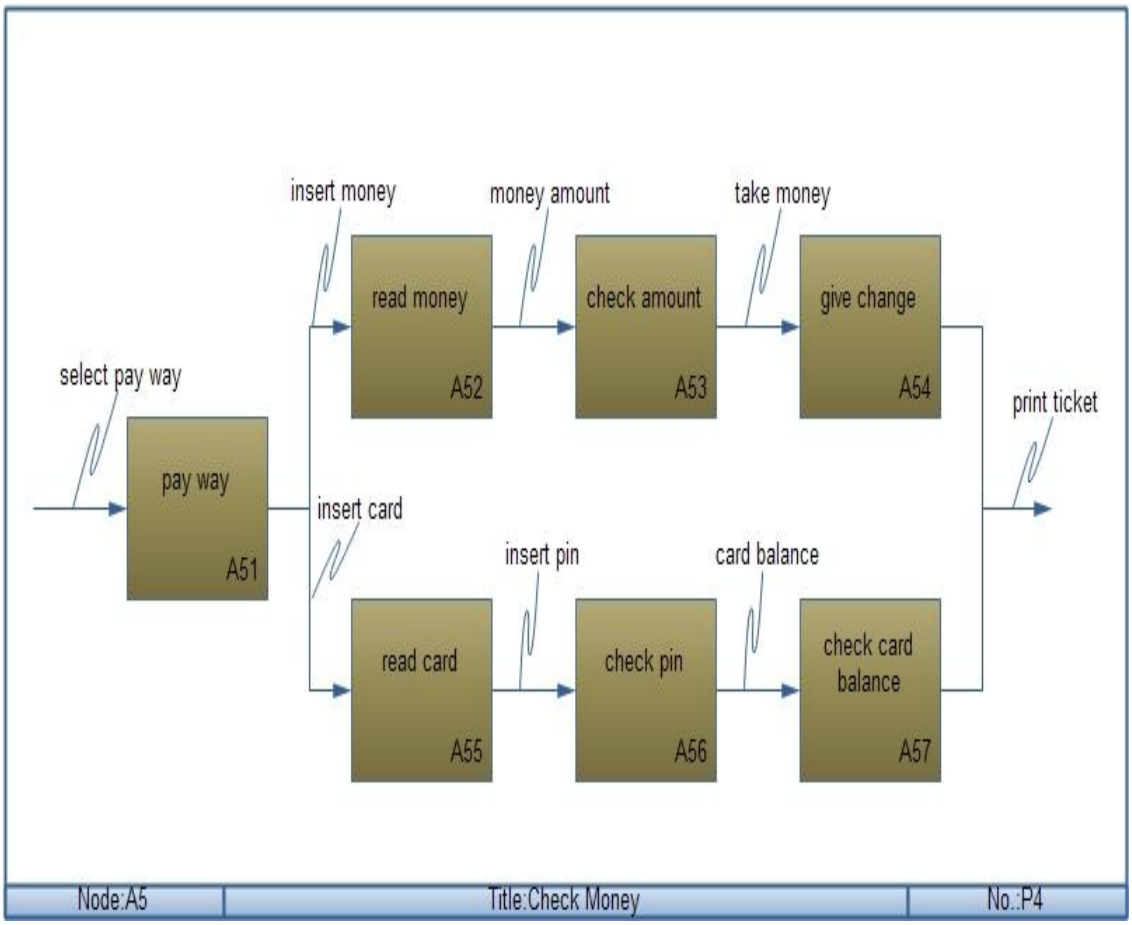
Εικόνα 3.3 Υποεπίπεδο Α3 – Τιμή *(Price)*

Στο υποεπίπεδο Α5 *(Check Money)* που φαίνεται στην εικόνα 3.4, όλες οι διεργασίες απόγονοι έχουν το ίδιο χρώμα *(μουσταρδί)* με τη γονική διεργασία Α5. Σ' αυτό το υποεπίπεδο, ο χρήστης αλληλεπιδρά για μια ακόμη φορά με τη μηχανή και καθορίζει τον τρόπο πληρωμής του εισιτηρίου, καθορίζοντας έτσι και τη διαδρομή της ροής της πληροφορίας. Λεπτομερέστερα, στη διεργασία Α51 *(pay way),* ο χρήστης πρέπει να επιλέξει αν επιθυμεί να πληρώσει το εισιτήριο με μετρητά είτε μέσω πιστωτικής κάρτας. Η απόφαση αυτή προσδιορίζει την μετέπειτα διαδρομή του εισιτηρίου.

- Αν ο τρόπος πληρωμής γίνει μέσω μετρητών χρημάτων, τότε ο χρήστης πρέπει να εισάγει στην ειδική σχισμή της μηχανής ένα χρηματικό ποσό μεγαλύτερο ή ίσο με εκείνο της συνολικής τιμής *(total price)* που είχε προκύψει στην προηγούμενη διεργασία Price. Στη διεργασία Α52 *(read money),* η μηχανή παίρνει και διαβάζει το χρηματικό ποσό που εισήγαγε ο χρήστης. Έπειτα, στη διεργασία Α53 *(check money),* η μηχανή ελέγχει αν το εισηγμένο χρηματικό ποσό επαρκεί για την αποπληρωμή του εισιτηρίου. Στη διεργασία Α54 *(give change),* γίνεται ο

υπολογισμός του χρηματικού ποσού που πρέπει να λάβει ο χρήστης σαν ρέστα και τυπώνεται το εισιτήριο του.

- Αν ο τρόπος πληρωμής γίνει μέσω πιστωτικής κάρτας, τότε ο χρήστης πρέπει να εισάγει στην ειδική σχισμή την πιστωτική κάρτα, ενώ η μηχανή στη διεργασία A55 *(read card),* διαβάζει τα στοιχεία της κάρτας. Στη συνέχεια πληκτρολογείται ο κωδικός *(pin)* της κάρτας από το χρήστη έτσι ώστε στη διεργασία A56 *(check pin),* να είναι εφικτός ο έλεγχος της εγκυρότητας του κωδικού αυτού. Τέλος, στη διεργασία A57 *(check card balance),* ελέγχεται η δυνατότητα εξόφλησης του εισιτηρίου μέσω της συγκεκριμένης κάρτας, αφαιρείται η συνολική του τιμή *(total price)* από το χρηματικό ποσό της κάρτας, επιστρέφεται η κάρτα στον ιδιοκτήτη της και το επιθυμητό εισιτήριο τυπώνεται.



Εικόνα 3.4 Υποεπίπεδο A5 – Έλεγχος Χρημάτων *(Check Money)*

Συνοπτικά η ακολουθία των βημάτων για την έκδοση εισιτηρίου φαίνεται στον πίνακα 3.1

Βήματα έκδοσης εισιτηρίου
Επιλογές A2 *(Options)* ▪ Επιλογή προορισμού *(select station)* ▪ Επιλογή τύπου εισιτηρίου *(select type)* ▪ Επιλογή ηλικίας επιβάτη *(select age)* ▪ Επιλογή κλάσης *(select class)*
Τιμή A3 *(Price)* ▪ Προσθήκη τιμής προορισμού *(add station price)* ▪ Προσθήκη τιμής τύπου *(add type price)* ▪ Προσθήκη τιμής ηλικίας επιβάτη *(add age price)* ▪ Προσθήκη τιμής κλάσης *(add class price)* ▪ Πολλαπλασιασμός επί τον αριθμό των εισιτηρίων *(multiply number of ticket)*
Τελική επιβεβαίωση A4 *(final confirm)*

Έλεγχος χρημάτων A5 *(Check Money)*
▪ Επιλογή τρόπου πληρωμής *(select pay way)*
▪ Εισαγωγή χρημάτων *(insert money)*
  ♦ Αναγνώριση χρημάτων *(read money)*
  ♦ Έλεγχος χρηματικού ποσού *(check amount)*
  ♦ Επιστροφή ρέστων *(give change)*
▪ Εισαγωγή κάρτας *(insert card)*
  ♦ Αναγνώριση κάρτας *(read card)*
  ♦ Έλεγχος Προσωπικού κωδικού *(check pin)*
  ♦ Έλεγχος διαθεσιμότητας *(check card balance)*

Παραλαβή εισιτηρίου A6 *(take ticket)*

Πίνακας 3.1 Περιγραφή βημάτων έκδοσης εισιτηρίου

3.2 Καταγραφή, πρόληψη & αντιμετώπιση ανθρώπινων λαθών

Μελετώντας το αρχικό μας μοντέλο, αλλά και πληθώρα άρθρων πάνω στο ίδιο αντικείμενο, έγινε καταγραφή των πιθανών ανθρώπινων λαθών, με στόχο την περαιτέρω μελέτη και ανάλυση τους για τη εύρεση μεθόδων πρόληψης ή ακόμα και αντιμετώπισης των ανθρώπινων λαθών. Στη συνέχεια, παραθέτονται τα ανθρώπινα λάθη που εντοπίστηκαν στο αρχικό μας μοντέλο, καθώς και τα άρθρα τα οποία έχουν ασχοληθεί με τα συγκεκριμένα λάθη. Επίσης, παρουσιάζονται ορισμένα ανθρώπινα λάθη και ανεπιθύμητες καταστάσεις του συστήματος, που αποτρέπονται από την συγκεκριμένη σχεδίαση του μοντέλου.

- Ένα από τα συχνότερα ανθρώπινα λάθη, εντοπίζεται στην εσφαλμένη επιλογή του προορισμού του επιβάτη, λάθος που εμφανίζεται και στα εξής άρθρα 1, 2, 3 και 4 (πίνακας 3.2).
- Ένα ακόμα ανθρώπινο λάθος, αφορά στη λανθασμένη επιλογή του τύπου του εισιτηρίου, αν δηλαδή το εισιτήριο θα είναι για απλή μετάβαση στον προεπιλεγμένο προορισμό ή αν θα είναι εισιτήριο με επιστροφή, λάθος που εμφανίζεται και στα εξής άρθρα 3 και 4 (πίνακας 3.2).
- Επίσης, έχει παρατηρηθεί ότι αρκετοί χρήστες κάνουν λάθος στην ηλικία του επιβάτη για τον οποίο προορίζεται το εισιτήριο, επιλέγουν, για παράδειγμα, εισιτήριο ενηλίκου, ενώ ουσιαστικά χρειάζονται εισιτήριο για φοιτητή ή παιδικό εισιτήριο. Το λάθος αυτό εμφανίζεται και στο εξής άρθρο 3 (πίνακας 3.2).
- Ανθρώπινο λάθος παρατηρήθηκε και κατά την επιλογή της κλάσης του εισιτηρίου, όπου για παράδειγμα ο χρήστης διάλεξε εισιτήριο πρώτης κλάσης, ενώ

ουσιαστικά ήθελε οικονομικό εισιτήριο. Το λάθος αυτό εμφανίστηκε και στο άρθρο 4 (πίνακας 3.2).

- Ένα επιπλέον ανθρώπινο λάθος, αναφέρεται στη λανθασμένη επιλογή του τρόπου πληρωμής του εισιτηρίου, λάθος που εμφανίζεται και στα εξής άρθρα 2 και 3 (πίνακας 3.2).
- Ακόμα, ανθρώπινο σφάλμα παρουσιάζεται κατά την εισαγωγή μη επαρκούς χρηματικού ποσού στην μηχανή, λάθος που εμφανίζεται και στο εξής άρθρο 3 (πίνακας 3.2).
- Επιπρόσθετα, ένα λάθος που εμφανίζεται συχνά είναι η μη αναμενόμενη υψηλή τιμή του εισιτηρίου που παρουσιάζεται στο χρήστη. Πιθανότατα ο χρήστης έκανε κάποια λάθος επιλογή στα προηγούμενα βήματα, χωρίς να τη συνειδητοποιήσει. Το λάθος αυτό εμφανίζεται και στο άρθρο 4 (πίνακας 3.2).
- Πολλές φορές, ο χρήστης εισάγει μεγαλύτερο χρηματικό ποσό από την τελική τιμή του εισιτηρίου που θέλει, σε μηχανή η οποία δεν δύναται να επιστρέψει ρέστα. Το λάθος αυτό εμφανίζεται και στο άρθρο 2 (πίνακας 3.2).
- Παράλληλα, υπάρχουν μηχανές στις οποίες ο χρήστης καλείται να επαναλάβει πολλές φορές την ίδια διαδικασία για την απόκτηση όμοιων εισιτηρίων, λάθος που εμφανίζεται και στο άρθρο 1 (πίνακας 3.2).
- Πολλοί είναι και οι χρήστες που λαμβάνουν το εισιτήριο και φεύγουν από τη μηχανή, έχοντας ξεχάσει να πάρουν τα ρέστα τους. Το λάθος αυτό εμφανίζεται και στο άρθρο 1 (πίνακας 3.2).
- Τέλος, σφάλματα εμφανίζονται και κατά την πληρωμή ενός εισιτηρίου με κάρτα. Τα λάθη αυτά οφείλονται είτε στη εισαγωγή μη έγκυρου προσωπικού κωδικού *(pin),* είτε στο μη επαρκές χρηματικό υπόλοιπο της κάρτας. Το λάθος αυτό εμφανίζεται και στο άρθρο 1 (πίνακας 3.2).

Αρίθμηση άρθρου	Τίτλος άρθρου
Άρθρο 1	Connell W.I., Blandford A. and Green T. 'CASSM and cognitive walkthrough: usability issues with ticket vending machines', Behaviour and Information Technology, 23, (2004), 307-320
Άρθρο 2	Connell W.I., 'Error analysis of ticket vending machines: comparing analytic and empirical data', Ergonomics, 41, (1998), 927-961
Άρθρο 3	Stanton N.A. and Baber C., 'Human error identification techniques applied to public technology: predictions compared with observed use' Ergonomics, 27, (1996), 119-131
Άρθρο 4	Verhoef L.W.M., 'Decision making of vending machine users' Ergonomics, 19, (1988), 103-109

Πίνακας 3.2 Τίτλοι άρθρων 1, 2, 3, και 4

Όλα τα παραπάνω ανθρώπινα λάθη, καθώς και οι ανεπιθύμητες καταστάσεις του συστήματος, παρουσιάζονται συνοπτικά στον Πίνακα 1 στο Παράρτημα Α. Ο πίνακας αυτός διαθέτει τρεις στήλες. Στη πρώτη στήλη, φαίνονται τα άρθρα στα οποία συναντήσαμε το ανθρώπινο λάθος, στη δεύτερη στήλη φαίνεται το λάθος στο οποίο αναφερόμαστε και στη τρίτη στήλη, περιγράφεται περιληπτικά ο τρόπος πρόληψης ή αντιμετώπισης του προαναφερθέν λάθους.

3.3 Μοντελοποίηση της μηχανής έκδοσης εισιτηρίων *(TVM)* με χρήση CPN

Η υλοποίηση των μοντέλων των δύο τύπων μηχανών TVM με τη χρήση έγχρωμων δικτύων Petri, βασίστηκε στα διαγράμματα της IDEF0, που περιγράφηκαν παραπάνω. Στη συνέχεια, αναφέρονται ορισμένοι από τους βασικότερους κανόνες που χρησιμοποιήθηκαν για τη μετατροπή των διαγραμμάτων. Κύριος κανόνας για τη μετατροπή των διαγραμμάτων σε έγχρωμα δίκτυα Petri, είναι ότι τα πλαίσια των διαγραμμάτων SADT μετατρέπονται σε ορθογώνια παραλληλόγραμμα δηλαδή

μεταβάσεις, οι οποίες διαβάζουν εντολές και εκτελούν εργασίες/λειτουργίες. Παράλληλα, τα βέλη της SADT μετατρέπονται σε τόξα που καταλήγουν σε κύκλους/ελλείψεις, δηλαδή τις θέσεις, που περιγράφουν πληροφορίες, εντολές και θέσεις αναμονής. Τέλος, οι οντότητες της SADT, όπως οι πρώτες ύλες, το προσωπικό, ή οι πληροφορίες, αναπαρίστανται ως μάρκες στο δίκτυο Petri.

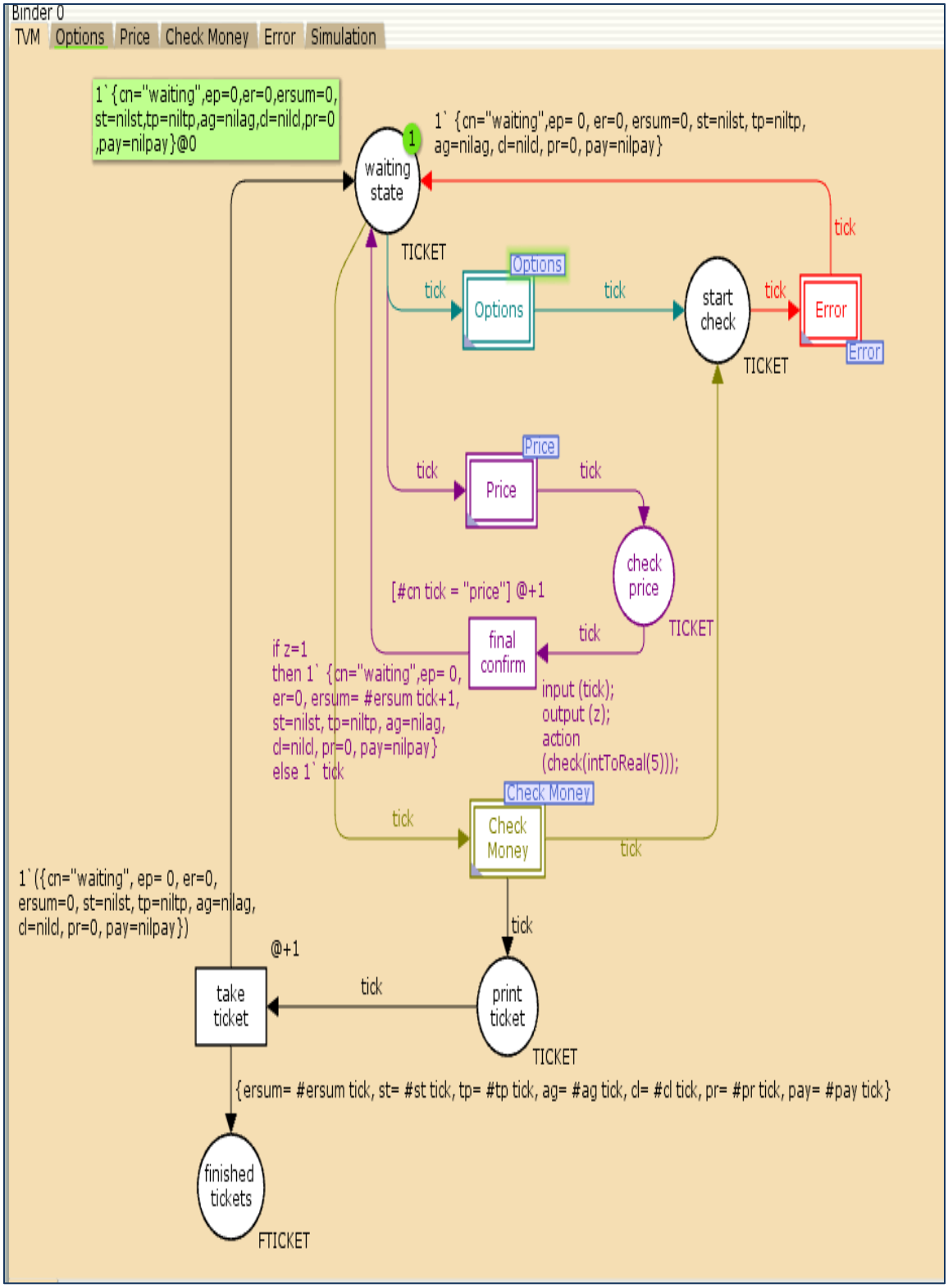
Για την βαθύτερη κατανόηση της μετατροπής των διαγραμμάτων της IDEF0 σε έγχρωμα δίκτυα Petri δημιουργήθηκε ο πίνακας 3.3, ο οποίος παρουσιάζει την αντιστοιχία των επιπέδων ανάμεσα στις δυο τεχνικές.

Τεχνική IDEF0	Έγχρωμα δίκτυα Petri
Επίπεδο A0 – Ticket Vending Machine	Επίπεδο 0 – TVM
Υποεπίπεδο A2 – Options	Υποεπίπεδο 1 – Options
Υποεπίπεδο A3 – Price	Υποεπίπεδο 1 – Price
Υποεπίπεδο A5 – Check Money	Υποεπίπεδο 1 – Check Money
--------	Υποεπίπεδο 1 – Error

Πίνακας 3.3 Αντιστοιχία διαγραμμάτων IDEF0 και έγχρωμων δικτύων Petri

Το υποεπίπεδο 1 Σφάλμα (Error) δεν αντιστοιχεί σε κανένα διάγραμμα της IDEF0, αφού στην αρχική μοντελοποίηση δεν είχαν ληφθεί υπόψη τα ανθρώπινα σφάλματα, ούτε οι ανεπιθύμητες καταστάσεις του συστήματος.

Η μοντελοποίηση της TVM με τη χρήση των έγχρωμων δικτύων Petri, εμπλουτίστηκε με τεχνικές πρόληψης και αντιμετώπισης των ανθρώπινων λαθών, αλλά και των ανεπιθύμητων καταστάσεων του συστήματος. Στην εικόνα 3.5 παρουσιάζεται το γενικό μοντέλο της TVM, το οποίο είναι όμοιο και για τους δύο τύπους μηχανών QFM και MFM. Στο μοντέλο αυτό εμφανίζονται οι μεταβάσεις Options, Price, final confirm, Check Money και take ticket, οι οποίες ήδη περιγράφηκαν και στο διάγραμμα της IDEF0, ενώ η μετάβαση Error εμφανίζεται σε αυτό το μοντέλο για πρώτη φορά.



Εικόνα 3.5 Επίπεδο 0 – Μηχανής έκδοσης εισιτηρίων (TMV)

Η πρώτη θέση είναι η κατάσταση αναμονής *(waiting state)* της μηχανής, η οποία είναι τύπου TICKET. Το colset TICKET (εικόνα 3.6) είναι μια εγγραφή *(record),* που ανήκει στους σύνθετους τύπους δεδομένων, αντιπροσωπεύει ένα μη ολοκληρωμένο εισιτήριο και προσδίδει ένα είδος μνήμης στο σύστημα. Αποτελείται από εννιά πεδία. Αναλυτικότερα, το πρώτο πεδίο cn *(κατάσταση/condition)*, είναι το σημείο στο οποίο κρατείται η κατάσταση από την οποία προήλθε το εισιτήριο και χρησιμεύει σαν ένα είδος μνήμης, έτσι ώστε αν γίνει κάποιο λάθος να μπορεί το σύστημα επιστρέψει ακριβώς ένα βήμα πίσω για να διορθωθεί το σφάλμα. Στο δεύτερο πεδίο ep *(πιθανότητα σφάλματος/error possibility),* αποθηκεύεται η πιθανότητα εμφάνισης ενός σφάλματος, η οποία είναι στενά συνδεδεμένη με τον αριθμό των επιλογών που έχει ο χρήστης. Στο τρίτο πεδίο er *(σφάλμα/error),* κρατείται ο αριθμός των σφαλμάτων που έγιναν στο συγκεκριμένο βήμα. Στο τέταρτο πεδίο ersum (σύνολο σφαλμάτων/error sum), αποθηκεύεται ο συνολικός αριθμός των σφαλμάτων που έγιναν κατά την έκδοση του εισιτηρίου και χρησιμεύει στη διεξαγωγή των συμπερασμάτων. Στο πέμπτο πεδίο tp *(τύπος/type),* κρατείται ο επιλεγμένος από το χρήστη τύπος του εισιτηρίου, ο οποίος παίρνει τις τιμές single, return, niltp που αντιστοιχούν σε εισιτήριο απλής μετάβασης, εισιτήριο μετ' επιστροφής είτε μη επιλεγμένο τύπο εισιτηρίου. Στο έκτο πεδίο ag *(ηλικία/age),* αποθηκεύεται η επιλεγμένη από το χρήστη ηλικία επιβάτη για τον οποίο προορίζεται το εισιτήριο, η οποία παίρνει τις τιμές adult, child, nilag που αντιστοιχούν σε εισιτήριο ενηλίκου, παιδικό εισιτήριο, είτε μη επιλεγμένη ηλικία του επιβάτη του εισιτηρίου. Στο έβδομο πεδίο cl *(κλάση/class),* κρατείται η επιλεγμένη από τον χρήστη κλάση του εισιτηρίου, η οποία παίρνει τις τιμές aclass, economy, nilcl που αντιστοιχούν σε εισιτήριο πρώτης θέσης, εισιτήριο οικονομικής θέσης, είτε μη επιλεγμένη κλάση εισιτηρίου. Στο όγδοο πεδίο pr

```
▼colset TICKET= record cn:CONDITION*
                       ep:INT*
                       er:INT*
                       ersum:INT*
                       st:STATION*
                       tp:TYPE*
                       ag:AGE*
                       cl:CLASS*
                       pr:INT*
                       pay:PAY timed;
```

Εικόνα 3.6 Εγγραφή TICKET

(τιμή/price), αποθηκεύεται η συνολική τιμή του εισιτηρίου του χρήστη και τέλος, στο ένατο πεδίο pay *(πληρωμή/pay way),* κρατείται ο επιλεγμένος από το χρήστη τρόπος πληρωμής του εισιτηρίου, ο οποίος παίρνει τις τιμές money, card, nilpay που αντιστοιχούν σε πληρωμή του εισιτηρίου με τη χρήση μετρητών, με χρήση κάρτας, είτε μη επιλεγμένο τρόπο πληρωμής του εισιτηρίου.

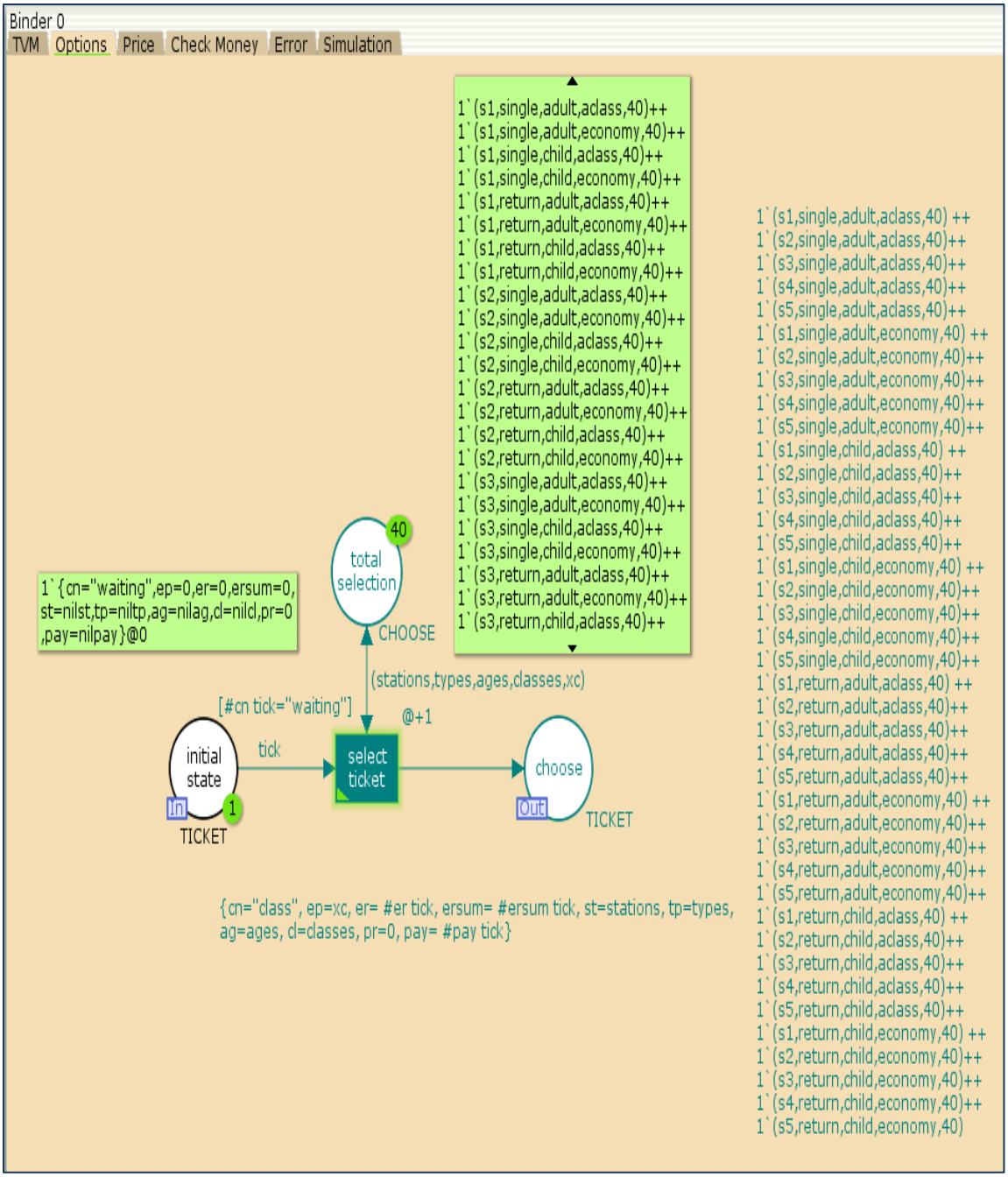
Αρχικά η μόνη ενεργοποιημένη θέση είναι η κατάσταση αναμονής *(waiting state)* της μηχανής (εικόνα 3.5), η μηχανή, δηλαδή, περιμένει μέχρι την εμφάνιση του επόμενου χρήστη. Στη συνέχεια, ενεργοποιείται η μετάβαση αντικατάστασης των επιλογών *(Options),* η οποία αντιπροσωπεύει το υποεπίπεδο Options, όπου ο χρήστης κάνει τις επιλογές που αφορούν τον προορισμό του, τον τύπο του εισιτηρίου, την ηλικία του επιβάτη και την κλάση του εισιτηρίου. Μετά το τέλος κάθε μιας από αυτές τις επιλογές, αν ο χρήστης έχει κάνει λάθος, του δίνεται η δυνατότητα, μέσω της μετάβασης αντικατάστασης των σφαλμάτων *(Error),* να επιστρέψει ένα ακριβώς βήμα πίσω και να κάνει τη σωστή επιλογή. Αφού τελειώσουν οι επιλογές, καθώς και ο έλεγχος των λαθών, ενεργοποιείται η μετάβαση αντικατάστασης τιμή *(Price),* η οποία αντιπροσωπεύει το υποεπίπεδο Price, όπου πραγματοποιείται ο υπολογισμός της τελικής τιμής του εισιτηρίου. Έπειτα, ενεργοποιείται η μετάβαση της τελικής επιβεβαίωσης *(final confirm),* όπου εμφανίζεται στο χρήστη η τελική τιμή του εισιτηρίου. Αν ο χρήστης δεν είναι σύμφωνος με αυτή την τιμή, παρέχεται η δυνατότητα να ακυρώσει τις προηγούμενες επιλογές του και να επιστρέψει στην αρχική κατάσταση αναμονής *(waiting state),* για να επιχειρήσει εκ νέου την έκδοση του επιθυμητού εισιτηρίου. Στην περίπτωση που ο χρήστης είναι σύμφωνος με την τελική τιμή, δίνει την τελική επιβεβαίωση και ενεργοποιείται η μετάβαση αντικατάστασης της πληρωμής *(Check Money),* όπου αρχικά ο χρήστης καλείται να επιλέξει τον τρόπο με τον οποίο επιθυμεί να πληρώσει το εισιτήριο του. Αν σε αυτό το σημείο ο χρήστης κάνει λάθος, μπορεί να επιστρέψει ένα βήμα πίσω, ενεργοποιείται για μια ακόμη φορά η μετάβαση αντικατάστασης των σφαλμάτων *(Error)* και να επιλέξει τον επιθυμητό τρόπο πληρωμής. Παράλληλα, πραγματοποιούνται ποικίλοι έλεγχοι που αφορούν στην ορθή αποπληρωμή του εισιτηρίου. Αφού

ολοκληρωθούν οι έλεγχοι το εισιτήριο εκτυπώνεται και παραδίδεται στο χρήστη. Το τελειωμένο εισιτήριο, το οποίο στο σημείο αυτό είναι τύπου FTICKET, φυλάσσεται στη θέση των τελειωμένων εισιτηρίων *(finished tickets)*. Η θέση αυτή, αποτελεί ένα είδος αποθήκης ολοκληρωμένων εισιτηρίων και βοηθάει στις μετέπειτα συγκρίσεις των δυο τύπων μηχανών, καθώς και στη διεξαγωγή συμπερασμάτων. Τέλος, η μηχανή επιστρέφει στην αρχική κατάσταση αναμονής *(waiting state),* αναμένοντας τον επόμενο χρήστη.

3.3.1 Περιγραφή μοντέλου μηχανής γρήγορης χρέωσης *(QFM)*

Η μηχανή γρήγορης χρέωσης QFM είναι μια από τις πιο διαδεδομένες μηχανές αυτόματης πώλησης εισιτηρίων. Η φιλοσοφία της βασίζεται στην ελαχιστοποίηση του χρόνου έκδοσης του εισιτηρίου. Ο χρήστης πρέπει να μπορεί να λάβει το επιθυμητό εισιτήριο κάνοντας όσο λιγότερα βήματα και αλληλεπιδρώντας το ελάχιστο δυνατό με τη μηχανή. Τα βασικά βήματα για την παραγωγή του εισιτηρίου είναι τέσσερα. Ο χρήστης επιλέγει από μια πληθώρα κουμπιών εκείνο που αντιστοιχεί στο εισιτήριο που χρειάζεται, έπειτα πατάει το κουμπί που αντιστοιχεί στον επιθυμητό τρόπο πληρωμής, εισάγει τα χρήματα ή την κάρτα και τέλος, παραλαμβάνει το εκτυπωμένο εισιτήριο.

Στην εικόνα 3.7 παρουσιάζεται το υποεπίπεδο 1 Επιλογές *(Options),* το οποίο αντιπροσωπεύει το πρώτο βήμα αλληλεπίδρασης του χρήστη με τη μηχανή. Στο σημείο αυτό ο χρήστης καλείται να διαλέξει με ένα πάτημα μια από τις 40 επιλογές – 5 προορισμοί επί 2 τύποι εισιτηρίου επί 2 τύποι για την ηλικία του επιβάτη επί 2 τύποι της κλάσης του εισιτηρίου (5x2x2x2=40) – που διαθέτει η μηχανή. Σε ένα μόνο βήμα, ο χρήστης επιλέγει ταυτόχρονα, τον προορισμό στον οποίο επιθυμεί να πάει, τον τύπο του εισιτηρίου του, δηλαδή αν το εισιτήριο του θα είναι εισιτήριο απλής μετάβασης ή εισιτήριο μετ' επιστροφής, την ηλικία του επιβάτη για τον οποίο προορίζεται το εισιτήριο, δηλαδή εισιτήριο ενήλικου ή παιδικό εισιτήριο, καθώς και την κλάση του εισιτηρίου, αν δηλαδή το εισιτήριο θα είναι οικονομικής θέσης ή πρώτης θέσης.



Εικόνα 3.7 Υποεπίπεδο 1 Επιλογές – Options (QFM)

Παρατηρώντας την εικόνα 3.7, βλέπουμε ότι καταρχάς είναι ενεργοποιημένη η αρχική κατάσταση *(initial state),* η οποία έχει μια μάρκα τύπου εισιτηρίου TICKET. Η μάρκα αυτή δηλώνει ότι έχει προέλθει από την κατάσταση αναμονής της μηχανής

(cn="waiting"), ενώ τα υπόλοιπα πεδία της δεν έχουν οριστεί ακόμα (εικόνα 3.8). Το @0 στο τέλος της μάρκας υποδηλώνει ότι η μάρκα καταφτάνει τη χρονική στιγμή μηδέν.

```
1`{cn="waiting",ep=0,er=0,ersum=0,
st=nilst,tp=niltp,ag=nilag,cl=nilcl,pr=0
,pay=nilpay}@0
```

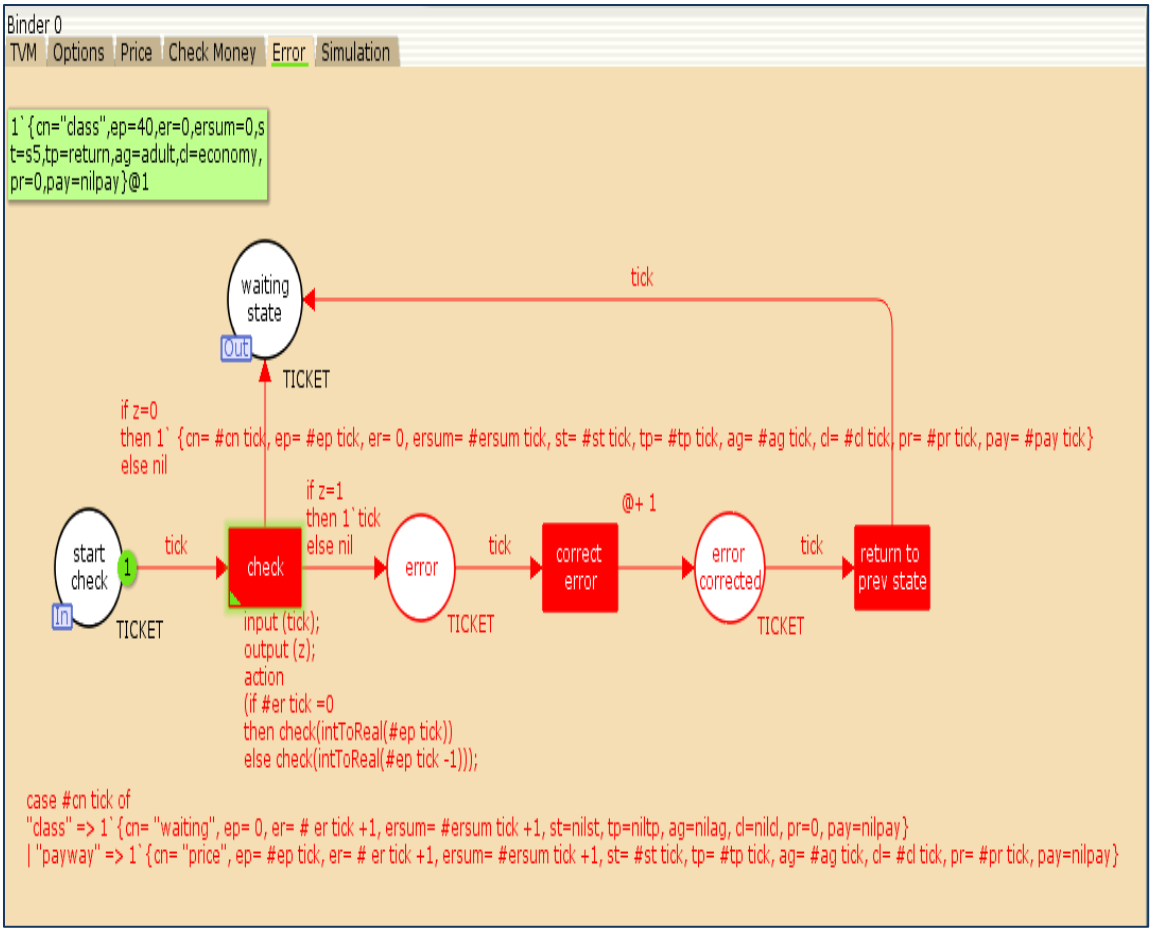
Εικόνα 3.8 Σήμανση αρχικής θέσης

Ενεργοποιημένη είναι και η θέση συνολικής επιλογής *(total selection),* η οποία περιέχει 40 μάρκες, τόσες δηλαδή όσες είναι οι συνολικές επιλογές του χρήστη. Τυχαία επιλέγεται μια από αυτές και θεωρείται ως την επιλογή που έκανε ο χρήστης. Οι δυο παραπάνω μάρκες, απαιτούνται για την ενεργοποίηση της μετάβασης επιλογή εισιτηρίου *(select ticket),* που ενεργοποιείται μόνο εφόσον υπάρχουν αυτές οι δύο μάρκες και η κατάσταση του εισιτηρίου είναι cn="waiting". Η συγκεκριμένη μετάβαση αποδίδει τιμές σε ορισμένα από τα πεδία του εισιτηρίου TICKET και συγκεκριμένα στα πεδία που αναφέρονται στην πιθανότητα σφάλματος *(ep)*, στο σφάλμα *(er)*, στο συνολικό σφάλμα *(ersum)*, στον προορισμό *(st)*, στον τύπο του εισιτηρίου *(tp)*, στην ηλικία του επιβάτη για τον οποίο προορίζεται το εισιτήριο *(ag)*, και στην κλάση του εισιτηρίου *(cl)*. Παρατηρείται, επίσης, ότι έχει αυξηθεί ο χρόνος κατά μια μονάδα (εικόνα 3.9).

```
1`{cn="class",ep=40,er=0,ersum=0,s
t=s2,tp=return,ag=adult,cl=aclass,pr
=0,pay=nilpay}@1
```

Εικόνα 3.9 Παράδειγμα σήμανσης τελικής θέσης του υποεπιπέδου 1 Επιλογές

Στη συνέχεια, ενεργοποιείται η θέση έναρξη ελέγχου *(start check)* του επιπέδου 0 TVM και ξεκινάει ο έλεγχος για τυχόν ανθρώπινο σφάλμα. Στο υποεπίπεδο 1 Σφάλμα (Error) (εικόνα 3.10) μοντελοποιείται το ανθρώπινο σφάλμα, με τη βοήθεια τριών συναρτήσεων που περιγράφονται αναλυτικά παρακάτω.



Εικόνα 3.10 Υποεπίπεδο 1 Σφάλμα – Error (QFM)

Στη μετάβαση έλεγχος *(check),* γίνεται χρήση του παρακάτω τμήματος κώδικα:

```
input (tick);
output (z);
action
(if #er tick = 0
then check(intToReal(#ep tick))
else check(intToReal(#ep tick -1)));
```

Ο κώδικας αυτός ευθύνεται για την παραγωγή λάθους, η οποία είναι στενά συνδεδεμένη με τον αριθμό των επιλογών που έχει ο χρήστης, δηλαδή όσο περισσότερες είναι οι επιλογές που δίνονται στο χρήστη, τόσο μεγαλύτερη είναι η πιθανότητα να κάνει κάποια λανθασμένη επιλογή. Το παραπάνω τμήμα κώδικα, ελέγχει αρχικά αν ο χρήστης έχει

ξανακάνει λάθος επιλογή (if #er tick = 0). Στην περίπτωση που ο χρήστης δεν έχει κάνει λάθος πρωτύτερα, δηλαδή #er tick = 0, καλείται η συνάρτηση check() με όρισμα το πλήθος των επιλογών του χρήστη (#ep tick), ενώ αν είχε κάνει λάθος #er tick > 0, καλείται η συνάρτηση check() με όρισμα το πλήθος των επιλογών του χρήστη μείον 1 (#ep tick -1), δηλαδή τις υπόλοιπες επιλογές εκτός από τη λανθασμένη που ήδη είχε διαλέξει. Είναι ένα είδος μνήμης όπου η μηχανή θυμάται ότι ο χρήστης έκανε κάποιο λάθος και μειώνει την πιθανότητα επόμενου λάθους. Αν παραχθεί λάθος η έξοδος παίρνει την τιμή 1 (output z=1), το εισιτήριο επιστρέφει ένα βήμα πίσω, δηλαδή στην κατάσταση αναμονής της μηχανής *(waiting sate),* περιμένοντας την εκ νέου επιλογή του χρήστη, ενώ αν δεν παραχθεί λάθος η έξοδος παίρνει την τιμή 0 (output z=0), η πορεία του εισιτηρίου συνεχίζει την κατάσταση όπου η μηχανή έχοντας ήδη αποθηκεύσει τα στοιχεία που της χρειάζονται ξεκινάει τον υπολογισμό της συνολικής τιμής του εισιτηρίου.

Η συνάρτηση check(y) είναι μια συνάρτηση τυχαίας παραγωγής λάθους, η οποία καλεί δύο συναρτήσεις: την uniform() και την prob():

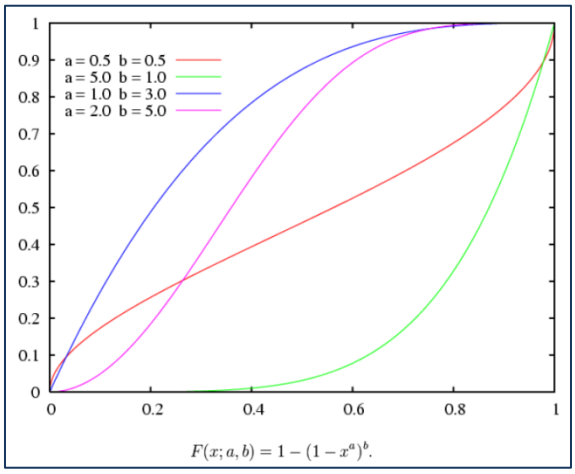
fun check(y)= if uniform(0.0,1.0)>prob(y) then 0 else 1;

Η uniform(0.0,1.0) παράγει ένα τυχαίο νούμερο από το 0.0 έως το 1.0 το οποίο συγκρίνεται με την έξοδο της συνάρτησης prob(). Η prob() επιστρέφει επίσης ένα νούμερο από το 0.0 έως το 1.0. Αν το νούμερο που παράγεται από την uniform() είναι μεγαλύτερο από εκείνο που παράγεται από την prob() τότε δεν παράγεται λάθος, ενώ στην αντίστροφη περίπτωση παράγεται. Η συνάρτηση uniform με την τυχαία παραγωγή του αριθμού στο περιθώριο που καθορίζουν τα ορίσματα της, βοηθάει στην αποδοτικότερη προσομοίωση της πραγματικότητας, ενώ παρέχει τη δυνατότητα αλλαγής των συγκεκριμένων περιθωρίων, μεταβάλλοντας κατά αυτόν τον τρόπο την πιθανότητα λάθους.

Η συνάρτηση prob() είναι μια σιγμοειδής συνάρτηση κατανομής της πιθανότητας λάθους η οποία εξαρτάται από το πλήθος των επιλογών που έχει ο χρήστης.

fun prob(y)= 1.0 - (1.0 - y / 200.0)*(1.0 - y / 200.0)*(1.0 - y / 200.0);

Όσες περισσότερες είναι οι επιλογές, τόσο μεγαλύτερη είναι και η πιθανότητα λάθους. Ανήκει στην οικογένεια συναρτήσεων κατανομής Kumaraswamy distribution[16] (εικόνα 3.11), ενώ η επιλογή της έγινε, έτσι ώστε η πιθανότητα παραγωγής σφάλματος να είναι σύμφωνη με τις πραγματικές τιμές που προήλθαν από την μέθοδο συνεντεύξεων των χρηστών σε πραγματικό περιβάλλον, όπως αυτές αναφέρονται στα άρθρα που μελετήθηκαν και αναφέρθηκαν στην παράγραφο 3.2. Ορισμένες ενδεικτικές τιμές τις φαίνονται στον πίνακα 3.4 που ακολουθεί.



Εικόνα 3.11 Kumaraswamy distribution

Αριθμός Επιλογών	Πιθανότητα Παραγωγής Σφάλματος (%)
2 επιλογές	3%
3 επιλογές	4,4%
5 επιλογές	7,3%
20 επιλογές	27%

Πίνακας 3.4 Ενδεικτικές τιμές συνάρτησης κατανομής Kumaraswamy

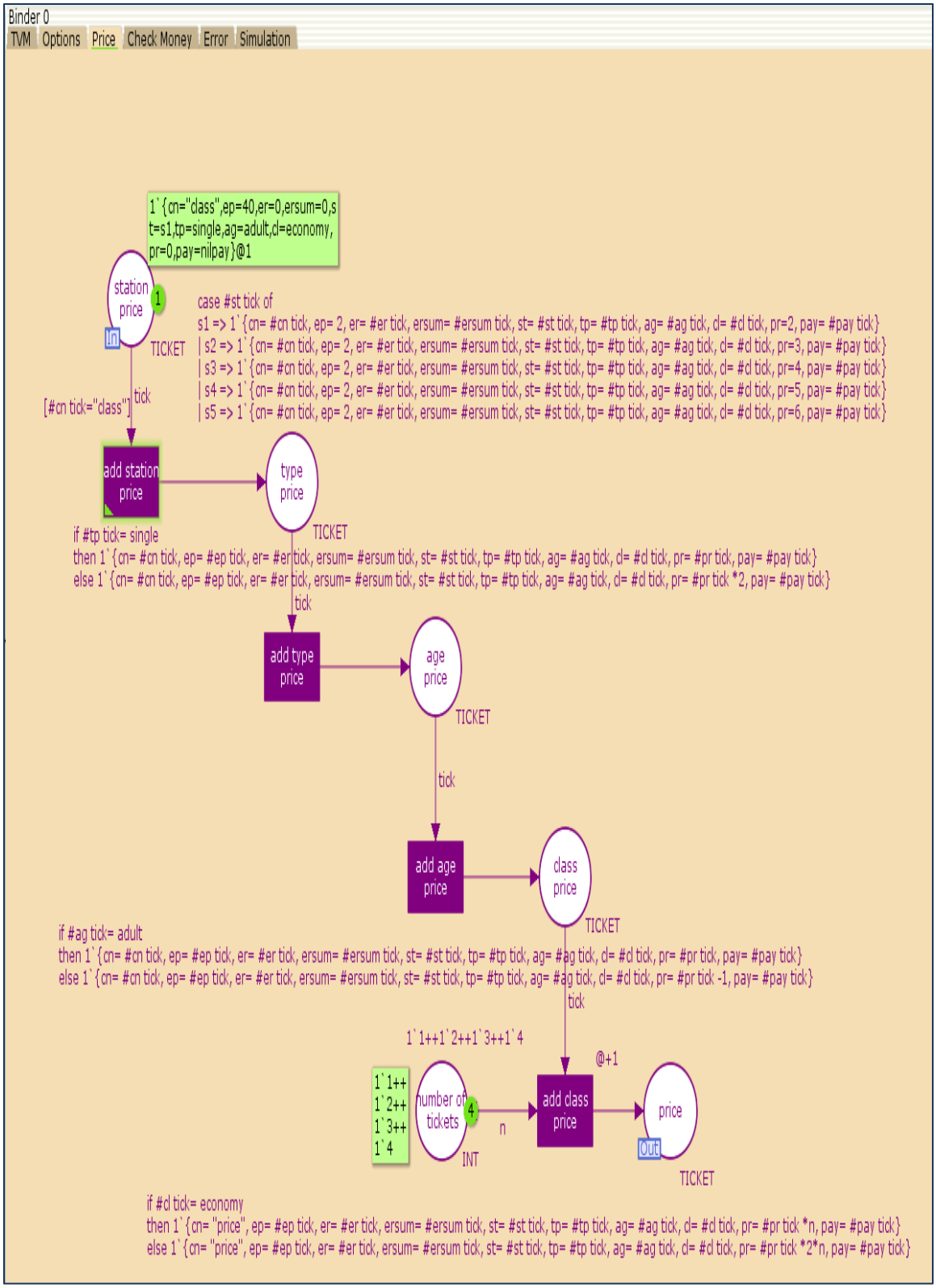
Όπως προαναφέρθηκε στη περίπτωση παραγωγής λάθους, η έξοδος παίρνει την τιμή 1 (output z=1) και ενεργοποιείται η μετάβαση διόρθωσης λάθους *(correct error),* η οποία ορίζει ως κατάσταση του εισιτηρίου την κατάσταση αναμονής της μηχανής *(cn="waiting"),* ενώ αρχικοποιεί τα υπόλοιπα πεδία του εισιτηρίου. Το εισιτήριο επιστρέφει στην κατάσταση αναμονής της μηχανής για να γίνει εκ νέου η επιλογή του χρήστη. Αντιθέτως, στην περίπτωση που δεν παράγεται λάθος, η έξοδος παίρνει την τιμή 0 (output z=0) και το εισιτήριο έχοντας αποθηκευμένες στα πεδία του τις επιλογές του χρήστη, κατευθύνεται προς το υποεπίπεδο 1 Τιμή (Price) (εικόνα 3.12).

[16] http://en.wikipedia.org/wiki/Kumaraswamy_distribution

Αφού ολοκληρωθεί η διαδικασία ελέγχου των σφαλμάτων, το εισιτήριο καταφτάνει στο υποεπίπεδο 1 Τιμή (Price), που φαίνεται στην εικόνα 3.12, στο οποίο γίνεται τμηματικά ο υπολογισμός της συνολικής τιμής του εισιτηρίου. Πρώτα, ενεργοποιείται η μετάβαση της προσθήκης της τιμής του προορισμού *(add station price),* όπου λαμβάνεται υπόψη η επιλογή του προορισμού του χρήστη. Αν ο χρήστης έχει επιλέξει ως προορισμό του το σταθμό s1, το εισιτήριο χρεώνεται με 2 μονάδες τιμής, αν έχει επιλέξει το σταθμό s2, το εισιτήριο χρεώνεται με 3 μονάδες τιμής, αν έχει επιλέξει το σταθμό s3, το εισιτήριο χρεώνεται με 4 μονάδες τιμής, αν έχει επιλέξει το σταθμό s4, το εισιτήριο χρεώνεται με 5 μονάδες τιμής και τέλος αν έχει επιλέξει το σταθμό s5, το εισιτήριο χρεώνεται με 6 μονάδες τιμής.

Στη συνέχεια, ενεργοποιείται η μετάβαση της προσθήκης της τιμής του τύπου του εισιτηρίου *(add type price).* Αν ο επιλεγμένος τύπος εισιτηρίου είναι εισιτήριο απλής μετάβασης, η προηγούμενη τιμή παραμένει ως έχει, ενώ στην περίπτωση που ο χρήστης έχει επιλέξει εισιτήριο μετ' επιστροφής η προηγούμενη τιμή διπλασιάζεται. Έπειτα, ενεργοποιείται η μετάβαση της προσθήκης της τιμής της ηλικίας του επιβάτη για τον οποίο προορίζεται το εισιτήριο *(add age price).* Στην περίπτωση που ο χρήστης έχει επιλέξει εισιτήριο ενηλίκου η τιμή παραμένει ως έχει, ενώ στην περίπτωση που έχει επιλέξει παιδικό εισιτήριο, τη παραπάνω τιμή του εισιτηρίου μειώνεται κατά 1 μονάδα τιμής.

Τέλος, ενεργοποιείται η μετάβαση της προσθήκης της τιμής της κλάσης του εισιτηρίου *(add class price).* Η μετάβαση αυτή συνδέεται και με τη θέση αριθμός εισιτηρίων *(number of tickets),* η οποία παράγει ένα τυχαίο ακέραιο από το 1 μέχρι το 4 και αντιπροσωπεύει τον αριθμό των όμοιων εισιτηρίων που επιλέγει ο χρήστης. Στην περίπτωση που ο χρήστης έχει επιλέξει εισιτήριο πρώτης θέσης, η τρέχουσα τιμή του εισιτηρίου διπλασιάζεται, ενώ πολλαπλασιάζεται και με τον αριθμό των εισιτηρίων, για να προκύψει η τελική συνολική τιμή του. Αν πάλι ο χρήστης έχει επιλέξει εισιτήριο οικονομικής θέσης, η τρέχουσα τιμή παραμένει ως έχει, ενώ πολλαπλασιάζεται και με τον αριθμό των εισιτηρίων, για να προκύψει η τελική συνολική τιμή του.



Εικόνα 3.12 Υποεπίπεδο 1 Τιμή – Price (QFM)

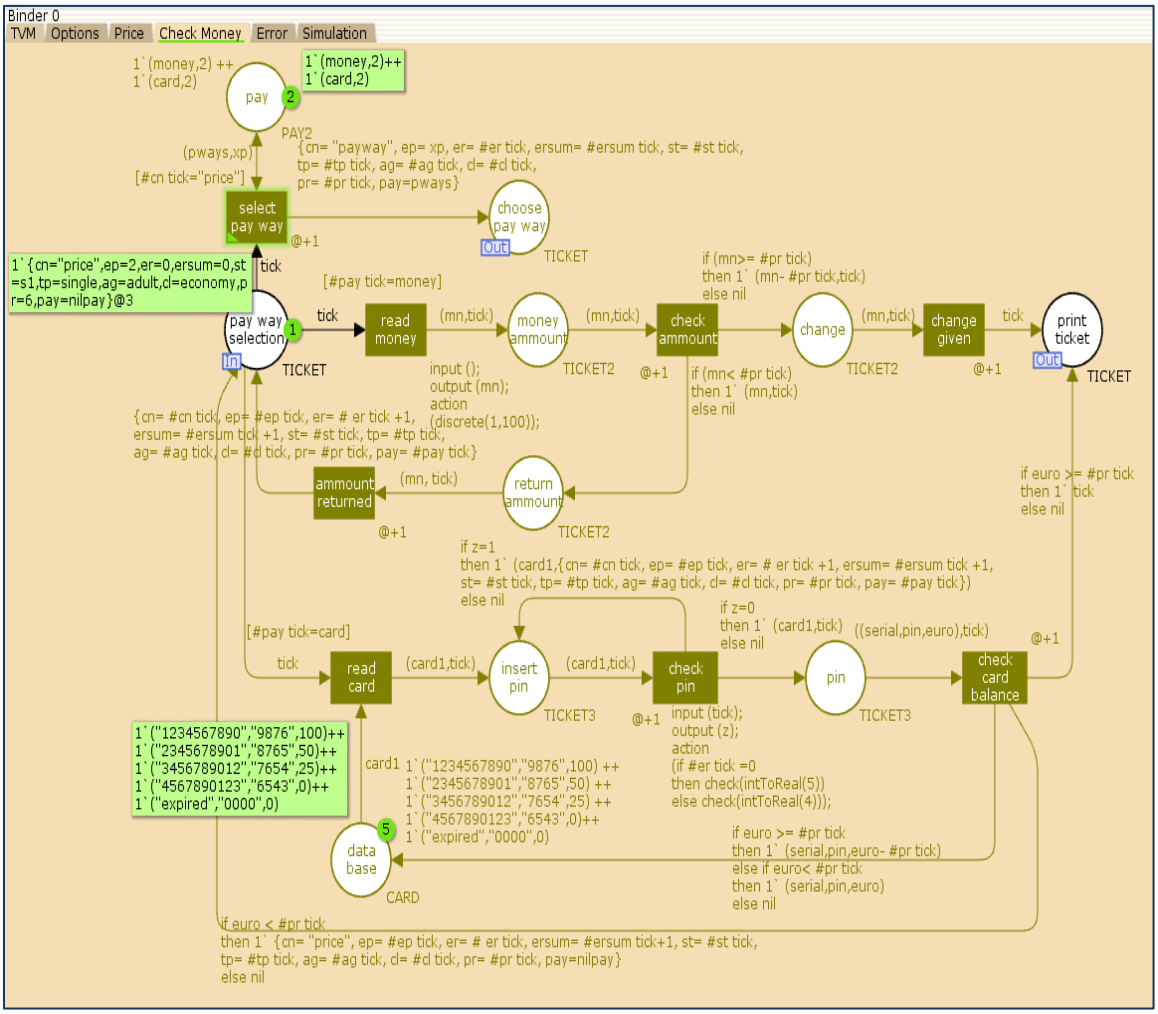
Αφού ολοκληρωθεί ο υπολογισμός της τελικής τιμής του εισιτηρίου, ενεργοποιείται η θέση ελέγχου της τιμής *(check price)* και η τελική τιμή παρουσιάζεται στο χρήστη. Υπάρχει η πιθανότητα, ο χρήστης να είχε κάνει κάποιο λάθος στα προηγούμενα βήματα, το οποίο δεν είχε συνειδητοποιήσει μέχρι αυτή τη στιγμή. Αν η τελική τιμή δεν είναι η αναμενόμενη, τότε ο χρήστης έχει τη δυνατότητα να ακυρώσει το τρέχων εισιτήριο και να επιστρέψει στην κατάσταση αναμονής της μηχανής *(waiting state),* για να κάνει εξ' αρχής μια νέα επιλογή. Όλη η διαδικασία της παραγωγής ή μη του λάθους εκτελείται στη μετάβαση της τελικής επιβεβαίωσης *(final confirm)* και επιτυγχάνεται με τη χρήση του εξής τμήματος κώδικα:

```
input (tick);
output (z);
action
(check(intToReal(5)));
```

Σαν όρισμα της συνάρτησης check(y), που έχει ήδη περιγραφεί αναλυτικά, επιλέχθηκε ο αριθμός 5, οποίο αντιστοιχεί σε πιθανότητα παραγωγής λάθους ίση με 7,3%. Αν τελικά παραχθεί ένα λάθος *(output z=1),* στο τόξο εξόδου της μετάβασης της τελικής επιβεβαίωσης, αρχικοποιούνται τα πεδία του εισιτηρίου και κατευθύνεται προς την κατάσταση αναμονής της μηχανής *(waiting state),* ενώ σε αντίθετη περίπτωση το εισιτήριο συνεχίζει κανονικά την πορεία του προς το υποεπίπεδο 1 Ελέγχου Χρημάτων (Check Money) (εικόνα 3.13).

Η επόμενη ενεργοποιημένη μετάβαση είναι η μετάβαση επιλογής του τρόπου πληρωμής του εισιτηρίου *(select pay way),* του υποεπιπέδου 1 Ελέγχου Χρημάτων (Check Money). Η μετάβαση αυτή ενεργοποιείται μόνο όταν η μάρκα που φτάνει σε αυτήν έχει προέλθει από την κατάσταση υπολογισμού της τελικής τιμής, δηλαδή όταν cn="price". Στο σημείο αυτό ο χρήστης επιλέγει τον τρόπο με τον οποίο επιθυμεί να πληρώσει το εισιτήριο του και στη συνέχεια γίνεται και πάλι έλεγχος της σωστής ή μη επιλογής του χρήστη, με τη βοήθεια του υποεπιπέδου 1 Σφάλμα (Error). Αν έχει συμβεί

σφάλμα, δηλαδή ο χρήστης έκανε μια λανθασμένη επιλογή, το εισιτήριο επιστρέφει ακριβώς ένα



Εικόνα 3.13 Υποεπίπεδο 1 Έλεγχος Χρημάτων – Check Money (QFM)

βήμα πίσω, στο σημείο δηλαδή όπου ο χρήστης επιλέγει για μια ακόμη φορά τον επιθυμητό τρόπο πληρωμής του εισιτηρίου. Η επιλογή του χρήστη είναι αυτή που καθορίζει την μετέπειτα πορεία του εισιτηρίου.

- Στην πρώτη περίπτωση, όπου ο χρήστης έχει επιλέξει να πληρώσει το εισιτήριο του με μετρητά, ενεργοποιείται η μετάβαση αναγνώριση χρημάτων *(read money),* όπου η μηχανή διαβάζει και αναγνωρίζει το χρηματικό ποσό, που μόλις εισήχθη από το χρήστη. Με την ενεργοποίηση της μετάβασης ελέγχου του ποσού *(check amount),* η μηχανή ελέγχει αν το εισηγμένο ποσό επαρκεί για την αποπληρωμή

του εισιτηρίου. Στην περίπτωση που επαρκεί, η μηχανή εισπράττει το ποσό που αντιστοιχεί στην τελική τιμή του εισιτηρίου και ενεργοποιείται η μετάβαση επιστροφής του υπόλοιπου ποσού *(change given),* η οποία επιστρέφει στο χρήστη τα ρέστα του. Στην περίπτωση όπου το εισηγμένο ποσό δεν επαρκεί για την εξόφληση του εισιτηρίου, ενεργοποιείται η μετάβαση επιστροφής του χρηματικού ποσού *(amount return),* επιστρέφεται στο χρήστη ολόκληρο το χρηματικό ποσό, ενώ η μηχανή επιστρέφει στην κατάσταση επιλογής του τρόπου πληρωμής, δίνοντας έτσι στο χρήστη την ευκαιρία να αλλάξει τρόπο πληρωμής του εισιτηρίου του, αν για παράδειγμα, τα μετρητά που διαθέτει δεν επαρκούν για την αποπληρωμή του επιθυμητού εισιτηρίου.

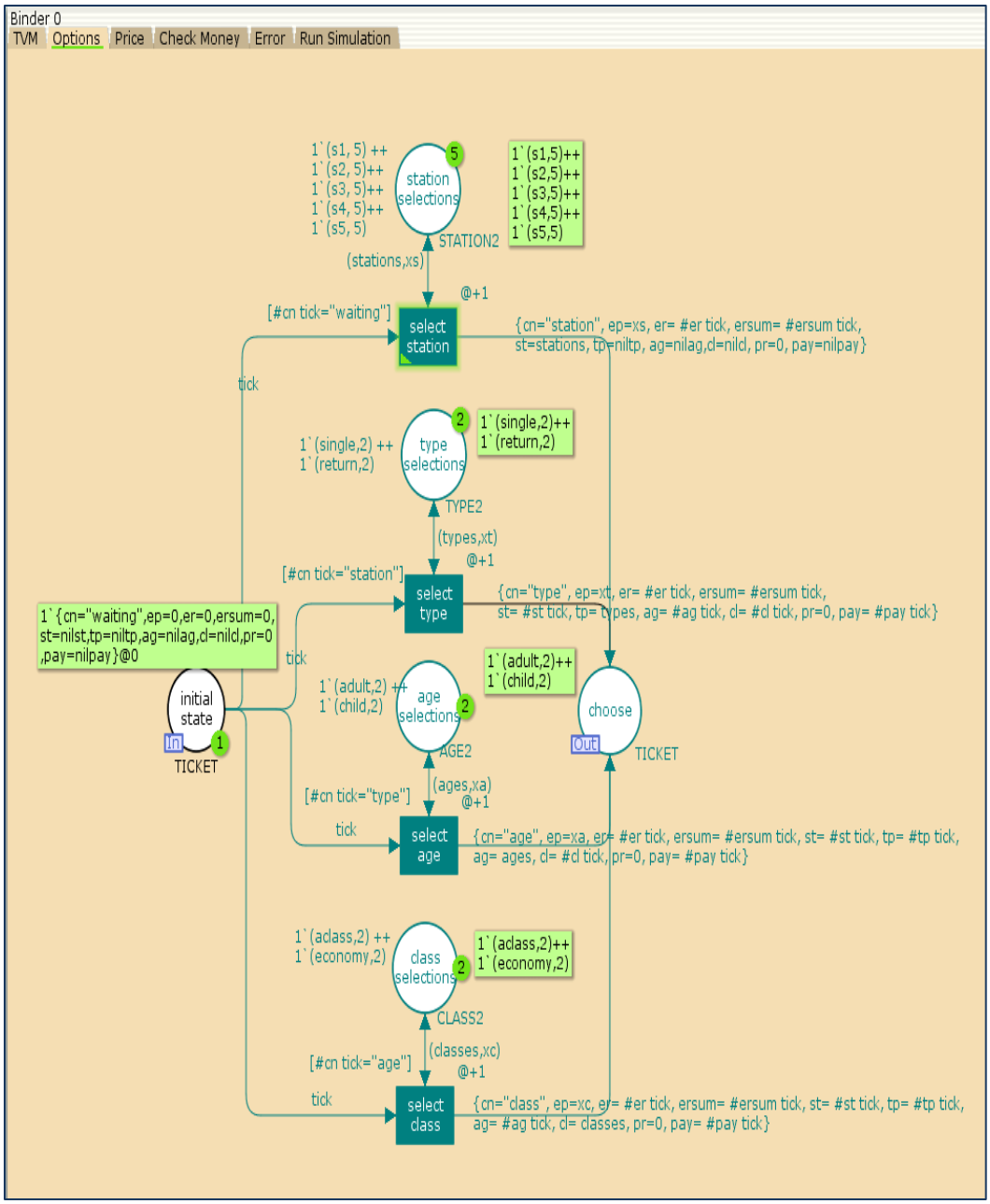
- Στην δεύτερη περίπτωση, όπου ο χρήστης επιλέγει να πληρώσει το εισιτήριο του με τη χρήση κάρτας, ενεργοποιείται η μετάβαση αναγνώριση κάρτας *(read card),* όπου η μηχανή διαβάζει και αναγνωρίζει την κάρτα, που μόλις εισήχθη από το χρήστη. Έπειτα, ενεργοποιείται η μετάβαση ελέγχου του προσωπικού κωδικού της κάρτας *(check pin),* στην οποία, η μηχανή ελέγχει αν ο χρήστης εισήγαγε τον σωστό προσωπικό κωδικό της κάρτας. Αν εισήχθη λανθασμένος κωδικός, η μηχανή επιστρέφει στη θέση εισαγωγής του προσωπικού κωδικού *(insert pin),* ενώ, αν ο κωδικός ήταν σωστός, ενεργοποιείται η μετάβαση ελέγχου διαθεσιμότητας της κάρτας *(card balance).* Στην περίπτωση που το χρηματικό ποσό της κάρτας επαρκεί, η μηχανή εισπράττει το ποσό που αντιστοιχεί στην τελική τιμή του εισιτηρίου και επιστρέφεται η κάρτα. Στην περίπτωση που το χρηματικό ποσό της κάρτας δεν επαρκεί για την εξόφληση του εισιτηρίου, επιστρέφεται η κάρτα, ενώ η μηχανή επιστρέφει στην κατάσταση επιλογής του τρόπου πληρωμής, δίνοντας έτσι στο χρήστη την ευκαιρία να αλλάξει τρόπο πληρωμής του εισιτηρίου του.

Τελειώνοντας, το εισιτήριο τυπώνεται και παραδίδεται στον χρήστη. Η μηχανή επιστρέφει στην κατάσταση αναμονής *(waiting state),* ενώ η θέση τελειωμένα εισιτήρια *(finished tickets),* χρησιμεύει ως αποθήκη, στην οποία φυλάσσονται τα ολοκληρωμένα

εισιτήρια. Μεγάλη είναι η χρησιμότητα της στη διεξαγωγή συμπερασμάτων, καθώς και στην αξιολόγηση του συστήματος.

3.3.2 Περιγραφή μοντέλου μηχανής πολλαπλής χρέωσης *(MFM)*

Η μηχανή πολλαπλής χρέωσης MFM αποτελεί έναν από τους πιο συνηθισμένους



Εικόνα 3.14 Υποεπίπεδο 1 Επιλογές – Options (MFM)

τύπους μηχανής αυτόματης πώλησης εισιτηρίων. Η φιλοσοφία της βασίζεται στην ελαχιστοποίηση του πιθανού σφάλματος κατά την έκδοση του εισιτηρίου. Ο χρήστης καλείται να εκτελέσει μεγάλο αριθμό επιλογών, που η κάθε μια αναφέρεται σε ένα αυστηρά καθορισμένο χαρακτηριστικό του εισιτηρίου. Στόχος της MFM είναι να αναλύσει διεξοδικά, ένα προς ένα, όλα τα χαρακτηριστικά του εισιτηρίου καθιστώντας τα απολύτως σαφή, αφήνοντας όσο το δυνατό μικρότερο περιθώριο λάθους στο χρήστη. Ο αριθμός των εκτελέσιμων βημάτων στα οποία ο χρήστης αλληλεπιδρά με τη μηχανή, είναι αρκετά μεγαλύτερος από εκείνον που απαιτείται για την έκδοση του αντίστοιχου εισιτηρίου με τη χρήση μιας μηχανής QFM.

Στην εικόνα 3.14 παρουσιάζεται το υποεπίπεδο 1 Επιλογές *(Options),* στο οποίο ο χρήστης καλείται να εκτελέσει ένα πλήθος επιλογών που σταδιακά θα τον οδηγήσουν στην απόκτηση του επιθυμητού εισιτηρίου. Στο συγκεκριμένο υποεπίπεδο, υπάρχουν 4 μεταβάσεις, όπου κάθε μια αναφέρονται σε ένα ακριβώς χαρακτηριστικό του εισιτηρίου και αντιστοιχεί σε μια επιλογή του χρήστη. Η μετάβαση επιλογή σταθμού προορισμού *(select station),* ενεργοποιείται μόνο όταν η κατάσταση της μάρκας TICKET, δηλώνει ότι η μάρκα προέρχεται από την κατάσταση αναμονής, δηλαδή cn=”waiting”. Η μετάβαση αυτή επιλέγει τυχαία έναν από τους 5 πιθανούς προορισμούς *(s1, s2, s3, s4, s5)*, ο οποίος θεωρείται ως την επιλογή του χρήστη και αποθηκεύεται στο πεδίο st του εισιτηρίου *(st=s4)* (εικόνα 3.15).

```
1`{cn="station",ep=5,er=0,ersum=0,
st=s4,tp=niltp,ag=nilag,cl=nilcl,pr=0,p
ay=nilpay}@1
```

Εικόνα 3.15 Παράδειγμα σήμανσης μάρκας μετά την επιλογή του σταθμού προορισμού υποεπίπεδο 1 Επιλογές

Η μετάβαση επιλογή τύπου του εισιτηρίου *(select type),* ενεργοποιείται μόνο όταν η κατάσταση της μάρκας TICKET, δηλώνει ότι η μάρκα προέρχεται από την κατάσταση επιλογής του σταθμού προορισμού, δηλαδή cn=”station”. Η μετάβαση αυτή επιλέγει τυχαία έναν από τους 2 πιθανούς τύπους εισιτηρίων *(single, return)*, ο οποίος θεωρείται

ως την επιλογή του χρήστη και αποθηκεύεται στο πεδίο tp του εισιτηρίου *(tp=single)* (εικόνα 3.16).

```
1`{cn="type",ep=2,er=0,ersum=0,st
=s4,tp=single,ag=nilag,cl=nilcl,pr=0,p
ay=nilpay}@2
```

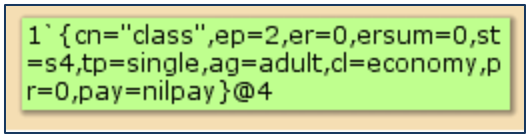
Εικόνα 3.16 Παράδειγμα σήμανσης μάρκας μετά την επιλογή του τύπου του εισιτηρίου υποεπίπεδο 1 Επιλογές

Η μετάβαση επιλογή ηλικίας του επιβάτη *(select age),* ενεργοποιείται μόνο όταν η κατάσταση της μάρκας TICKET, δηλώνει ότι η μάρκα προέρχεται από την κατάσταση επιλογής του τύπου του εισιτηρίου, δηλαδή cn="type". Η μετάβαση αυτή επιλέγει τυχαία έναν από τους 2 πιθανούς τύπους εισιτηρίων *(adult, child)*, ο οποίος θεωρείται ως την επιλογή του χρήστη και αποθηκεύεται στο πεδίο ag του εισιτηρίου *(ag=adult)* (εικόνα 3.17).

```
1`{cn="age",ep=2,er=0,ersum=0,st=
s4,tp=single,ag=adult,cl=nilcl,pr=0,pa
y=nilpay}@3
```

Εικόνα 3.17 Παράδειγμα σήμανσης μάρκας μετά την επιλογή της ηλικίας του επιβάτη υποεπίπεδο 1 Επιλογές

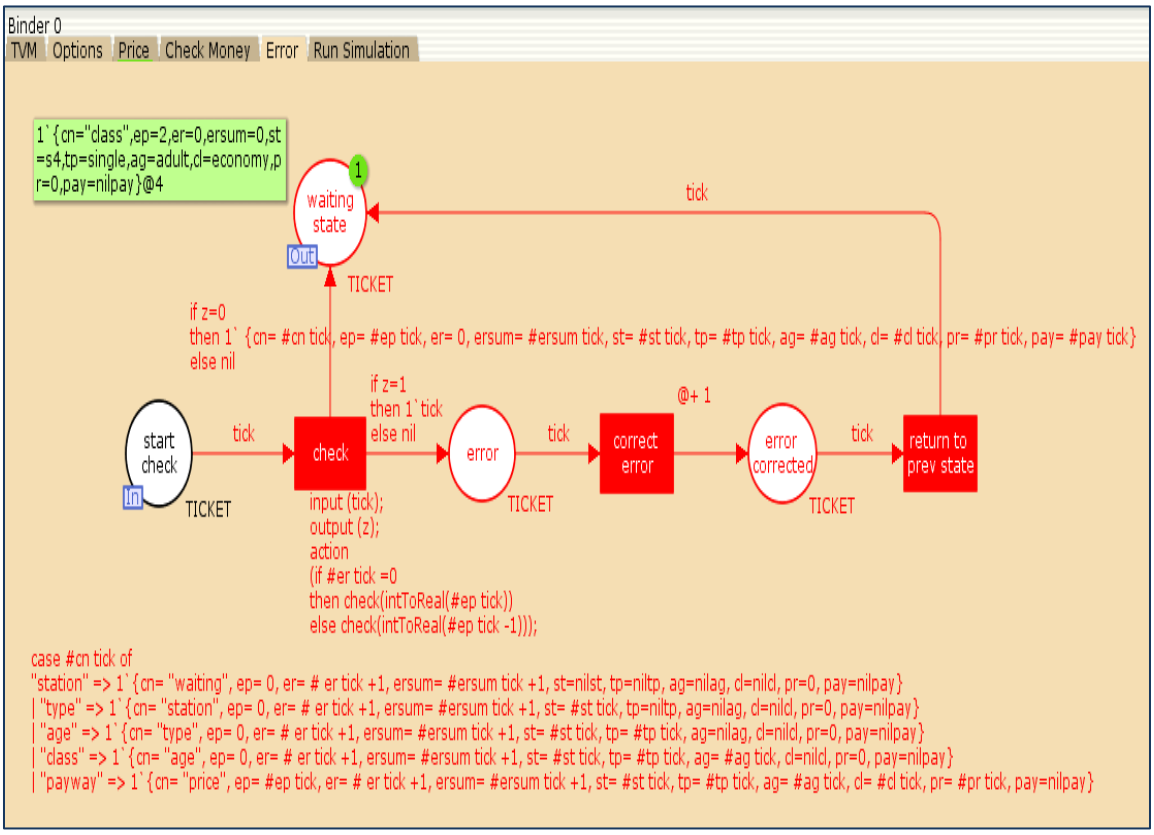
Η τελευταία μετάβαση, του υποεπιπέδου1 Επιλογές, επιλογή κλάσης *(select class),* ενεργοποιείται μόνο όταν η κατάσταση της μάρκας TICKET, δηλώνει ότι η μάρκα προέρχεται από την κατάσταση επιλογής ηλικίας του επιβάτη, δηλαδή cn="age". Η μετάβαση αυτή επιλέγει τυχαία μια από τους 2 πιθανές κλάσεις εισιτηρίων *(aclass, economy)*, η οποία θεωρείται ως την επιλογή του χρήστη και αποθηκεύεται στο πεδίο cl του εισιτηρίου *(cl=economy)* (εικόνα 3.18).



Εικόνα 3.18 Παράδειγμα σήμανσης μάρκας μετά την επιλογή της κλάσης του εισιτηρίου υποεπίπεδο 1 Επιλογές

Μετά το τέλος κάθε μιας από τις μεταβάσεις του υποεπιπέδου 1 Επιλογές *(Options),* γίνεται έλεγχος για τυχόν λανθασμένη επιλογή του χρήστη. Ο έλεγχος αυτός υλοποιείται στο υποεπίπεδο 1 Σφάλμα *(Error)* που φαίνεται στην εικόνα 3.19. Το ανθρώπινο σφάλμα μοντελοποιείται και σε αυτή την περίπτωση με τη χρήση των συναρτήσεων check(y), uniform(0.0,1.0) και prob(y) που περιγράφηκαν ήδη αναλυτικά στην παράγραφο 3.3.1.

```
fun check(y)= if uniform(0.0,1.0)>prob(y) then 0 else 1;
fun prob(y)= 1.0 - (1.0 - y / 200.0)*(1.0 - y / 200.0)*(1.0 - y / 200.0);
```



Εικόνα 3.19 Υποεπίπεδο 1 Σφάλμα – Error (MFM)

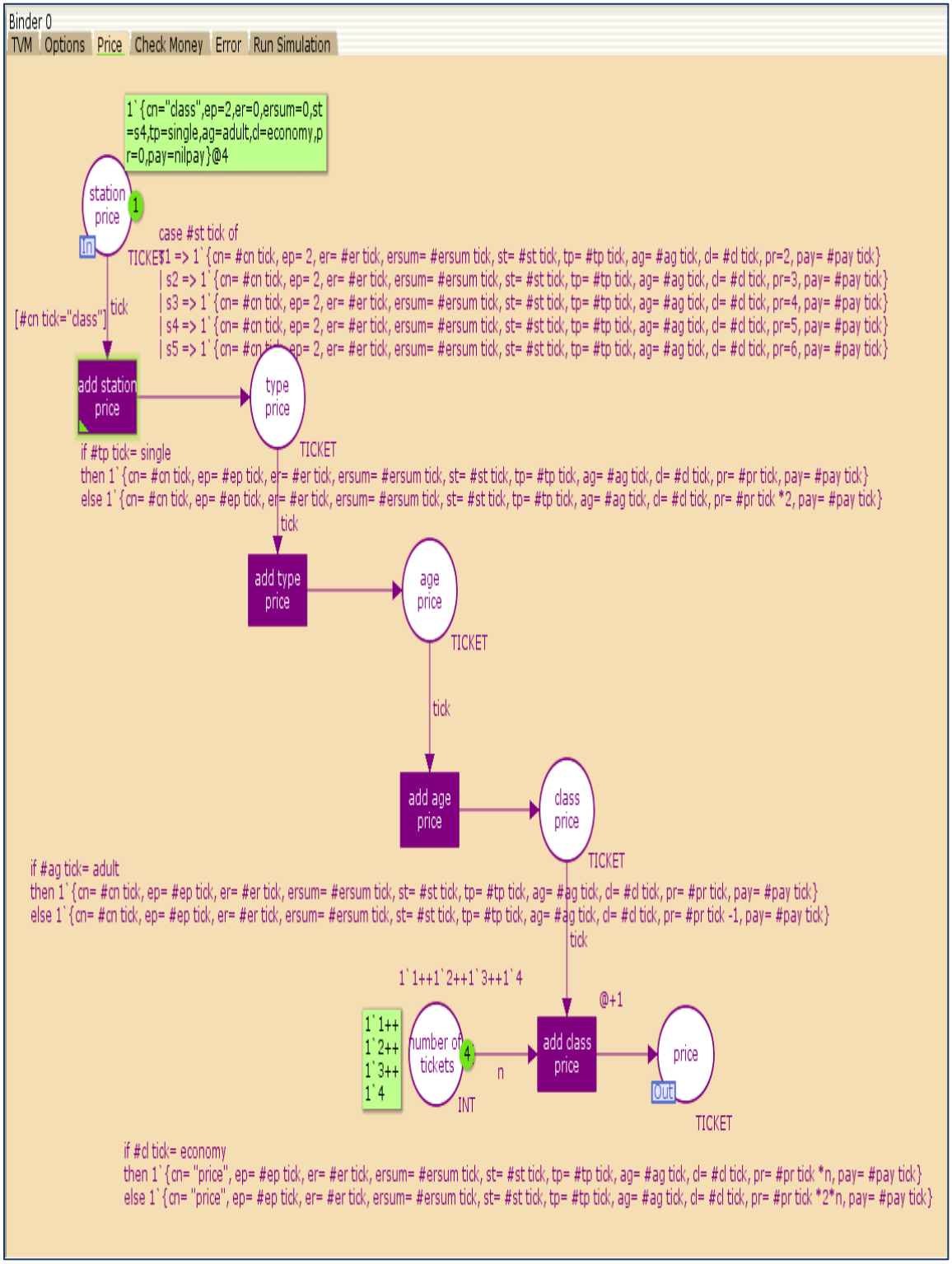
Όρισμα της συνάρτησης check(y) είναι ο αριθμός των επιλογών του χρήστη. Στην μηχανή MFM ο αριθμός αυτός είναι μικρός (από 2 έως 5) σε αντίθεση με ότι συμβαίνει στη μηχανή QFM (40 επιλογές), γεγονός στο οποίο οφείλεται και η μικρή πιθανότητα παραγωγής λάθους. Σε περίπτωση που παραχθεί ένα τυχαίο λάθος, ενεργοποιείται η μετάβαση διόρθωσης λάθους *(correct error)*. Κατά την διόρθωση γίνεται έλεγχος της κατάστασης *(cn)* από την οποία προήλθε η μάρκα του εισιτηρίου. Ο προορισμός από τον οποίο προήλθε η μάρκα, φανερώνει το στάδιο στο οποίο έλαβε χώρα το λάθος. Η λανθασμένη επιλογή αρχικοποιείται, ενώ όλες οι υπόλοιπες παραμένουν ως είχαν. Το εισιτήριο επιστρέφει ακριβώς στην αρχή του βήματος, στο οποίο εντοπίστηκε το λάθος και αναμένει από τον χρήστη να κάνει εκ νέου την επιθυμητή επιλογή του. Στο παρακάτω τμήμα κώδικα, περιγράφεται η αρχικοποίηση των πεδίων της μάρκας, στα οποία εμφανίστηκε το λάθος.

```
case #cn tick of
"station" => 1`{cn= "waiting", ep= 0, er= # er tick +1, ersum= #ersum tick +1, st=nilst,
tp=niltp, ag=nilag, cl=nilcl, pr=0, pay=nilpay}
| "type" => 1`{cn= "station", ep= 0, er= # er tick +1, ersum= #ersum tick +1, st= #st tick,
tp=niltp, ag=nilag, cl=nilcl, pr=0, pay=nilpay}
| "age" => 1`{cn= "type", ep= 0, er= # er tick +1, ersum= #ersum tick +1, st= #st tick, tp=
#tp tick, ag=nilag, cl=nilcl, pr=0, pay=nilpay}
| "class" => 1`{cn= "age", ep= 0, er= # er tick +1, ersum= #ersum tick +1, st= #st tick, tp=
#tp tick, ag= #ag tick, cl=nilcl, pr=0, pay=nilpay}
| "payway" => 1`{cn= "price", ep= #ep tick, er= # er tick +1, ersum= #ersum tick +1, st=
#st tick, tp= #tp tick, ag= #ag tick, cl= #cl tick, pr= #pr tick, pay=nilpay}
```

Αν, για παράδειγμα, η κατάσταση από την οποία προήλθε το εισιτήριο ήταν η κατάσταση επιλογής του τύπου του εισιτηρίου *(#cn tick="type"),* τότε αρχικοποιείται μόνο το πεδίο που αναφέρεται σε αυτή *(tp=niltp)*, ενώ τα υπόλοιπα πεδία παραμένουν ως είχαν.

Αφού ολοκληρωθεί η διαδικασία ελέγχου των σφαλμάτων, το εισιτήριο καταφτάνει στο υποεπίπεδο 1 Τιμή (Price), που φαίνεται στην εικόνα 3.20, στο οποίο γίνεται σταδιακά ο υπολογισμός της τελικής τιμής του εισιτηρίου. Παρόλο που το υποεπίπεδο

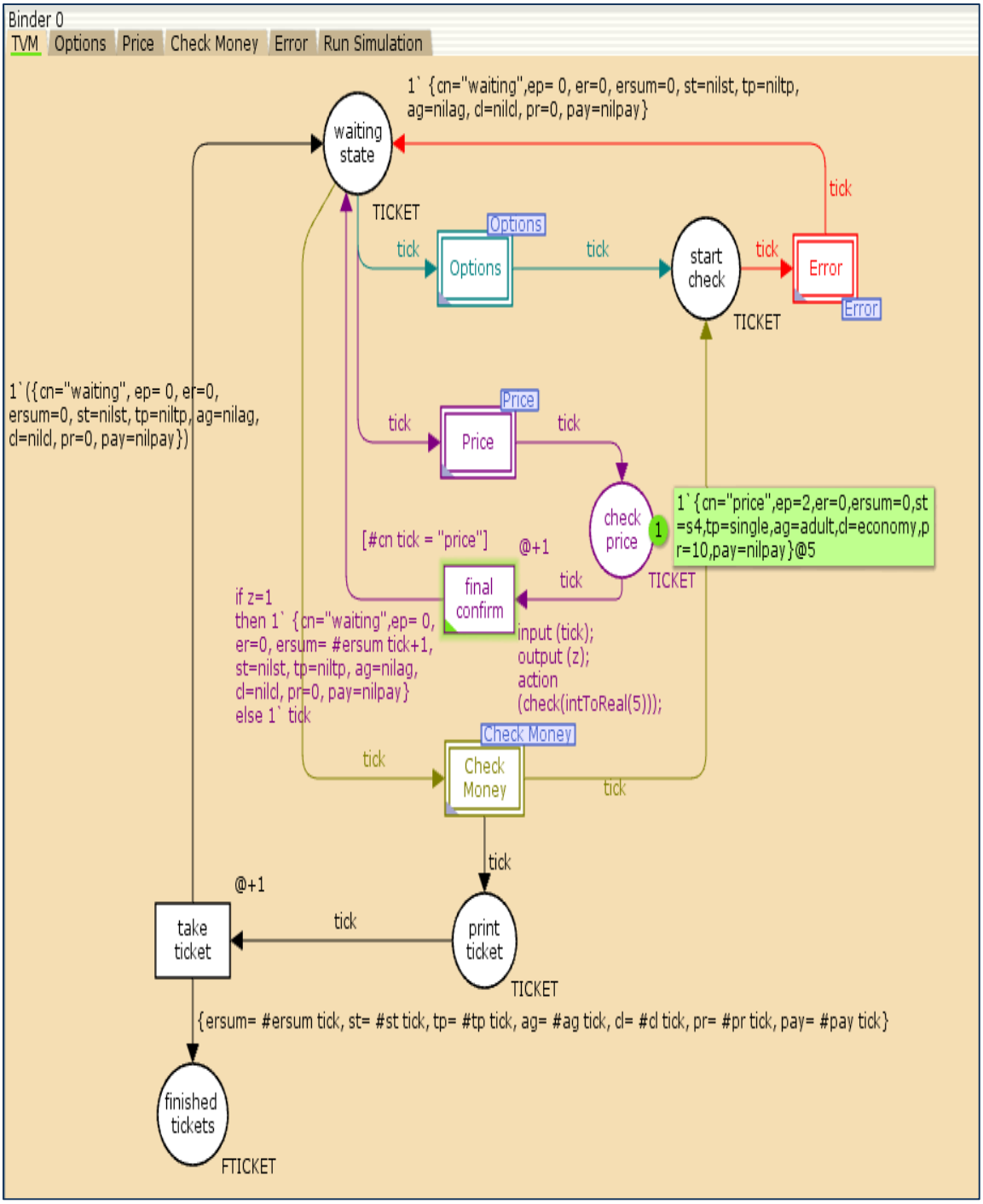
αυτό, έχει αναλυθεί διεξοδικά στην παράγραφο 3.3.1, υπενθυμίζονται, εν συντομία, οι



Εικόνα 3.20 Υποεπίπεδο 1 Τιμή – Price (MFM)

βασικές του λειτουργίες. Πρώτα, ενεργοποιείται η μετάβαση της προσθήκης της τιμής του προορισμού *(add station price),* όπου αναλόγως την επιλογή προορισμού του χρήστη, προστίθεται στην τιμή του εισιτηρίου το αντίστοιχο ποσό που αναλογεί στην συγκεκριμένη επιλογή. Στη συνέχεια, ενεργοποιείται η μετάβαση της προσθήκης της τιμής του τύπου του εισιτηρίου *(add type price)*, όπου η προηγούμενη τιμή αυξάνεται μόνο εφόσον ο χρήστης επιλέξει εισιτήριο μετ' επιστροφής. Έπειτα, ενεργοποιείται η μετάβαση της προσθήκης της τιμής της ηλικίας του επιβάτη για τον οποίο προορίζεται το εισιτήριο *(add age price)*, όπου η προηγούμενη τιμή του εισιτηρίου μειώνεται μόνο εφόσον ο χρήστης επιλέξει παιδικό εισιτήριο. Τέλος, ενεργοποιείται η μετάβαση της προσθήκης της τιμής της κλάσης του εισιτηρίου *(add class price),* όπου η προηγούμενη τιμή του εισιτηρίου αυξάνεται μόνο εφόσον ο χρήστης έχει επιλέξει εισιτήριο πρώτης θέσης. Η μετάβαση αυτή συνδέεται και με τη θέση αριθμός εισιτηρίων *(number of tickets),* η οποία αντιπροσωπεύει τον αριθμό των όμοιων εισιτηρίων που επιλέγει ο χρήστης. Η τελική τιμή του εισιτηρίου προκύπτει αφού πολλαπλασιαστεί η τρέχουσα τιμή με τον αριθμό των όμοιων εισιτηρίων που επιθυμεί ο χρήστης.

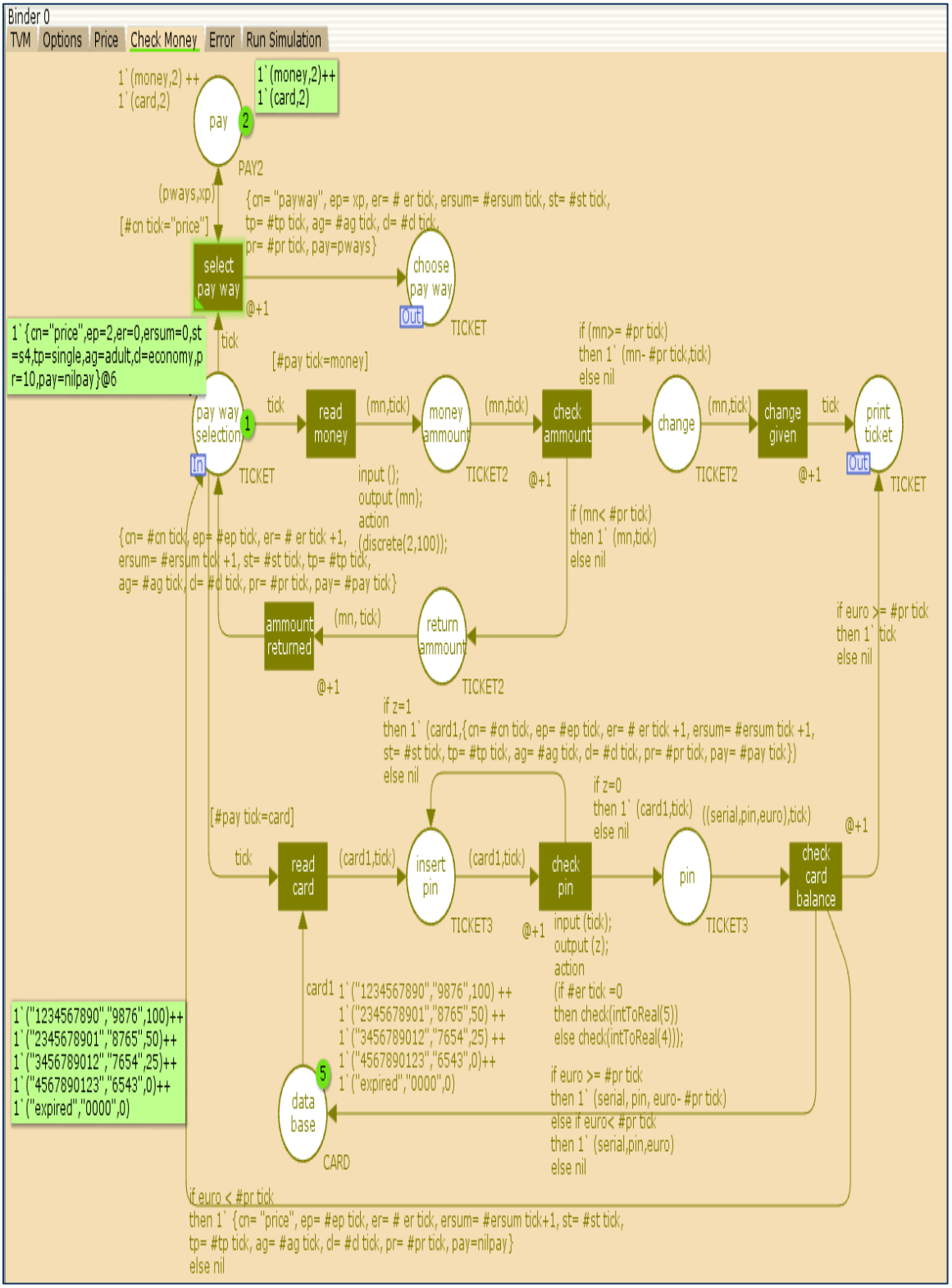
Αφού ολοκληρωθεί ο υπολογισμός της τελικής τιμής του εισιτηρίου, ενεργοποιείται η θέση ελέγχου της τιμής *(check price)* και η τελική τιμή παρουσιάζεται στο χρήστη όπως φαίνεται στην εικόνα 3.21. Πιθανότατα, ο χρήστης να είχε κάνει κάποιο λάθος στα προηγούμενα βήματα, το οποίο δεν είχε συνειδητοποιήσει μέχρι αυτή τη στιγμή. Αν η τελική τιμή δεν είναι η αναμενόμενη, τότε ο χρήστης έχει τη δυνατότητα να ακυρώσει το τρέχων εισιτήριο και να επιστρέψει στην κατάσταση αναμονής της μηχανής *(waiting state),* για να κάνει εξ' αρχής μια νέα επιλογή. Όλη η διαδικασία της παραγωγής ή μη του λάθους εκτελείται στη μετάβαση της τελικής επιβεβαίωσης *(final confirm)* και αναλύεται διεξοδικά στην παράγραφο 3.3.1.



Εικόνα 3.21 Επίπεδο 0 τελική επιβεβαίωση – final confirm (MFM)

Η επόμενη ενεργοποιημένη μετάβαση είναι η μετάβαση επιλογής του τρόπου πληρωμής του εισιτηρίου *(select pay way),* του υποεπιπέδου 1 Ελέγχου Χρημάτων (Check Money) που φαίνεται στην εικόνα 3.22. Η μετάβαση αυτή ενεργοποιείται μόνο όταν η μάρκα που φτάνει σε αυτήν έχει προέλθει από την κατάσταση υπολογισμού της

τελικής τιμής, δηλαδή όταν cn ="price". Στο σημείο αυτό ο χρήστης επιλέγει τον τρόπο

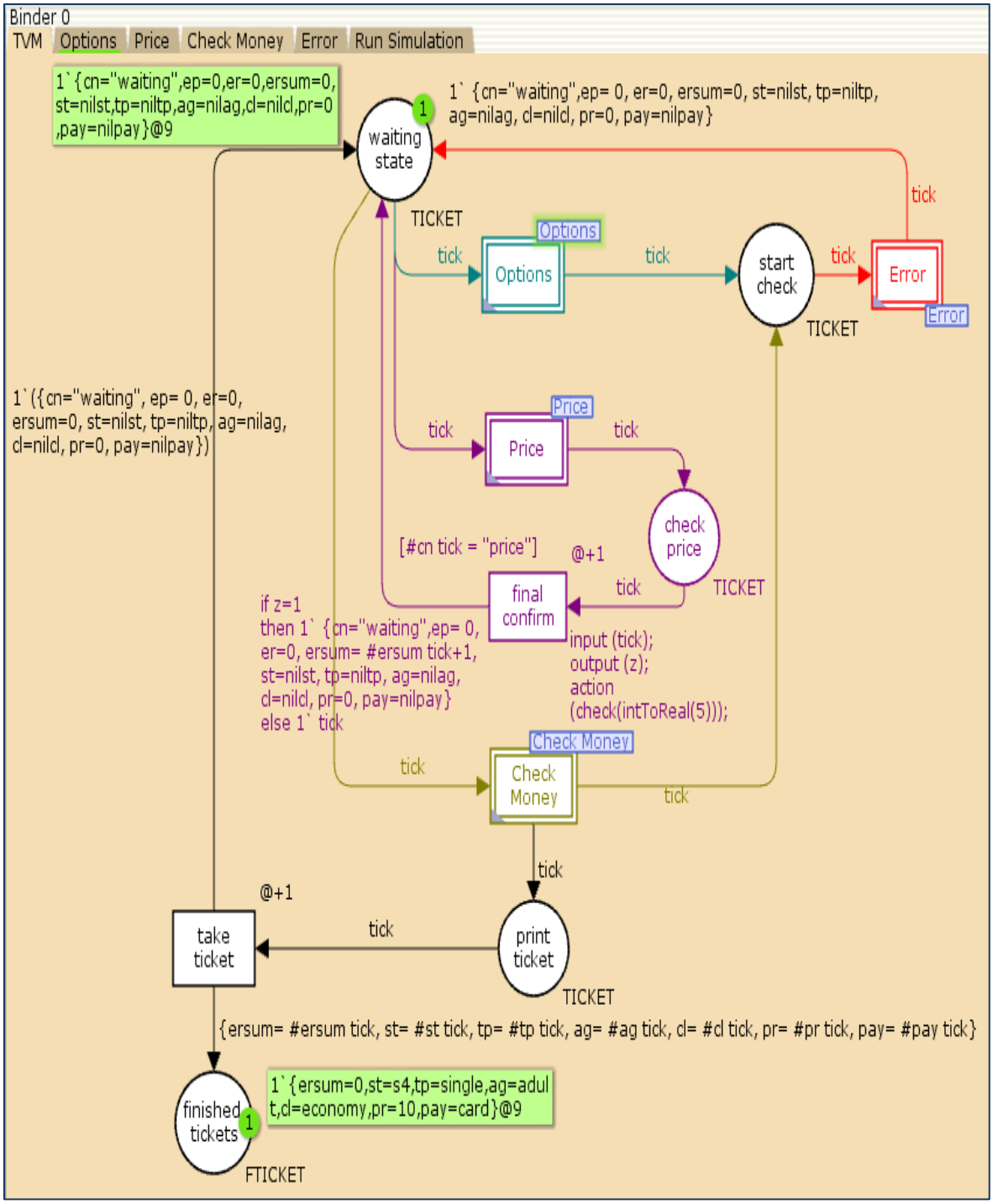


Εικόνα 3.22 Υποεπίπεδο 1 Έλεγχος Χρημάτων – Check Money (MFM)

με τον οποίο επιθυμεί να πληρώσει το εισιτήριο του, επιλογή που καθορίζει την μετέπειτα πορεία του εισιτηρίου και στη συνέχεια γίνεται και πάλι έλεγχος της σωστής ή μη επιλογής του χρήστη, με τη βοήθεια του υποεπιπέδου 1 Σφάλμα (Error). Η διαδικασία αποπληρωμής του εισιτηρίου αναλύεται διεξοδικά στην παράγραφο 3.3.1 περιγράφεται, όμως, εν συντομία και παρακάτω.

- Σε περίπτωση, που ο χρήστης επιλέξει να πληρώσει το εισιτήριο του με μετρητά, η μηχανή ελέγχει αν το εισηγμένο ποσό επαρκεί για την αποπληρωμή του εισιτηρίου. Στην περίπτωση που επαρκεί, η μηχανή εισπράττει το ποσό που αντιστοιχεί στην τελική τιμή του εισιτηρίου και επιστρέφεται το υπόλοιπο ποσό στο χρήστη, ενώ αν το ποσό δεν επαρκεί για την εξόφληση του εισιτηρίου, επιστρέφεται εξ ολοκλήρου στο χρήστη.
- Σε περίπτωση, που ο χρήστης επιλέγει να πληρώσει το εισιτήριο του με τη χρήση κάρτας, η μηχανή ελέγχει αν ο χρήστης εισήγαγε τον σωστό προσωπικό κωδικό *(pin)* της κάρτας. Αν εισήχθη λανθασμένος κωδικός, η μηχανή ζητάει από τον χρήστη να εισάγει τον σωστό κωδικό. Στη συνέχεια, γίνεται έλεγχος της διαθεσιμότητας της κάρτας. Αν το χρηματικό ποσό της κάρτας επαρκεί, η μηχανή εισπράττει το ποσό που αντιστοιχεί στην τελική τιμή του εισιτηρίου και επιστρέφεται η κάρτα, ενώ αν δεν επαρκεί, επιστρέφεται η κάρτα.

Τελειώνοντας, το εισιτήριο τυπώνεται και παραδίδεται στον χρήστη. Η μηχανή επιστρέφει στην κατάσταση αναμονής *(waiting state),* ενώ η θέση τελειωμένα εισιτήρια *(finished tickets)* όπως φαίνεται και στην εικόνα 3.23*,* χρησιμεύει ως αποθήκη, στην οποία φυλάσσονται τα ολοκληρωμένα εισιτήρια. Μεγάλη είναι η χρησιμότητα της στη διεξαγωγή συμπερασμάτων, καθώς και στην αξιολόγηση του συστήματος.



Εικόνα 3.23 Επίπεδο 0 Αποθήκη ολοκληρωμένων εισιτηρίων – finished tickets (MFM)

4 Προσομοίωση

4.1 Ανάλυση απόδοσης συστημάτων & προσομοίωση

Η ανάγκη για τον υπολογισμό και την πρόβλεψη της απόδοσης συστημάτων εμφανίζεται σε όλες τις φάσεις ανάπτυξης και λειτουργίας τους, από το αρχικό στάδιο του σχεδιασμού, μέχρι την καθημερινή χρήση τους. Η αξιολόγηση της απόδοσης ενός συστήματος αποτελεί το επίκεντρο στη σχεδίαση, την ανάπτυξη και τη διαμόρφωση του συστήματος. Πολλές φορές δεν αρκεί ένα σύστημα να λειτουργεί χωρίς προβλήματα, αλλά θα πρέπει και να λειτουργεί αποτελεσματικά. Η ανάλυση της απόδοσης ενός συστήματος αποτελεί την κύρια μέθοδο αξιολόγησης των ήδη σχεδιασμένων συστημάτων, αλλά και την κύρια μέθοδο σύγκρισης εναλλακτικών μορφών τους, με σκοπό την εύρεση της βέλτιστης μορφής του συστήματος αυτού. Η προσομοίωση (*simulation*) είναι η πιο διαδεδομένη μέθοδος ανάλυσης ενός συστήματος και εξοικείωσης με τα χαρακτηριστικά του.

Η προσομοίωση αποτελεί μία πειραματική μέθοδο που έχει ως σκοπό τη βελτίωση συστημάτων, την ανάλυση της ευαισθησίας τους και τη μελέτη της λειτουργίας τους. Ως πειραματική μέθοδος εξαρτάται πολύ από την πιστότητα του μοντέλου του συστήματος που χρησιμοποιείται, καθώς και από την επιλογή εκείνων των παραμέτρων που απαιτούνται για την εξαγωγή αξιόπιστων και χρήσιμων συμπερασμάτων. Κατά την προσομοίωση γίνεται αναπαράσταση μιας διεργασίας με τη βοήθεια ενός μοντέλου. Η αναπαράσταση αυτή συνήθως είναι οικονομικότερη, ταχύτερη, λιγότερο επικίνδυνη και περισσότερο προσιτή από την πραγματική διεργασία. Η προσομοίωση είναι, επίσης, πολύ χρήσιμη στις περιπτώσεις που μπορούμε να πάρουμε μόνο προσεγγιστική αναλυτική λύση, οπότε θέλουμε να επιβεβαιώσουμε την προσέγγιση, όπως για παράδειγμα, να βρούμε το σχετικό λάθος. Οι κατηγορίες της προσομοίωσης που εφαρμόζονται στα δίκτυα Petri είναι οι εξής:

- **Προσομοίωση δραστηριοτήτων,** κατά τη οποία γίνεται εκτίμηση του μοντέλου σε σταθερά χρονικά διαστήματα, ανεξάρτητα από τις αλλαγές που συμβαίνουν σ' αυτό.

- **Προσομοίωση γεγονότων,** κατά τη οποία γίνεται εκτίμηση του μοντέλου μόνον όταν αλλάξει η τιμή κάποιας παραμέτρου του μοντέλου.
- **Προσομοίωση διεργασιών,** κατά τη οποία προσφέρεται ένα υψηλότερο επίπεδο άποψης του συστήματος, επιτρέποντας στον σχεδιαστή να μοντελοποιεί κάθε επί μέρους τμήμα του μοντέλου, ανεξάρτητα από τα άλλα. Τα γεγονότα που είναι εσωτερικά σε κάθε τμήμα απομονώνονται από τα γεγονότα που συμβαίνουν σε άλλα τμήματα του μοντέλου.

Η ανάλυση απόδοσης ενός συστήματος με τη χρήση της προσομοίωσης συμπεριλαμβάνει τη στατιστική διερεύνηση και την απεικόνιση των εξόδων του συστήματος, την εξερεύνηση μεγάλων ομάδων δεδομένων καθώς και την επαλήθευση και επικύρωση των πειραμάτων της προσομοίωσης. Προκειμένου να γίνει η προσομοίωση ενός δικτύου Petri, απαιτείται ένας σχετικά μεγάλος όγκος πληροφοριών για την απόδοση του συστήματος, όπως είναι το μήκος των ουρών αναμονής *(queue length)*, ο χρόνος απόδοσης *(response time)* και η ρυθμοαπόδοση *(throughput).* Οι πληροφορίες αυτές εξάγονται από το μοντέλο, με εξέταση της σήμανσης του δικτύου *(marking of places)* και των εμφανιζόμενων δεσμεύσεων των στοιχείων *(binding process),* κατά τη διάρκεια της προσομοίωσης[17]. Μερικά παραδείγματα μέτρων απόδοσης που υπολογίζονται με την εξαγωγή στοιχείων από τη σήμανση του δικτύου, είναι ο αριθμός των μαρκών σε μια θέση, (στην παρούσα διατριβή, ο αριθμός των ολοκληρωμένων εισιτηρίων στην θέση τελειωμένα εισιτήρια *(finished tickets),* του επιπέδου 0 TVM), καθώς και ο αριθμός των στοιχείων σε μια εγγραφή, (στην παρούσα διατριβή, ο συνολικός αριθμός των σφαλμάτων *(ersum),* κατά την παραγωγή ενός εισιτηρίου). Άλλα παραδείγματα μέτρων απόδοσης που υπολογίζονται με την εξαγωγή δεδομένων από τη δέσμευση στοιχείων και πληροφοριών του δικτύου, είναι η χρονική διάρκεια της διαδρομής ενός προϊόντος στο δίκτυο, για παράδειγμα, ενός εισιτηρίου. Στην περίπτωση αυτή, ο χρόνος έκδοσης του εισιτηρίου, θα πρέπει, προφανώς, να

[17] http://wiki.daimi.au.dk/cpntools-help/introduction_to_performan.wiki?cmd=get&anchor=Introduction+to+Performance+Analysis

συμπεριληφθεί στις πληροφορίες που συνδέονται με αυτό. Προκειμένου να μοντελοποιηθεί ο χρόνος που απαιτούν ορισμένες δραστηριότητες, χρησιμοποιήθηκε ο χρονισμός σε όλες τις διεργασίες του δικτύου Petri.

Γενικότερα, οι μετρήσιμες ποσότητες που χαρακτηρίζουν την απόδοση ενός συστήματος, μπορούν να ομαδοποιηθούν σε δύο κατηγορίες. Τα "μέτρα χρηστικότητας" *(user oriented measures)* και τα "μέτρα αξιολόγησης του συστήματος" *(system oriented measures).* Τα μέτρα χρηστικότητας περιλαμβάνουν:

- **Τον χρόνο ολοκλήρωσης** *(turnaround time),* από την υποβολή μιας δουλειάς μέχρι τη στιγμή που γίνονται διαθέσιμα τα αποτελέσματα. Στην παρούσα διατριβή είναι ο χρόνος που απαιτείται από την έναρξη έως την έκδοση του εισιτηρίου.
- **Τον χρόνο απόκρισης** *(response time),* από την άφιξη μίας απαίτησης *(request)* μέχρι την ικανοποίησή της από το σύστημα.
- **Τον χρόνος αντίδραση του συστήματος** *(system reaction time),* από την άφιξη στο σύστημα μίας απαίτησης, μέχρι τη στιγμή που θα πάρει την πρώτη μερίδα χρόνου εξυπηρέτησης *(time slice).*

Όλες οι παραπάνω ποσότητες θεωρούνται τυχαίες μεταβλητές και κατά συνέπεια, μελετάται η κατανομή τους, η μέση τιμή τους, η διακύμανσή τους και άλλες σχετικές ποσότητες. Τα μέτρα αξιολόγησης του συστήματος περιλαμβάνουν:

- **Την ρυθμοαπόδοση** *(throughput),* που ορίζεται ως ο μέσος αριθμός εργασιών που εξυπηρετούνται από το σύστημα στη μονάδα του χρόνου και αποτελεί ένα μέτρο της παραγωγικότητας του συστήματος.
- **Τη χρησιμοποίηση** *(utilization)* ενός πόρου, είναι το κλάσμα του χρόνου που ο πόρος αυτός είναι απασχολημένος.

4.2 Προσομοίωση μηχανών QFM & MFM

Η παρούσα διατριβή ασχολείται με τη μοντελοποίηση και τη σύγκριση των δύο τύπων μηχανών αυτόματης πώλησης εισιτηρίων (QFM και MFM), δίνοντας έμφαση στην

μέτρηση της απόδοσης των συστημάτων κάτω από διαφορετικές συνθήκες λειτουργίας, αλλά και στην πρόληψη των ανθρώπινων λαθών.

Υπάρχουν διάφορες μέθοδοι αξιολόγησης της απόδοσης ενός παραγωγικού συστήματος, οι οποίες σχετίζονται με το πλαίσιο που έχει τεθεί για την μοντελοποίηση του συστήματος, τους στόχους που πρέπει να επιτευχθούν με την αξιολόγηση αυτή και τους περιορισμούς που θέτει το ίδιο το σύστημα. Εκτός από την παραγωγικότητα του συστήματος, υπάρχει και ένα πλήθος δεδομένων που μπορούν να την επηρεάζουν σημαντικά, είτε να δίνουν ενδείξεις για την ορθή πορεία της. Τέτοιου είδους δεδομένα είναι ο αριθμός των λαθών που παράγονται κατά τη διαδικασία παραγωγής ενός εισιτηρίου, καθώς και ο χρόνος επεξεργασίας κάθε εισιτηρίου.

Η πρόληψη των ανθρώπινων λαθών επιτυγχάνεται με την ενεργοποίηση συγκεκριμένων αναδράσεων *(feedback)* που προκαλούν την επανάληψη είτε τμήματος των εργασιών στο οποίο εντοπίστηκε το λάθος, όπως, για παράδειγμα, συμβαίνει στο υποεπίπεδο 1 Σφάλμα (Error) είτε ενός μεγαλύτερου μέρους της διαδικασίας που περιλαμβάνει και προηγούμενες μεταβάσεις, εκτός από αυτή που έλαβε χώρα το ανθρώπινο λάθος, όπως, για παράδειγμα, συμβαίνει στην μετάβαση της τελικής επιβεβαίωσης *(final confirm)*. Στη δεύτερη περίπτωση, οι εν λόγω μεταβάσεις θεωρούνται αλληλένδετες με τη μετάβαση που εμφανίστηκε το ανθρώπινο σφάλμα και επιβάλλεται να επαναληφθούν.

Μετά τη δημιουργία και την επαλήθευση του ορθού τρόπου λειτουργίας του υπό εξέταση μοντέλου, επιτακτική είναι η ανάγκη για λήψη των αποφάσεων πριν την έναρξη της οποιασδήποτε μελέτης ή προσομοίωσης. Το πειραματικό στάδιο ασχολείται με τον καθορισμό των σεναρίων, τα οποία πρόκειται να προσομοιωθούν, αλλά και τον τρόπο με τον οποίο κάθε σενάριο θα προσομοιωθεί στη μελέτη. Ο αναλυτής θα πρέπει να αποφασίσει πόσες προσομοιώσεις θα γίνουν για το κάθε σενάριο της προσομοίωσης, αλλά και πόσο θα διαρκέσουν οι προσομοιώσεις αυτές. Τόσο η χρονική διάρκεια, όσο και ο αριθμός των δοκιμών των προσομοιώσεων αυτών, μπορούν να έχουν σημαντικό

αντίκτυπο στις εκτιμήσεις των μέτρων απόδοσης. Η λήψη αυθαίρετων αποφάσεων συχνά οδηγεί σε σπατάλη χρόνου, με αποτέλεσμα την παραγωγή μη χρήσιμων ή ακόμα και λανθασμένων αποτελεσμάτων.

Για τη συγκέντρωση των απαραίτητων δεδομένων, που θα χρησιμοποιηθούν στη μέτρηση της απόδοσης των μηχανών αυτόματης πώλησης εισιτηρίων, έγινε καταγραφή του πλήθους των ολοκληρωμένων εισιτηρίων στην αποθήκη *(buffer)* των τελειωμένων εισιτηρίων *(finished tickets),* αλλά και καταγραφή ορισμένων στοιχείων των εισιτηρίων αυτών, όπως, για παράδειγμα, τον αριθμό των λαθών που έλαβαν χώρα κατά την έκδοση του κάθε εισιτηρίου. Ελεγκτικοί καταγραφείς *(monitors),* χρησιμοποιήθηκαν για να εξάγουν τα στοιχεία που προκύπτουν κατά την προσομοίωση από το δίκτυο Petri. Κάθε μέτρο απόδοσης υπολογίζεται από ένα καταγραφέα δεδομένων, ενώ ένας καταγραφέας μπορεί να υπολογίσει περισσότερα από ένα μέτρα απόδοσης. Για την καταγραφή του χρόνου παραγωγής, καθώς και του πλήθους των λαθών του κάθε εισιτηρίου, υλοποιήθηκαν οι καταγραφείς One button Finished Ticket Monitor (εικόνα 4.1) και Finished Ticket Observation για τις μηχανές QFM και MFM αντίστοιχα. Οι δύο αυτοί καταγραφείς είναι όμοιοι και είναι τύπου Write in file monitors, καταγράφουν δηλαδή σε αρχεία τύπου .txt, τα ζητούμενα μέτρα απόδοσης. Στη συνέχεια, με τη βοήθεια της εφαρμογής των λογιστικών φύλλων *(Microsoft Office Excel)*, μπορούμε να υπολογίσουμε τον χρόνο παραμονής του κάθε εισιτηρίου στο σύστημα, το μέσο λάθος καθώς και το μέσο χρόνο ολοκλήρωσης ανά εισιτήριο. Τέλος, με τους καταγραφείς One Button Stop Simulation και Stop Simulation, οι οποίοι είναι τύπου Break point monitors, καθορίζεται η χρονική διάρκεια της κάθε προσομοίωσης.

```
▼Monitors
 ▼One Button Finished Ticket Monitor
  ▶Type: Write in file
  ▶Nodes ordered by pages
  ▶Init
  ▼Predicate
     fun pred (bindelem) =
     let
      fun predBindElem (TVM'take_ticket (1, {tick})) = true
          | predBindElem _ = false
     in
      predBindElem bindelem
     end
  ▼Observer
     fun obs (bindelem) =
     let
      fun obsBindElem (TVM'take_ticket (1, {tick})) = Int.toString(intTime()) ^ " "^(Int.toString(#ersum tick))^"\n"
          | obsBindElem _ = ""
     in
      obsBindElem bindelem
     end
  ▶Stop
 ▼One Button Stop Simulation
    Type: Break point
  ▼Nodes ordered by pages
   ▼TVM
      take_ticket (transition)
  ▼Predicate
     fun pred (bindelem) =
     let
      fun predBindElem (TVM'take_ticket (1, {tick})) = intTime()> 5000
          | predBindElem _ = false
     in
      predBindElem bindelem
     end
```

Εικόνα 4.1 Καταγραφείς για την μηχανή QFM

5 Πειράματα

5.1 Περιγραφή πειραμάτων

Στο κεφάλαιο αυτό γίνεται σύγκριση των δύο τύπων μηχανών αυτόματης πώλησης εισιτηρίων (QFM και MFM), στη μοντελοποίηση των οποίων έχουν ενσωματωθεί η πρόληψη και η αντιμετώπιση των ανθρώπινων λαθών, καθώς και των ανεπιθύμητων καταστάσεων του συστήματος. Τα μοντέλα στα οποία βασίστηκε η σύγκριση των δύο μηχανών είναι εκείνα που περιγράφηκαν αναλυτικά στις παραγράφους 3.3.1 και 3.3.2. Έμφαση δίνεται στο μέσο χρόνο που απαιτείται για να παραχθεί ένα εισιτήριο, από τη στιγμή δηλαδή, που ο χρήστης φθάνει στη μηχανή έως να παραλάβει το τυπωμένο εισιτήριο στα χέρια του, αλλά και στο μέσο ανθρώπινο λάθος κατά την παραγωγή του εισιτηρίου αυτού.

Στη συνέχεια, παραθέτονται συνοπτικά τα ανθρώπινα λάθη, που προσμετρούνται και εξετάζονται στο σύστημά μας, καθώς και τα σημεία του μοντέλου, στα οποία λαμβάνει χώρα η αντιμετώπιση αυτών.

- Ένα από τα συχνότερα ανθρώπινα λάθη, εντοπίζεται στην εσφαλμένη επιλογή του προορισμού του επιβάτη, λάθος που αντιμετωπίζεται στο υποεπίπεδο 1 Σφάλμα (Error).
- Ένα ακόμα ανθρώπινο λάθος, αφορά στη λανθασμένη επιλογή του τύπου του εισιτηρίου, αν, δηλαδή, το εισιτήριο θα είναι για απλή μετάβαση στον προεπιλεγμένο προορισμό ή αν θα είναι εισιτήριο με επιστροφή, λάθος που αντιμετωπίζεται στο υποεπίπεδο 1 Σφάλμα (Error).
- Επίσης, έχει παρατηρηθεί ότι αρκετοί χρήστες κάνουν λάθος στην ηλικία του επιβάτη για τον οποίο προορίζεται το εισιτήριο, επιλέγουν για παράδειγμα εισιτήριο ενηλίκου, ενώ ουσιαστικά χρειάζονται εισιτήριο για φοιτητή ή παιδικό εισιτήριο, λάθος που αντιμετωπίζεται στο υποεπίπεδο 1 Σφάλμα (Error).
- Ανθρώπινο λάθος παρατηρήθηκε και κατά την επιλογή της κλάσης του εισιτηρίου, όπου για παράδειγμα ο χρήστης διάλεξε εισιτήριο πρώτης κλάσης, ενώ

ουσιαστικά ήθελε οικονομικό εισιτήριο, λάθος που αντιμετωπίζεται στο υποεπίπεδο 1 Σφάλμα (Error).

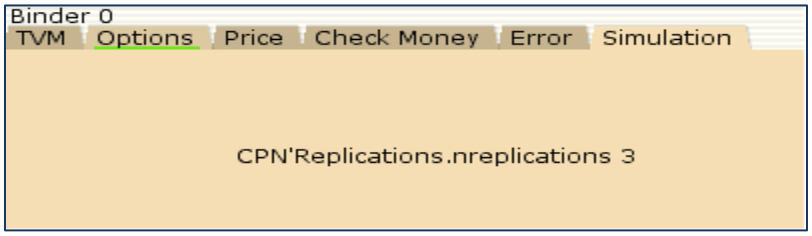
- Ένα επιπλέον ανθρώπινο λάθος, αναφέρεται στη λανθασμένη επιλογή του τρόπου πληρωμής του εισιτηρίου, λάθος που αντιμετωπίζεται στο υποεπίπεδο 1 Σφάλμα (Error).
- Ακόμα, ανθρώπινο σφάλμα παρουσιάζεται κατά την εισαγωγή μη επαρκούς χρηματικού ποσού στην μηχανή, λάθος που αντιμετωπίζεται στο υποεπίπεδο 1 Έλεγχος Χρημάτων (Check Money).
- Επιπρόσθετα, ένα λάθος που εμφανίζεται συχνά είναι η μη αναμενόμενη υψηλή τιμή του εισιτηρίου που παρουσιάζεται στο χρήστη. Πιθανότατα ο χρήστης έκανε κάποια λάθος επιλογή στα προηγούμενα βήματα, χωρίς να τη συνειδητοποιήσει. Το λάθος αυτό αντιμετωπίζεται στο επίπεδο 0 TVM.
- Τέλος, σφάλματα εμφανίζονται και κατά την πληρωμή ενός εισιτηρίου με κάρτα. Τα λάθη αυτά οφείλονται είτε στη εισαγωγή μη έγκυρου προσωπικού κωδικού *(pin),* είτε στο μη επαρκή χρηματικό υπόλοιπο της κάρτας. Τα λάθη αυτά αντιμετωπίζεται στο υποεπίπεδο 1 Έλεγχος Χρημάτων (Check Money).

Το πειραματικό στάδιο ασχολείται με τον καθορισμό των σεναρίων, τα οποία πρόκειται να προσομοιωθούν, αλλά και τον τρόπο με τον οποίο κάθε σενάριο θα προσομοιωθεί στη μελέτη. Τα πειράματα που πραγματοποιούνται, εξετάζουν την επίδραση που έχει η σταδιακή αύξηση των επιλογών του χρήστη, στο μέσο χρόνο που απαιτείται για την παραγωγή ενός εισιτήριου, αλλά και στο μέσο ανθρώπινο λάθος κατά την παραγωγή του εισιτηρίου αυτού. Συγκεκριμένα, τα πειράματα που πραγματοποιούνται αφορούν:

- 32 επιλογές του χρήστη (2 προορισμοί x 2 τύποι εισιτηρίου x 2 ηλικίες επιβάτη x 2 κλάσεις εισιτηρίου x 2 τρόποι πληρωμής).
- 48 επιλογές του χρήστη (3 προορισμοί x 2 τύποι εισιτηρίου x 2 ηλικίες επιβάτη x 2 κλάσεις εισιτηρίου x 2 τρόποι πληρωμής).

- 80 επιλογές του χρήστη (5 προορισμοί x 2 τύποι εισιτηρίου x 2 ηλικίες επιβάτη x 2 κλάσεις εισιτηρίου x 2 τρόποι πληρωμής).
- 144 επιλογές του χρήστη (6 προορισμοί x 2 τύποι εισιτηρίου x 3 ηλικίες επιβάτη x 2 κλάσεις εισιτηρίου x 2 τρόποι πληρωμής).
- 192 επιλογές του χρήστη (8 προορισμοί x 2 τύποι εισιτηρίου x 3 ηλικίες επιβάτη x 2 κλάσεις εισιτηρίου x 2 τρόποι πληρωμής).
- 240 επιλογές του χρήστη (10 προορισμοί x 2 τύποι εισιτηρίου x 3 ηλικίες επιβάτη x 2 κλάσεις εισιτηρίου x 2 τρόποι πληρωμής).

Προκειμένου να ολοκληρωθεί το κάθε πείραμα, πραγματοποιούνται τρεις επαναληπτικές προσομοιώσεις, οι οποίες υλοποιούνται με τη βοήθεια του τμήματος κώδικα που φαίνεται στην εικόνα 5.1. Η χρονική διάρκεια της κάθε μιας από τις προσομοιώσεις ορίζεται στις 5000 χρονικές μερίδες *(time slice),* η υλοποίηση της οποίας γίνεται με χρήση του καταγραφέα Stop Simulation, ο οποίος είναι τύπου Break point monitors και φαίνεται στην εικόνα 5.2.



Εικόνα 5.1 Κώδικας προσομοιώσεων

```
▼Stop Simulation
   Type: Break point
  ▼Nodes ordered by pages
    ▼TVM
       take_ticket (transition)
  ▼Predicate
     fun pred (bindelem) =
     let
       fun predBindElem (TVM'take_ticket (1, {tick})) = intTime()> 5000
           | predBindElem _ = false
     in
       predBindElem bindelem
     end
```

Εικόνα 5.2 Κώδικας καταγραφέα Stop Simulation

5.2 Σύγκριση QFM & MFM για 32 δυνατές επιλογές

Στον πίνακα 5.1 παρουσιάζεται ένα τμήμα από τα πειραματικά αποτελέσματα των τριών προσομοιώσεων που πραγματοποιήθηκαν στη μηχανή QFM ενώ οι δυνατές επιλογές του χρήστη ήταν 32, δηλαδή 2 προορισμοί x 2 τύποι εισιτηρίου x 2 ηλικίες επιβάτη x 2 κλάσεις εισιτηρίου x 2 τρόποι πληρωμής.

Simulation 1			Simulation 2			Simulation 3		
Χρόνος Εξόδου	Αριθμός Λαθών	Χρόνος Εισιτηρίου	Χρόνος Εξόδου	Αριθμός Λαθών	Χρόνος Εισιτηρίου	Χρόνος Εξόδου	Αριθμός Λαθών	Χρόνος Εισιτηρίου
5	0	5	5	0	5	5	0	5
12	1	7	13	2	8	16	3	11
22	0	10	19	0	6	24	0	8
34	0	12	29	1	10	41	4	17
44	1	10	42	2	13	51	0	10
53	3	9	48	0	6	57	0	6
59	0	6	54	0	6	63	0	6
71	2	12	60	0	6	73	0	10
77	0	6	66	0	6	79	0	6
83	0	6	73	1	7	89	1	10
91	0	8	79	0	6	97	1	8
108	2	17	85	0	6	105	1	8
116	0	8	103	2	18	115	2	10
124	1	8	111	2	8	121	0	6
133	1	9	119	1	8	131	0	10
139	0	6	129	2	10	137	0	6
145	0	6	139	1	10	147	1	10
153	2	8	148	2	9	155	0	8
161	1	8	154	0	6	162	1	7
168	1	7	169	3	15	170	0	8

Πίνακας 5.1 Πειραματικά αποτελέσματα μηχανής QFM για 32 δυνατές επιλογές

Στον πίνακα 5.2 παρουσιάζεται ένα τμήμα από τα πειραματικά αποτελέσματα των τριών προσομοιώσεων που πραγματοποιήθηκαν στη μηχανή MFM ενώ οι δυνατές επιλογές του χρήστη ήταν 32, δηλαδή 2 προορισμοί x 2 τύποι εισιτηρίου x 2 ηλικίες επιβάτη x 2 κλάσεις εισιτηρίου x 2 τρόποι πληρωμής.

Simulation 1			Simulation 2			Simulation 3		
Χρόνος Εξόδου	Αριθμός Λαθών	Χρόνος Εισιτηρίου	Χρόνος Εξόδου	Αριθμός Λαθών	Χρόνος Εισιτηρίου	Χρόνος Εξόδου	Αριθμός Λαθών	Χρόνος Εισιτηρίου
8	0	8	8	0	8	10	2	10
19	0	11	18	1	10	19	0	9
34	2	15	29	0	11	34	6	15
43	0	9	41	1	12	43	0	9
52	0	9	50	0	9	54	2	11
61	0	9	60	1	10	65	1	11
70	0	9	72	2	12	76	1	11
79	0	9	87	0	15	85	0	9
98	2	19	96	0	9	96	1	11
107	0	9	105	0	9	109	0	13
116	0	9	116	0	11	124	0	15
125	0	9	125	0	9	135	0	11
134	0	9	136	0	11	144	0	9
143	0	9	145	0	9	153	0	9
152	0	9	156	1	11	172	0	19
163	0	11	165	0	9	198	5	26
172	0	9	178	0	13	209	0	11
189	0	17	190	3	12	234	2	25
198	0	9	207	1	17	243	0	9
209	1	11	222	0	15	252	0	9

Πίνακας 5.2 Πειραματικά αποτελέσματα μηχανής MFM για 32 δυνατές επιλογές

Τα συνολικά πειραματικά αποτελέσματα συγκεντρώνονται στον πίνακα 5.3 που φαίνεται παρακάτω. Στην ίδια χρονική διάρκεια 15000 μερίδες χρόνου, στη μηχανή QFM παράγονται 26,4% περισσότερα εισιτήρια από στη μηχανή MFM. Το μέσο λάθος όμως των εισιτηρίων που παράγονται από τη μηχανή QFM εντοπίζεται ότι είναι μεγαλύτερο κατά 35,7% από εκείνο των εισιτηρίων που παράγονται στη μηχανή MFM.

32 επιλογές	Αριθμός εισιτηρίων	Μέσος χρόνος έκδοσης εισιτηρίου	Μέσο λάθος ανά εισιτήριο
QFM	1567	9.59	0.95
MFM	1240	12.11	0.70

Πίνακας 5.3 Συνολικά πειραματικά αποτελέσματα για 32 δυνατές επιλογές

5.3 Σύγκριση QFM & MFM για 48 δυνατές επιλογές

Στον πίνακα 5.4 παρουσιάζεται ένα τμήμα από τα πειραματικά αποτελέσματα των τριών προσομοιώσεων που πραγματοποιήθηκαν στη μηχανή QFM ενώ οι δυνατές επιλογές του χρήστη ήταν 48, δηλαδή 3 προορισμοί x 2 τύποι εισιτηρίου x 2 ηλικίες επιβάτη x 2 κλάσεις εισιτηρίου x 2 τρόποι πληρωμής.

Simulation 1			Simulation 2			Simulation 3		
Χρόνος Εξόδου	Αριθμός Λαθών	Χρόνος Εισιτηρίου	Χρόνος Εξόδου	Αριθμός Λαθών	Χρόνος Εισιτηρίου	Χρόνος Εξόδου	Αριθμός Λαθών	Χρόνος Εισιτηρίου
5	0	5	5	0	5	7	0	7
15	2	10	11	0	6	20	2	13
21	0	6	17	0	6	35	2	15
27	0	6	27	3	10	41	0	6
46	2	19	41	4	14	49	1	8
52	0	6	47	0	6	58	1	9
58	0	6	57	2	10	64	0	6
64	0	6	63	0	6	70	0	6
70	0	6	71	0	8	76	1	6
82	3	12	84	3	13	87	2	11
88	0	6	93	0	9	93	0	6
94	0	6	99	0	6	104	1	11
100	0	6	107	2	8	112	1	8
114	4	14	113	0	6	118	0	6
120	0	6	122	0	9	126	1	8
128	0	8	138	1	16	132	0	6
142	1	14	144	0	6	144	2	12
148	0	6	150	0	6	150	0	6
156	1	8	156	0	6	160	0	10
162	0	6	166	2	10	166	0	6

Πίνακας 5.4 Πειραματικά αποτελέσματα μηχανής QFM για 48 δυνατές επιλογές

Στον πίνακα 5.5 παρουσιάζεται ένα τμήμα από τα πειραματικά αποτελέσματα των τριών προσομοιώσεων που πραγματοποιήθηκαν στη μηχανή MFM ενώ οι δυνατές επιλογές του χρήστη ήταν 48, δηλαδή 3 προορισμοί x 2 τύποι εισιτηρίου x 2 ηλικίες επιβάτη x 2 κλάσεις εισιτηρίου x 2 τρόποι πληρωμής.

Simulation 1			Simulation 2			Simulation 3		
Χρόνος Εξόδου	Αριθμός Λαθών	Χρόνος Εισιτηρίου	Χρόνος Εξόδου	Αριθμός Λαθών	Χρόνος Εισιτηρίου	Χρόνος Εξόδου	Αριθμός Λαθών	Χρόνος Εισιτηρίου
10	0	10	8	0	8	9	1	9
19	0	9	19	0	11	18	0	9
32	2	13	28	0	9	27	0	9
43	1	11	37	1	9	36	0	9
52	0	9	50	0	13	47	1	11
63	2	11	67	2	17	56	0	9
72	0	9	76	0	9	67	0	11
81	0	9	87	1	11	84	1	17
92	1	11	98	1	11	95	1	11
103	1	11	107	0	9	107	2	12
112	0	9	116	0	9	118	0	11
121	0	9	125	0	9	127	0	9
132	0	11	136	0	11	136	0	9
145	0	13	147	0	11	145	0	9
155	1	10	156	0	9	154	0	9
164	1	9	165	0	9	164	1	10
173	0	9	176	1	11	173	0	9
184	0	11	188	1	12	182	0	9
193	0	9	199	0	11	191	0	9
210	5	17	208	0	9	202	1	11

Πίνακας 5.5 Πειραματικά αποτελέσματα μηχανής MFM για 48 δυνατές επιλογές

Τα συνολικά πειραματικά αποτελέσματα συγκεντρώνονται στον πίνακα 5.6 που φαίνεται παρακάτω. Στην ίδια χρονική διάρκεια 15000 μερίδες χρόνου, στη μηχανή QFM παράγονται 24,5% περισσότερα εισιτήρια από στη μηχανή MFM. Το μέσο λάθος όμως των εισιτηρίων που παράγονται από τη μηχανή QFM εντοπίζεται ότι είναι μεγαλύτερο κατά 66,2% από εκείνο των εισιτηρίων που παράγονται στη μηχανή MFM.

48 επιλογές	Αριθμός εισιτηρίων	Μέσος χρόνος έκδοσης εισιτηρίου	Μέσο λάθος ανά εισιτήριο
QFM	1529	9.83	1.08
MFM	1228	12.23	0.65

Πίνακας 5.6 Συνολικά πειραματικά αποτελέσματα για 48 δυνατές επιλογές

5.4 Σύγκριση QFM & MFM για 80 δυνατές επιλογές

Στον πίνακα 5.7 παρουσιάζεται ένα τμήμα από τα πειραματικά αποτελέσματα των τριών προσομοιώσεων που πραγματοποιήθηκαν στη μηχανή QFM ενώ οι δυνατές επιλογές του χρήστη ήταν 80, δηλαδή 5 προορισμοί x 2 τύποι εισιτηρίου x 2 ηλικίες επιβάτη x 2 κλάσεις εισιτηρίου x 2 τρόποι πληρωμής.

Simulation 1			Simulation 2			Simulation 3		
Χρόνος Εξόδου	Αριθμός Λαθών	Χρόνος Εισιτηρίου	Χρόνος Εξόδου	Αριθμός Λαθών	Χρόνος Εισιτηρίου	Χρόνος Εξόδου	Αριθμός Λαθών	Χρόνος Εισιτηρίου
8	1	8	10	0	10	16	3	16
16	1	8	18	1	8	22	0	6
27	2	11	40	8	22	40	5	18
43	2	16	52	3	12	53	2	13
49	0	6	69	2	17	63	1	10
59	1	10	75	0	6	69	0	6
67	1	8	83	0	8	85	8	16
81	3	14	93	3	10	98	2	13
89	1	8	103	2	10	104	0	6
95	0	6	113	2	10	121	4	17
117	5	22	124	1	11	134	2	13
127	1	10	139	2	15	143	1	9
133	0	6	147	1	8	149	0	6
139	0	6	153	0	6	157	0	8
145	0	6	159	0	6	163	0	6
151	0	6	171	2	12	171	1	8
163	3	12	185	4	14	181	1	10
173	1	10	196	3	11	187	0	6
184	0	11	202	0	6	197	3	10
194	2	10	208	0	6	203	0	6

Πίνακας 5.7 Πειραματικά αποτελέσματα μηχανής QFM για 80 δυνατές επιλογές

Στον πίνακα 5.8 παρουσιάζεται ένα τμήμα από τα πειραματικά αποτελέσματα των τριών προσομοιώσεων που πραγματοποιήθηκαν στη μηχανή MFM ενώ οι δυνατές επιλογές του χρήστη ήταν 80, δηλαδή 5 προορισμοί x 2 τύποι εισιτηρίου x 2 ηλικίες επιβάτη x 2 κλάσεις εισιτηρίου x 2 τρόποι πληρωμής.

Simulation 1			Simulation 2			Simulation 3		
Χρόνος Εξόδου	Αριθμός Λαθών	Χρόνος Εισιτηρίου	Χρόνος Εξόδου	Αριθμός Λαθών	Χρόνος Εισιτηρίου	Χρόνος Εξόδου	Αριθμός Λαθών	Χρόνος Εισιτηρίου
8	0	8	10	0	10	16	1	16
19	0	11	19	0	9	27	0	11
34	2	15	32	1	13	40	1	13
43	0	9	41	0	9	51	2	11
52	0	9	50	0	9	60	0	9
61	0	9	61	0	11	71	1	11
70	0	9	70	0	9	81	1	10
79	0	9	79	0	9	92	0	11
98	2	19	88	0	9	111	2	19
107	0	9	97	0	9	121	1	10
116	0	9	108	1	11	132	0	11
125	0	9	119	1	11	145	0	13
134	0	9	128	0	9	154	0	9
143	0	9	141	0	13	176	5	22
154	1	11	150	0	9	189	1	13
165	1	11	159	0	9	200	0	11
176	1	11	174	1	15	209	0	9
185	0	9	185	0	11	218	1	9
194	0	9	202	5	17	233	2	15
204	1	10	211	0	9	242	0	9

Πίνακας 5.8 Πειραματικά αποτελέσματα μηχανής MFM για 80 δυνατές επιλογές

Τα συνολικά πειραματικά αποτελέσματα συγκεντρώνονται στον πίνακα 5.9 που φαίνεται παρακάτω. Στην ίδια χρονική διάρκεια 15000 μερίδες χρόνου, στη μηχανή QFM παράγονται 11,8% περισσότερα εισιτήρια από στη μηχανή MFM. Το μέσο λάθος, όμως, των εισιτηρίων που παράγονται από τη μηχανή QFM εντοπίζεται ότι είναι μεγαλύτερο κατά 106,2% από εκείνο των εισιτηρίων που παράγονται στη μηχανή MFM.

80 επιλογές	Αριθμός εισιτηρίων	Μέσος χρόνος έκδοσης εισιτηρίου	Μέσο λάθος ανά εισιτήριο
QFM	1369	10.98	1.67
MFM	1224	12.27	0.81

Πίνακας 5.9 Συνολικά πειραματικά αποτελέσματα για 80 δυνατές επιλογές

5.5 Σύγκριση QFM & MFM για 144 δυνατές επιλογές

Στον πίνακα 5.10 παρουσιάζεται ένα τμήμα από τα πειραματικά αποτελέσματα των τριών προσομοιώσεων που πραγματοποιήθηκαν στη μηχανή QFM ενώ οι δυνατές επιλογές του χρήστη ήταν 144, δηλαδή 6 προορισμοί x 2 τύποι εισιτηρίου x 3 ηλικίες επιβάτη x 2 κλάσεις εισιτηρίου x 2 τρόποι πληρωμής.

Simulation 1			Simulation 2			Simulation 3		
Χρόνος Εξόδου	Αριθμός Λαθών	Χρόνος Εισιτηρίου	Χρόνος Εξόδου	Αριθμός Λαθών	Χρόνος Εισιτηρίου	Χρόνος Εξόδου	Αριθμός Λαθών	Χρόνος Εισιτηρίου
5	0	5	11	3	11	11	3	11
21	5	16	21	2	10	51	17	40
29	1	8	29	1	8	57	0	6
52	6	23	35	0	6	63	0	6
60	1	8	58	7	23	81	5	18
69	2	9	64	0	6	91	1	10
89	7	20	70	0	6	100	2	9
103	4	14	82	3	12	135	13	35
132	10	29	92	1	10	161	11	26
138	0	6	100	1	8	169	1	8
144	0	6	108	0	8	226	24	57
162	6	18	123	3	15	248	7	22
170	1	8	135	2	12	292	16	44
200	11	30	141	0	6	302	2	10
212	2	12	162	7	21	314	3	12
224	2	12	177	3	15	338	9	24
232	1	8	183	0	6	344	0	6
244	3	12	194	3	11	350	0	6
253	3	9	200	0	6	360	0	10
283	8	30	208	1	8	367	1	7

Πίνακας 5.10 Πειραματικά αποτελέσματα μηχανής QFM για 144 δυνατές επιλογές

Στον πίνακα 5.11 παρουσιάζεται ένα τμήμα από τα πειραματικά αποτελέσματα των τριών προσομοιώσεων που πραγματοποιήθηκαν στη μηχανή MFM ενώ οι δυνατές επιλογές του χρήστη ήταν 144, δηλαδή 6 προορισμοί x 2 τύποι εισιτηρίου x 3 ηλικίες επιβάτη x 2 κλάσεις εισιτηρίου x 2 τρόποι πληρωμής

Simulation 1			Simulation 2			Simulation 3		
Χρόνος Εξόδου	Αριθμός Λαθών	Χρόνος Εισιτηρίου	Χρόνος Εξόδου	Αριθμός Λαθών	Χρόνος Εισιτηρίου	Χρόνος Εξόδου	Αριθμός Λαθών	Χρόνος Εισιτηρίου
8	0	8	8	0	8	10	0	10
19	1	11	17	0	9	19	0	9
44	1	25	26	0	9	34	1	15
54	1	10	35	0	9	44	1	10
63	0	9	44	0	9	53	0	9
72	0	9	53	0	9	62	1	9
81	0	9	62	0	9	71	0	9
98	2	17	73	0	11	85	2	14
108	1	10	82	0	9	98	0	13
117	0	9	96	1	14	109	0	11
126	0	9	109	2	13	120	1	11
141	0	15	122	0	13	131	1	11
151	1	10	133	0	11	140	0	9
161	1	10	142	0	9	155	1	15
172	0	11	153	0	11	164	0	9
181	0	9	168	0	15	173	0	9
192	0	11	177	0	9	182	0	9
205	1	13	190	1	13	195	2	13
216	1	11	199	0	9	210	0	15
239	1	23	208	0	9	219	0	9

Πίνακας 5.11 Πειραματικά αποτελέσματα μηχανής MFM για 144 δυνατές επιλογές

Τα συνολικά πειραματικά αποτελέσματα συγκεντρώνονται στον πίνακα 5.12 που φαίνεται παρακάτω. Στην ίδια χρονική διάρκεια 15000 μερίδες χρόνου, στη μηχανή QFM παράγονται 18,3% λιγότερα εισιτήρια από στη μηχανή MFM. Το μέσο λάθος των εισιτηρίων που παράγονται από τη μηχανή QFM εντοπίζεται ότι είναι μεγαλύτερο κατά 324,7% από εκείνο των εισιτηρίων που παράγονται στη μηχανή MFM.

144 επιλογές	Αριθμός εισιτηρίων	Μέσος χρόνος έκδοσης εισιτηρίου	Μέσο λάθος ανά εισιτήριο
QFM	991	15.19	3.78
MFM	1213	12.37	0.89

Πίνακας 5.12 Συνολικά πειραματικά αποτελέσματα για 144 δυνατές επιλογές

5.6 Σύγκριση QFM & MFM για 192 δυνατές επιλογές

Στον πίνακα 5.13 παρουσιάζεται ένα τμήμα από τα πειραματικά αποτελέσματα των τριών προσομοιώσεων που πραγματοποιήθηκαν στη μηχανή QFM ενώ οι δυνατές επιλογές του χρήστη ήταν 192, δηλαδή 8 προορισμοί x 2 τύποι εισιτηρίου x 3 ηλικίες επιβάτη x 2 κλάσεις εισιτηρίου x 2 τρόποι πληρωμής.

Simulation 1			Simulation 2			Simulation 3		
Χρόνος Εξόδου	Αριθμός Λαθών	Χρόνος Εισιτηρίου	Χρόνος Εξόδου	Αριθμός Λαθών	Χρόνος Εισιτηρίου	Χρόνος Εξόδου	Αριθμός Λαθών	Χρόνος Εισιτηρίου
5	0	5	9	2	9	7	1	7
21	5	16	15	0	6	17	2	10
100	35	79	24	2	9	43	9	26
146	20	46	52	10	28	57	3	14
154	1	8	74	8	22	90	12	33
194	17	40	90	5	16	107	6	17
216	8	22	100	2	10	124	3	17
230	4	14	126	8	26	131	1	7
260	9	30	138	1	12	151	7	20
278	6	18	157	7	19	159	1	8
298	7	20	167	2	10	165	0	6
322	9	24	180	4	13	181	5	16
330	1	8	188	0	8	197	4	16
340	2	10	247	24	59	217	7	20
350	2	10	253	0	6	229	2	12
366	3	16	261	1	8	243	4	14
376	3	10	309	21	48	267	9	24
416	16	40	327	6	18	287	6	20
432	2	16	333	0	6	295	1	8
456	9	24	343	2	10	315	7	20

Πίνακας 5.13 Πειραματικά αποτελέσματα μηχανής QFM για 192 δυνατές επιλογές

Στον πίνακα 5.14 παρουσιάζεται ένα τμήμα από τα πειραματικά αποτελέσματα των τριών προσομοιώσεων που πραγματοποιήθηκαν στη μηχανή MFM ενώ οι δυνατές επιλογές του χρήστη ήταν 192, δηλαδή 8 προορισμοί x 2 τύποι εισιτηρίου x 3 ηλικίες επιβάτη x 2 κλάσεις εισιτηρίου x 2 τρόποι πληρωμής.

Simulation 1			Simulation 2			Simulation 3		
Χρόνος Εξόδου	Αριθμός Λαθών	Χρόνος Εισιτηρίου	Χρόνος Εξόδου	Αριθμός Λαθών	Χρόνος Εισιτηρίου	Χρόνος Εξόδου	Αριθμός Λαθών	Χρόνος Εισιτηρίου
10	0	10	8	0	8	8	0	8
19	0	9	21	0	13	19	0	11
32	2	13	30	0	9	31	1	12
43	1	11	39	0	9	41	1	10
52	0	9	54	0	15	54	2	13
63	2	11	63	0	9	63	0	9
72	0	9	74	1	11	72	0	9
81	0	9	83	0	9	81	1	9
92	1	11	92	0	9	90	0	9
103	1	11	101	0	9	99	0	9
112	0	9	112	0	11	116	1	17
121	0	9	123	1	11	125	0	9
132	0	11	134	1	11	134	0	9
145	0	13	143	0	9	149	0	15
155	1	10	155	1	12	158	0	9
164	1	9	164	0	9	169	0	11
173	0	9	177	1	13	180	1	11
186	1	13	188	1	11	193	1	13
195	0	9	197	0	9	202	1	9
205	1	10	206	0	9	213	0	11

Πίνακας 5.14 Πειραματικά αποτελέσματα μηχανής MFM για 192 δυνατές επιλογές

Τα συνολικά πειραματικά αποτελέσματα συγκεντρώνονται στον πίνακα 5.15 που φαίνεται παρακάτω. Στην ίδια χρονική διάρκεια 15000 μερίδες χρόνου, στη μηχανή QFM παράγονται 44,3% λιγότερα εισιτήρια από στη μηχανή MFM. Το μέσο λάθος των εισιτηρίων που παράγονται από τη μηχανή QFM εντοπίζεται ότι είναι μεγαλύτερο κατά 698,9% από εκείνο των εισιτηρίων που παράγονται στη μηχανή MFM.

192 επιλογές	Αριθμός εισιτηρίων	Μέσος χρόνος έκδοσης εισιτηρίου	Μέσο λάθος ανά εισιτήριο
QFM	674	22.35	7.27
MFM	1209	12.42	0.91

Πίνακας 5.15 Συνολικά πειραματικά αποτελέσματα για 192 δυνατές επιλογές

5.7 Σύγκριση QFM & MFM για 240 δυνατές επιλογές

Στον πίνακα 5.16 παρουσιάζεται ένα τμήμα από τα πειραματικά αποτελέσματα των τριών προσομοιώσεων που πραγματοποιήθηκαν στη μηχανή QFM ενώ οι δυνατές επιλογές του χρήστη ήταν 240, δηλαδή 10 προορισμοί x 2 τύποι εισιτηρίου x 3 ηλικίες επιβάτη x 2 κλάσεις εισιτηρίου x 2 τρόποι πληρωμής.

Simulation 1			Simulation 2			Simulation 3		
Χρόνος Εξόδου	Αριθμός Λαθών	Χρόνος Εισιτηρίου	Χρόνος Εξόδου	Αριθμός Λαθών	Χρόνος Εισιτηρίου	Χρόνος Εξόδου	Αριθμός Λαθών	Χρόνος Εισιτηρίου
5	0	5	39	17	39	51	23	51
23	6	18	67	9	28	68	4	17
100	34	77	87	7	20	323	122	255
146	20	46	110	9	23	353	12	30
238	42	92	121	3	11	387	13	34
276	16	38	159	13	38	445	26	58
294	6	18	214	23	55	472	9	27
334	17	40	249	15	35	480	0	8
384	22	50	269	7	20	518	16	38
400	7	16	391	58	122	524	0	6
406	0	6	456	28	65	620	45	96
450	19	44	462	0	6	658	16	38
462	3	12	473	3	11	702	19	44
490	11	28	545	33	72	722	7	20
500	2	10	561	5	16	738	5	16
518	7	18	571	2	10	808	32	70
526	0	8	602	12	31	846	16	38
581	23	55	648	20	46	852	0	6
703	58	122	660	3	12	876	9	24
711	1	8	673	2	13	922	19	46

Πίνακας 5.16 Πειραματικά αποτελέσματα μηχανής QFM για 240 δυνατές επιλογές

Στον πίνακα 5.17 παρουσιάζεται ένα τμήμα από τα πειραματικά αποτελέσματα των τριών προσομοιώσεων που πραγματοποιήθηκαν στη μηχανή MFM ενώ οι δυνατές επιλογές του χρήστη ήταν 240, δηλαδή 10 προορισμοί x 2 τύποι εισιτηρίου x 3 ηλικίες επιβάτη x 2 κλάσεις εισιτηρίου x 2 τρόποι πληρωμής.

Simulation 1			Simulation 2			Simulation 3		
Χρόνος Εξόδου	Αριθμός Λαθών	Χρόνος Εισιτηρίου	Χρόνος Εξόδου	Αριθμός Λαθών	Χρόνος Εισιτηρίου	Χρόνος Εξόδου	Αριθμός Λαθών	Χρόνος Εισιτηρίου
10	0	10	8	0	8	14	1	14
19	0	9	19	0	11	23	0	9
32	2	13	30	1	11	33	1	10
43	1	11	43	1	13	44	1	11
52	0	9	52	0	9	53	0	9
63	2	11	68	1	16	70	1	17
72	0	9	81	0	13	85	2	15
81	0	9	91	1	10	94	0	9
92	1	11	102	0	11	105	1	11
103	1	11	113	2	11	131	17	26
112	0	9	124	1	11	146	5	15
121	0	9	133	0	9	161	0	15
132	0	11	142	0	9	170	0	9
145	0	13	152	1	10	183	0	13
155	1	10	161	0	9	192	0	9
164	1	9	174	2	13	201	0	9
173	0	9	183	0	9	216	1	15
186	1	13	192	0	9	225	0	9
195	0	9	207	1	15	234	1	9
205	1	10	216	0	9	245	0	11

Πίνακας 5.17 Πειραματικά αποτελέσματα μηχανής MFM για 240 δυνατές επιλογές

Τα συνολικά πειραματικά αποτελέσματα συγκεντρώνονται στον πίνακα 5.18 που φαίνεται παρακάτω. Στην ίδια χρονική διάρκεια 15000 μερίδες χρόνου, στη μηχανή QFM παράγονται 69,9% λιγότερα εισιτήρια από στη μηχανή MFM. Το μέσο λάθος των εισιτηρίων που παράγονται από τη μηχανή QFM εντοπίζεται ότι είναι μεγαλύτερο κατά 1777,2% από εκείνο των εισιτηρίων που παράγονται στη μηχανή MFM.

240 επιλογές	Αριθμός εισιτηρίων	Μέσος χρόνος έκδοσης εισιτηρίου	Μέσο λάθος ανά εισιτήριο
QFM	360	41.99	17.27
MFM	1196	12.58	0.92

Πίνακας 5.18 Συνολικά πειραματικά αποτελέσματα για 240 δυνατές επιλογές

6 Συμπεράσματα & Μελλοντική Έρευνα

6.1 Συμπεράσματα & τελική αξιολόγηση

Από την μελέτη των πειραματικών αποτελεσμάτων προκύπτουν τα δεδομένα που θα χρησιμοποιηθούν για την τελική αξιολόγηση των δύο τύπων μηχανών αυτόματης πώλησης εισιτηρίων. Στους πίνακες 6.1 και 6.2 παρουσιάζονται συγκεντρωτικά τα αποτελέσματα για κάθε τύπο μηχανής.

QFM	Αριθμός εισιτηρίων	Μέσος χρόνος έκδοσης εισιτηρίου	Μέσο λάθος ανά εισιτήριο
32 επιλογές	1567	9.59	0.95
48 επιλογές	1529	9.83	1.08
80 επιλογές	1369	10.98	1.67
144 επιλογές	991	15.19	3.78
192 επιλογές	674	22.35	7.27
240 επιλογές	360	41.99	17.27

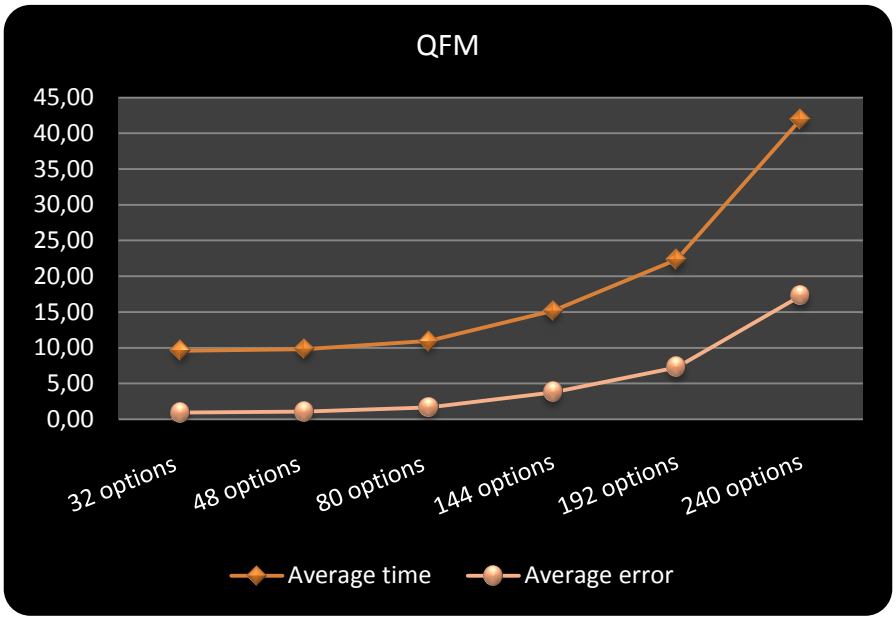
Πίνακας 6.1 Συγκεντρωτικά αποτελέσματα για την μηχανή QFM

MFM	Αριθμός εισιτηρίων	Μέσος χρόνος έκδοσης εισιτηρίου	Μέσο λάθος ανά εισιτήριο
32 επιλογές	1240	12.11	0.70
48 επιλογές	1228	12.23	0.65
80 επιλογές	1224	12.27	0.81
144 επιλογές	1213	12.37	0.89
192 επιλογές	1209	12.42	0.91
240 επιλογές	1196	12.58	0.92

Πίνακας 6.2 Συγκεντρωτικά αποτελέσματα για την μηχανή MFM

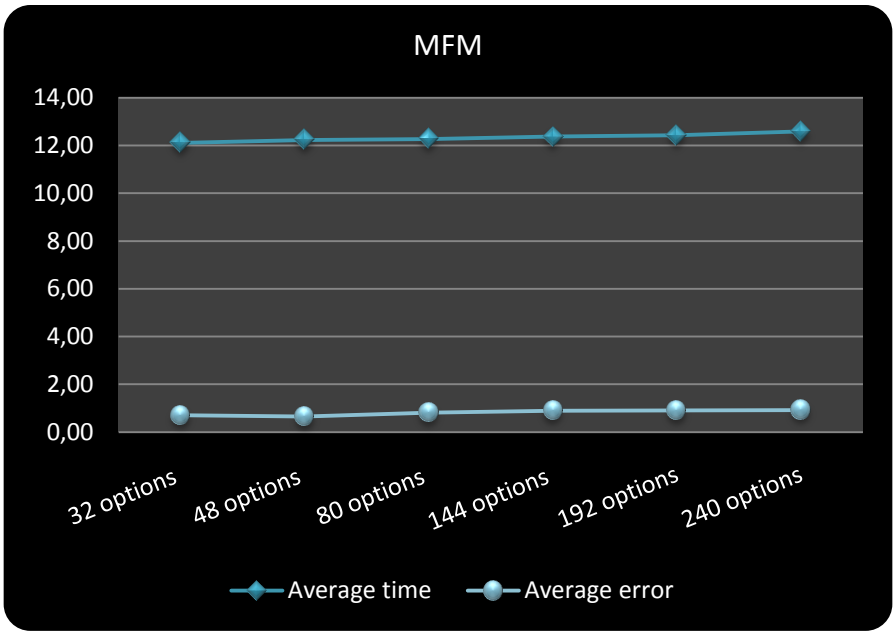
Προκειμένου να γίνουν πιο ευδιάκριτα αλλά και κατανοητά όλα τα παραπάνω συγκεντρωτικά στοιχεία, απεικονίζονται και με τη βοήθεια γραφικών παραστάσεων. Στη γραφική παράσταση της εικόνας 6.1 απεικονίζεται, ο ρυθμός αύξησης του μέσου χρόνου έκδοσης ενός εισιτηρίου, αλλά και ο ρυθμός αύξησης του μέσου λάθους ανά εισιτήριο, σε μια μηχανή QFM ανάλογα με την αύξηση του αριθμού των επιλογών που είναι διαθέσιμες

στο χρήστη. Είναι εμφανές ότι η κλίση και των δύο ρυθμών απόδοσης αυξάνεται, όταν οι διαθέσιμες επιλογές του χρήστη είναι περισσότερες από 100.



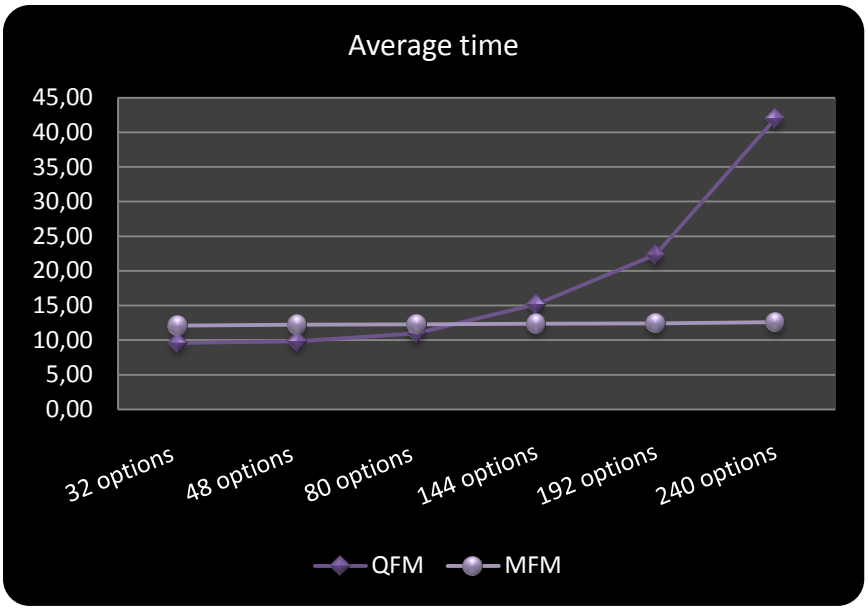
Εικόνα 6.1 Συγκεντρωτικά αποτελέσματα για τη μηχανή QFM

Στη γραφική παράσταση της εικόνας 6.2 απεικονίζεται ο ρυθμός αύξησης του μέσου χρόνου έκδοσης ενός εισιτηρίου, αλλά και ο ρυθμός αύξησης του μέσου λάθους ανά εισιτήριο, σε μια μηχανή MFM ανάλογα με την αύξηση του αριθμού των επιλογών που είναι διαθέσιμες στο χρήστη. Σε αντίθεση με την προηγούμενη γραφική παράσταση (γραφική παράσταση 6.1), είναι φανερό ότι στη συγκεκριμένη γραφική παράσταση, η κλίση και των δύο ρυθμών απόδοσης αυξάνεται κατά ένα πολύ μικρό ποσοστό, παραμένοντας σχεδόν σταθερή, όσο και αν πληθαίνουν οι διαθέσιμες επιλογές του χρήστη.



Εικόνα 6.2 Συγκεντρωτικά αποτελέσματα για τη μηχανή MFM

Στη γραφική παράσταση της εικόνας 6.3 απεικονίζονται, στο ίδιο διάγραμμα, ο ρυθμός αύξησης του μέσου χρόνου έκδοσης ενός εισιτηρίου στις δύο μηχανές, καθώς αυξάνονται οι διαθέσιμες επιλογές του χρήστη. Συγκρίνοντας το μέσο χρόνο που απαιτείται για την έκδοση ενός εισιτηρίου στις μηχανές QFM και MFM, καταλήγουμε στο συμπέρασμα ότι για λιγότερες από 100 διαθέσιμες επιλογές είναι προτιμότερη η χρήση της μηχανής QFM, ενώ για περισσότερες επιλογές, καταλληλότερη κρίνεται η χρήση της μηχανής MFM.



Εικόνα 6.3 Συγκεντρωτικά αποτελέσματα μέσου χρόνου έκδοσης εισιτηρίου

Στη γραφική παράσταση της εικόνας 6.4 απεικονίζονται, στο ίδιο διάγραμμα, ο ρυθμός αύξησης του μέσου λάθους ανά εισιτήριου στις δύο μηχανές, καθώς αυξάνονται οι διαθέσιμες επιλογές του χρήστη.

Εικόνα 6.4 Συγκεντρωτικά αποτελέσματα μέσου λάθους ανά εισιτήριο

Συγκρίνοντας το μέσο λάθος στις μηχανές QFM και MFM, καταλήγουμε στο συμπέρασμα ότι είναι προτιμότερη η χρήση της μηχανής MFM, αφού το μέσο λάθος διατηρείται σχεδόν σταθερό, ανεξάρτητα από την αύξηση του πλήθους των διαθέσιμων επιλογών του χρήστη.

Συμπερασματικά, στην παρούσα διατριβή μοντελοποιήθηκαν οι λειτουργίες ενός μηχανήματος αυτόματης πώλησης εισιτηρίων. Μετά από μελέτη ποικίλων συνεντεύξεων με χρήστες, αφού καταγράφηκαν τα πιθανά ανθρώπινα λάθη, καθώς και οι ανεπιθύμητες καταστάσεις του συστήματος, επινοήθηκαν τρόποι πρόληψης και αντιμετώπισης αυτών. Επιπρόσθετα, δόθηκε έμφαση στα σχεδιαστικά χαρακτηριστικά της μηχανής που αποτρέπουν, είτε μειώνουν δυνητικά τα λάθη. Συγκεκριμένα, προσομοιώθηκαν δύο τύποι μηχανών αυτόματης πώλησης εισιτηρίων, QFM και MFM. Στον πρώτο τύπο, ο χρήστης για να κατορθώσει το στόχο του, να παραλάβει δηλαδή το επιθυμητό εισιτήριο, αρκεί η επιλογή δυο μόνο κουμπιών, ενώ στον δεύτερο τύπο για την επίτευξη του στόχου του, πρέπει να πατήσει πολλαπλά κουμπιά.

Μελετώντας και αναλύοντας όλα τα παραπάνω πειραματικά αποτελέσματα, καταφέρνουμε να αξιολογήσουμε τις δύο μηχανές. Στην περίπτωση όπου οι διαθέσιμες επιλογές του χρήστη είναι οι ελάχιστες δυνατές, δηλαδή 32, συμπεραίνουμε ότι οι δύο μηχανές είναι εξίσου αξιόπιστες. Αυτό συμβαίνει γιατί, η μηχανή QFM υπερτερεί έναντι της μηχανής MFM όσο αφορά στο χρόνο έκδοσης του κάθε εισιτηρίου, αφού στο ίδιο χρονικό διάστημα μπορεί να παράγει 26,4% περισσότερα εισιτήρια από τη μηχανή MFM, αλλά και η μηχανή MFM υπερέχει έναντι της QFM, αφού το μέσο λάθος των εισιτηρίων που παράγονται από τη μηχανή QFM εντοπίζεται ότι είναι μεγαλύτερο κατά 35,7% από εκείνο των εισιτηρίων που παράγονται στη μηχανή MFM. Για τον παραπάνω λόγο, οι δύο μηχανές παραμένουν εξίσου αξιόπιστες, καθώς οι διαθέσιμες επιλογές του χρήστη αυξάνονται περίπου έως τις 80. Στην περίπτωση, όμως, που οι επιλογές του χρήστη αυξηθούν πάνω από το όριο των 80, τα παραπάνω δεδομένα ανατρέπονται. Ο μέσος χρόνος παραγωγής ενός εισιτηρίου στη μηχανή QFM αρχίζει να αυξάνεται δραματικά, λόγω του μεγάλου αριθμού ανθρώπινων λαθών που λαμβάνουν χώρα. Η επιλογή ενός

κουμπιού, ανάμεσα στα 120 (120 επιλογές για τα χαρακτηριστικά του εισιτηρίου x 2 τρόποι πληρωμής = 240 επιλογές), αποτελεί ένα πραγματικά δύσκολο εγχείρημα για το χρήστη. Αποδεικνύεται λοιπόν, ότι για μεγάλο αριθμό διαθέσιμων επιλογών, η χρήση της μηχανής QFM καθίσταται απαγορευτική.

6.2 Μελλοντικές κατευθύνσεις έρευνας

Αναζητώντας μελλοντικές επεκτάσεις για το παρόν σύστημα, θα μπορούσε να δημιουργηθεί μια διεπαφή αλληλεπίδρασης με τους χρήστες *(graphical user interface),* έτσι ώστε οι δύο μηχανές αυτόματης πώλησης εισιτηρίων να δοκιμαστούν σε πραγματικές συνθήκες λειτουργίας και η αξιολόγηση αυτών να γίνει βάση της συλλογής πραγματικών δεδομένων. Επιπρόσθετα, θα ήταν δυνατή η κατηγοριοποίηση των χρηστών σε συγκεκριμένες ομάδες και η σύνδεση κάθε ομάδας με μια διαφορετική συνάρτηση πιθανότητας παραγωγής ανθρώπινου λάθους. Με τον τρόπο αυτό, ομάδες χρηστών όπως παιδιά, τουρίστες ή μεθυσμένοι θα είχαν μεγαλύτερη πιθανότητα να εκτελέσουν κάποια λανθασμένη επιλογή, από ομάδες χρηστών όπως οι εργαζόμενοι που καθημερινά κάνουν το ίδιο δρομολόγιο. Κατά συνέπεια, αν το ποσοστό των καθημερινών χρηστών ήταν μεγαλύτερο, μια μηχανή τύπου QFM θα ήταν περισσότερο λειτουργική, αφού η έκδοση του επιθυμητού εισιτηρίου θα ήταν ταχύτερη, ενώ μια μηχανή τύπου MFM θα διευκόλυνε τις ομάδες χρηστών που δεν έχουν μεγάλη πρότερη εμπειρία, μειώνοντας την πιθανότητα ανθρώπινου λάθους.

Βιβλιογραφία

Βιβλία:

- Proth J.M. and Xie X., 'Petri Nets: A tool for design and management of manufacturing systems', Wiley, Chichester UK 1996
- Ρουμελιώτης Εμ., 'Τόμος Ά, Μοντελοποίηση και Προσομοίωση', Πάτρα 2001

Άρθρα:

- Aalst W.M.P., 'The Application of Petri Nets to Workflow Management' The Journal of Circuits - Systems and Computers, 8, (1998), 21-66
- Connell W.I., Blandford A. and Green T. 'CASSM and cognitive walkthrough: usability issues with ticket vending machines', Behaviour and Information Technology, 23, (2004), 307-320.
- Connell W.I., 'Error analysis of ticket vending machines: comparing analytic and empirical data', Ergonomics, 41, (1998), 927-961.
- Murata T., 'Petri Nets: Properties, Analysis and Applications' IEEE, 77, (1989), 541-580.
- Salimifard K., 'Petri net-based modeling of workflow systems: An overview' European Journal of Operations Research, 134, (2001), 664–676
- Stanton N.A. and Baber C., 'Human error identification techniques applied to public technology: predictions compared with observed use' Ergonomics, 27, (1996), 119-131.
- Verhoef L.W.M., 'Decision making of vending machine users' Ergonomics, 19, (1988), 103-109.

Δικτυακοί τόποι:

- http://en.wikipedia.org/wiki/IDEF
- http://en.wikipedia.org/wiki/Kumaraswamy_distribution

- http://en.wikipedia.org/wiki/Petri_net
- http://en.wikipedia.org/wiki/SADT
- http://en.wikipedia.org/wiki/Workflow
- http://wiki.daimi.au.dk/cpntools-help/introduction_to_performan.wiki?cmd=get&anchor=Introduction+to+Performance+Analysis

Παράρτημα Α

Πίνακας 1

Ο πίνακας αυτός διαθέτει τρεις στήλες. Στη πρώτη στήλη, φαίνονται τα άρθρα στα οποία συναντήσαμε το ανθρώπινο λάθος, στη δεύτερη στήλη φαίνεται το λάθος στο οποίο αναφερόμαστε και στη τρίτη στήλη, περιγράφεται περιληπτικά ο τρόπος πρόληψης ή αντιμετώπισης του προαναφερθέν λάθους.

Πίνακας Πρόληψης - Αντιμετώπισης Ανθρώπινων Λαθών		
Πηγή	Ανθρώπινο Λάθος	Πρόληψη - Αντιμετώπιση
• Stanton, N.A. and Baber, C. 1996, 'Human error identification techniques applied to public technology: predictions compared with observed use' **• Verhoef, L.W.M 1988 'Decision making of vending machine users'** **• Connell, W. I. 1998 'Error analysis of ticket vending machines: comparing analytic and empirical data'** **• Connell, W. I., Blandford, A. and Green, T. 2004 CASSM and cognitive walkthrough: usability issues with ticket vending machines**	Error in station selection (destination)	Το εισιτήριο μετά την επιλογή του σταθμού περνάει από την κατάσταση 'start check' που οδηγεί στη μετάβαση 'error'. Στην μετάβαση αυτή γίνεται χρήση του κώδικα: input (tick); output (z); action (if #er tick =0 then check(intToReal(#ep tick)) else check(intToReal(#ep tick - 1))); Ο κώδικας αυτός ελέγχει αρχικά αν ο χρήστης έχει ξανακάνει λάθος στην επιλογή του σταθμού. Στην περίπτωση που ο χρήστης δεν έχει κάνει λάθος πρωτύτερα, καλείται η συνάρτηση check() με όρισμα το πλήθος των επιλογών των σταθμών (#ep tick), ενώ αν είχε κάνει λάθος, καλείται η συνάρτηση check() με όρισμα το πλήθος των επιλογών των σταθμών μείον 1 (#ep tick -1). Είναι ένα είδος μνήμης όπου η μηχανή θυμάται ότι ο χρήστης έκανε λάθος και μειώνει την πιθανότητα επόμενου λάθους. Αν παραχθεί λάθος, το

		εισιτήριο επιστρέφει ένα βήμα πίσω, δηλαδή στην κατάσταση αναμονής επιλογής του σταθμού από το χρήστη, ενώ αν δεν παραχθεί η πορεία του εισιτηρίου συνεχίζει την κατάσταση όπου ο χρήστης πρέπει να επιλέξει τον τύπο του εισιτηρίου που επιθυμεί (επόμενο βήμα).
		
		fun check(y)= if uniform(0.0,1.0)>prob(y) then 0 else 1;
		
		Η συνάρτηση check(y) είναι μια συνάρτηση *τυχαίας παραγωγής λάθους*. Καλεί δύο συναρτήσεις: την uniform() και την prob().
		Η uniform(0.0,1.0) παράγει ένα τυχαίο νούμερο από το 0.0 έως το 1.0 το οποίο συγκρίνεται με την έξοδο από τη συνάρτηση prob() η οποία επιστρέφει επίσης ένα νούμερο από το 0.0 έως το 1.0. Αν το νούμερο που παράγεται από την uniform() είναι μεγαλύτερο από εκείνο που παράγεται από την prob() τότε δεν παράγεται λάθος, ενώ στην αντίστροφη περίπτωση παράγεται.
		
		fun prob(y)= 1.0 - (1.0 - y / 200.0)*(1.0 - y / 200.0)*(1.0 - y / 200.0);
		
		Η συνάρτηση prob() είναι μια σιγμοειδής συνάρτηση κατανομής της πιθανότητας λάθους η οποία εξαρτάται από το πλήθος των επιλογών που έχει ο χρήστης. Όσες περισσότερες είναι οι επιλογές τόσο μεγαλύτερη είναι και η πιθανότητα λάθους.

• Stanton, N.A. and Baber, C. 1996, 'Human error identification techniques applied to public technology: predictions compared with	Error in ticket type selection (single, return)	Γίνεται και πάλι χρήση της συνάρτησης check(). Αν παραχθεί λάθος, το εισιτήριο επιστρέφει στην κατάσταση όπου έχει επιλεγεί ο σταθμός

observed use' **• Verhoef, L.W.M 1988 'Decision making of vending machine users'**		και αναμένεται από το χρήστη και επιλέξει και τον τύπο του εισιτηρίου που επιθυμεί.
• Stanton, N.A. and Baber, C. 1996, 'Human error identification techniques applied to public technology: predictions compared with observed use'	Error in passenger age selection (adult, child, student)	Γίνεται και πάλι χρήση της συνάρτησης check(). Αν παραχθεί λάθος, το εισιτήριο επιστρέφει στην κατάσταση όπου έχει επιλεγεί ο τύπος του εισιτηρίου και αναμένεται από το χρήστη και επιλέξει και την ηλικία στην οποία ανταποκρίνεται το εισιτήριο που επιθυμεί.
• Stanton, N.A. and Baber, C. 1996, 'Human error identification techniques applied to public technology: predictions compared with observed use' **• Connell, W. I. 1998 'Error analysis of ticket vending machines: comparing analytic and empirical data'**	Error in pay way selection (money, card)	Υπάρχουν δύο επιλογές στον τρόπο με τον οποίο ο χρήστης μπορεί να πληρώσει το εισιτήριο του. Είτε με μετρητά είτε με κάρτα. Και στην επιλογή αυτή γίνεται χρήση της συνάρτησης check(). Αν παραχθεί λάθος, το εισιτήριο επιστρέφει στην κατάσταση όπου έχει γίνει η τελική επιβεβαίωση από το χρήστη και αναμένεται να επιλέξει εκ νέου τον τρόπο πληρωμής του εισιτηρίου.
• Stanton, N.A. and Baber, C. 1996, 'Human error identification techniques applied to public technology: predictions compared with observed use'	Error in money insertion (not enough money)	Αν ο χρήστης επιλέξει να πληρώσει με μετρητά χρησιμοποιείται ο εξής κώδικας: if (mn>= #pr tick) then 1` (mn- #pr tick,tick) else nil Ο κώδικας αυτός ελέγχει αν τα χρήματα που εισήγαγε ο χρήστης στη μηχανή επαρκούν για την αποπληρωμή του εισιτηρίου. Αν τα χρήματα δεν επαρκούν, επιστρέφονται πίσω στο χρήστη και η μηχανή περιμένει από το χρήστη να εισάγει νέο ποσό που να καλύπτει την αξία του εισιτηρίου που έχει επιλέξει.
• Verhoef, L.W.M 1988 'Decision making of vending machine users'	Error with the price (not expected so expensive ticket)	Αφού ο χρήστης επιλέξει το σταθμό, τον τύπο, την ηλικία, την κλάση και τον αριθμό των εισιτηρίων που επιθυμεί,

		υπολογίζεται η τιμή του εισιτηρίου. Όταν εμφανίζεται η τιμή του εισιτηρίου στο χρήστη, μπορεί να είναι μεγαλύτερη από την αναμενόμενη, πιθανότατα γιατί ο χρήστης έκανε στα προηγούμενα βήματα κάποιο λάθος που δεν συνειδητοποίησε. Η μηχανή του δίνει τη δυνατότητα να ακυρώσει το εισιτήριο μέσω της μετάβασης 'final confirm' και να επιστρέψει στην αρχική κατάσταση αναμονής της μηχανής για να κάνει εξ' αρχής όλες τις επιλογές που επιθυμεί. Στη μετάβαση 'final confirm' χρησιμοποιείται και πάλι η συνάρτηση check() όπου παράγει την πιθανότητα λάθους.
• Verhoef, L.W.M 1988 'Decision making of vending machine users'	Error in ticket class (a class, economy)	Γίνεται και πάλι χρήση της συνάρτησης check(). Αν παραχθεί λάθος, το εισιτήριο επιστρέφει στην κατάσταση όπου έχει επιλεγεί ο ηλικία στην οποία ανταποκρίνεται το εισιτήριο και αναμένεται από το χρήστη και επιλέξει και την κλάση του εισιτηρίου που επιθυμεί.
• Connell, W. I. 1998 'Error analysis of ticket vending machines: comparing analytic and empirical data'	More than ticket price inserted when no change given	Στη μηχανή που έχουμε κατασκευάσει υπάρχει η δυνατότητα επιστροφής του υπόλοιπου ποσού (ρέστα) από την αγορά του επιθυμητού εισιτηρίου.
• Connell, W. I., Blandford, A. and Green, T. 2004 CASSM and cognitive walkthrough: usability issues with ticket vending machines	Repeated transactions for multiple ticket (number of tickets)	Ένα βήμα πριν τον τελικό υπολογισμό της τιμής του εισιτηρίου υπάρχει η κατάσταση 'number of tickets' η οποία καθορίζει τον αριθμό τον εισιτηρίων που επιθυμεί ο χρήστης.
• Connell, W. I., Blandford, A. and Green, T. 2004 CASSM and cognitive walkthrough: usability issues with ticket vending machines	Users forgetting take their change at the end of the transaction	Στη μηχανή που έχουμε κατασκευάσει πρώτα επιστρέφονται τα ρέστα και στη συνέχεια τυπώνεται το εισιτήριο του χρήστη.
• Connell, W. I., Blandford, A. and Green, T. 2004 CASSM and	Cards acceptance status Cards of denominations	Στην περίπτωση που ο χρήστης επιλέξει την κάρτα ως τρόπο

cognitive walkthrough: usability issues with ticket vending machines		πληρωμής γίνονται οι έλεγχοι που ακολουθούν. Αρχικά η μηχανή διαβάζει την κάρτα και περιμένει από το χρήστη την εισαγωγή του προσωπικού κωδικού (pin) της κάρτας. Με τη χρήση της συνάρτησης check() παρέχεται η πιθανότητα, ο χρήστης να έβαλε λάθος κωδικό. Τότε η μηχανή επιστρέφει στην κατάσταση αναμονής εισαγωγής νέου προσωπικού κωδικού. Αφού ο χρήστης εισάγει το σωστό κωδικό, η μηχανή ελέγχει αν το υπόλοιπο της κάρτας επαρκεί για την αναπλήρωση του εισιτηρίου. Αν επαρκεί τα χρήματα αφαιρούνται από την κάρτα και το εισιτήριο τυπώνεται. Στην περίπτωση που το υπόλοιπο της κάρτας δεν επαρκεί η κάρτα επιστρέφεται και η μηχανή μεταβαίνει στην κατάσταση αναμονής επιλογής του τρόπου πληρωμής. Ο χρήστης έχει τη δυνατότητα να επιλέξει να χρησιμοποιήσει μια νέα κάρτα ή να πληρώσει το εισιτήριο του με μετρητά.