

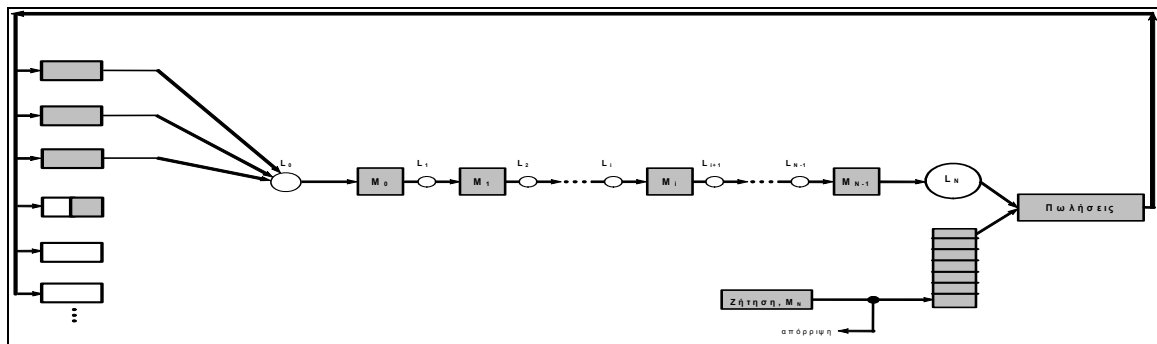


ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ &
ΔΙΟΙΚΗΣΗΣ

Πολιτικές προμηθειών, παραγωγής και πωλήσεων σε
γραμμές CONWIP

Κουδιγκέλης-Δαλάκης Κωνσταντίνος

Διπλωματική Εργασία



Τριμελής εξεταστική επιτροπή:
Κουϊκόγλου Βασίλειος (επιβλέπων)
Νικολός Ιωάννης
Βλάσσης Νικόλαος

Χανιά, 2008

Copyright 2008 υπό Κουδιγκέλη-Δαλάκη Κωνσταντίνο

Πρόλογος

Στο σημείο αυτό νιώθω την ανάγκη να ευχαριστήσω κάποιους ανθρώπους που με στήριξαν στην πορεία μου ως εδώ.

Πρώτα από όλα οφείλω ένα μεγάλο ευχαριστώ στους γονείς μου, Νίκο και Χαρούλα, και στην αδερφή μου, Ελευθερία, για την αγάπη, την υποστήριξη και την δύναμη που μου δίνουν να πετύχαινω τους στόχους μου. Ακόμη, θα ήθελα να ευχαριστήσω τον επιβλέποντα της διπλωματικής, Καθηγητή Βασίλειο Κουικόγλου τόσο για την εμπιστοσύνη που μου έδειξε, όσο και για την πολύτιμη βοήθεια, και καθοδήγηση κατά τη διάρκεια της εκπόνησης της διπλωματικής. Νιώθω ευγνώμων που συνεργάστηκα μαζί του και πραγματικά του χρωστάω πολλά.

Περιεχόμενα

1. Εισαγωγή και Βιβλιογραφική Ανασκόπηση.....	1
1.1 Συστήματα ώθησης (push systems)	2
1.2 Συστήματα έλξης (pull systems)	3
1.3 Σύγκριση μοντέλων ώθησης και έλξης.....	4
1.4 Πολιτική CONWIP	5
1.5 Σύγκριση συστήματος πολιτικής CONWIP με συστήματα πολιτικής Kanban ..	6
1.6 Σύγκριση συστήματος πολιτικής CONWIP με συστήματα ώθησης.....	6
1.7 Πολιτικές Παραγγελιών – Έλεγχος και Διαχείριση Αποθεμάτων	7
1.8 Συστήματα Συνεχούς Επιθεώρησης.....	8
1.8.1 Σταθερή ποσότητα παραγγελίας (s, Q)	8
1.8.2 Μεταβλητή ποσότητα παραγγελίας (s, S)	9
1.9 Σύστημα Περιοδικής Επιθεώρησης	9
1.9.1 Περιοδική επιθεώρηση (R, S)	9
1.9.2 Περιοδική επιθεώρηση (R, s, S).....	10
1.10 Πολιτικές Αποδοχής ή Απόρριψης Παραγγελιών.....	10
1.11 Σύνοψη της Διπλωματικής Εργασίας.....	12
2. Περιγραφή του Συστήματος	13
2.1 Γραμμή Παραγωγής	13
2.2 Ζήτηση και Πώληση	13
2.3 Παραγγελίες πρώτων υλών	14
2.4 Λίστα αναμονής πελατών	15
2.5 Λειτουργικές Παράμετροι.....	17
2.6 Παράμετροι Κόστους	18
3. Περιγραφή Αλγορίθμου	20
3.1 Αλγόριθμος Προσομοίωσης Διακριτών Γεγονότων	20
3.1.1 Βασική Δομή του Αλγορίθμου	20
3.1.2 Αφιξη ζήτησης	21
3.1.3 Παραγωγή ενός κομματιού από την M_{N-1}	21
3.1.4 Αφιξη μιας παραγγελίας	22
3.1.5 Παραγωγή ενός κομματιού από μια ενδιάμεση μηχανή	22
3.2 Αλγόριθμος εύρεσης της βέλτιστης πολιτικής.....	24
3.2.1 Συνάρτηση κέρδους	24
3.2.2 Στατιστικά συνάρτησης κέρδους	26
4. Αριθμητικά Αποτελέσματα	27
4.1 Σύστημα M/M/1/S+C.....	27
4.1.1 Παρατηρήσεις-Συμπεράσματα.....	31
4.2 Γραμμή παραγωγής με σύστημα CONWIP	32
4.2.1 Παρατηρήσεις-Συμπεράσματα.....	36
5. Συμπεράσματα και Επεκτάσεις	38
Βιβλιογραφία	39
Παράρτημα: Κώδικας του αλγορίθμου	41

1. Εισαγωγή και Βιβλιογραφική Ανασκόπηση

Το περιβάλλον των επιχειρήσεων λόγω του παγκόσμιου ανταγωνισμού αλλάζει ραγδαία. Η αλλαγή αυτή είναι ορατή τόσο στις πωλήσεις όσο και στις τεχνολογίες μεταποίησης. Όσον αφορά τις πωλήσεις παρατηρούνται μικρότεροι κύκλοι ζωής προϊόντων, αλλαγές στα κανάλια των πωλήσεων (με σημαντικότερη την τάση πολλών επιχειρήσεων να εντάξουν το Διαδίκτυο στα κανάλια πωλήσεων που διαθέτουν) και τέλος αλλαγές στα χαρακτηριστικά του σύγχρονου καταναλωτή. Όσον αφορά τις αλλαγές στις τεχνολογίες μεταποίησης, αυτές γίνονται ορατές μέσα από τα ευέλικτα συστήματα παραγωγής, την ολοένα αυξανόμενη τάση αυτοματοποίησης πολλών εργασιών στο χώρο της επιχείρησης καθώς και από τις νέες και καινοτόμες ψηφιακές τεχνολογίες που έχουν εισέλθει στο χώρο της εφοδιαστικής και έχουν επιφέρει θεαματικές αλλαγές. Από τα παραπάνω σε συνδυασμό με τις εξελίξεις στις επιστημονικές περιοχές της πληροφορικής, της διοίκησης επιχειρήσεων, των συστημάτων παραγωγής, της επιχειρησιακής έρευνας και της οικονομίας και την αξιοποίηση αυτών προς όφελος των επιχειρήσεων, γίνεται κατανοητό ότι οι αλλαγές αυτές επηρεάζουν σημαντικά και την οργάνωση των διαφόρων συστημάτων παραγωγής.

Στην παρούσα εργασία εξετάζεται το σύνθετο πρόβλημα της εύρεσης βέλτιστων πολιτικών παραγωγής, τροφοδοσίας με πρώτες ύλες και διαχείρισης πελατών ενός συστήματος παραγωγής με σειριακή διασύνδεση μηχανών (γραμμή παραγωγής). Υιοθετούνται συγκεκριμένες οικογένειες πολιτικών και από αυτές, με τη βοήθεια ενός αλγορίθμου προσομοίωσης που αναπτύσσεται, προσδιορίζονται εκείνες που μεγιστοποιούν τον μέσο ρυθμό κερδοφορίας του συστήματος.

Βασικός στόχος των συστημάτων παραγωγής είναι να βελτιώσουν την παραγωγική διαδικασία ούτως ώστε αυτή να γίνει πιο αποτελεσματική και κερδοφόρος. Σύμφωνα με τους Spearman, Woodruff and Hopp [1] ένα αποτελεσματικό σύστημα παραγωγής είναι αυτό που παράγει ποιοτικά κομμάτια, την κατάλληλη χρονική στιγμή και σε ανταγωνιστικό κόστος. Στην διεθνή βιβλιογραφία επικρατούν δύο διαφορετικές απόψεις για τον τρόπο με τον οποίο μια βιομηχανία μπορεί να επιτύχει τους τρεις παραπάνω στόχους.

1.1 Συστήματα ώθησης (push systems)

Από την μία πλευρά υπάρχουν αυτοί που πιστεύουν ότι η περαιτέρω ανάπτυξη και βελτίωση των συστημάτων παραγωγής θα επέλθει μέσω της υιοθέτησης τεχνικών και μεθοδολογιών από τον τομέα της ενοποιημένης παραγωγής με την βοήθεια υπολογιστή (Computer Integrated Manufacturing, CIM). Το CIM είναι μια πολιτική παραγωγής στην οποία ολόκληρη η διαδικασία παραγωγής ελέγχεται από υπολογιστή. Ο Προγραμματισμός Απαιτήσεων Υλικών (Material Requirements Planning, MRP) αποτελεί μια από τις βασικές συνιστώσες του CIM. Το MRP ανήκει στην κατηγορία των μοντέλων ώθησης, στα οποία οι διάφορες εργασίες προγραμματίζονται με βάση τη σχέση που υπάρχει ανάμεσα στην προβλεπόμενη ζήτηση του τελικού προϊόντος και των επιμέρους υλικών από τα οποία αυτό αποτελείται. Είναι μια τεχνική υπολογισμού των ποσοτήτων των απαιτούμενων συστατικών υλικών έτσι ώστε να παραχθεί το τελικό προϊόν στον προγραμματισμένο χρόνο. Μέσα σε ένα σύστημα πολιτικής MRP διατηρούνται σε πραγματικό χρόνο στοιχεία του διαθέσιμου αποθέματος και παράγοντες σχεδιασμού, όπως είναι οι χρόνοι παράδοσης και παραγωγής και το απόθεμα ασφαλείας. Το σύστημα πολιτικής MRP παίρνει ένα συνολικό πλάνο παραγωγής και το μεταφράζει, μέσω δενδροειδών δομών (κατάλογος υλικών, bill of materials), σε επιμέρους υλικά που απαιτούνται, υπολογίζοντας τις ποσότητες και τη χρονική στιγμή κατά την οποία αυτά θα είναι απαραίτητα.

Μια από τις βασικές παραδοχές της πολιτικής MRP είναι ότι η διαθέσιμη δυναμικότητα του συστήματος είναι απεριόριστη. Με άλλα λόγια η πολιτική MRP μπορεί να προσδιορίσει τι "πρέπει" να παραχθεί, όχι όμως και τι "μπορεί" να παραχθεί. Η πολιτική MRPII λύνει αυτό το πρόβλημα, γιατί ουσιαστικά αποτελείται από δύο συστήματα: ένα σύστημα MRP και ένα σύστημα προγραμματισμού απαιτήσεων πόρων (Capacity Requirements Planning, CRP). Από τη στιγμή που έχει καθοριστεί από το MRP τί θα παραχθεί, το CRP χρησιμοποιεί στοιχεία σχετικά με τον διαθέσιμο παραγωγικό εξοπλισμό για να υπολογίσει τις απαιτήσεις δυναμικότητας παραγωγής. Σήμερα, τα συστήματα ERP (Enterprise Resource Planning) έρχονται να δώσουν ένα συνολικό σχεδιασμό των πόρων της επιχείρησης, περιλαμβάνοντας και το σχεδιασμό των οικονομικών πόρων.

Ορισμένα από τα βασικά πλεονεκτήματα των συστημάτων που εφαρμόζουν πολιτικές MRP και MRPII είναι ότι πετυχαίνουν μείωση των επιπέδων αποθέματος και, σε πολλές περιπτώσεις, βελτιώνουν τα επίπεδα έγκαιρης εξυπηρέτησης πελατών (customer service) σε ένα σύστημα παραγωγής. Από την άλλη μεριά ορισμένα από τα βασικά μειονεκτήματά τους είναι ότι δεν κατασκευάζουν πάντα εφικτά πλάνα

παραγωγής και αυτή η αδυναμία τους μπορεί να γίνει αντιληπτή όταν είναι ήδη πλέον αργά. Παράλληλα, τα συστήματα αυτά θεωρούν σταθερούς χρόνους παραμονής των διάφορων προϊόντων/υλικών εντός του συστήματος παραγωγής (lead time) οι οποίοι είναι ανεξάρτητοι από την δυναμικότητα του συστήματος. Στις περισσότερες των περιπτώσεων γίνεται υπερεκτίμηση του χρόνου αυτού. Συνεπώς, ενώ λογικά κάποιος θα περίμενε οι διάφορες εργασίες εντός του συστήματος να έχουν τελειώσει νωρίτερα από ότι αναμενόταν και τα έτοιμα πλέον προϊόντα να δημιουργούν ένα διαθέσιμο απόθεμα, στην πραγματικότητα τις περισσότερες φορές η απαισιόδοξη εκτίμηση του χρόνου αυτού οδηγεί στο να έχουμε περισσότερα ημιτελή προϊόντα ή υλικά εντός του συστήματος παραγωγής (WIP, Work in Progress) καθώς και υψηλότερες τιμές του αποθέματος έτοιμων προϊόντων (FGI, Finished Good Inventory).

1.2 Συστήματα έλξης (pull systems)

Σε αντίθεση με την υιοθέτηση μεθοδολογιών από τον τομέα του CIM, κυρίως οι Ιαπωνικές επιχειρήσεις έχουν υιοθετήσει άλλες μεθόδους και τεχνικές προκειμένου να βελτιώσουν την παραγωγική τους διαδικασία. Οι επιχειρήσεις αυτές υιοθέτησαν μια εκ διαμέτρου αντίθετη πολιτική για την εφοδιαστική τους αλυσίδα: αντί τα προϊόντα/υπηρεσίες να παράγονται προκειμένου να ικανοποιήσουν μελλοντική ζήτηση από τον πελάτη (μοντέλο ώθησης), η παραγωγή καθορίζεται ή όπως λέγεται «καθοδηγείται/ρυθμίζεται» από την τελική ζήτηση (μοντέλο έλξης). Στα μοντέλα αυτά η ολοκλήρωση μιας εργασίας κατά την διάρκεια της παραγωγής «πυροδοτεί» την έναρξη μιας άλλης.

Μια από τις πλέον γνωστές εφαρμογές του μοντέλου έλξης είναι η πολιτική Kanban. Η πολιτική αυτή αναπτύχθηκε για πρώτη φορά από τον Dr.Taichi Ohno, Διευθυντή της αυτοκινητοβιομηχανίας Toyota. Η λέξη Kanban στα Ιαπωνικά σημαίνει «κάρτα», η οποία και χρησιμοποιείται στο σύστημα παραγωγής στο οποίο εφαρμόζεται η πολιτική αυτή προκειμένου είτε να ενεργοποιήσει την διαδικασία έναρξης παραγωγής του συστήματος, είτε να ενεργοποιήσει την έναρξη μίας εργασίας. Η πολιτική Kanban βασίζεται σε ένα οπτικό σήμα που ακολουθεί ένα κιβώτιο εξαρτημάτων από την στιγμή που γεμίσει μέχρι τη στιγμή που αδειάσει. Ρυθμίζει τη ροή στο σύστημα παραγωγής σηματοδοτώντας την παραγωγή και τις παραδόσεις σε προηγούμενα στάδια. Επίσης τα συστήματα που εφαρμόζουν πολιτική Kanban ειδοποιούν τους χειριστές τους πότε υπάρχει πρόβλημα, που εμφανίζεται αυτό το πρόβλημα καθώς και ποιος είναι αρμόδιος να το επιλύσει. Ο αριθμός των καρτών προσδιορίζει το επίπεδο WIP του συστήματος.

Τα βασικά πλεονεκτήματα ενός συστήματος πολιτικής Kanban, είναι ότι πετυχαίνουν μείωση των επιπέδων αποθέματος και των χρόνων παραμονής των προϊόντων/υπηρεσιών εντός του συστήματος (lead time). Ωστόσο, το βασικό μειονέκτημα της συγκεκριμένης πολιτικής είναι ότι δεν μπορεί να εφαρμοστεί σε πολλά συστήματα παραγωγής καθώς αυτά θα πρέπει να ικανοποιούν κάποιες προϋποθέσεις. Μερικές από αυτές είναι να υπάρχει αξιόπιστος εξοπλισμός, καλή ποιότητα προϊόντων και σταθερό εργατικό δυναμικό. Παράλληλα απαιτείται η αυστηρή εφαρμογή των διαδικασιών παραγωγής καθώς και αυστηρή πειθαρχία της ακολουθίας των διαδικασιών και των κανόνων παραγωγής.

Πολλές φορές συστήματα παραγωγής που εφαρμόζουν πολιτική Kanban συγχέονται με συστήματα παραγωγής που εφαρμόζουν πολιτικές JIT (Just In Time). Ωστόσο, οι δύο αυτές πολιτικές δεν είναι ταυτόσημες. Συγκεκριμένα, εφαρμόζοντας μια επιχείρηση την πολιτική JIT δημιουργεί ένα σύνθετο σύστημα παραγωγής το οποίο αποτελείται από ένα σύστημα πολιτικής Kanban, ένα σύστημα ελέγχου ολικής ποιότητας και προϋποθέτει την ενεργή συμμετοχή όσων υπαλλήλων εργάζονται σε αυτό το σύστημα. Ουσιαστικά τα συστήματα αυτού του τύπου στηρίζονται στις ακόλουθες τρεις αρχές: την μείωση του κόστους παραγωγής, την αποφυγή της σπατάλης και την αναγνώριση των ικανοτήτων των εργαζομένων.

Βασικά πλεονεκτήματα ενός συστήματος που εφαρμόζει την πολιτική JIT είναι τα μειωμένα επίπεδα αποθεμάτων εντός του συστήματος (WIP) και ο μειωμένος χρόνος ροής των διαφόρων εργασιών του συστήματος, δηλαδή ο χρόνος που μεσολαβεί μεταξύ της έναρξης μιας εργασίας και της ολοκλήρωσης της. Αυτή η μείωση έχει ως αποτέλεσμα αφενός την μείωση του κόστους παραγωγής και αφετέρου την αύξηση της κατανόησης από τους πελάτες, σε περίπτωση που αυτοί αργήσουν να εξυπηρετηθούν.

1.3 Σύγκριση μοντέλων ώθησης και έλξης

Τα συστήματα τύπου ώθησης (π.χ. MRP) κυρίως λόγω της ευκολίας στην εφαρμογή τους υιοθετούνται από πολλές περισσότερες βιομηχανικές και παραγωγικές μονάδες από ότι τα συστήματα τύπου έλξης (π.χ. Kanban). Ωστόσο, τα συστήματα τύπου έλξης, όταν αυτά μπορούν να εφαρμοστούν, φαίνεται να εμφανίζουν ανώτερα αποτελέσματα σε σχέση με τα αντίστοιχα ώθησης. Στην διεθνή βιβλιογραφία έχουν δημοσιευθεί αρκετές εργασίες που συγκρίνουν τα δύο αυτά μοντέλα, όπως των Spearman, Woodruff and Hopp [1], και Spearman and Zazanis [2]. Η ανωτερότητα

των συστημάτων έλξης έναντι αυτών της ώθησης οφείλεται στους τρεις παρακάτω λόγους:

1. Τα συστήματα έλξης χαρακτηρίζονται από μικρότερη συμφόρηση σε σχέση με αυτή των συστημάτων ώθησης.
2. Ο έλεγχος παραγωγής των συστημάτων έλξης είναι πιο εύκολος σε σχέση με τον έλεγχο των συστημάτων ώθησης.
3. Στα συστήματα έλξης ο αριθμός των κομματιών που βρίσκονται εντός του συστήματος (WIP) βρίσκεται εντός κάποιων ορίων (ανώτατης και κατώτατης τιμής). Στα συστήματα ώθησης, τα αποθέματα μπορεί να αυξάνονται χωρίς όριο.

Παρά την ανωτερότητα των συστημάτων έλξης, δεν τυγχάνουν ευρείας εφαρμογής από τις επιχειρήσεις στα συστήματα παραγωγής, κυρίως λόγω των ειδικών προϋποθέσεων που αυτά απαιτούν προκειμένου να εφαρμοσθούν. Για τον λόγο αυτό τα τελευταία χρόνια γίνονται προσπάθειες συνδυασμού των παραπάνω δύο συστημάτων (έλξης και ώθησης) σε ένα σύστημα το οποίο να διαθέτει τα πλεονεκτήματα των συστημάτων έλξης και παράλληλα να έχει ένα μεγαλύτερο φάσμα εφαρμογής στα διάφορα συστήματα παραγωγής.

1.4 Πολιτική CONWIP

Μια πρώτη προσπάθεια επίλυσης του παραπάνω προβλήματος έγινε από τον Hall [3] ο οποίος ανέπτυξε την πολιτική synchro-MRP. Σε αυτό το σύστημα οι εργασίες προγραμματίζονταν από ένα σύστημα πολιτικής MRP, αλλά το σύστημα αυτό δεν μπορούσε να ξεκινήσει εάν δεν είχε λάβει πρώτα την εξουσιοδότηση (εντολή) να ξεκινήσει από ένα σύστημα πολιτικής Kanban. Η πολιτική αυτή αποτελεί προάγγελο της πολιτικής CONWIP (CONstant Work In Process) η οποία και παρουσιάστηκε από τους Spearman, Woodruff και Hopp [1] το 1990.

Σε ένα σύστημα σταθερού αποθέματος (CONWIP) οι μηχανές παράγουν όποτε μπορούν και εφόσον έχουν υλικά. Ωστόσο κάθε φορά που για οποιονδήποτε λόγο κομμάτι εγκαταλείπει το σύστημα, αυτό αναπληρώνεται άμεσα με πρώτες ύλες. Για παράδειγμα, όταν πωλείται ένα τελικό προϊόν ή απορρίπτεται ένα ελαττωματικό κομμάτι τότε αμέσως εισάγονται πρώτες ύλες στις αρχικές αποθήκες. Σε κάθε μια από

αυτές μπαίνουν τόσες πρώτες ύλες όσες απαιτούνται για την παραγωγή ενός τελικού έτοιμου προϊόντος. Με αυτό τον τρόπο μπορεί να κρατηθεί σταθερό το σύνολο των «εν δυνάμει» έτοιμων προϊόντων και κατά επέκταση να ελεγχθούν και οι χωρητικότητες των ενδιάμεσων αποθηκών έτσι ώστε να μην δημιουργηθούν προβλήματα στην παραγωγή. Επομένως με την πολιτική CONWIP διαμορφώνεται ένα κλειστό σύστημα ελέγχου των αποθεμάτων σε όλη την γραμμή παραγωγής.

1.5 Σύγκριση συστήματος πολιτικής CONWIP με συστήματα πολιτικής Kanban

Το CONWIP μπορεί να χαρακτηριστεί ως μία γενικευμένη μορφή του Kanban. Όπως και το Kanban έτσι και το CONWIP για την σωστή λειτουργία του στηρίζεται στις λεγόμενες «κάρτες». Η διαφορά των δύο συστημάτων είναι ότι στο Kanban οι κάρτες μπορεί να είναι τοποθετημένες σε διάφορα σημεία εντός του συστήματος και μέσω αυτών να πυροδοτείται η έναρξη κάποιας συγκεκριμένης εργασίας. Αντίθετα στο CONWIP μια κάρτα προσκολλάται σε ένα κομμάτι το οποίο βρίσκεται στην αρχή της γραμμής παραγωγής. Όταν το κομμάτι αυτό φτάσει στο τέλος της γραμμής παραγωγής η κάρτα αφαιρείται και επιστρέφει στην αρχή της γραμμής παραγωγής όπου και περιμένει σε μια λίστα αναμονής καρτών έως ότου προσκολληθεί ξανά σε επόμενο κομμάτι. Ο τρόπος εξαγωγής των καρτών από την λίστα βασίζεται στην λειτουργία FIFO (First In First Out). Παράλληλα, στο CONWIP οι διάφορες εργασίες εντός του συστήματος παραγωγής, από την στιγμή που έχουν λάβει την εξουσιοδότηση (εντολή) από την κάρτα στην αρχή της γραμμής παραγωγής υλοποιούνται βάσει της μεθόδου ώθησης.

1.6 Σύγκριση συστήματος πολιτικής CONWIP με συστήματα ώθησης

Το σύστημα CONWIP παρουσιάζει τρία σημαντικά πλεονεκτήματα έναντι των συστημάτων ώθησης. Σε περίπτωση που η ζήτηση υπερβεί την δυναμικότητα του συστήματος, το σύστημα ώθησης θα συνωστισθεί από πρώτες ύλες που αντιστοιχούν στις παραγγελίες που εκκρεμούν. Αντίθετα, το CONWIP θα λειτουργήσει κανονικά έχοντας ένα προκαθορισμένο απόθεμα του οποίου το ύψος είναι ανεξάρτητο από τις παραγγελίες οι οποίες πρέπει να ικανοποιηθούν. Ο χειρισμός μιας τέτοιας λίστας είναι πιο εύκολος σε σχέση με τον χειρισμό του WIP του συστήματος. Επιπλέον, σε αντίθεση με τα συστήματα ώθησης, το CONWIP μας παρέχει την δυνατότητα

συνεχούς ελέγχου του πλήθους των κομματιών που βρίσκονται εντός του συστήματος, των κομματιών δηλαδή που ακόμα είναι στο στάδιο της παραγωγής ή που βρίσκονται στην αποθήκη έτοιμων προϊόντων και είναι διαθέσιμα προς πώληση. Τέλος με την πολιτική CONWIP διατηρείται πεπερασμένο το κόστος αποθέματος, το οποίο συχνά αποτελεί μια πολύ σημαντική συνιστώσα του κόστους παραγωγής.

1.7 Πολιτικές Παραγγελιών – Έλεγχος και Διαχείριση Αποθεμάτων

Όταν ένα προϊόν πωλείται, γίνεται προμήθεια πρώτων υλών για παραγωγή νέων προϊόντων. Το κόστος και τα προβλήματα στη διανομή πρώτων υλών καθιστούν απαγορευτική για μια επιχείρηση τη συνεχή τοποθέτηση παραγγελιών. Για το λόγο αυτό οι επιχειρήσεις προβαίνουν σε μαζικές παραγγελίες πρώτων υλών σε τακτά χρονικά διαστήματα, ώστε να έχουν αρκετό απόθεμα για να παράγουν και να καλύψουν την ζήτηση.

Από την άλλη πλευρά η ύπαρξη και συντήρηση αποθέματος συνεπάγεται ένα κόστος για την επιχείρηση (λόγω δέσμευσης κεφαλαίου, πληρωμής ασφαλιστρών, έξοδα λειτουργίας αποθήκης κλπ.). Στόχος μιας επιχείρησης είναι να προσδιορίσει την κατάλληλη ποσότητα παραγγελίας ώστε να ελαχιστοποιείται το κόστος αποθέματος και συγχρόνως να αποφεύγονται ελλείψεις πρώτων υλών.

Με την θεωρία ελέγχου αποθεμάτων (inventory control) μπορεί να γίνει αποτελεσματική διαχείριση του ύψους των αποθεμάτων πρώτων υλών, με σκοπό την απρόσκοπτη λειτουργία του συστήματος για ορισμένο επίπεδο της ζήτησης με το ελάχιστο δυνατό κόστος. Η δημιουργία αποθεμάτων μπορεί είτε να είναι σχεδιασμένη με σκοπό να εξομαλύνει τις παρουσιαζόμενες διαφορές μεταξύ της προσφοράς και της ζήτησης του αγαθού, είτε να προκύψει ως αποτέλεσμα διαφόρων παραγόντων, όπως κακός προγραμματισμός ή έκτακτα φαινόμενα (π.χ. βλάβες μηχανών). Ένας σωστός σχεδιασμός διαχείρισης αποθεμάτων αποσυνδέει το παραγωγικό σύστημα από τις διακυμάνσεις της ζήτησης, διατηρεί ομαλή ροή κομματιών στο σύστημα, αυξάνει το ρυθμό παραγωγής και ελαττώνει το κόστος.

Το βασικό πρόβλημα με το οποίο ασχολείται η θεωρία αποθεμάτων είναι ο προσδιορισμός του μεγέθους των παραγγελιών πρώτων υλών καθώς και ο χρονικός προγραμματισμός τους. Μας ενδιαφέρει δηλαδή, με απλούς όρους, πόσο και πότε πρέπει να παραγγείλουμε. Για την μελέτη ενός προβλήματος αποθεμάτων θα πρέπει

να είναι γνωστές οι ιδιαιτερότητες της δομής του συστήματος, αφού στις ποικίλες εφαρμογές, τα πλαίσια λειτουργίας διαφέρουν ουσιωδώς.

Οι βασικοί παράγοντες που επηρεάζουν την ποσότητα αποθέματος σε ένα σύστημα παραγωγής είναι οι ακόλουθοι:

- Ο ρυθμός ζήτησης, που ορίζεται ως το πλήθος των προϊόντων που ζητούνται προς κατανάλωση ανά χρονική μονάδα.
- Ο χρόνος που μεσολαβεί μεταξύ παραγγελίας και παράδοσης (lead time). Σε ορισμένα μοντέλα ο χρόνος αυτός είναι μηδενικός. Στην πράξη είναι ένας θετικός αριθμός ή τυχαία μεταβλητή.
- Ο κύκλος παραγγελίας, ορίζεται ως το διάστημα μεταξύ δύο παραγγελιών.
- Το απόθεμα ασφαλείας, είναι το απόθεμα που διατηρείται προκειμένου να αντιμετωπισθεί υψηλότερη ζήτηση από την προβλεπόμενη κατά την διάρκεια του χρόνου μεταξύ παραγγελίας και παράδοσης.
- Οι εκπτώσεις, που δίνονται από τους προμηθευτές πρώτων υλών στο σύστημα παραγωγής και είναι ανάλογες της ποσότητας παραγγελίας.

Μια πρώτη διάκριση των μοντέλων αποθεμάτων γίνεται βάσει της ζήτησης του προϊόντος. Τα μοντέλα στα οποία η ζήτηση είναι γνωστή με βεβαιότητα αναφέρονται ως **αιτιοκρατικά ή ντετερμινιστικά** σε αντίθεση με τα μοντέλα στα οποία η ζήτηση είναι γνωστή μόνο πιθανοθεωρητικά, που αναφέρονται ως **στοχαστικά**. Επιπλέον τα μοντέλα διακρίνονται σε **συνεχούς και σε περιοδικής επιθεώρησης** ανάλογα με το αν είναι δυνατή η συνεχής παρακολούθηση του ύψους του αποθέματος ή όχι. Στην συνέχεια ακολουθεί μια σύνοψη των πολιτικών ελέγχου αποθεμάτων που εφαρμόζονται στην πράξη (Hadley and Whitin [6] και Zipkin [8]).

1.8 Συστήματα Συνεχούς Επιθεώρησης

1.8.1 Σταθερή ποσότητα παραγγελίας (s , Q)

Στην πολιτική (s , Q) παραγγέλλονται Q πρώτες ύλες όταν το απόθεμα πέσει στην τιμή s , η οποία ονομάζεται στάθμη αναπαραγγελίας. Παράλληλα, η ποσότητα παραγγελίας Q παραμένει σταθερή στον χρόνο καθώς στο σύστημα αυτό η ζήτηση

είναι ομοιόμορφη, με την έννοια ότι παραμένει σταθερή από την μια περίοδο στην άλλη. Στην ιδεατή περίπτωση εφαρμογής μίας πολιτικής (s, Q) , η κατάσταση του αποθέματος παρακολουθείται συνεχώς. Το σύστημα διαχείρισης παραγγελιών λειτουργεί ως εξής: Εάν κατά την τρέχουσα επιθεώρηση του συστήματος η κατάσταση του διαθέσιμου αποθέματος είναι ίση ή μικρότερη από την στάθμη αναπαραγγελίας s , τότε δίνει την εντολή για παραγγελία ποσότητας Q .

Με τον όρο διαθέσιμο απόθεμα εννοούμε τον συνολικό αριθμό των πρώτων υλών, ημικατεργασμένων κομματιών και των μονάδων προϊόντος που είναι άμεσα διαθέσιμο προς πώληση, μειωμένο κατά το μέγεθος της ανικανοποίητης ζήτησης. Η ανικανοποίητη ζήτηση γίνεται να εμφανιστεί μόνο όταν το απόθεμα προϊόντων τα οποία είναι άμεσα προς πώληση είναι μηδενικό, οπότε εξαιτίας της ζήτησης που δεν έχει ικανοποιηθεί, το διαθέσιμο απόθεμα γίνεται αρνητικό.

1.8.2 Μεταβλητή ποσότητα παραγγελίας (s, S)

Στην πολιτική (s, S) , η χρονική στιγμή κατά την οποία ξεκινά η διαδικασία αναπλήρωσης των πρώτων υλών εξαρτάται πάλι από τη στάθμη αναπαραγγελίας s . Επιπλέον η ποσότητα παραγγελίας μπορεί να μεταβάλλεται στον χρόνο, καθώς στο παρόν σύστημα η ζήτηση είναι δυναμική, δηλαδή είναι διαφορετική από την μια περίοδο στην άλλη. Κάθε φορά που θα γίνεται μια παραγγελία η ποσότητα αυτής θα πρέπει να είναι τέτοια ώστε το σύστημα να αποκτήσει ξανά διαθέσιμο απόθεμα ίσο με S . Δηλαδή, παραγγέλλουμε ποσότητα $Q = S - s$.

1.9 Σύστημα Περιοδικής Επιθεώρησης

1.9.1 Περιοδική επιθεώρηση (R, S)

Στην πολιτική (R, S) το σύστημα επιθεώρησης ακολουθεί τον ακόλουθο κανόνα απόφασης: Σε σταθερά και συγκεκριμένα χρονικά διαστήματα $0, R, 2R, \dots$, δίνει εντολή για την παραγγελία ποσότητας η οποία απαιτείται προκειμένου το σύστημα να αποκτήσει ξανά διαθέσιμο απόθεμα S . Η ποσότητα της παραγγελίας εξαρτάται από την ζήτηση καθώς και από το απόθεμα το οποίο μετρήθηκε τις προηγούμενες χρονικές περιόδους.

1.9.2 Περιοδική επιθεώρηση (R, s, S)

Στην πολιτική (R, s, S), η χρονική στιγμή κατά την οποία ξεκινά η έναρξη της διαδικασίας αναπλήρωσης των πρώτων υλών, εξαρτάται από τη στάθμη αναπαραγγελίας s η οποία ελέγχεται ανά περίοδο επιθεώρησης R . Και σε αυτή την πολιτική η ποσότητα παραγγελίας μπορεί να μεταβάλλεται στο χρόνο. Ωστόσο, κάθε φορά που θα γίνεται μια παραγγελία, η ποσότητα αυτής θα πρέπει να είναι τέτοια ώστε το σύστημα να αποκτήσει ξανά διαθέσιμο απόθεμα ίσο με S . Δηλαδή, παραγγέλλουμε ποσότητα $Q = S - s$. Το σύστημα αυτό είναι το πιο γενικευμένο σύστημα επιθεώρησης από τα προηγούμενα τρία που αναφέρθηκαν στην παρούσα μελέτη και αποτελεί έναν συνδυασμό αυτών. Για παράδειγμα, εάν το $R=1$ (αν ο χρόνος είναι διακριτός) ή $R \rightarrow 0$ (αν ο χρόνος είναι συνεχής) και η ποσότητα που ζητείται κάθε φορά είναι σταθερή τότε πρόκειται για πολιτική (s, Q). Επίσης εάν το $R=1$ ή $R \rightarrow 0$ και η ποσότητα που ζητείται κάθε φορά δεν είναι σταθερή αλλά μεταβάλλεται τότε πρόκειται για πολιτική (s, S).

1.10 Πολιτικές Αποδοχής ή Απόρριψης Παραγγελιών

Ένα σύστημα παραγωγής προκειμένου να λάβει αποφάσεις σχετικά με την απόρριψη ή την αποδοχή νέων παραγγελιών θα πρέπει κάθε χρονική στιγμή να γνωρίζει το πλήθος του διαθέσιμου αποθέματος το οποίο βρίσκεται εντός του συστήματος παραγωγής, καθώς και το πλήθος των εκκρεμών παραγγελιών. Με τον όρο διαθέσιμο απόθεμα εννοούμε το πλήθος των προϊόντων που βρίσκονται εντός του συστήματος παραγωγής και τα οποία είναι άμεσα διαθέσιμα για την ικανοποίηση της ζήτησης.

Στην διεθνή βιβλιογραφία υπάρχουν αρκετές εργασίες που εξετάζουν τη δομή της βέλτιστης πολιτικής ελέγχου αποδοχής πελατών. Σε ορισμένες από αυτές, όπως στις εργασίες των Stidham [4] και Stidham και Weber [5], χρησιμοποιούνται αναμονητικά συστήματα για τη μαθηματική περιγραφή των συστημάτων παραγωγής και στοχαστικός δυναμικός προγραμματισμός για την λήψη της βέλτιστης απόφασης (αποδοχή ή απόρριψη παραγγελίας).

Η πολιτική παραγγελιών που θα υιοθετήσει ένα σύστημα παραγωγής, εξαρτάται άμεσα από την ιδιομορφία με την οποία αυτό χαρακτηρίζεται, από το μοντέλο παραγωγής το οποίο χρησιμοποιεί, καθώς και από την πολιτική διαχείρισης αποθεμάτων που έχει υιοθετήσει.

Οι δύο βασικότερες αλλά και εκ διαμέτρου αντίθετες πολιτικές παραγγελιών είναι οι ακόλουθες:

- *Complete Backordering*: Πολιτική πλήρους αποδοχής της μη ικανοποιημένης ζήτησης (όπου το έλλειμμα βάσης είναι ∞). Το έλλειμμα βάσης είναι ένα κατώφλι το οποίο αφορά το πλήθος της ανικανοποίητης ζήτησης.
- *Lost Sales*: Πολιτική πλήρους απόρριψης της μη ικανοποιημένης ζήτησης (όπου το έλλειμμα βάσης είναι 0). Δηλαδή όταν δεν υπάρχει ετοιμοπαράδοτο προϊόν εντός του συστήματος παραγωγής.

Η φιλοσοφία των παραπάνω δύο πολιτικών παραγγελιών αναπτύσσεται αναλυτικότερα σε βιβλία και άρθρα των Hadley and Whitin [6], Smith [7], και Zipkin [8].

Με την πλήρη αποδοχή της μη άμεσα ικανοποιημένης ζήτησης δεν τίθεται φραγμός στο πλήθος των καθυστερημένων παραγγελιών ενώ αντίθετα με την πολιτική πλήρους απόρριψης το πλήθος των εκκρεμών παραγγελιών είναι μηδέν. Μια ενδιαμέση πολιτική ελέγχου των παραγγελιών είναι η πολιτική μερικής αποδοχής της μη ικανοποιημένης ζήτησης (PLS, Partly Lost Sales). Η πολιτική αυτή είναι τύπου κατωφλιού και απορρίπτει όλες τις παραγγελίες όταν η μη ικανοποιημένη ζήτηση γίνει ίση με ένα κατώφλι που ονομάζεται έλλειμμα βάσης. Η πολιτική μερικής αποδοχής παραγγελιών είναι γενικότερη των πλήρους αποδοχής και πλήρους απόρριψης αφού για ακραίες τιμές ∞ και 0 του ελλείμματος βάσης εκφυλίζεται σε μια από αυτές. Η πολιτική αυτή εξετάζεται για πρώτη φορά στις εργασίες Caldentey [9] και Kouikoglou and Phillis [10] σε συνδυασμό με πολιτικές αποθεμάτων τύπου $(S-1, S)$ για συστήματα που έχουν μόνο μια μηχανή.

Μια εναλλακτική πολιτική ελέγχου όταν δεν υπάρχει διαθέσιμο απόθεμα είναι η αποδοχή ή απόρριψη παραγγελιών με τυχαίο τρόπο (π.χ πραγματοποιώντας ένα πείραμα Bernoulli) ανεξαρτήτως από το πλήθος των παραγγελιών που ήδη εκκρεμούν (Moinzadeh [11]). Αυτή η πολιτική ονομάζεται τυχαίας αποδοχής (RAC, Randomized Admission Control) και μέχρι τώρα χρησιμοποιείται μόνο σε συστήματα μίας μηχανής.

Μια άλλη εναλλακτική πολιτική ελέγχου για την αποδοχή ή απόρριψη πελατών είναι αυτή που διαχωρίζει τους πελάτες σε κλάσεις σημαντικότητας, επομένως και

προτεραιότητας στην εξυπηρέτησή τους. Τέτοιες εργασίες έχουν δημοσιευθεί από τον Ha [12], [13].

Σε όλες τις παραπάνω εργασίες εξετάζονται συστήματα παραγωγής που έχουν μόνο μια μηχανή. Η πολιτική της μερικής αποδοχής παραγγελιών σε συνδυασμό με πολιτικές αποθεμάτων ($S-1$, S) έχει εφαρμοσθεί σε δίκτυα παραγωγής στην εργασία [14].

1.11 Σύνοψη της Διπλωματικής Εργασίας

Σε αυτήν την διπλωματική εργασία εξετάζονται γραμμές παραγωγής στις οποίες ο έλεγχος γίνεται συνδυάζοντας τρεις πολιτικές:

1. Πολιτική τύπου CONWIP: Το άθροισμα της στάθμης αποθέματος του συστήματος και της στάθμης παραγγελιών πρώτων υλών που δεν έχουν φθάσει ακόμη είναι σταθερό.
2. Πολιτική παραγγελιών πρώτων υλών: Παραγγέλλονται πρώτες ύλες κατά παρτίδες μεγέθους Q .
3. Πολιτική απόρριψης ή αποδοχής παραγγελιών: Όταν δεν υπάρχουν έτοιμα προϊόντα, οι παραγγελίες γίνονται δεκτές μέχρις ότου το πλήθος προϊόντων που υπολείπεται να παραχθούν (ώστε να ικανοποιηθούν όλοι οι πελάτες που είναι σε αναμονή) φθάσει σε κάποιο άνω όριο, το έλλειμμα βάσης. Πέραν αυτού του ορίου, οι αφικνούμενες παραγγελίες δεν γίνονται δεκτές από το σύστημα.

Αναπτύσσεται ένας αλγόριθμος προσομοίωσης γραμμών παραγωγής μέσω του οποίου προσδιορίζονται οι παράμετροι ελέγχου (στάθμη CONWIP, ποσότητα παραγγελίας Q , έλλειμμα βάσης) οι οποίες μεγιστοποιούν το μέσο ρυθμό κέρδους του συστήματος.

2. Περιγραφή του Συστήματος

Το σύστημα παραγωγής το οποίο μελετάται στην παρούσα εργασία παρουσιάζεται στο Σχήμα.1 και περιγράφεται από τις ακόλουθες συνιστώσες:

2.1 Γραμμή Παραγωγής

Εξετάζονται συστήματα παραγωγής CONWIP που αποτελούνται από μια γραμμή παραγωγής με μηχανές συνδεδεμένες σε σειρά μέσω ενδιάμεσων αποθηκών, καθώς και από ενδιάμεσες αποθήκες. Κάθε μηχανή ακολουθείται από μια αποθήκη πεπερασμένης χωρητικότητας. Στην τελευταία αποθήκη αποθηκεύονται τα έτοιμα προϊόντα του συστήματος. Η αποθήκη αυτή σε αντίθεση με όλες τις άλλες θεωρείται ότι έχει αρκετή χωρητικότητα ώστε να δεχθεί όλα τα ημιτελή κομμάτια που υπάρχουν στο σύστημα. Οι χρόνοι επεξεργασίας ενός κομματιού από τη κάθε μηχανή του συστήματος είναι εν γένει τυχαίοι και υπάρχει δυνατότητα επιλογής μιας από τις ακόλουθες κατανομές: ομοιόμορφη, εκθετική, σταθερή ή Weibul. Κάθε μηχανή μπορεί να έχει οιαδήποτε από αυτές τις κατανομές με διαφορετικές παραμέτρους κατανομής από τις άλλες μηχανές.

2.2 Ζήτηση και Πώληση

Όταν παράγεται τελικό προϊόν, μεταφέρεται στην αποθήκη έτοιμων προϊόντων όπου αναμένει πελάτη που θα το αγοράσει. Εισάγονται δηλαδή δύο έννοιες, η ζήτηση και η πώληση.

Όσον αφορά την ζήτηση, αυτή προσομοιώνεται μέσω μιας εικονικής μηχανής η οποία βρίσκεται μετά την αποθήκη των έτοιμων προϊόντων και αποτελεί την τελευταία μηχανή του συστήματος παραγωγής. Η διαδικασία της ζήτησης μπορεί να θεωρηθεί ως μια μηχανή που παράγει και αυτή με κάποια τυχειότητα. Ωστόσο, αυτό που παράγει δεν είναι τίποτα άλλο από πελάτες που έρχονται και είναι υποψήφιοι αγοραστές. Αντίστοιχα με τις άλλες μηχανές, υπάρχει η δυνατότητα επιλογής από ένα πλήθος κατανομών (ομοιόμορφη, εκθετική, σταθερή, Weibul) της κατανομής την οποία θα ακολουθεί ο χρόνος της μηχανής της ζήτησης. Τέλος, ο αριθμός των προϊόντων τα οποία ζητούνται ανά παραγγελία (παρτίδα), δεν είναι πάντα ο ίδιος

αλλά υπάρχει δυνατότητα επιλογής του μεγέθους της παρτίδας που μπορεί να παραγγείλει ένας πελάτης σύμφωνα με γνωστές πιθανότητες που μπορεί να είναι διαθέσιμες από στατιστική ανάλυση του ιστορικού πωλήσεων της επιχείρησης.

Όσον αφορά την πώληση, στο συγκεκριμένο σύστημα παραγωγής ακολουθείται η πολιτική μερικής αποδοχής της ζήτησης, η φιλοσοφία της οποίας αναπτύχθηκε στο προηγούμενο κεφάλαιο.

2.3 Παραγγελίες πρώτων υλών

Στο παρόν σύστημα παραγωγής χρησιμοποιείται μια στοχαστική πολιτική επιθεώρησης παραγγελιών συνεχούς χρόνου. Η πολιτική αυτή είναι ένας συνδυασμός των πολιτικών μεταβλητής ποσότητας παραγγελίας (s, S), σταθερής ποσότητας παραγγελίας (s, Q) και της πολιτικής CONWIP.

Σε ένα σύστημα CONWIP το S ισούται με το πλήθος των ημιτελών κομματιών και των έτοιμων προϊόντων τα οποία βρίσκονται εντός του συστήματος παραγωγής. Δηλαδή, πρόκειται για το άθροισμα των έτοιμων προϊόντων τα οποία είναι άμεσα διαθέσιμα για την ικανοποίηση της ζήτησης, των κομματιών τα οποία βρίσκονται στις ενδιάμεσες αποθήκες του συστήματος, των κομματιών που βρίσκονται στις μηχανές και υφίστανται επεξεργασία, καθώς και των πρώτων υλών.

Η φιλοσοφία της πολιτικής CONWIP, όπως ήδη έχει αναφερθεί, είναι ότι όταν ένα κομμάτι πωληθεί και επομένως εγκαταλείψει το σύστημα, αυτό θα πρέπει να αναπληρωθεί άμεσα με πρώτες ύλες. Η φιλοσοφία αυτή σε συνδυασμό με τις πολιτικές παραγγελίας πρώτων υλών (s, S) και (s, Q) οδηγούν στην δημιουργία μιας καινούργιας πολιτικής τύπου $(s, S)=(S-Q, S)$.

Στην νέα αυτή πολιτική παραγγελίας πρώτων υλών η στάθμη αναπαραγγελίας είναι η $S-Q$ και η ποσότητα παραγγελίας είναι ίση με S . Το S στην συγκεκριμένη όμως περίπτωση είναι το άθροισμα του πραγματικού αποθέματος (πρώτες ύλες, έτοιμα προϊόντα και κομμάτια τα οποία βρίσκονται στο στάδιο της παραγωγής) και των παραγγελιών πρώτων υλών που δεν έχουν φθάσει ακόμα στο σύστημα.

Οι παραγγελίες πρώτων υλών φθάνουν στο σύστημα μέσω μιας εικονικής μηχανής η οποία τοποθετείται στην αρχή του συστήματος παραγωγής. Ο ρόλος της μηχανής αυτής είναι να πραγματοποιεί αφίξεις παραγγελιών πρώτων υλών κατά παρτίδες μεγέθους Q . Αντίστοιχα με τις υπόλοιπες μηχανές του συστήματος, ο χρόνος αυτός

είναι εν γένει τυχαίος. Υπάρχει η δυνατότητα επιλογής από ένα πλήθος κατανομών (ομοιόμορφη, εκθετική, σταθερή, Weibul) της κατανομής την οποία θα ακολουθεί ο χρόνος της μηχανής άφιξης των παραγγελιών πρώτων υλών.

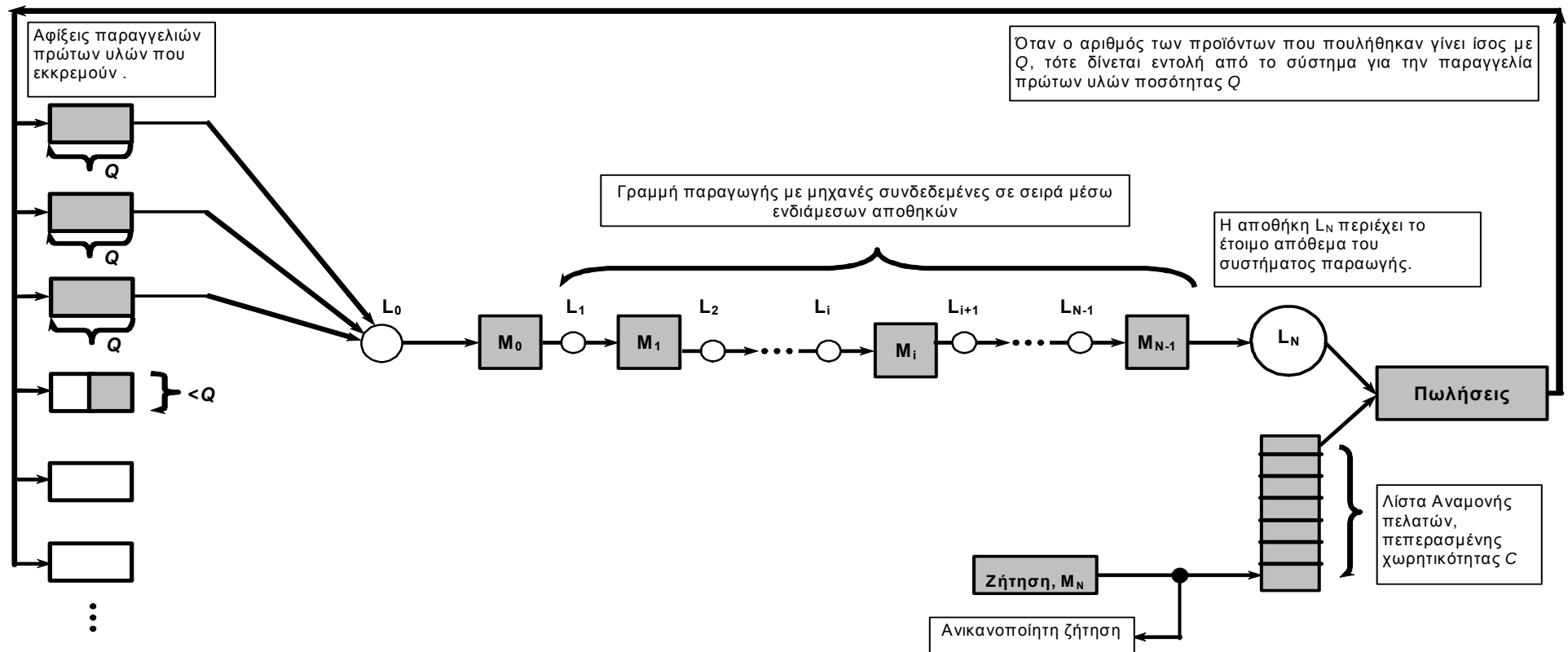
Η βασική διαφορά της μηχανής αυτής από τις υπόλοιπες μηχανές του συστήματος είναι ότι έχει την δυνατότητα να επεξεργάζεται παραπάνω από ένα κομμάτια (παραγγελίες) την φορά. Είναι δηλαδή σαν ένα σύστημα παράλληλων μηχανών, κάθε μια από τις οποίες αντιστοιχεί στο φορτηγό (ή άλλο μέσον) το οποίο πρόκειται να φέρει πρώτες ύλες στο σύστημα. Μέσω της διαδικασίας αυτής διασφαλίζεται ότι η παραγγελία που θα φθάσει στο σύστημα θα είναι αυτή που θα έχει τον μικρότερο χρόνο άφιξης από το σύνολο των παραγγελιών που θα έχουν γίνει μέχρι μια δεδομένη χρονική στιγμή από το σύστημα.

2.4 Λίστα αναμονής πελατών

Προκειμένου το σύστημα να μπορεί να εφαρμόσει την πολιτική της μερικής ικανοποίησης της ζήτησης, πρέπει να διαθέτει μια λίστα αναμονής πελατών. Η λίστα αυτή έχει πεπερασμένη χωρητικότητα λόγω του ότι έχουμε θεωρήσει ότι το πλήθος των κομματιών (ή παραγγελιών) που εκκρεμούν δεν μπορεί να υπερβεί το έλλειμμα βάσης. Η λειτουργία της λίστας αυτής στηρίζεται στην αρχή FIFO (First In First Out). Δηλαδή, η προτεραιότητα που τηρείται για την εξυπηρέτηση ενός πελάτη από το σύστημα είναι βάσει της σειράς που μπήκε στην λίστα και ειδικότερα, ο πελάτης που μπήκε πρώτος στην λίστα αναμονής, πρώτος θα εξυπηρετηθεί.

Επιπλέον έχουμε θεωρήσει ότι όλοι οι πελάτες ανήκουν σε μια και μόνο κλάση σημαντικότητας. Επομένως, όλοι οι πελάτες υπόκεινται στα ίδια κριτήρια όσον αφορά το αν τελικά θα εξυπηρετηθούν ή όχι από το συγκεκριμένο σύστημα παραγωγής. Άλλες πολιτικές εξυπηρέτησης μπορούν εύκολα να περιγραφούν κάνοντας μικρές επεκτάσεις στο μοντέλο προσομοίωσης που αναπτύσσεται στην εργασία αυτή.

Το Σχήμα 1 δείχνει μια αναπαράσταση του συγκεκριμένου συστήματος παραγωγής.



Σχήμα. 1. Ενδεικτικό σύστημα τύπου CONWIP στο οποίο εφαρμόζεται η πολιτική της μερικής αποδοχής της μη ικανοποιημένης ζήτησης και στην οποία οι παραγγελίες φθάνουν στο σύστημα κατά παρτίδες ποσότητας Q . Η εντολή για παραγωγή πρώτων υλών από το σύστημα θα δοθεί μόνο όταν ο αριθμός των προϊόντων που έχουν πουληθεί γίνει ίσος με την ποσότητα Q .

Εκτός από τις τέσσερις παραπάνω συνιστώσες οι οποίες χαρακτηρίζουν το συγκεκριμένο σύστημα παραγωγής υπάρχουν και κάποιες παράμετροι οι οποίες επηρεάζουν τον τρόπο λειτουργίας του. Οι παράμετροι αυτές αναπτύσσονται αναλυτικά στις ακόλουθες ενότητες.

2.5 Λειτουργικές Παράμετροι

Οι λειτουργικές παράμετροι του συστήματος αφορούν όλους εκείνους τους παράγοντες (π.χ χρόνοι γεγονότων, κατάσταση μηχανών) οι οποίοι επηρεάζουν την λειτουργία των μηχανών στο σύστημα.

Οι μηχανές της γραμμής παραγωγής θα συμβολίζονται ως M . Το πλήθος των μηχανών θα συμβολίζεται N . Όπως ήδη έχει αναφερθεί παραπάνω, το συγκεκριμένο σύστημα παραγωγής αποτελείται από ένα πλήθος πραγματικών μηχανών, οι οποίες και επεξεργάζονται τα διάφορα προϊόντα και από δύο εικονικές μηχανές. Η μια προσομοιώνει την διαδικασία της ζήτησης και η δεύτερη την διαδικασία της παραγγελίας πρώτων υλών από το σύστημα. Αν ο συνολικός αριθμός των μηχανών του συστήματος, συμπεριλαμβάνοντας το πλήθος των πραγματικών και των εικονικών μηχανών του συστήματος, είναι ίσος με N , τότε:

- Η μηχανή που προσομοιώνει την διαδικασία των παραγγελιών είναι η μηδενική και συμβολίζεται ως M_0 . Η «μηχανή» αυτή στην πραγματικότητα απαρτίζεται από πολλές μηχανές, όσες είναι οι παραγγελίες πρώτων υλών που εκκρεμούν.
- Η μηχανή της ζήτησης, αποτελεί την τελευταία μηχανή του συστήματος και επομένως συμβολίζεται ως M_N .
- Οι υπόλοιπες μηχανές, οι οποίες βρίσκονται μεταξύ των M_0 και M_N αφορούν τις ενδιάμεσες μηχανές του συστήματος και συμβολίζονται ως M_i , όπου το i υποδηλώνει τον αριθμό της μηχανής αυτής. Προφανώς, η τελευταία μηχανή της γραμμής παραγωγής είναι η M_{N-1} . Μόλις ένα κομμάτι επεξεργαστεί και από αυτή την μηχανή, θεωρείται προϊόν το οποίο είναι άμεσα διαθέσιμο προς πώληση.

Κάθε μια από τις συνολικά N μηχανές του συστήματος έχει έναν χρόνο κατεργασίας. Ο χρόνος αυτός μπορεί να:

- Είναι σταθερός και ίσος με con_i .
- Είναι εκθετικά κατανομημένος με μέση τιμή μ_i .
- Είναι ομοιόμορφος στο διάστημα $[a_i, \beta_i]$.
- Ακολουθεί την κατανομή Weibul με παραμέτρους $[a_i, \beta_i, \gamma_i]$.

Επιπλέον, πριν από κάθε μηχανή υπάρχει μια αποθήκη. Οι ενδιάμεσες αυτές αποθήκες συμβολίζονται ως L_i , όπου το i , υποδηλώνει τον αριθμό της μηχανής στην οποία η αποθήκη αυτή ανήκει. Οπότε για παράδειγμα το L_1 σημαίνει ότι η αποθήκη αυτή ανήκει στην μηχανή M_1 . Ως χωρητικότητα της αποθήκης έχει θεωρηθεί το άθροισμα των κομματιών που βρίσκονται εντός της αποθήκης και του κομματιού που μπορεί να βρίσκεται εντός της αντίστοιχης μηχανής και υφίσταται επεξεργασία. Επομένως η μέγιστη δυνατή χωρητικότητα της αποθήκης i ορίζεται η ποσότητα $L_i + 1$.

Τέλος, υπάρχει μια μεταβλητή s_i η οποία χαρακτηρίζει την τρέχουσα κατάσταση της i μηχανής. Η s_i λαμβάνει τιμή 0, 1, ή 2 σύμφωνα με τα ακόλουθα:

- Όταν το s_i είναι ίσο με 0, σημαίνει ότι η μηχανή αυτή είναι αποστερημένη («πεινάει»). Δηλαδή, η αποθήκη της μηχανής αυτής είναι άδεια και επομένως δεν έχει κομμάτια για να την τροφοδοτήσει ούτως ώστε να δουλέψει.
- Όταν το s_i είναι ίσο με 1, σημαίνει ότι η μηχανή αυτή δουλεύει, δηλαδή επεξεργάζεται ένα κομμάτι.
- Όταν το s_i είναι ίσο με 2, σημαίνει ότι η μηχανή αυτή είναι αποκλεισμένη («μπλοκαρισμένη»). Δηλαδή, αν και έχει ολοκληρώσει την επεξεργασία ενός κομματιού δεν μπορεί να το στείλει στην επόμενη αποθήκη καθώς το πλήθος των κομματιών που ήδη βρίσκονται σε αυτή ισούται με την μέγιστη δυνατή χωρητικότητά της.

2.6 Παράμετροι Κόστους

Πέραν των λειτουργικών παραμέτρων που εξετάστηκαν παραπάνω υπάρχουν και παράμετροι κόστους οι οποίες θα διαμορφώσουν στο τέλος το κατά πόσο μια πολιτική με μια συγκεκριμένη ποσότητα αναπαραγγελίας Q , με μια λίστα αναμονής πελατών χωρητικότητας C και τέλος με αρχικό απόθεμα $I \cdot Q$ είναι συμφέρουσα. Σε

μια γραμμή παραγωγής υπάρχουν κάποιες παράμετροι που συνδέονται με το κέρδος και κάποιες που συνδέονται με το κόστος. Το I δηλώνει το πολλαπλάσιο του Q και πρέπει να είναι ακέραιος αριθμός. Δεν έχει νόημα να θεωρήσουμε ως απόθεμα CONWIP το $3.5 \cdot Q$ για παράδειγμα, όταν οι παραγγελίες φθάνουν κατά Q .

Στο συγκεκριμένο σύστημα παραγωγής στο κέρδος συμβάλλει μόνο η τιμή πώλησης που διαμορφώνεται για το τελικό προϊόν. Αν από την τιμή πώλησης αφαιρεθούν κάποια κόστη, σχετιζόμενα με το λειτουργικό κόστος ανά προϊόν κατά την παραγωγική διαδικασία ενός κομματιού (π.χ. κόστος πρώτων υλών, κόστος λειτουργίας μηχανών ανά προϊόν κ.α.) διαμορφώνεται το τελικό καθαρό κέρδος ανά μονάδα προϊόντος.

Όμως πέραν των κερδών υπάρχουν και τα κόστη. Κάθε δίκτυο παραγωγής οποιασδήποτε μορφής επιβαρύνεται με το κόστος αποθέματος ανά μονάδα χρόνου. Πρόκειται για το κόστος που οφείλεται αφενός στην δέσμευση κεφαλαίου εξαιτίας της ύπαρξης αποθεμάτων και αφετέρου στα έξοδα συντήρησης αυτών των αποθεμάτων στις αποθήκες του δικτύου παραγωγής κατά την παραγωγική διαδικασία.

Τέλος, στο συγκεκριμένο σύστημα είναι δυνατόν να υπάρξει ανικανοποίητη ζήτηση. Στην περίπτωση αυτή θα υπάρχει ένα κόστος καθυστέρησης ικανοποίησης της παραγγελίας ενός πελάτη. Αυτό το κόστος μπορεί να είναι είτε άμεσο (ρήτρες καθυστέρησης), είτε έμμεσο (δυσφορία πελατών). Με αυτό τον τρόπο διαμορφώνεται ένα κόστος ανικανοποίητης ζήτησης ανά μονάδα χρόνου.

3. Περιγραφή Αλγορίθμου

Το συγκεκριμένο σύστημα παραγωγής αποτελείται από δυο αλγόριθμους. Ο ένας προσομοιώνει τον τρόπο λειτουργίας του συστήματος παραγωγής και ο δεύτερος αναζητεί και παρουσιάζει την βέλτιστη πολιτική μέσα από ένα πλήθος δυνατών πολιτικών που μπορούν να εφαρμοστούν στο συγκεκριμένο σύστημα. Στην συνέχεια ακολουθεί αναλυτική περιγραφή του τρόπου λειτουργίας των παραπάνω δύο αλγορίθμων.

3.1 Αλγόριθμος Προσομοίωσης Διακριτών Γεγονότων

3.1.1 Βασική Δομή του Αλγορίθμου

Η βασική φιλοσοφία λειτουργίας του συγκεκριμένου αλγορίθμου προσομοίωσης είναι η εξής:

- Αρχικά, βρίσκει όλα τα πιθανά επόμενα γεγονότα που πρόκειται να συμβούν στις μηχανές του συστήματος τις επόμενες χρονικές περιόδους από αυτή που βρίσκεται τώρα. Τα γεγονότα ορίζονται παρακάτω.
- Στην συνέχεια, για το κάθε πιθανό επόμενο γεγονός υπολογίζει την χρονική στιγμή κατά την οποία αυτό αναμένεται να εμφανιστεί.
- Τέλος, ορίζει ως επόμενο γεγονός του συστήματος, το γεγονός εκείνο το οποίο πρόκειται να πραγματοποιηθεί πιο σύντομα από τα υπόλοιπα πιθανά επόμενα γεγονότα.

Τα πιθανά επόμενα γεγονότα τα οποία μπορεί να εμφανιστούν στο συγκεκριμένο σύστημα παραγωγής είναι τα ακόλουθα:

1. Άφιξη ενός πελάτη από την μηχανή M_N , ο οποίος ζητάει συγκεκριμένη ποσότητα παραγγελίας,
2. Παραγωγή ενός έτοιμου προϊόντος από την μηχανή M_{N-1} .
3. Άφιξη μιας παραγγελίας πρώτων υλών, ποσότητας Q , από την μηχανή M_0 .
4. Παραγωγή ενός κομματιού από μια από τις ενδιάμεσες μηχανές M_i του συστήματος παραγωγής.

Η χρονική στιγμή κατά την οποία θα πραγματοποιηθεί ένα από τα παραπάνω γεγονότα, συμβολίζεται T_i , όπου το i αντιστοιχεί στον αριθμό της μηχανής που θα παραγάγει ένα γεγονός. Δηλαδή, το T_N σημαίνει ότι η μηχανή M_N την χρονική στιγμή T_N θα φέρει στο σύστημα έναν πελάτη. Επιπλέον, ο τρέχων χρόνος του συστήματος συμβολίζεται ως t , ενώ ο χρόνος που διαρκεί η προσομοίωση ως $TSIM$.

Στην συνέχεια περιγράφεται η διαδικασία που ακολουθείται για την προσομοίωση καθενός από τα παραπάνω τέσσερα πιθανά γεγονότα του συστήματος παραγωγής.

3.1.2 Άφιξη ζήτησης

Την χρονική στιγμή που ένας πελάτης αφικνείται στο σύστημα παραγωγής εξετάζεται αν μπορεί να γίνει δεκτός και επομένως να εξυπηρετηθεί. Για να γίνει ένας πελάτης δεκτός στο σύστημα παραγωγής, θα πρέπει να ικανοποιείται μια από τις ακόλουθες δύο συνθήκες:

- Είτε η ποσότητα που ζητήθηκε να μπορεί να καλυφθεί άμεσα από τον αριθμό των έτοιμων προϊόντων που βρίσκονται στην τελευταία αποθήκη του συστήματος και τα οποία είναι άμεσα προς πώληση.
- Είτε η ποσότητα αυτή να χωράει να εισέλθει στην λίστα αναμονής πελατών. Στην περίπτωση που χωράει εισέρχεται στην τελευταία θέση της λίστας αναμονής και περιμένει να εξυπηρετηθούν πρώτα όλες οι προηγούμενες από αυτή παραγγελίες.

Εάν καμία από τις δύο παραπάνω συνθήκες δεν ισχύει, τότε ο πελάτης αυτός δεν γίνεται δεκτός από το σύστημα και το εγκαταλείπει, αυξάνοντας τον αριθμό των ανικανοποίητων πελατών κατά ένα και τον αριθμό της ανικανοποίητης ζήτησης κατά την ποσότητα την οποία ζήτησε και δεν ικανοποιήθηκε.

3.1.3 Παραγωγή ενός κομματιού από την M_{N-1}

Όταν ένα προϊόν παράγεται από την τελευταία μηχανή της γραμμής παραγωγής του συστήματος ελέγχονται οι ακόλουθες δύο περιπτώσεις:

- Εάν η λίστα αναμονής πελατών είναι άδεια, τότε το κομμάτι αυτό εισέρχεται στην τελευταία αποθήκη του συστήματος και είναι άμεσα έτοιμο για πώληση, όταν αυτή παρουσιαστεί.

- Εάν η λίστα αναμονής πελατών δεν είναι άδεια, τότε το κομμάτι αυτό ικανοποιεί μερικώς την ποσότητα που ζητήθηκε από τον πελάτη που βρίσκεται πρώτος στην λίστα αναμονής. Δηλαδή, συμβαίνει μια εικονική πώληση ενός κομματιού. Ωστόσο, στην πραγματικότητα δεν συμβαίνει καμία πώληση μέχρι την στιγμή που το σύστημα θα παραγάγει όλη την ποσότητα την οποία αρχικά είχε ζητήσει ο πελάτης που βρίσκεται τώρα πρώτος στην λίστα αναμονής και εξυπηρετείτε μερικώς.

3.1.4 Άφιξη μιας παραγγελίας

Προκειμένου να γίνει η άφιξη μιας παραγγελίας πρώτα θα πρέπει να έχει δοθεί η εντολή για παραγγελία. Η διαδικασία που ακολουθείται για το πότε θα δοθεί εντολή για παραγγελία καθώς και η διαδικασία που ακολουθείται για την προσομοίωση της άφιξης της παραγγελίας αυτής στο σύστημα περιγράφονται στις ακόλουθες παραγράφους.

Κάθε φορά που γίνεται μια πώληση από το σύστημα, είτε αν αυτή είναι πραγματική, είτε αν είναι εικονική, μια μεταβλητή, έστω $temp$, αυξάνει την τιμή της κατά την ποσότητα της παραγγελίας η οποία πουλήθηκε. Όταν και μόνο όταν, η τιμή της μεταβλητής αυτής γίνει ίση με τη σταθερή ποσότητα παραγγελίας, έστω Q , τότε δίνεται η εντολή για την παραγγελία αυτής της ποσότητας και η μεταβλητή $temp$ μηδενίζεται, προκειμένου η διαδικασία αυτή να επαναληφθεί.

Αφού έχει δοθεί η εντολή για μια παραγγελία, αυτή εισέρχεται καταρχάς στην αποθήκη L_0 , όπου στην συνέχεια λαμβάνει από την μηχανή M_0 τον αναμενόμενο χρόνο άφιξης της. Όπως αναφέρθηκε και στο προηγούμενο κεφάλαιο, η μηχανή αυτή έχει την δυνατότητα να επεξεργάζεται πολλά κομμάτια ταυτόχρονα. Ωστόσο, ο χρόνος επεξεργασίας για κάθε κομμάτι είναι διαφορετικός. Επομένως, την μηχανή αυτή την εγκαταλείπει πρώτα το κομμάτι το οποίο απαιτεί τον μικρότερο χρόνο επεξεργασίας από την μηχανή. Με αυτό τον τρόπο εξασφαλίζεται ότι στο σύστημα οι παραγγελίες των πρώτων υλών θα φθάνουν βάσει του χρόνου που προσδιορίζει το ποια από το σύνολο των παραγγελιών πρόκειται να αφιχθεί πιο γρήγορα στο σύστημα.

3.1.5 Παραγωγή ενός κομματιού από μια ενδιάμεση μηχανή

Στην περίπτωση που ένα κομμάτι έχει παραχθεί από μια ενδιάμεση μηχανή M_i , θα πρέπει να ελεγχθούν οι ακόλουθες περιπτώσεις:

1. Αρχικά, ελέγχεται η κατάσταση της M_{i+1} μηχανής.
 - a) Εάν το s_{i+1} είναι ίσο με μηδέν, τότε το κομμάτι αυτό εισέρχεται στην μηχανή M_{i+1} και αυτή ενεργοποιείται και αρχίζει την διαδικασία της επεξεργασίας. Επομένως, το s_{i+1} γίνεται ίσο με 1 και το L_{i+1} αυξάνει την τιμή του κατά ένα και από μηδέν γίνεται ίσο με 1. Παράλληλα, υπολογίζεται ο αναμενόμενος χρόνος T_{i+1} , δηλαδή η χρονική στιγμή κατά την οποία η M_{i+1} θα έχει ολοκληρώσει την επεξεργασία αυτού του κομματιού.
 - b) Εάν το L_{i+1} είναι ίσο με την μέγιστη δυνατή χωρητικότητα του, τότε το κομμάτι που έχει παραχθεί από την M_i δεν μπορεί να εισέλθει στην επόμενη αποθήκη και περιμένει στην δικιά του (L_i) έως ότου το L_{i+1} να μειωθεί κατά ένα. Επιπλέον, ο χρόνος παραγωγής επόμενου κομματιού από την M_i γίνεται προσωρινά ίσος με άπειρο, πρακτικά δηλαδή ίσος με τον χρόνο προσομοίωσης του συστήματος συν ένα ($TSIM + 1$) και δεν συνεχίζει στο βήμα 2 που ακολουθεί.
 - c) Εάν ούτε το a) αλλά ούτε και το b) ισχύουν, τότε το L_{i+1} αυξάνει την τιμή του κατά ένα.
2. Στην συνέχεια ελέγχεται η κατάσταση της μηχανής M_{i-1} .
 - a) Εάν το s_{i-1} είναι ίσο με 2, τότε το s_{i-1} γίνεται ίσο με 1 και η μηχανή αυτή ενεργοποιείται (ξεμπλοκάρει), οπότε και διώχνει το κομμάτι που είναι έτοιμο να πάει στην επόμενη αποθήκη, δηλαδή την L_i . Παράλληλα, η χωρητικότητα της L_{i-1} μειώνεται κατά 1, ποσότητα κατά την οποία αυξάνεται η L_i . Ο χρόνος για την αναχώρηση του κομματιού από την M_{i-1} είναι ίσος με τον τρέχοντα χρόνο της προσομοίωσης. Δηλαδή, $T_{i-1} = t$. Τέλος, ο χρόνος παραγωγής ενός νέου κομματιού από την M_i μηχανή γίνεται $T_i = t +$ (τυχαίος χρόνος κατεργασίας).
 - b) Εάν η χωρητικότητα της αποθήκης L_i είναι ίση με μηδέν, τότε το s_i γίνεται ίσο με μηδέν και ο χρόνος παραγωγής ενός νέου κομματιού από αυτή την μηχανή γίνεται προσωρινά ίσος με $TSIM + 1$.
 - c) Τέλος, εάν δεν ισχύει ένα εκ των a) και b), τότε η χωρητικότητα της αποθήκης L_i μειώνεται κατά ένα και υπολογίζεται ο χρόνος

παραγωγής ενός νέου κομματιού από την M_i μηχανή, ο οποίος θα είναι ίσος με $T_i = t + (\text{τυχαίος χρόνος κατεργασίας})$.

Για την δημιουργία τυχαίων χρόνων κατεργασιών χρησιμοποιούνται κατάλληλες γεννήτριες τυχαίων αριθμών [16].

3.2 Αλγόριθμος εύρεσης της βέλτιστης πολιτικής

Σκοπός του αλγορίθμου αυτού είναι η εύρεση της βέλτιστης πολιτικής μέσα από ένα πλήθος δυνατών πολιτικών τις οποίες το συγκεκριμένο σύστημα μπορεί να εφαρμόσει. Οι δυνατές αυτές πολιτικές αυτές αφορούν διάφορους συνδυασμούς των C , Q και I . Το C συμβολίζει το έλλειμμα βάσης, δηλαδή την μέγιστη δυνατή χωρητικότητα, σε ποσότητα παραγγελιών, της λίστας αναμονής πελατών που πρόκειται να εξυπηρετηθούν. Το Q , συμβολίζει τη σταθερή ποσότητα παραγγελίας και το I , το πόσες ποσότητες Q μπορεί να χωρέσει το σύστημα παραγωγής. Στην αρχή, δηλαδή την χρονική στιγμή $t = 0$, το γινόμενο $I \cdot Q$ αποτελεί το αρχικό απόθεμα του συστήματος.

3.2.1 Συνάρτηση κέρδους

Η βέλτιστη πολιτική θα είναι αυτή η οποία μεγιστοποιεί την συνάρτηση κέρδους του συγκεκριμένου συστήματος παραγωγής. Επομένως, προτού αναπτυχθεί ο τρόπος υλοποίησης του παραπάνω αλγορίθμου περιγράφεται η διαδικασία υπολογισμού της συνάρτησης κέρδους του συστήματος. Το κέρδος/ζημία για το συγκεκριμένο σύστημα παραγωγής υπολογίζεται από την κάτωθι συνάρτηση:

$$\text{Κέρδος} = \frac{[(\text{Πωλήσεις} \times P) - (NUC \times D) - [(I + NQA) \times A] - [(\bar{L} + L_{\text{ελλος}} \times TSIM) \times h] - [(\bar{B} + B_{\text{ελλος}} \times TSIM) \times b]]}{TSIM}$$

Συμβολισμοί:

Πωλήσεις	Ο συνολικός αριθμός κομματιών τα οποία πουλήθηκαν από το σύστημα κατά την διάρκεια του χρόνου προσομοίωσης $TSIM$
$TSIM$	Ο χρόνος που διαρκεί η προσομοίωση του συστήματος
NUC	Το εμβαδόν του πλήθους ανικανοποίητων πελατών του συστήματος στο διάστημα $[0, TSIM]$
NQA	Το πόσες φορές η M_0 έφερε Q ποσότητες στο σύστημα παραγωγής κατά την διάρκεια του χρόνου προσομοίωσης

$L_{\pi\lambda o\varsigma}$	Το συνολικό απόθεμα το οποίο βρίσκεται στο σύστημα παραγωγής μετά το τέλος της προσομοίωσης
$B_{\pi\lambda o\varsigma}$	Η συνολική ποσότητα των παραγγελιών που έχουν απομείνει στην λίστα αναμονής των πελατών μετά το τέλος της προσομοίωσης
$\bar{L}$	Το εμβαδόν του αποθέματος του συστήματος στο διάστημα $[0, TSIM]$
$\bar{B}$	Το εμβαδόν του πλήθους πελατών που βρίσκονται στην λίστα αναμονής στο διάστημα $[0, TSIM]$
P	Το καθαρό κέρδος/μονάδα προϊόντος που πουλήθηκε.
A	Το σταθερό κόστος φορτωτικής, το οποίο είναι ανεξάρτητο από την ποσότητα Q
h	Το κόστος αποθήκευσης ενός κομματιού
b	Το κόστος της καθυστερημένης ικανοποίησης της ζήτησης
D	Το κόστος δυσφήμισης/ανικανοποίητο πελάτη

Στην συνέχεια παρουσιάζεται ο τρόπος υπολογισμού των $\bar{L}$ και $\bar{B}$.

- Στην αρχή της προσομοίωσης $\bar{L}=0$ και $\bar{B}=0$.
- Αν κάποια χρονική στιγμή t έρθει μια ποσότητα Q από πρώτες ύλες στο σύστημα, τότε: $\bar{L}=\bar{L}+Q \cdot t$.
- Αν κάποια χρονική στιγμή t η M_N (ζήτηση) φέρει έναν πελάτη ο οποίος ζητάει μια ποσότητα, έστω $order$, και η ποσότητα αυτή μπορεί να ικανοποιηθεί άμεσα, τότε: $\bar{L}=\bar{L}+order \cdot t$ και $\bar{B}=\bar{B}-order \cdot t$.
- Αν κάποια χρονική στιγμή t η M_{N-1} φέρει ένα έτοιμο προϊόν και με αυτό το κομμάτι ο πελάτης που βρίσκεται πρώτος στην λίστα αναμονής εξυπηρετηθεί πλήρως, δηλαδή με όλη την ποσότητα ($order$) την οποία είχε ζητήσει την χρονική στιγμή που μπήκε στην λίστα αναμονής, τότε: $\bar{L}=\bar{L}+order \cdot t$ και $\bar{B}=\bar{B}-order \cdot t$.

Το συνολικό απόθεμα ($L_{\pi\lambda o\varsigma}$) το οποίο βρίσκεται στο σύστημα παραγωγής μετά το τέλος της προσομοίωσης, υπολογίζεται με την βοήθεια του κάτωθι τύπου:

$$L_{\pi\lambda o\varsigma} = \sum_{i=1}^{i=N} L_i$$

3.2.2 Στατιστικά συνάρτησης κέρδους

Ο αλγόριθμος εύρεσης της βέλτιστης πολιτικής υπολογίζει το κέρδος/ζημία για κάθε πιθανό συνδυασμό (C, Q, I) και στην συνέχεια λαμβάνει ως βέλτιστη πολιτική του συστήματος αυτή που παρουσιάζεται να έχει το μεγαλύτερο κέρδος. Ουσιαστικά, εκτελεί έναν μεγάλο αριθμό προσομοιώσεων χρησιμοποιώντας κάθε φορά τον αλγόριθμο προσομοίωσης που αναπτύχθηκε παραπάνω. Για κάθε συνδυασμό (C, Q, I) εκτελούνται συνολικά 10 προσομοιώσεις με διαφορετικούς τυχαίους αριθμούς. Ωστόσο, όλοι οι πιθανοί συνδυασμοί (C, Q, I) χρησιμοποιούν κοινούς τυχαίους αριθμούς. Δηλαδή η πρώτη προσομοίωση του (C, Q, I) χρησιμοποιεί ίδιους τυχαίους αριθμούς με την πρώτη του (C', Q', I') , κ.ο.κ.

Για κάθε πιθανό συνδυασμό (C, Q, I) υπολογίζεται το μέσο κέρδος που προκύπτει από τις 10 προσομοιώσεις. Οπότε στο σημείο αυτό είναι σημαντικό να αναπτυχθεί η διαδικασία υπολογισμού των διαστημάτων εμπιστοσύνης του μέσου κέρδους.

Κάθε μια από τις 10 προσομοιώσεις δίνει μια τιμή Κέρδος_i , βάσει της συνάρτησης κέρδους που αναπτύχθηκε παραπάνω. Στην συνέχεια, υπολογίζεται το μέσο κέρδος από τις 10 προσομοιώσεις, σύμφωνα με την ακόλουθη σχέση:

$$\overline{\text{Κέρδος}} = \sum_{i=1}^{10} \text{Κέρδος}_i$$

Ακολούθως υπολογίζεται η διασπορά του κέρδους.

$$S^2 = \frac{\sum_{i=1}^{10} (\text{Κέρδος}_i - \overline{\text{Κέρδος}})^2}{9}$$

Τέλος, με πιθανότητα εμπιστοσύνης $1 - \alpha = 95\%$ υπολογίζεται η ακτίνα του διαστήματος στατιστικού λάθους. Ο τύπος υπολογισμού της ποσότητας αυτής είναι ο ακόλουθος:

$$\Sigma\phi\acute{\alpha}\lambda\mu\alpha = Z_{\alpha/2} \times \sqrt{S^2 / 10}$$

όπου $Z_{\alpha/2}$ είναι το σημείο της τυποποιημένης κανονικής κατανομής δεξιά από το οποίο υπάρχει μόλις το 2.5% των τιμών της κατανομής [16]. Συνεπώς το διάστημα εμπιστοσύνης για το μέσο κέρδος είναι $\overline{\text{Κέρδος}} \pm \Sigma\phi\acute{\alpha}\lambda\mu\alpha$.

4. Αριθμητικά Αποτελέσματα

Μετά την περιγραφή του συγκεκριμένου συστήματος παραγωγής και την ανάλυση της λειτουργίας των αλγορίθμων προσομοίωσης και εύρεσης της βέλτιστης πολιτικής, διερευνώνται κάποια από τα αποτελέσματα τα οποία έδωσαν οι παραπάνω δύο αλγόριθμοι, κάτω από διαφορετικές συνθήκες και για διαφορετικά συστήματα παραγωγής. Τα αποτελέσματα αυτά καθώς και η ανάλυση τους παρουσιάζονται στην συνέχεια.

4.1 Σύστημα $M/M/1/S+C$

Πρώτα ελέγχουμε την ακρίβεια του αλγορίθμου προσομοίωσης συγκρίνοντας τον με ένα μοντέλο για συστήματα μιας μηχανής το οποίο επιδέχεται αναλυτικής επίλυσης.

Ένα σύστημα $M/M/1/S+C$ είναι ένα σύστημα στο οποίο η διαδικασία αφίξεων είναι Poisson, οι χρόνοι εξυπηρέτησης ακολουθούν εκθετική κατανομή, έχει έναν εξυπηρετητή και ο συνολικός αριθμός κομματιών που βρίσκονται εντός του συστήματος δεν μπορεί να είναι μεγαλύτερος από την τιμή $S+C$. Στο σύστημα αυτό οι χρόνοι άφιξης πρώτων υλών είναι ίσοι με 0 και οι παραγγελίες πρώτων υλών γίνονται κατά παρτίδες $Q=1$.

Προκειμένου να προσομοιώσουμε το συγκεκριμένο σύστημα θεωρούμε ότι το σύστημα αυτό αποτελείται από τρεις μηχανές ($N=2$). Η μηδενική μηχανή του συστήματος, M_0 , αποτελεί την μηχανή που θα φέρνει πρώτες ύλες στο σύστημα. Ο χρόνος επεξεργασίας της μηχανής αυτής ακολουθεί την ομοιόμορφη κατανομή με τους συντελεστές α_0 και β_0 να είναι ίσοι με μηδέν. Δηλαδή, την χρονική στιγμή που δίνεται εντολή για μια παραγγελία αυτή φθάνει στο σύστημα ακαριαία. Η μηχανή M_1 , αποτελεί την μηχανή που επεξεργάζεται τις πρώτες ύλες και έχει την δυνατότητα να επεξεργάζεται ένα κομμάτι την φορά. Παράλληλα, ο χρόνος κατεργασίας αυτού του κομματιού ακολουθεί την εκθετική κατανομή με συντελεστή $\lambda = 1$. Τέλος, υπάρχει και η μηχανή M_2 η οποία προσομοιώνει την ζήτηση. Ο χρόνος άφιξης ενός πελάτη ακολουθεί την εκθετική κατανομή με συντελεστή $\mu = 1$. Κάθε φορά είναι δυνατόν να έρθει μόνο ένας πελάτης. Όσον αφορά τις χωρητικότητες των αποθηκών

του συστήματος αυτές έχουν άπειρη χωρητικότητα, ενώ η μέγιστη δυνατή χωρητικότητα της λίστας αναμονής πελατών είναι ίση με C . Τέλος, ως S για το σύστημα αυτό θεωρείται το άθροισμα της ποσότητας των πρώτων υλών, της ποσότητας των έτοιμων προϊόντων καθώς και του κομματιού που μπορεί να βρίσκεται εντός της μηχανής και να επεξεργάζεται.

Το σύστημα αυτό λόγω της απλοϊκότητας του μπορεί να αναλυθεί χρησιμοποιώντας συναρτήσεις του EXCEL. Προκειμένου όμως να γίνει αυτό θα πρέπει πρώτα να υπολογισθούν όλες οι πιθανότητες των πιθανών καταστάσεων στις οποίες το συγκεκριμένο σύστημα μπορεί να μεταβεί. Ο πίνακας των πιθανών καταστάσεων καθώς και των πιθανοτήτων εμφάνισής τους παρουσιάζεται στην συνέχεια στον Πίνακα 1.

Πίνακας 1. Οι καταστάσεις του συστήματος και οι πιθανότητες εμφάνισής τους

Καταστάσεις	Πρώτες Ύλες	Έτοιμα Προϊόντα	Πελάτες στην Λίστα Αναμονής	Πιθανότητα Εμφάνισης
0	0	S	0	P_0
1	1	S-1	0	$\rho \times P_0$
2	2	S-2	0	$\rho^2 \times P_0$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
n	n	S-n	0	$\rho^n \times P_0$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
S-1	S-1	1	0	$\rho^{S-1} \times P_0$
S	S	0	0	$\rho^S \times P_0$
S+1	S	0	1	$\rho^{S+1} \times P_0$
S+2	S	0	2	$\rho^{S+2} \times P_0$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
n	S	0	n-S	$\rho^n \times P_0$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
S+C	S	0	C	$\rho^{S+C} \times P_0$

$$\text{όπου, } \rho = \frac{\lambda}{\mu} \text{ και } P_0 = \begin{cases} \frac{1}{S+C+1}, \rho = 1 \\ \frac{1-\rho}{1-\rho^{S+C+1}}, \rho \neq 1 \end{cases}$$

Για τον υπολογισμό της συνάρτησης κέρδους αυτού του συστήματος χρειάζεται να υπολογιστούν τα ακόλουθα:

1. Το μέσο πλήθος κομματιών στο σύστημα. Το πλήθος αυτό είναι ίσο με το S του συστήματος.

2. Μέσο πλήθος ανικανοποιήτων παραγγελιών. Το μέγεθος αυτό συμβολίζεται ως B και υπολογίζεται με την βοήθεια της ακόλουθης σχέσης:

$$B = \begin{cases} \frac{C(C+1)}{2(S+C+1)}, \rho = 1 \\ P_0 \rho^{S+1} \frac{d}{d\rho} \left(\frac{1-\rho^{C+1}}{1-\rho} \right), \rho \neq 1 \end{cases}$$

3. Το μέσο ρυθμό παραγωγής κομματιών (TH) από το σύστημα, ο οποίος ισούται με: το μέσο ρυθμό πωλήσεων και υπολογίζεται από την σχέση:

$$TH = \mu(1 - P_0) = \lambda(1 - P_{S+C})$$

Οι παράμετροι αυτοί καθώς και οι παράμετροι κέρδους και κόστους του συστήματος διαμορφώνουν την ακόλουθη συνάρτηση κέρδους για το συγκεκριμένο σύστημα παραγωγής η οποία και παρουσιάζεται στον κάτωθι τύπο.

$$\text{Κέρδος} = TH \cdot P - (S \cdot h) - (B \cdot b) - (TH \cdot A) - (\lambda \cdot P_{S+C} \cdot D),$$

όπου, P είναι το καθαρό κέρδος πώλησης ενός έτοιμου κομματιού από το σύστημα, h είναι το κόστος αποθήκευσης ενός κομματιού από το σύστημα, b το κόστος που σχετίζεται με μη ικανοποίηση μιας παραγγελίας, A το σταθερό κόστος φορτωτικής για την προμήθεια μιας μονάδας πρώτης ύλης και τέλος το D είναι το κόστος δυσφήμισης από έναν ανικανοποίητο πελάτη.

Οι τιμές που λαμβάνουν οι παραπάνω παράμετροι κέρδους και κόστους είναι ενδεικτικές και συνοψίζονται στον Πίνακα 2.

Πίνακας 2. Τιμές παραμέτρων της συνάρτησης κέρδους

P	h	b	A	D
100	10	10	10	300

Στην συνέχεια ακολουθούν δύο συγκριτικοί πίνακες μεταξύ των αποτελεσμάτων που προκύπτουν από το φύλλο EXCEL και αυτών που προέκυψαν από τον αλγόριθμο προσομοίωσης του συστήματος παραγωγής για ενδεικτικές τιμές (C , S). Για την εξαγωγή των παρακάτω δεδομένων χρησιμοποιήθηκε $TSIM = 100000$ μονάδες χρόνου, $\lambda = \mu = 1$ για τον Πίνακα 3 και $\mu \neq \lambda$ ($\lambda = 0.8$ και $\mu = 1$) για τον Πίνακα 4.

Πίνακας 3. Αποτελέσματα προσομοίωσης για $\lambda = \mu = 1$

(C,S)	EXCEL	ΚΩΔΙΚΑΣ	(+)	(-)
(1,1)	-53.333	-53.271	-52.589	-53.953
(1,2)	-30.000	-30.055	-29.483	-30.619
(1,3)	-20.000	-20.035	-19.661	-20.409
(1,4)	-16.667	-16.694	-16.099	-17.288
(1,5)	-17.142	-17.109	-16.631	-17.588
(1,6)	-20.000	-19.840	-19.235	-20.446
(1,7)	-24.444	-24.484	-23.947	-25.021
(1,8)	-30.000	-29.921	-29.516	-30.325
(2,1)	-25.000	-25.046	-24.464	-25.628
(3,1)	-10.000	-10.033	-9.623	-10.443
(4,1)	-1.667	-1.684	-1.072	-2.297
(5,1)	2.855	2.902	3.460	2.345
(6,1)	5.000	5.177	5.902	4.452
(7,1)	5.555	5.495	6.217	4.773
(8,1)	5.000	5.030	5.852	4.207
(9,1)	3.636	3.605	4.827	2.382
(2,2)	-14.000	-14.035	-13.643	-14.426
(2,3)	-10.000	-10.025	-9.411	-10.639
(3,2)	-5.000	-5.020	-4.385	-5.654
(5,3)	0.000	-0.062	0.618	-0.743
(3,4)	-6.250	-6.075	-5.424	-6.726
(6,2)	3.333	3.754	3.977	2.572
(3,7)	-20.909	-20.930	-19.972	-21.888
(5,8)	-28,571	-28,587	-27,849	-29,324

Πίνακας 4. Αποτελέσματα προσομοίωσης για $\mu \neq \lambda$ ($\lambda = 0.8$ και $\mu = 1$)

(C,S)	EXCEL	ΚΩΔΙΚΑΣ	(+)	(-)
(1,1)	-22.459	-22.492	-21.669	-23.314
(1,2)	-4.281	-4.247	-3.613	-4.882
(1,3)	2.079	2.0498	2.459	1.639
(1,4)	2.553	2.573	3.276	1.871
(1,5)	-0.318	-0.278	0.446	-1.003
(1,6)	-5.262	-5.241	-4.849	-5.632
(1,7)	-11.570	-11.617	-11.004	-12.229
(1,8)	-18.816	-18.878	-18.196	-19.560
(2,1)	2.249	2.307	2.902	1.712
(3,1)	15.378	15.403	16.141	14.667
(4,1)	22.894	22.912	23.593	22.232
(5,1)	27.346	27.267	28.225	26.309
(6,1)	30.000	29.825	30.476	29.174
(7,1)	31.550	31.612	32.835	30.390
(8,1)	32.405	32.441	33.046	31.835
(9,1)	33.946	33.787	34.359	33.216
(10,1)	33.837	33.895	34.373	33.416
(11,1)	33.606	33.429	34.024	32.834
(12,1)	32.760	32.785	33.159	32.411
(13,1)	32.990	32.831	33.222	32.439
(2,2)	9.033	9.079	9.558	8.601
(2,3)	9.777	9.796	10.169	9.422
(3,2)	16.447	16.474	17.011	15.937
(5,3)	19.030	18.852	19.257	18.448
(3,4)	11.742	11.778	12.392	11.164
(6,2)	26.123	26.183	26.905	25.461
(3,7)	-10.531	-10.500	-9.942	-11.057
(5,8)	-22.572	-22.516	-21.935	-23.098

Με (+) και (-) συμβολίζεται η μέγιστη και η ελάχιστη τιμή της συνάρτησης κέρδους, λόγω της διασποράς.

4.1.1 Παρατηρήσεις-Συμπεράσματα

Από την παρατήρηση των αποτελεσμάτων των δύο παραπάνω συγκριτικών πινάκων προέκυψαν τα ακόλουθα συμπεράσματα:

Διαπιστώνεται η εγκυρότητα του αλγορίθμου προσομοίωσης που αναπτύχθηκε στα πλαίσια της παρούσας εργασίας. Αυτό προκύπτει από την συμφωνία μεταξύ των αποτελεσμάτων που προκύπτουν από το φύλλο EXCEL με τα αντίστοιχα που εξάγονται από τον αλγόριθμο προσομοίωσης. Επιλέον παρατηρείται ότι τα αποτελέσματα που προκύπτουν από τον αλγόριθμο προσομοίωσης παρουσιάζουν μικρή διασπορά από την μέση τιμή του κέρδους.

Και στις δυο περιπτώσεις ($\mu = \lambda$ και $\mu \neq \lambda$) παρατηρείται ότι όταν το C είναι σταθερό και ίσο με 1 και το S του συστήματος αυξάνεται, τότε το κέρδος αυξάνει συνεχώς έως ότου αποκτήσει μια μέγιστη τιμή για κάποιο συγκεκριμένο S . Στην συνέχεια καθώς το S αυξάνεται το κέρδος αρχίζει να μειώνεται. Αντίστοιχα αποτελέσματα παρατηρούνται και στην περίπτωση που το S παραμένει σταθερό και ίσο με 1 και το C αυξάνεται συνεχώς. Το γεγονός αυτό αποτελεί μια ένδειξη για πιθανή ύπαρξη ολικού ακρότατου το οποίο προκύπτει για κάποιο συνδυασμό των C και S και μεγιστοποιεί την συνάρτηση κέρδους του συστήματος παραγωγής. Επομένως, είναι πολύ πιθανό στο συγκεκριμένο σύστημα παραγωγής να πρέπει να εφαρμοστεί μια πολιτική τύπου κατωφλιού.

Η τιμή του κέρδους εκτός από τους παράγοντες κόστους επηρεάζεται επίσης και από τον μέσο ρυθμό της άφιξης πελατών (μ) καθώς και από τον μέσο ρυθμό άφιξης των πρώτων υλών (λ) στο σύστημα. Στην συγκεκριμένη περίπτωση και για τις συγκεκριμένες τιμές των παραμέτρων κόστους, το σύστημα με $\mu > \lambda$ παρουσιάζεται να είναι πιο κερδοφόρο από το αντίστοιχο σύστημα με $\mu = \lambda$.

Ένα από τα βασικά πλεονέκτημα του αλγορίθμου προσομοίωσης που αναπτύχθηκε στην παρούσα εργασία είναι η δυνατότητα που έχει να βρίσκει και να εμφανίζει την βέλτιστη πολιτική (C, S) που θα πρέπει να εφαρμοστεί προκειμένου να μεγιστοποιηθεί η συνάρτηση κέρδους του συστήματος. Για τις παραπάνω δύο περιπτώσεις ($\mu = \lambda$ και $\mu \neq \lambda$) για το σύστημα M/M/1/C+S οι βέλτιστες πολιτικές είναι οι ακόλουθες:

- Για $\mu = \lambda$: $C = 7$, $I = 1$ η οποία αποφέρει κέρδος 5.495 χρηματικών μονάδων.
- Για $\mu \neq \lambda$: $C = 9$, $I = 1$ η οποία αποφέρει κέρδος 33.787 χρηματικών μονάδων.

Η εύρεση των βέλτιστων πολιτικών για τις συγκεκριμένες δυο περιπτώσεις μπορεί επίσης να προκύψει παρατηρώντας τα αποτελέσματα των παραπάνω πινάκων. Και σε αυτή την περίπτωση τα αποτελέσματα που προκύπτουν είναι ίδια με αυτά που παρουσίασε ο αλγόριθμος εύρεσης της βέλτιστης πολιτικής του συστήματος παραγωγής.

4.2 Γραμμή παραγωγής με σύστημα CONWIP

Το σύστημα που μελετήθηκε στην προηγούμενη ενότητα αφορούσε την πλέον απλοϊκή περίπτωση συστήματος παραγωγής. Ένα σύνηδες σύστημα παραγωγής αποτελείται εκτός των άλλων και από μια γραμμή παραγωγής, στην οποία και λαμβάνει χώρα η παραγωγική διαδικασία του συστήματος.

Στην ενότητα αυτή αναπτύσσεται και μελετάται ένα σύστημα CONWIP με γραμμή παραγωγής. Πρόκειται για ένα σύστημα παραγωγής το οποίο αποτελείται από 5 μηχανές ($N=4$). Οι τρεις είναι οι πραγματικές και αποτελούν την γραμμή παραγωγής του συστήματος και οι άλλες δύο είναι εικονικές και προσομοιώνουν την διαδικασία άφιξης των πρώτων υλών στο σύστημα και την ζήτηση. Το σύστημα αποτελείται επίσης από ένα ισάριθμο πλήθος ενδιάμεσων αποθηκών των οποίων οι χωρητικότητες έχουν θεωρηθεί ότι είναι αρκετά μεγάλες ούτως ώστε να μην συμβεί να γεμίσουν.

Στο σύστημα αυτό εφαρμόζεται η πολιτική της μερικής αποδοχής της μη ικανοποιημένης ζήτησης (PLS). Επομένως, το συγκεκριμένο σύστημα διαθέτει μια λίστα αναμονής πελατών μέγιστης χωρητικότητας C . Μια σημαντική παράμετρος του συστήματος είναι ότι θεωρεί ότι είναι δυνατό να έρχεται μόνο ένας πελάτης την φορά, ο οποίος και θα ζητάει ποσότητα ίση με 1 κομμάτι. Εντολή για παραγγελιά πρώτων υλών δίνεται μόνο στην περίπτωση που πραγματοποιηθούν Q πωλήσεις προϊόντων από το σύστημα. Στην περίπτωση αυτή δίνεται άμεση εντολή για την αναπλήρωση αυτής της ποσότητας Q που μόλις πουλήθηκε. Τέλος, το σύστημα προκειμένου να ξεκινήσει να παράγει προϊόντα έχει αρχικό απόθεμα ίσο με $I \cdot Q$ το οποίο και βρίσκεται στην αποθήκη L_1 της μηχανής M_1 .

Σε αντίθεση με το σύστημα $M/M/1/S+C$, το συγκεκριμένο σύστημα παραγωγής δεν μπορεί να αναλυθεί χρησιμοποιώντας συναρτήσεις του EXCEL. Ο τρόπος μέσω του οποίου θα επιτευχθεί η προσομοίωση του συγκεκριμένου συστήματος είναι με την χρήση του αλγορίθμου προσομοίωσης που αναπτύχθηκε στην παρούσα εργασία.

Σκοπός της ενότητας αυτής είναι η εξεύρεση, για συγκεκριμένα δεδομένα, της βέλτιστης λύσης του συγκεκριμένου συστήματος παραγωγής και στην συνέχεια η σύγκρισή της με τις βέλτιστες λύσεις του αντίστοιχου συστήματος όταν σε αυτό

εφαρμόζονται οι πολιτικές της πλήρους αποδοχής της μη ικανοποιημένης ζήτησης (CB) και της πλήρους απόρριψης της μη ικανοποιημένης ζήτησης (LS).

Το σύστημα προσομοιώθηκε για τα δεδομένα του Πίνακα 5:

Πίνακας 5. Τιμές παραμέτρων της συνάρτησης κέρδους

P	h	b	A	D
100	10	10	100	300

Παράλληλα, θεωρήθηκε ότι οι χρόνοι κατεργασίας ενός κομματιού για κάθε μηχανή του συστήματος παραγωγής ακολουθούν την εκθετική κατανομή με μέση τιμή ίση με 10. Επομένως, το σύστημα προσομοιώθηκε για την περίπτωση $\mu = \lambda$.

Ο τρόπος με τον οποίο υλοποιήθηκε η προσομοίωση περιγράφεται στα ακόλουθα τρία βήματα:

- Για $0 \leq C \leq 20$, $1 \leq Q \leq 20$, $0 \leq I \leq 5$ και για $TSIM = 10000$ μονάδες χρόνου πραγματοποιήθηκε η πρώτη προσομοίωση και υπολογίστηκε η βέλτιστη πολιτική (C_1, Q_1, I_1) του συστήματος.
- Στην συνέχεια, προσομοιώνονται οι πολιτικές (C, Q, I) για $C_1 - 2 \leq C \leq C_1 + 2$, $Q_1 - 2 \leq Q \leq Q_1 + 2$, $I_1 - 1 \leq I \leq I_1 + 1$ και για $TSIM = 100000$ μονάδες χρόνου υπολογίζεται η νέα βέλτιστη πολιτική (C_2, Q_2, I_2) .
- Τέλος, για (C, Q, I) τέτοια ώστε $C_2 - 1 \leq C \leq C_2 + 1$, $Q_2 - 1 \leq Q \leq Q_2 + 1$, $I_2 - 1 \leq I \leq I_2 + 1$ και για $TSIM = 1000000$ μονάδες χρόνου υπολογίζεται η τελική και βέλτιστη λύση (C_3, Q_3, I_3) του συγκεκριμένου συστήματος παραγωγής.

Ο συνολικός χρόνος που απαιτήθηκε για την προσομοίωση του συγκεκριμένου συστήματος παραγωγής ανήλθε στις 4 ώρες και πραγματοποιήθηκε σε υπολογιστή με επεξεργαστή Intel Cetrino Duo στα 1,99GHz.

Στην συνέχεια παρατίθεται ο Πίνακας 6 με τις βέλτιστες πολιτικές που προέκυψαν από την διαδικασία των τριών συνεχών προσομοιώσεων του συστήματος.

Πίνακας 6. Βέλτιστες πολιτικές

	<i>TSIM=10.000</i>	<i>TSIM=100.000</i>	<i>TSIM=1.000.000</i>
(C, Q, I)	(8, 13, 3)	(7, 13, 3)	(7, 13, 3)
Μέσο Κέρδος	262.961	263.014	263.832
Max Κέρδος	268.414	264.862	264.352
Min Κέρδος	257.507	261.165	263.312
Απόκλιση	5.453	1.848	0.519

Από τον παραπάνω πίνακα προκύπτει ότι η βέλτιστη πολιτική για το συγκεκριμένο σύστημα είναι η $(C, Q, I) = (7, 13, 3)$ η οποία και αποφέρει το μέγιστο δυνατό μέσο κέρδος, το οποίο είναι ίσο με 263.832 χρηματικές μονάδες.

Σε αυτό το σημείο θα πρέπει να τονιστεί ότι, σταματήσαμε να κάνουμε προσομοιώσεις για χρόνο προσομοίωσης $TSIM=1000000$ μονάδες χρόνου για τον εξής ακόλουθο λόγο. Για το συγκεκριμένο $TSIM$, η αμέσως χειρότερη πολιτική που προέκυψε από τον αλγόριθμο προσομοίωσης είχε μέγιστο δυνατό κέρδος ίσο με 262.853 χρηματικές μονάδες, το οποίο σε κάθε περίπτωση είναι μικρότερο της ελάχιστης δυνατής τιμής της βέλτιστης λύσης του συστήματος και η οποία είναι ίση με 263.312 χρηματικές μονάδες. Επομένως, φαίνεται ότι για τα συγκεκριμένα δεδομένα η βέλτιστη αυτή λύση αποτελεί την ολική βέλτιστη λύση του συστήματος παραγωγής. Σε διαφορετική περίπτωση, μη ύπαρξης ολικού βέλτιστου, θα έπρεπε να αυξηθεί ο χρόνος προσομοίωσης και να ξαναεπαναληφθεί η όλη διαδικασία, προκειμένου να μειωθεί περαιτέρω η απόκλιση γύρω από την τιμή του μέσου κέρδους, ούτως ώστε τα αποτελέσματα τα οποία θα προκύψουν να είναι πιο αξιόπιστα.

Στην συνέχεια θα γίνει μια σύγκριση των βέλτιστων συνδυασμών (C, Q, I) που μεγιστοποιούν την συνάρτηση κέρδους του συστήματος για τις τρεις πολιτικές αποδοχής πελατών (PLS, LS και CB) που μπορούν να εφαρμοστούν σε ένα σύστημα παραγωγής. Ωστόσο, προκειμένου αυτό να συμβεί θα πρέπει το σύστημα να προσομοιωθεί για $\mu > \lambda$, συνθήκη η οποία αποτελεί βασική προϋπόθεση για την εφαρμογή της πολιτικής CB.

Στην πολιτική της πλήρους απόρριψης της μη ικανοποιημένης ζήτησης (lost sales) αν το σύστημα δεν μπορεί να ικανοποιήσει άμεσα έναν πελάτη, τότε τον χάνει. Δηλαδή, το C του συστήματος παραγωγής είναι ίσο με μηδέν. Αντίθετα, στην πολιτική της πλήρους αποδοχής της μη ικανοποιημένης ζήτησης (complete backordering) το C του συστήματος θεωρείται ότι είναι άπειρο. Στην

πραγματικότητα, είναι αρκετά μεγάλο ούτως ώστε να κάνει δεκτούς όλους τους πελάτες που φθάνουν στο σύστημα παραγωγής.

Οι βέλτιστες στρατηγικές καθώς και οι βέλτιστες τιμές του μέσου κέρδους για κάθε μια από τις παραπάνω πολιτικές για $\mu > \lambda$ ($\mu = 10$ και $\lambda = 9$) παρουσιάζονται στον ακόλουθο συγκριτικό Πίνακα 7.

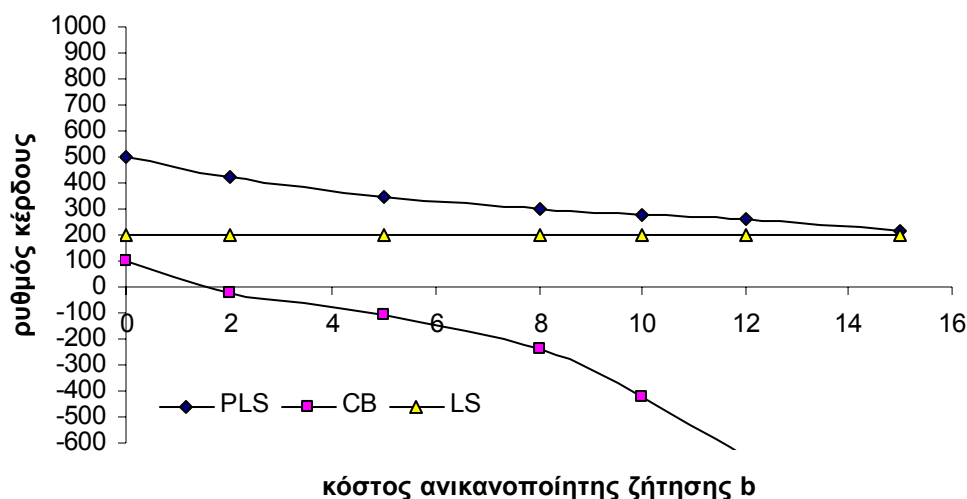
Πίνακας 7. Συγκριτικά Αποτελέσματα

	LS	PLS	CB
Μέσο Κέρδος	201.952	275.553	-420.238
(C, Q, I)	(0, 12, 2)	(8, 13, 3)	(100, 8, 2)

Από τον πίνακα αυτό παρατηρούμε ότι η καλύτερη πολιτική βάσει του μέσου κέρδους είναι αυτή της μερικής αποδοχής της μη ικανοποιημένης ζήτησης (PLS). Ωστόσο, αυτό και μόνο το κριτήριο δεν αρκεί προκειμένου να θεωρήσουμε την πολιτική PLS ως την πλέον καλύτερη και αποτελεσματικότερη έναντι των πολιτικών LS και CB.

Ένα πολύ σημαντικό κριτήριο το οποίο πρέπει να ληφθεί υπόψη είναι η ευαισθησία που παρουσιάζουν οι τρεις αυτές πολιτικές σε μεταβολές των παραμέτρων οι οποίες την επηρεάζουν. Παραδείγματα τέτοιων παραμέτρων είναι ο ρυθμός ζήτησης μ , το κόστος αποθέματος h , το κόστος ανικανοποίητης ζήτησης b κ.α. Στην συγκεκριμένη εργασία, θα γίνει μια ανάλυση ευαισθησίας των τριών πολιτικών αποδοχής πελατών σε ένα σύστημα παραγωγής, συναρτήσει της τιμής του κόστους ανικανοποίητης ζήτησης b .

Για διάφορες τιμές της μεταβλητής b και για κάθε μια από τις τρεις πολιτικές εκτελέσαμε τον αλγόριθμο προσομοίωσης με σκοπό να διαπιστώσουμε ποια είναι η βέλτιστη πολιτική και το αντίστοιχο βέλτιστο κέρδος του συστήματος παραγωγής για τις διάφορες τιμές του b . Τα αποτελέσματα από την συγκεκριμένη μελέτη συνοψίζονται στο Σχήμα 2 το οποίο παρουσιάζεται στην συνέχεια.



Σχήμα 2. Ρυθμός κέρδους –Κόστος ανικανοποίητης ζήτησης b

4.2.1 Παρατηρήσεις-Συμπεράσματα

Από το παραπάνω σχήμα διαπιστώνουμε ότι η πλέον ευαίσθητη από τις τρεις πολιτικές σε τυχόν μεταβολές της τιμής του κόστους b , είναι η πολιτική CB. Το γεγονός αυτό ήταν αναμενόμενο καθώς η πολιτική αυτή δέχεται όλους τους πελάτες που επισκέπτονται το σύστημά της. Επομένως, ο αριθμός των πελατών στην λίστα αναμονής θα είναι αισθητά μεγαλύτερος από τον αντίστοιχο που θα υπάρχει σε συστήματα που εφαρμόζεται η πολιτική PLS. Ως εκ τούτου, μια πιθανή αύξηση του κόστους ύπαρξης πελατών στην λίστα αναμονής θα έχει ως αποτέλεσμα την ραγδαία μείωση του ρυθμού κέρδους της.

Επίσης, παρατηρούμε και γραφικά πλέον ότι οι πολιτικές CB και LS είναι δυο εκ διαμέτρου αντίθετες πολιτικές. Συγκεκριμένα, η πολιτική LS παραμένει αμετάβλητη στις όποιες μεταβολές της τιμής b . Το γεγονός αυτό οφείλεται στην μη ύπαρξη λίστας αναμονής πελατών σε συστήματα που εφαρμόζουν την πολιτική LS.

Όσον αφορά την πολιτική PLS αυτή φαίνεται να μειώνεται με εκθετικό τρόπο και καθώς το b μεγαλώνει αυτή τείνει στην πολιτική LS. Το γεγονός αυτό οφείλεται στο ότι καθώς η τιμή του b αυξάνεται ο ρυθμός του κέρδους συνεχώς και μειώνεται και κάνει την ύπαρξη της λίστας αναμονής πελατών μια υπηρεσία με πολύ υψηλό κόστος λειτουργίας. Επομένως, το σύστημα ωθείται, καθώς το συμφέρει οικονομικά, στο να δέχεται όλο και λιγότερους πελάτες στην λίστα αυτή, μειώνοντας σταδιακά την χωρητικότητά της, η οποία καθώς το b τείνει στο άπειρο αυτή τείνει στο μηδέν.

Τέλος, από το παραπάνω γίνεται ακόμα πιο σαφές ότι η πολιτική PLS αποτελεί έναν συνδυασμό των δυο ακραίων πολιτικών CB και LS και ότι σε καμία περίπτωση δεν μπορεί να οδηγήσει σε αποτελέσματα χειρότερα από αυτά που θα προέκυπταν εάν εφαρμοζόταν μεμονωμένα κάποια από τις πολιτικές CB και LS, τόσο σε επίπεδο ευαισθησίας όσο και σε επίπεδο κερδοφορίας.

5. Συμπεράσματα και Επεκτάσεις

Στην παρούσα εργασία μελετήθηκε ένα σύστημα παραγωγής στο οποίο εφαρμόζεται ένας συνδυασμός της πολιτικής διαχείρισης αποθεμάτων τύπου CONWIP, της πολιτικής της μερικής αποδοχής της μη ικανοποιημένης ζήτησης και στο οποίο οι πρώτες ύλες παραγγέλλονται κατά παρτίδες Q .

Οι παραπάνω πολιτικές εφαρμόστηκαν για συστήματα παραγωγής τα οποία διαθέτουν γραμμή παραγωγής με μηχανές συνδεδεμένες σε σειρά μέσω ενδιάμεσων αποθηκών. Ωστόσο, η μορφή ενός τέτοιου συστήματος θεωρείται ξεπερασμένη στις σύγχρονες και μεγάλες βιομηχανικές μονάδες οι οποίες έχουν εισάγει στην καθημερινή λειτουργία τους τον έλεγχο ποιότητας και έχουν αναγάγει σε βασική προτεραιότητά τους την διασφάλιση της παραγωγής άρτιων και αξιόπιστων προϊόντων.

Μια πιθανή μελλοντική εφαρμογή της παρούσας πολιτικής θα ήταν για ένα σύστημα παραγωγής το οποίο εκτός των άλλων θα διαθέτει και σταθμούς ελέγχου ποιότητας. Στόχος της νέας αυτής πολιτικής θα ήταν να προσδιορίσει την βέλτιστη κατανομή των σημείων ελέγχου επιθεώρησης, η οποία θα μεγιστοποιεί την συνάρτηση κέρδους του συστήματος παραγωγής, μειώνοντας παράλληλα τον αριθμό των ελαττωματικών κομματιών τα οποία διατίθενται προς πώληση από το σύστημα.

Τέλος, η πολιτική αυτή θα γινόταν πιο ρεαλιστική εάν διέθετε παραπάνω από μια λίστα αναμονής πελατών. Σε αυτή την περίπτωση, οι πελάτες χωρίζονται σε ομάδες σημαντικότητας, οι οποίες και καθορίζουν την προτεραιότητα εξυπηρέτησης τους από το σύστημα. Επομένως, είναι πιθανό μια βιομηχανική μονάδα, αν έχει έτοιμα προϊόντα στην διάθεση της και παράλληλα φθάσει στο σύστημα ένας πελάτης που ανήκει σε ομάδα πελατών χαμηλής σημαντικότητας, να μην τον δεχτεί στο σύστημα, αν και μπορεί να τον ικανοποιήσει, προκειμένου να είναι σε θέση να ανταποκριθεί σε τυχόν μελλοντική άφιξη ενός πελάτη που ανήκει σε ομάδα υψηλής σημαντικότητας.

Μπορεί λοιπόν η παρούσα εργασία να αποτελέσει αφετηρία για την ανάπτυξη μιας νέας πολιτικής η οποία συν των άλλων θα διαχειρίζεται αφενός τους σταθμούς ελέγχου ποιότητας του συστήματος και αφετέρου τον τρόπο εξυπηρέτησης πελατών οι οποίοι ανήκουν σε διαφορετικές κλάσεις σημαντικότητας.

Βιβλιογραφία

- [1] Spearman, M. L., Woodruff, D. L., and Hopp, W. J., 1990. Conwip: a pull alternative to Kanban. *International Journal of Production Research*, 28, 879–894.
- [2] Spearman, M. L., and Zazanis, M. A., 1992. Push and pull production systems: issues and comparisons. *Operations Research*, 40, 521–532.
- [3] Hall W.R., 1981, Driving the Productivity Machine: Production Planning and Control in Japan (Falls Church, Virginia: American Production and Inventory Control Society).
- [4] Stidham, S. Jr (1985) Optimal control of admission to a queueing system. *IEEE Transactions on Automatic Control*, 30, 705-713.
- [5] Stidham, S. Jr and Weber, R.R. (1993) A survey of Markov decision models for control of networks of queues. *Queueing Systems*, 13, 291-314.
- [6] Hadley, G., and Whitin, T.M., 1963. *Analysis of Inventory Systems*. Prentice Hall, Englewood Cliffs, NJ.
- [7] Smith, S.A., 1977. Optimal inventories for an $(S - 1, S)$ system with no backorders. *Management Science*, 23, 522–528.
- [8] Zipkin, P.H., 2000. *Foundations of Inventory Management*. McGraw-Hill, New York, NY.
- [9] Caldentey, R.A., 2001. Analyzing the make-to stock queue in the supply chain and e-business settings. Ph.D. dissertation, Sloan School of Management, MIT, Massachusetts.
- [10] Kouikoglou, V.S., and Phillis, Y.A., 2002. Design of product specifications and control policies in a single-stage production system. *IIE Transactions*, 34, 590–600.
- [11] Moinzadeh, K. (1989) Operating characteristics of the $(S-1,S)$ inventory system with partial backorders and constant resupply times. *Management Science*, 35, 472-477
- [12] Ha, A.Y., (1997) Stock-Rationing policy for a make-to-stock production system with two priority classes and backordering. *Yale School of Management*, Yale University, New Haven, Connecticut, USA.

- [13] Ha, A.Y., (1997) Inventory rationing in a make-to-stock production system with several demand classes and lost sales. *Yale School of Management*, Yale University, New Haven, Connecticut, USA.
- [14] Ioannidis, S., Kouikoglou, V.S., and Phillis, Y.A. (2004) Analysis of admission and inventory control for production networks. *IEEE Transactions*, 18, 353-358.
- [15] Jonson, L. A., and Montgomery, D.C., *Operations Research in Production Planning, Scheduling, and Inventory Control*, Wiley, 1974
- [16] Β. Κουϊκόγλου, *Προσομοίωση*, Σημειώσεις μαθήματος, Πολυτεχνείο Κρήτης, 2006
- [17] Β. Κουϊκόγλου, *Προγραμματισμός Απαιτήσεων Υλικών*, Σημειώσεις μαθήματος, Πολυτεχνείο Κρήτης, 2002
- [18] Γ. Φίλης, *Συστήματα Παραγωγής*, Σημειώσεις μαθήματος, Πολυτεχνείο Κρήτης, 2006
- [19] Γ. Φίλης, *Δίκτυα Παραγωγής C.A.M.*, Σημειώσεις μαθήματος, Πολυτεχνείο Κρήτης, 2003

Παράρτημα: Κώδικας του αλγορίθμου

Ο κώδικας που ακολουθεί αναπτύχθηκε σε γλώσσα προγραμματισμού C++ και προσομοιώνει συστήματα παραγωγής που εφαρμόζουν την πολιτική που αναπτύχθηκε στην παρούσα εργασία.

```
#include<stdio.h>
#include<stdlib.h>
#include<math.h>
float distrib_function(float P[],int i, int nqd,float F[]);
long zinext(long znext);
double unext(long zo);
long initial_Zo(int i);
double proceeding_time(int i,int temp_1);
double u_function(int i,int temp_1);
int orders_function(float P[],int qd[],float F[],int nqd,int temp_1);
void machine_zero(double t);
void machines_among_zero_and_N(int i,int temp_1,double t);
void machine_N(int temp_1,double t);
void zthsh(int temp_1,double t,float P[],int qd[],int nqd,float F[]);
int next_event();
void function_of_Q_order(int temp_10,double t,int temp_11);
int temp_3,temp_4,temp_8,posothta_paraggelias,Q,min;
int N,metritis_listas,c,current_list;
long TSIM;
long etoimo_apothema;
double anikanopoihth_zthsh,anikanopoihtoipelates;
double L_meso,B_meso,sales;
long NQA,total_demand;
int *s,*number,*LISTA_1;
long *Z,*L,*C;
double *T,*expo_b,*UNI_a,*UNI_b,*con,*wei_a,*wei_b,*wei_c;
double *LISTA_2;
FILE *infile;
FILE *outfile;
int main ()
{
    int i,var,nqd,Qmin,Qmax,temp_1;
    int Cmin,Cmax,ISmax,min_time,arrivals;
    int temp_9,B_end,L_end;
    int I,metavlhth,metavlhth_1;
    int k,current;
    int ISIM,NSIM,C_1,Q_1,I_1;
    double t,total_profit,current_kerdos,metavlhth_2,variable;
    double Final_kerdos,S_2,diasthma_lathous;
    double Final_kerdos_syn,Final_kerdos_plyn;
    float NP,h,b,DC,A;
    char a;
    int * qd, *TEMP;
    float *P;
    double *Kerdos;
    float *F;
    double PROFIT[27],SYN[27],PLYN[27];
    int CCC[27],QQQ[27],III[27];

    if((infile=fopen("input_file.txt","r"))==NULL)
    {
        printf("cannot input_file.txt\n\n");
        exit(1);
    }
    if((outfile=fopen("xxx.txt","w"))==NULL)
    {
```

```

        printf("cannot results.txt\n\n");
        exit(1);
    }

    do
    {
        fscanf(infile, "%c", &a);
        printf("%c", a);
    }while(a!=':' && a!='. ');
    printf("\n");
    fscanf(infile, "%d", &N);
    printf("%d\n", N);
    do
    {
        fscanf(infile, "%c", &a);
        printf("%c", a);
    }while(a!=':' && a!='. ');
    printf("\n");
    fscanf(infile, "%d", &TSIM);
    printf("%d\n", TSIM);
    do
    {
        fscanf(infile, "%c", &a);
        printf("%c", a);
    }while(a!=':' && a!='. ');
    printf("\n");

    C=(long*)malloc((N+1)*sizeof(int));
    if(C==NULL)
    {
        printf("memory allocation error:1\n");
        exit(1);
    }
    for(i=0;i<N;i=i+1)
    {
        fscanf(infile, "%d", &C[i]);
        printf("%d\n", C[i]);
    }
    do
    {
        fscanf(infile, "%c", &a);
        printf("%c", a);
    }while(a!=':' && a!='. ');
    printf("\n");
    fscanf(infile, "%d", &Qmin);
    printf("%d\n", Qmin);
    fscanf(infile, "%d", &Qmax);
    printf("%d\n", Qmax);
    do
    {
        fscanf(infile, "%c", &a);
        printf("%c", a);
    }while(a!=':' && a!='. ');
    printf("\n");
    fscanf(infile, "%d", &Cmin);
    printf("%d\n", Cmin);
    fscanf(infile, "%d", &Cmax);
    printf("%d\n", Cmax);
    do
    {
        fscanf(infile, "%c", &a);
        printf("%c", a);
    }while(a!=':' && a!='. ');
    printf("\n");
    fscanf(infile, "%d", &ISmax);
    printf("%d\n", ISmax);

    for(i=0;i<5;i=i+1)
    {
        do
        {
            fscanf(infile, "%c", &a);
            printf("%c", a);
        }while(a!=':' && a!='. ');
        printf("\n");
    }
}

```

```

number=(int*)malloc((N+1)*sizeof(int));
if(number==NULL)
{
    printf("memory allocation error:2\n");
    exit(1);
}

for(i=0;i<N+1;i=i+1)
{
    fscanf(infile,"%d",&var);
    printf("%d\n",var);
    if(var==0)
        number[i]=0;
    if(var==1)
        number[i]=1;
    if(var==2)
        number[i]=2;
    if(var==3)
        number[i]=3;
}
do
{
    fscanf(infile,"%c",&a);
    printf("%c",a);
}while(a!=':' && a!='. ');
printf("\n");

expo_b=(double*)calloc((N+2),sizeof(int));
if(expo_b==NULL)
{
    printf("memory allocation error:3\n");
    exit(1);
}
UNI_a=(double*)calloc((N+2),sizeof(int));
if(UNI_a==NULL)
{
    printf("memory allocation error:4\n");
    exit(1);
}
UNI_b=(double*)calloc((N+2),sizeof(int));
if(UNI_b==NULL)
{
    printf("memory allocation error:5\n");
    exit(1);
}
con=(double*)calloc((N+2),sizeof(int));
if(con==NULL)
{
    printf("memory allocation error:6\n");
    exit(1);
}
wei_a=(double*)calloc((N+2),sizeof(int));
if(wei_a==NULL)
{
    printf("memory allocation error:7\n");
    exit(1);
}
wei_b=(double*)calloc((N+2),sizeof(int));
if(wei_b==NULL)
{
    printf("memory allocation error:8\n");
    exit(1);
}
wei_c=(double*)calloc((N+2),sizeof(int));
if(wei_c==NULL)
{
    printf("memory allocation error:9\n");
    exit(1);
}
i=0;
do
{
    if(number[i]==0)
    {
        fscanf(infile,"%lf",&expo_b[i]);
        printf("expo_b=%lf\n",expo_b[i]);
        i=i+1;
    }
}

```

```

    }
    if(number[i]==1)
    {
        fscanf(infile,"%lf",&UNI_a[i]);
        fscanf(infile,"%lf",&UNI_b[i]);
        printf("UNI_a=%lf\n",UNI_a[i]);
        printf("UNI_b=%lf\n",UNI_b[i]);
        i=i+1;
    }
    if(number[i]==2)
    {
        fscanf(infile,"%lf",&con[i]);
        printf("%lf\n",con[i]);
        i=i+1;
    }
    if(number[i]==3)
    {
        fscanf(infile,"%lf",&wei_a[i]);
        fscanf(infile,"%lf",&wei_b[i]);
        fscanf(infile,"%lf",&wei_c[i]);
        printf("%lf\n",wei_a[i]);
        printf("%lf\n",wei_b[i]);
        printf("%lf\n",wei_c[i]);
        i=i+1;
    }
}while(i<N+1);
do
{
    fscanf(infile,"%c",&a);
    printf("%c",a);
}while(a!=':' && a!='. ');
printf("\n");
fscanf(infile,"%d",&nqd);
printf("%d\n",nqd);
do
{
    fscanf(infile,"%c",&a);
    printf("%c",a);
}while(a!=':' && a!='. ');
printf("\n");
qd=(int*)calloc((nqd+1),sizeof(int));
if(qd==NULL)
{
    printf("memory allocation error:10\n");
    exit(1);
}
P=(float*)calloc((nqd+1),sizeof(int));
if(P==NULL)
{
    printf("memory allocation error:11\n");
    exit(1);
}

for(i=0;i<nqd;i=i+1)
{
    fscanf(infile,"%d",&qd[i]);
    fscanf(infile,"%f",&P[i]);
    printf("%d\n",qd[i]);
    printf("%f\n",P[i]);
}
printf("\n");
do
{
    fscanf(infile,"%c",&a);
    printf("%c",a);
}while(a!=':' && a!='. ');
printf("\n");
fscanf(infile,"%d",&NSIM);
printf("NSIM:%d\n",NSIM);
do
{
    fscanf(infile,"%c",&a);
    printf("%c",a);
}while(a!=':' && a!='. ');
printf("\n");
fscanf(infile,"%f",&NP);
printf("Net Profit:%f\n",NP);

```

```

do
{
    fscanf(infile, "%c", &a);
    printf("%c", a);
}while(a!=':' && a!='. ');
printf("\n");
fscanf(infile, "%f", &h);
printf("h:%f\n", h);
do
{
    fscanf(infile, "%c", &a);
    printf("%c", a);
}while(a!=':' && a!='. ');
printf("\n");
fscanf(infile, "%f", &b);
printf("b:%f\n", b);
do
{
    fscanf(infile, "%c", &a);
    printf("%c", a);
}while(a!=':' && a!='. ');
printf("\n");
fscanf(infile, "%f", &DC);
printf("Defamation Cost:%f\n", DC);
do
{
    fscanf(infile, "%c", &a);
    printf("%c", a);
}while(a!=':' && a!='. ');
printf("\n");
fscanf(infile, "%f", &A);
printf("A:%f\n", A);
printf("\n\n");
Z=(long*)calloc((N+1), sizeof(int));
if(Z==NULL)
{
    printf("memory allocation error:12\n");
    exit(1);
}
TEMP=(int*)calloc((N+2), sizeof(int));
if(TEMP==NULL)
{
    printf("memory allocation error:17\n");
    exit(1);
}
Kerdos=(double*)calloc((NSIM+1), sizeof(int));
if(Kerdos==NULL)
{
    printf("memory allocation error:24\n");
    exit(1);
}
s=(int*)calloc((N+1), sizeof(int));
if(s==NULL)
{
    printf("memory allocation error:14\n");
    exit(1);
}
L=(long*)calloc((N+1), sizeof(int));
if(L==NULL)
{
    printf("memory allocation error:15\n");
    exit(1);
}
T=(double*)calloc((N+2), sizeof(int));
if(T==NULL)
{
    printf("memory allocation error:16\n");
    exit(1);
}
F=(float*)calloc((nqd+1), sizeof(int));
if(F==NULL)
{
    printf("memory allocation error:21\n");
    exit(1);
}
LISTA_1=(int*)calloc((1000+1), sizeof(int));
if(LISTA_1==NULL)

```

```

{
    printf("memory allocation error:222\n");
    exit(1);
}
LISTA_2=(double*)calloc((ISmax+1),sizeof(int));
if(LISTA_2==NULL)
{
    printf("memory allocation error:111\n");
    exit(1);
}

current_kerdos=-999999999;

k=0;
metavlhth=0;
for(i=0;i<27;i=i+1)
{
    PROFIT[i]=-999999999;
    SYN[i]=-99999999;
    PLYN[i]=-9999999;
    CCC[i]=88;
    QQQ[i]=88;
    III[i]=88;
}
for(c=Cmin;c<=Cmax;c=c+1)
{
    for(Q=Qmin;Q<=Qmax;Q=Q+1)
    {
        for(I=1;I<=ISmax;I=I+1)
        {
            temp_1=0;
            for(i=0;i<N+1;i=i+1)
            {
                TEMP[i]=0;
            }

            S_2=0;
            Final_kerdos=0;
            diasthma_lathous=0;

            for(ISIM=0;ISIM<NSIM;ISIM=ISIM+1)
            {

                total_profit=0;
                Kerdos[ISIM]=0;
                for(i=0;i<N+1;i=i+1)
                {
                    s[i]=0;
                    L[i]=0;
                    T[i]=TSIM+1;
                }
                for(i=0;i<Cmax;i=i+1)
                {
                    LISTA_1[i]=0;
                }
                for(i=0;i<ISmax;i=i+1)
                {
                    LISTA_2[i]=TSIM+1;
                }

                temp_3=0,temp_4=0,temp_8=0,temp_9=0,t=0,arrivals=0,metritis_listas=0;
                anikanopoihth_zhthsh=0,anikanopoihtoi_pelates=0,posothta_paraggelias=0,
                etoimo_apothema=0;

total_demand=0,L_meso=0,L_end=0,B_meso=0,B_end=0,sales=0,current_list=0;
NQA=0,current=0;

L[1]=I*Q;
do
{
    if(temp_9==0)
    {
        temp_1=TEMP[N];
        TEMP[N]=1;
        T[N]=proceeding_time(N,temp_1);
    }
}

```

```

        for(i=1;i<N;i=i+1)
        {
            if(L[i]>0)
            {
                min_time=i;
                current=1;
            }
        }
        if(current==1)
        {
            temp_1=TEMP[min_time];
            TEMP[min_time]=1;
T[min_time]=proceeding_time(min_time,temp_1);
        }
        min_time=next_event();
        t=T[min_time];
    }
    if(min_time==0)
    {
        machine_zero(t);

        if(s[1]==0)
        {
            temp_1=TEMP[1];
            TEMP[1]=1;
            s[1]=1;

                                T[1]=t+proceeding_time(1,temp_1);

        }
    }
    if(min_time!=0 && min_time!=N-1 && min_time!=N)
    {
        temp_1=TEMP[min_time];
        TEMP[min_time]=1;
        i=min_time;

                                machines_among_zero_and_N(i,temp_1,t);

    }

    if(min_time==N-1)
    {
        temp_1=TEMP[N-1];
        TEMP[N-1]=1;
        machine_N(temp_1,t);
        if(posothta_paraggelias>=Q)
        {
            temp_1=TEMP[0];
            TEMP[0]=1;

                                function_of_Q_order(min_time,t,temp_1);
                                posothta_paraggelias=posothta_paraggelias-
Q;
        }
    }

    if(min_time==N)
    {
        temp_1=TEMP[N];
        TEMP[N]=1;
        arrivals=arrivals+1;

        zhthsh(temp_1,t,P,qd,nqd,F);
        if(posothta_paraggelias>=Q)
        {
            temp_1=TEMP[0];
            TEMP[0]=1;

                                function_of_Q_order(min_time,t,temp_1);
                                posothta_paraggelias=posothta_paraggelias-
Q;
        }
    }

    }
    for(i=0;i<N+1;i=i+1)
    min_time=next_event();

    t=T[min_time];

```

```

temp_9=20;

}while(t<=TSIM);

for(i=1;i<temp_3;i=i+1)
{
    B_end=B_end+LISTA_1[i];
}
B_end=B_end+current_list;

for(i=1;i<N;i=i+1)
{
    L_end=L_end+L[i];
}
L_end=L_end+etoimo_apothema;

total_profit=(double)((sales*NP) - (anikanopoihtoi_pelates*DC) - ((I+NQA)*A)-
((L_meso + L_end*TSIM)*h) - ((B_meso + B_end*TSIM)*b));
Kerdos[ISIM]=(double)(total_profit/(TSIM));

}

k=k+1;
printf("epanalhpsd:%d\n",k);
for(i=0;i<NSIM;i=i+1)
{
    Final_kerdos=Final_kerdos+Kerdos[i];
}
Final_kerdos=Final_kerdos/NSIM;

for(i=0;i<NSIM;i=i+1)
{
    S_2=S_2+((Kerdos[i]-Final_kerdos)*(Kerdos[i]-Final_kerdos));
}

for(i=26;i>-1;i=i-1)
{
    metavlhth_2=PROFIT[i];
    if(Final_kerdos>metavlhth_2
    {
        metavlhth=i;
        metavlhth_1=0;
    }
}
if(metavlhth_1==0)
{
    for(i=26;i>metavlhth;i=i-1)
    {
        PROFIT[i]=PROFIT[i-1];
        SYN[i]=SYN[i-1];
        PLYN[i]=PLYN[i-1];
        CCC[i]=CCC[i-1];
        QQQ[i]=QQQ[i-1];
        III[i]=III[i-1];
    }
    PROFIT[metavlhth]=Final_kerdos;
    SYN[metavlhth]=Final_kerdos+diasthma_lathous;
    PLYN[metavlhth]=Final_kerdos-diasthma_lathous;
    CCC[metavlhth]=c;
    QQQ[metavlhth]=Q;
    III[metavlhth]=I;
}
else
    metavlhth_1=1;

if(Final_kerdos>current_kerdos)
{
    current_kerdos=Final_kerdos;
    C_1=c;
    Q_1=Q;
}

```

```

I_1=I;
Final_kerdos_syn=Final_kerdos+diasthma_lathous;
Final_kerdos_plyn=Final_kerdos-diasthma_lathous;
variable=diasthma_lathous;

}
}

}
}
printf("Final_minimum_kerdos:=%lf\n\n",current_kerdos);
printf("diasthma_lathous=%lf\n\n",variable);
printf("Final_kerdos(+):=%lf\n\n",Final_kerdos_syn);
printf("Final_kerdos(-):=%lf\n\n",Final_kerdos_plyn);
printf("for C=%d    Q=%d    I=%d\n",C_1,Q_1,I_1);
for(i=0;i<27;i=i+1)
{
    fprintf(outfile,"for C=%d,    Q=%d    I=%d\n",CCC[i],QQQ[i],III[i]);
    fprintf(outfile,"PROFIT[%d]=%lf ,    (+)=%lf\n\n",i,PROFIT[i],SYN[i],PLYN[i]);
}
fclose(infile);
fclose(outfile);
return (0);
}
float distrib_function(float P[],int i, int nqd,float F[])
{
    int j,k;
    for(k=0;k<nqd+1;k=k+1)
        F[k]=0;
    F[0]=0;
    F[1]=P[0];
    for(j=2;j<=i;j=j+1)
    {
        F[j]=F[j-1]+P[j-1];
    }
    return F[i];
}
long zinext(long znext)
{
    long XHI,XALO,LEFTLO,FHI,K;
    long zo;
    XHI = znext/65536;
    XALO = (znext-XHI*65536)*24112;
    LEFTLO = XALO/65536;
    FHI = XHI*24112+LEFTLO;
    K = FHI/32768;
    znext = (((XALO-LEFTLO*65536)-2147483647)+(FHI-K*32768)*65536)+K;
    if ( znext < 0 )
    {
        znext =znext + 2147483647;
    }
    XHI = znext/65536;
    XALO = (znext-XHI*65536)*26143;
    LEFTLO = XALO/65536;
    FHI = XHI*26143+LEFTLO;
    K = FHI/32768;
    znext = (((XALO-LEFTLO*65536)-2147483647)+(FHI-K*32768)*65536)+K;
    if ( znext < 0 )
    {
        znext =znext + 2147483647;
    }
    zo=znext;
    return zo;
}
double unext(long zo)
{
    double u;
    u=(double)(2*(zo/256)+1)/16777216.0;
    return u;
}
long initial_Zo(int i)
{
    int j;
    long zo;
    zo=12345678;
    for(j=0;j<i;j=j+1)

```

```

    {
        zo=(long)fmod(715.0*zo,2147483647.0);
        zo=(long)fmod(1058.0*zo,2147483647.0);
        zo=(long)fmod(1385.0*zo,2147483647.0);
    }
    return zo;
}
double proceeding_time(int i,int temp_1)
{
    double u, X;
    if(number[i]==0)
    {
        u=u_function(i,temp_1);
        X=-((expo_b[i])*log (u));
    }
    if(number[i]==1)
    {
        u=u_function(i,temp_1);
        X=UNI_a[i]+(UNI_b[i]-UNI_a[i])*u;
    }
    if(number[i]==2)
    {
        u=u_function(i,temp_1);
        X=con[i];
    }
    if(number[i]==3)
    {
        u=u_function(i,temp_1);
        X=wei_a[i]+wei_b[i]+wei_c[i];
    }
    return (X);
}

double u_function(int i,int temp_1)
{
    long zo,znext=0;
    double u;

    if(temp_1==0)
    {
        zo=initial_Zo(i);
        u=unext(zo);
        Z[i]=zo;
    }
    if(temp_1!=0)
    {
        znext=znext(Z[i]);
        Z[i]=znext;
        u=unext(Z[i]);
    }
    return (u);
}

int orders_function(float P[],int qd[],float F[],int nqd,int temp_1)
{
    int orders=-5,k;
    double u;
    double value_min,value_max;
    u=u_function(N,temp_1);
    k=0;
    do
    {
        value_min=distrib_function(P,k,nqd,F);
        value_max=distrib_function(P,k+1,nqd,F);
        if(u>=value_min && u<value_max)
        {
            orders=qd[k];
            k=nqd;
        }
        k=k+1;
    }while(k<nqd);
    return (orders);
}

void machine_zero(double t)
{
    double z;
    int i;

```

```

NQA=NQA+1;
L_meso=L_meso-(Q*t);
L[1]=L[1]+Q;
temp_4=temp_4-1;
for(i=min;i<temp_4;i=i+1)
{
    LISTA_2[i]=LISTA_2[i+1];
}

LISTA_2[temp_4]=TSIM+1;
z=9999999999;
for(i=0;i<temp_4;i=i+1)
{
    if(LISTA_2[i]<z)
    {
        z=LISTA_2[i];
        min=i;
        T[0]=LISTA_2[i];
    }
}
if(temp_4==0)
    T[0]=TSIM+1;
}

void machines_among_zero_and_N(int i,int temp_1,double t)
{
    int temp=0;
    if(s[i+1]==0)
    {
        s[i+1]=1;
        L[i+1]=1;
        T[i+1]=t+proceeding_time(i+1,temp_1);
    }
    else if (L[i+1]==C[i+1])
    {
        s[i]=2;
        T[i]=(double)(TSIM+1);
        temp=5;
    }

    }
    else
    {
        L[i+1]=L[i+1]+1;
    }
    if(temp==0)
    {
        if(s[i-1]==2)
        {
            s[i-1]=1;
            T[i-1]=t;
            T[i]=t+proceeding_time(i,temp_1);
            L[i]=L[i]-1;
        }
        else if(L[i]==1)
        {
            s[i]=0;
            T[i]=(double)(TSIM+1);
            L[i]=0;
        }
        else
        {
            T[i]=t+proceeding_time(i,temp_1);
            L[i]=L[i]-1;
        }
    }
}

void machine_N(int temp_1,double t)
{
    int i;
    if(temp_8==0)
    {
        etoimo_apothema=etoimo_apothema+1;
    }
    else
    {
        LISTA_1[0]=LISTA_1[0]-1;
        metritis_listas=metritis_listas-1;
        posothta_paraggelias=posothta_paraggelias+1;
    }
}

```

```

temp_8=1;
if(LISTA_1[0]==0)
{
    L_meso=L_meso+(current_list*t);

    B_meso=B_meso+(current_list*t);
    sales=sales+current_list;

    temp_3=temp_3-1;
    if(metritis_listas>0)
    {
        for(i=0;i<temp_3;i=i+1)
        {
            LISTA_1[i]=LISTA_1[i+1];
        }
        LISTA_1[temp_3]=0;
        current_list=LISTA_1[0];
        temp_8=1;
    }
    else
    {
        temp_8=0;
    }
}

}
L[N-1]=L[N-1]-1;
if(L[N-1]==0)
{
    s[N-1]=0;
    T[N-1]=(double)(TSIM+1);
}
else if(s[N-2]==2)
{
    s[N-2]=1;
    T[N-2]=t;
    T[N-1]=t+proceeding_time(N-1,temp_1);
}
else
{
    T[N-1]=t+proceeding_time(N-1,temp_1);
}
}

void zhtsh(int temp_1,double t,float P[],int qd[],int nqd,float F[])
{
    int demand;
    demand=orders_function(P,qd,F,nqd,temp_1);
    total_demand=total_demand+demand;
    if((etoimo_apothema-demand)>=0 && temp_8==0)
    {
        etoimo_apothema=etoimo_apothema-demand;
        posothta_paraggelias=posothta_paraggelias+demand;
        T[N]=t+proceeding_time(N,temp_1);
        sales=sales+demand;

        L_meso=L_meso+(demand*t);
    }
    else
    {
        if((c-metritis_listas-demand)>=0)
        {
            metritis_listas=metritis_listas+demand;
            B_meso=B_meso-(demand*t);
            temp_8=1;
            if(temp_3==0)
            {
                current_list=demand;
            }

            LISTA_1[temp_3]=demand;
            temp_3=temp_3+1;
            T[N]=t+proceeding_time(N,temp_1);
        }
        else
    }
}

```

```

        {
            anikanopoihth_zhthsh=anikanopoihth_zhthsh+demand;
            anikanopoihtoipelates=anikanopoihtoipelates+1;
            T[N]=t+proceeding_time(N, temp_1);
        }
    }
}
int next_event()
{
    int i,min_time,time;
    double t;
    t=(double)TSIM+1;
    for(i=0;i<N+1;i=i+1)
    {
        if(T[i]<=t)
        {
            time=i;
            t=T[i];
        }
    }
    min_time=time;
    return(min_time);
}
void function_of_Q_order(int temp_10,double t,int temp_11)
{
    double z,temp;
    int i;

    temp=t+proceeding_time(0,temp_11);
    LISTA_2[temp_4]=temp;

    temp_4=temp_4+1;

    z=9999999999;
    for(i=0;i<temp_4;i=i+1)
    {
        if(LISTA_2[i]<z)
        {
            z=LISTA_2[i];
            min=i;
            T[0]=LISTA_2[i];
        }
    }
}
}

```