ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΡΗΤΗΣ
ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΕΠΙΣΤΗΜΗΣ ΥΠΟΛΟΓΙΣΤΩΝ

ΤΟ ΠΡΟΒΛΗΜΑ ΤΟΥ ΣΥΝΟΛΙΚΟΥ ΧΡΟΝΟΥ ΠΕΡΑΤΩΣΗΣ Ν ΕΡΓΑΣΙΩΝ ΣΕ ΜΙΑ ΜΗΧΑΝΗ ΜΕ ΧΡΟΝΟΥΣ ΕΞΑΡΜΩΣΗΣ

Νίκος Τσατσάκης

Μεταπτυχιακή Εργασία

ΗΡΑΚΛΕΙΟ, ΚΡΗΤΗ
ΜΑΪΟΣ 1993

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΡΗΤΗΣ
ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΕΠΙΣΤΗΜΗΣ ΥΠΟΛΟΓΙΣΤΩΝ

ΤΟ ΠΡΟΒΛΗΜΑ ΤΟΥ ΣΥΝΟΛΙΚΟΥ ΧΡΟΝΟΥ ΠΕΡΑΤΩΣΗΣ Ν ΕΡΓΑΣΙΩΝ ΣΕ ΜΙΑ ΜΗΧΑΝΗ ΜΕ ΧΡΟΝΟΥΣ ΕΞΑΡΜΩΣΗΣ

Εργασία που υποβλήθηκε από τον
ΝΙΚΟ ΜΙΧΑΗΛ ΤΣΑΤΣΑΚΗ
ως μερική απαίτηση για την απόκτηση του
ΜΕΤΑΠΤΥΧΙΑΚΟΥ ΔΙΠΛΩΜΑΤΟΣ ΕΞΕΙΔΙΚΕΥΣΗΣ

Ηράκλειο, Μάιος 1993

Συγγραφέας : . . . .
Τμήμα Επιστ. Υπολογιστών, 6 Μαΐου 1993

Εισηγητική Επιτροπή

. . . .
Αναπλ. Καθ. Πάνος Κωνσταντόπουλος, Επόπτης

. . . .
Καθηγητής Στέλιος Ορφανουδάκης, Μέλος

. . . .
Αναπλ. Καθ. Καίτη Χούστη, Μέλος

Δεκτή : . . . .
Αναπλ. Καθ. Πάνος Κωνσταντόπουλος,
Πρόεδρος Επιτρ. Μεταπτυχιακών Σπουδών

Το πρόβλημα του Συνολικού Χρόνου Περάτωσης n Εργασιών σε Μια Μηχανή με Χρόνους Εξάρμωσης

Νίκος Τσατσάκης
Μεταπτυχιακή Εργασία

Τμήμα Επιστήμης Υπολογιστών
Πανεπιστήμιο Κρήτης

Περίληψη

Στην παρούσα εργασία μελετούμε το πρόβλημα της ελαχιστοποίησης του συνολικού χρόνου περάτωσης n εργασιών σε μια μηχανή παρουσία χρόνων εξάρμωσης. Όλες οι εργασίες έχουν καλά καθορισμένους χρόνους επεξεργασίας, όχι κατ' ανάγκη ταυτοτικούς, είναι διαθέσιμες τη χρονική στιγμή μηδέν και κατανέμονται σε B διαφορετικές ομάδες. Για την προσαρμογή της μηχανής προκειμένου να υποδεχθεί εργασία διαφορετικής ομάδας από την τρέχουσα, απαιτείται χρόνος, ο οποίος χαρακτηρίζεται ως χρόνος εξάρμωσης. Στην μελέτη μας ασχολούμαστε κυρίως με την περίπτωση χρόνων εξάρμωσης ανεξάρτητων ακολουθίας. Η εκτέλεση μιας εργασίας δεν διακόπτεται ποτέ, ενώ δεν επιβάλλεται η διαδοχική εκτέλεση όλων των εργασιών που ανήκουν σε μια ομάδα.

Το πρόβλημα του συνολικού χρόνου περάτωσης n εργασιών σε μια μηχανή, εμφανίζεται συχνά στο λεπτομερειακό προγραμματισμό παραγωγής. Οι αλγόριθμοι επίλυσης του προβλήματος που αναπτύσσονται στην παρούσα εργασία έχουν ενταχθεί σ' ένα διαλογικό σύστημα προγραμματισμού παραγωγής, στα πλαίσια του οποίου οι βέλτιστες ή σχεδόν βέλτιστες λύσεις που παράγουν ελέγχονται από το χρήστη του συστήματος που αποφασίζει αν θα αποτελέσουν αυτές ή κάποιες τροποποιήσεις τους το τελικό πρόγραμμα παραγωγής.

Για την επίλυση του προβλήματος λαμβάνουμε υπόψη την κατανομή εργασιών σε ομάδες και κατασκευάζουμε υποσύνολα αυτών, τα οποία εκτελούνται διαδοχικά σε κάποια βέλτιστη διάταξη. Τέτοια υποσύνολα τα ονομάζουμε παρτίδες και τα διακρίνουμε σε πρωτογενείς (ΠΠ), δευτερογενείς (ΔΠ) και γενικευμένες (ΓΠ). Εξάγουμε προτάσεις σχετικές με τις παραπάνω οντότητες και χαρακτηριστικά των βέλτιστων λύσεων. Το πρόβλημα που μελετούμε είναι πολύ απλό όταν δεν έχουμε χρόνους εξάρμωσης ή είναι γνωστές οι μέγιστες ΓΠ που εμφανίζονται σε βέλτιστη λύση (λύνεται σε χρόνο O(nlogn)). Όταν έχουμε χρόνους εξάρμωσης το πρόβλημα δυσκολεύει αρκετά, αλλά η πλοκή του δεν έχει ακόμα προδιοριστεί.

Η δική μας συνεισφορά στην επίλυση του προβλήματος συνίσταται στην κατασκευή τριών αλγορίθμων: ενός ακριβούς και δύο προσεγγιστικών. Ο ακριβής είναι αλγόριθμος δυναμικού προγραμματισμού που ακολουθεί κατά βάση τα σχήματα των Ψαραύτη και Monma και Potts. Περιορίζεται αρκετά από τις απαιτήσεις μνήμης κυρίως και χρόνου δευτερευόντως όταν ο αριθμός των ομάδων είναι μεγάλος. Η χρήση των προτάσεων που σχετίζονται με παρτίδες κατά την κατασκευή των καταστάσεων του προβλήματος ελαττώνει σημαντικά τις απαιτήσεις σε μνήμη, με αποτέλεσμα να είναι δυνατή η επίλυση προβλημάτων με μερικές εκατοντάδες (200-300) εργασίες, κατανεμημένες σε μερικές δεκάδες (10-30) ομάδων. Για την επίλυση προβλημάτων μεγαλύτερου μεγέθους κατασκευάσαμε δύο προσεγγιστικούς αλγορίθμους. Ο πρώτος ξεκινώντας από μια αρχική λύση που έχει αρκετές από τις ιδιότητες μιας κατηγορίας βέλτιστων, συνενώνει ΔΠ σε ΓΠ αν κάτι τέτοιο βελτιώνει την αρχική λύση, φροντίζοντας παράλληλα να διατηρεί σε ισχύ ένα κανόνα σχετικό με τη βέλτιστη διάταξη μέγιστων ΓΠ. Ο αλγόριθμος αυτός είναι πολυωνυμικός και διαπιστώθηκε πειραματικά ότι δίνει λύσεις με χείριστη απόκλιση από το βέλτιστο μικρότερη του 2.4%. Πολλά από τα προβλήματα του πειραματικού δείγματος (˜ 73%) λύθηκαν βέλτιστα, ενώ η μέση απόκλιση των υπολοίπων από το βέλτιστο είναι 0.18%. Την κατασκευή του δεύτερου προσεγγιστικού αλγορίθμου ενέπνευσε η μελέτη και υλοποίηση ενός αλγορίθμου των Ahn και Hyun για το ίδιο πρόβλημα, παράλληλα με την πειραματική μελέτη της συμπεριφοράς του πρώτου προσεγγιστικού αλγορίθμου. Ο καινούργιος αυτός αλγόριθμος είναι στην ουσία μια παράθεση i) του πρώτου προσεγγιστικού αλγορίθμου, ii) μιας διαδικασίας ανάλυσης των γενικευμένων παρτίδων που προκύπτουν από το (i) στις ΠΠ που τις απαρτίζουν, και (iii) του βήματος θέσει-μεταθέσεων που περιέχεται στον αλγόριθμο των Ahn και Hyun. Ο αλγόριθμος αυτός είναι επίσης πολυωνυμικός, ως παράθεση δυο πολυωνυμικών αλγορίθμων και μιας πολυωνυμικής διαδικασίας. Παρατηρήθηκε ότι ο αλγόριθμος αυτός, δίνει βέλτιστη λύση στο σύνολο του δείγματος των προβλημάτων που κατασκευάστηκαν για την πειραματική μελέτη των προσεγγιστικών αλγορίθμων.

Επόπτης : Πάνος Κωνσταντόπουλος
Αναπλ. Καθ. Επιστήμης Υπολογιστών, Παν. Κρήτης

The Problem of Total Completion Time of n Jobs
on One Machine with Set-Up Times

Nikos Tsatsakis
Master of Science Thesis

Department of Computer Science
University of Crete

Abstract

We consider the problem of minimizing the total completion time of n jobs on one machine with set-up times. All jobs have deterministic processing times, not necessarily identical, are available at time zero and belong to B different groups. Sequence independent set-up times are needed to adjust the machine when switching from jobs in one group to another. No preemption is allowed and jobs in a group need not be processed consecutively.

The problem of total completion time of jobs on one machine often arises in detailed production planning. The algorithms developed here are embedded in an interactive production planning system, in the context of which the optimal or near-optimal solutions they produce can be modified before being committed as production schedules.

The distribution of jobs into groups is taken into account for the solution of the problem and we try to construct subsets of jobs that are executed consecutively in an optimal sequence. Such subsets are called lots and are distinguished into first-order lots (FL), second-order lots (SL) and generalized lots (GL). We obtain propositions for the above entities and characteristics of the optimal solutions. The problem we study is very simple when no set-up times are considered, or when maximal generalized lots are known (it is solved in O(nlogn) time). In the presence of set-up times the problem becomes more difficult, and its complexity has not been determined so far.

Our contribution to the solution of the problem is the construction of three algorithms, one exact and two approximate. The exact one is a dynamic programming algorithm, eventually overwhelmed mostly by memory requirements as the number of groups grows. Using the proposition

related to lot construction reduces memory requirements significantly and enables the solution of problems including 200-300 jobs distributed in 10-30 groups. On the other hand, the approximate algorithms can solve problems of larger dimensions. The first starts with an initial solution sharing several properties of the optimal and improves on it by combining SL into GL and maintaining a rule related to an optimal ordering of maximal GL. This algorithm is polynomial and has been experimentally observed to produce solutions with worst deviation from the optimum less than 2.4%. Many of the test problems (~ 73%) were solved optimally, while the average deviation from the optimum of the rest was 0.18%. The second approximate algorithm was inspired by the study and implementation of the polynomial algorithm of Ahn and Hyun for the same problem, along with the experimental study of our first approximate algorithm . The second algorithm is actually an apposition of i) the first approximate algorithm, ii) a procedure that analyzes the maximal GL of (i) into the constituent FL and iii) the algorithm of Ahn and Hyun, without the initial-sequence- construction step. This algorithm is also polynomial as apposition of two polynomial algorithms and one polynomial procedure. It has been observed to yield optimal solutions for the entire set of test problems.

Supervisor : Panos Constantopoulos
Associate Professor of Computer Science, Univ. of Crete

στην οικογένειά μου
και τους δασκάλους μου

Ευχαριστίες

Θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή μου κ. Πάνο Κωνσταντόπουλο, για τις πολύτιμες συμβουλές του στην εκπόνηση της εργασίας μου, την καθοδήγηση, την υπομονή του και τη συμπαράστασή του.

Επίσης θα ήθελα να ευχαριστήσω τα μέλη της ομάδας Συστημάτων Στήριξης Αποφάσεων για τις συζητήσεις και τις υποδείξεις τους, κατά τις τακτικές εβδομαδιαίες συναντήσεις μας. Ιδιαίτερα ευχαριστώ τον Θαλή Γεωργίου και τον Κοσμά Χαριτωνίδη για τίς ώρες που αφιέρωσαν στην ανταλλαγή απόψεων και ιδεών.

Τέλος θα ήθελα να ευχαριστήσω το Ινστιτούτο Πληροφορικής του Ι.Τ.Ε. για την οικονομική ενίσχυση και την υλικοτεχνική υποστήριξη που μου παρείχε κατά την διάρκεια των μεταπτυχιακών σπουδών μου.

Περιεχόμενα

Περίληψη
Ευχαριστίες
Περιεχόμενα

1. **Εισαγωγή** 1
   1.1 Ταξινόμηση των προβλημάτων χρονικού προγραμματισμού - Γενικό σχήμα περιγραφής 1
   1.2 Το πρόβλημα $n/1/seq-indep/f\,(\pi)$ 2
   1.3 Το πρόβλημα των διατεταγμένων κατηγοριών 4
   1.4 Το πρόβλημα $TCTS_j$ 4
   1.5 Εφαρμογές των προβλημάτων χρονικού προγραμματισμού στον προγραμματισμό παραγωγής 6
   1.6 Ορολογία - Συμβολισμοί 8

2. **Βιβλιογραφική ανασκόπηση** 10
   2.1 Γενική ανασκόπηση 10
   2.2 Οι εργασίες των Monma-Potts και Ψαραύτη 15
   2.3 Η εργασία των Ahn-Hyun 16

3. **Ανάλυση του προβλήματος** $TCTS_j$ 18
   3.1 Το $TCTS_j$ ως πρόβλημα διατεταγμένων κατηγοριών 18
   3.2 Υπολογιστική πλοκή του $TCTS_j$ 20
   3.3 Ορισμός παρτίδων στο $TCTS_j$ - Διάταξη μέγιστων παρτίδων 21
   3.4 Πρωτογενείς παρτίδες στο $TCTS_j$ 23
   3.5 Δευτερογενείς παρτίδες στο $TCTS_j$ - Ορισμός χαρακτηριστικών λόγων 31
   3.6 Γενικευμένες παρτίδες στο $TCTS_j$ 42

4. **Πρώτος προσεγγιστικός αλγόριθμος για το** $TCTS_j$ 45
   4.1 Περιγραφή - Γενική ιδέα 45
   4.2 Βήματα του πρώτου προσεγγιστικού αλγορίθμου 47

4.3 Σύνοψη του πρώτου προσεγγιστικού αλγορίθμου 54
4.4 Υπολογιστική πλοκή του πρώτου προσεγγιστικού αλγορίθμου 56

5. **Ακριβής αλγόριθμος για το** $TCTS_j$ 58
5.1 Περιγραφή 58
5.2 Πλοκή - Αναποτελεσματικότητα χρήσης - Βελτίωση 60

6. **Δεύτερος προσεγγιστικός αλγόριθμος για το** $TCTS_j$ 61
6.1 Αστοχία του πρώτου προσεγγιστικού αλγορίθμου 61
6.2 Αστοχία του προσεγγιστικού αλγορίθμου των Ahn και Hyun 62
6.3 Η ιδέα του δεύτερου προσεγγιστικού αλγορίθμου 63
6.4 Βήματα του δεύτερου προσεγγιστικού αλγορίθμου 64
6.5 Υπολογιστική πλοκή του δεύτερου προσεγγιστικού αλγορίθμου ... 65

7. **Υλοποίηση** 66
7.1 Περιβάλλον υλοποίησης 66
7.2 Υλοποίηση των αλγορίθμων 66
7.2.1 Υλοποίηση των προσεγγιστικών αλγορίθμων 67
7.2.2 Υλοποίηση του ακριβούς αλγορίθμου 68

8. **Πειραματική αξιολόγηση αλγορίθμων** 69
8.1 Κριτήρια αξιολόγησης 69
8.2 Κατασκευή προβλημάτων δοκιμών 69
8.3 Παράθεση και σχολιασμός αποτελεσμάτων 70

9. **Επίλογος** 76
9.1 Συμπεράσματα 76
9.2 Μελλοντικές επεκτάσεις - βελτιώσεις 77
9.3 Εφαρμογή στη λύση συγγενών προβλημάτων 78

Παραπομπές 78

Άλλη Βιβλιογραφία 80

1. Εισαγωγή

Μια σημαντική κλάση συνδυαστικών προβλημάτων αποτελούν τα προβλήματα χρονικού προγραμματισμού. Αυτά έχουν ως στόχο τη βέλτιστη διάταξη ενός συνόλου εργασιών, ώστε η εκτέλεσή τους να ακολουθεί τυχόντες περιορισμούς που τίθενται κατά περίπτωση (προτεραιότητες, εξαρτήσεις, προθεσμίες κ.ά.), και να δίνει βέλτιστη τιμή σε κάποιο, δεδομένο, κριτήριο επίδοσης ή συνάρτηση κόστους (συνολικός χρόνος περάτωσης, μέγιστος χρόνος ροής, αριθμός καθυστερημένων εργασιών, συνολική καθυστέρηση κ.ά.). Η εκτέλεση των εργασιών γίνεται με χρήση ενός συνόλου πόρων (μηχανές, άνθρωποι, εργαλεία κ.ά.), που είναι επίσης καθοριστικό για το εκάστοτε πρόβλημα.

Η ενασχόληση πολλών ερευνητών με προβλήματα χρονικού προγραμματισμού, έγκειται σε δύο κυρίως λόγους. Αφενός αποτελούν πρόκληση γιατί αν και είναι πολύ απλά στη διατύπωσή τους, είναι στην πλειοψηφία τους προβλήματα NP-hard, και αφετέρου παρουσιάζουν άμεσο πρακτικό ενδιαφέρον, εφόσον εμφανίζονται στο λεπτομερειακό προγραμματισμό παραγωγής σε επιχειρήσεις και στον χρονοπρογραμματισμό εργασιών σε υπολογιστικά συστήματα.

Η δική μας ενασχόληση με ένα από τα προβλήματα χρονικού προγραμματισμού, απέδωσε έναν αλγόριθμο δυναμικού προγραμματισμού που λύνει προβλήματα πραγματικών διαστάσεων (πολύ μεγαλύτερα απ' ότι λύνονταν μέχρι τώρα με ανάλογους αλγορίθμους), και δύο πολυωνυμικούς ευρηματικούς αλγορίθμους οι οποίοι δίνουν με μεγάλη πιθανότητα βέλτιστη λύση, σε ακόμη μεγαλύτερα προβλήματα.

1.1. Ταξινόμηση των προβλημάτων χρονικού προγραμματισμού - Γενικό σχήμα περιγραφής

Η ταξινόμηση των προβλημάτων χρονικού προγραμματισμού μπορεί να γίνει με βάσει τις οντότητες των προβλημάτων, τα μεγέθη που σχετίζονται με αυτές, τους πόρους που χρησιμοποιούνται, τα κριτήρια βελτίστου που λαμβάνονται υπόψη, τους περιορισμούς που επιβάλλονται κλπ.

Επειδή όμως ένα συγκεκριμένο πρόβλημα μπορεί να ανήκει σε διάφορες κατηγορίες ταυτόχρονα, είναι απαραίτητος ο ορισμός ενός γενικού σχήματος περιγραφής των προβλημάτων. Ένα τέτοιο σχήμα, ευρύτατα χρησιμοποιούμενο από τους ερευνητές που ασχολούνται με χρονικό προγραμματισμό είναι το εξής:

α | β | γ | δ όπου

α: ο αριθμός των εργασιών που πρέπει να προγραμματιστούν,

β: ο αριθμός των μηχανών (ή άλλων πόρων) που χρησιμοποιούνται,

γ: περιορισμοί πάνω στις οντότητες και τα μεγέθη του προβλήματος, και

δ το κριτήριο (ή κριτήρια) βελτιστοποίησης.

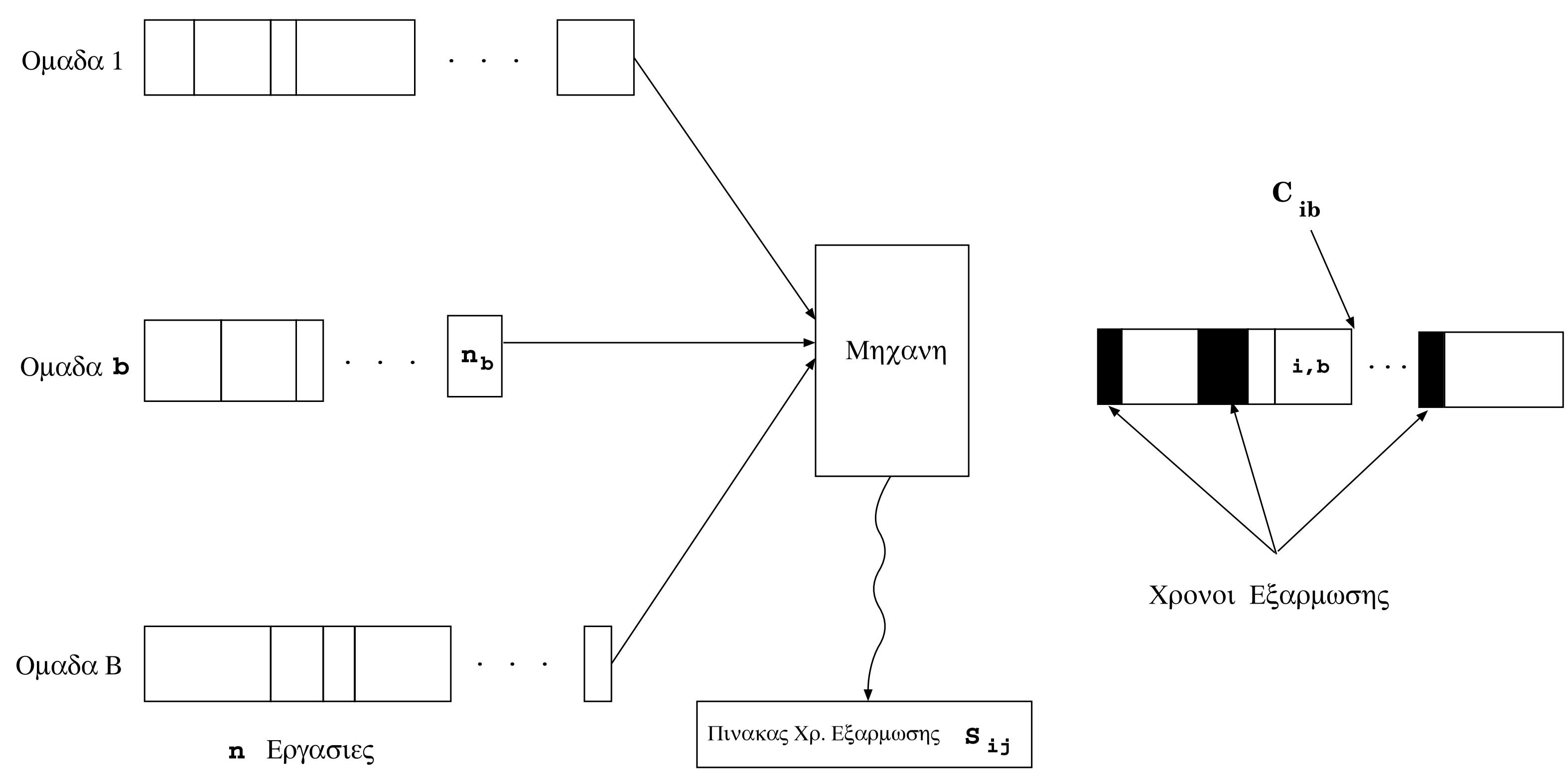
Στο σχήμα αυτό το πεδίο γ μπορεί να είναι κενό, ή να αποτελείται από έναν ή περισσότερους περιορισμούς. Επίσης το δ μπορεί να περιέχει περισσότερα του ενός κριτήρια βελτίστου. (Συνήθως αν αυτά χωρίζονται με κόμμα πχ ($\sum C_i, T_{\max}$) είναι ισοδύναμα, ενώ αν χωρίζονται με βέλος πχ ($\sum C_i \rightarrow T_{\max}$), το αριστερό του βέλους θεωρείται πρωτεύον και το δεξιό δευτερεύον [Th89].)

Για παράδειγμα το $n\,|\,1\,|\,prec-con, seq-indep\,|\,T_{\max}$ είναι το πρόβλημα ελαχιστοποίησης της μέγιστης καθυστέρησης n εργασιών σε μια μηχανή, όταν υπάρχουν χρόνοι εξάρμωσης ανεξάρτητοι ακολουθίας και επιβάλλονται περιορισμοί διαδοχής στις εργασίες. Βέβαια μπορούμε να παρατηρήσουμε ότι η περιγραφή $n\,|\,1\,|\,prec-con, seq-indep\,|\,T_{\max}$ δεν δίνει, για παράδειγμα, το είδος των περιορισμών διαδοχής, ούτε μας λέει κατά πόσον είναι καλά καθορισμένα (στατικά) τα μεγέθη του προβλήματος, όμως δίνει πολύ συνοπτικά τα πιο βασικά χαρακτηριστικά του. Όλες οι υπόλοιπες πληροφορίες μπορεί να δοθούν σε λεπτομερέστερη περιγραφή του.

1.2. Το πρόβλημα $n\,|\,1\,|\,seq-indep\,|\,f(\pi)$

Το πρόβλημα $n\,|\,1\,|\,seq-indep\,|\,f(\pi)$ έχει ως εξής (σχήμα 1.1) :

Τη χρονική στιγμή 0 δίνονται n εργασίες οι οποίες πρέπει να εκτελεστούν σε μια μηχανή. Οι εργασίες αυτές κατανέμονται σε B διαφορετικές ομάδες. Κάθε ομάδα b, με 1≤b≤B έχει n_b το πλήθος εργασίες αυθαίρετα σημειωμένες 1η, 2η, ..., n_b-οστή. Η i-οστή εργασία της ομάδας b απαιτεί χρόνο επεξεργασίας $p_{ib}>0$. Χρόνος εξάρμωσης $s_{b,c}$ απαιτείται όταν μια εργασία της ομάδας c πρέπει να



Σχημα 1.1 : ΠΕΡΙΓΡΑΦΗ του **n | 1 | setup time | f(π)**

εκτελεστεί αμέσως μετά από εργασία της ομάδας b. Επίσης ένας αρχικός χρόνος εξάρμωσης $s_{0,c}$ απαιτείται αν μια εργασία της ομάδας c, εκτελεστεί πρώτη στη μηχανή . Οι χρόνοι εξάρμωσης είναι ανεξάρτητοι ακολουθίας, δηλαδή $s_{b,c} = s_c$ για κάθε 0≤b≤B και 1≤c≤B με b≠c. Αν η μηχανή αρχίσει την επεξεργασία της εργασίας i δεν μπορεί να συνεχίσει με άλλη εργασία πριν από την ολοκλήρωση της i. Τέλος το πρόβλημα περιλαμβάνει ένα κριτήριο επίδοσης ή συνάρτηση κόστους, βάσει της οποίας αξιολογούμε κάθε διάταξη εργασιών. Στόχος μας είναι η εύρεση μιας διάταξης π^* που να βελτιστοποιεί το κριτήριο επίδοσης (ισοδύναμα να ελαχιστοποιεί τη συνάρτηση κόστους) $f(\pi)$.

1.3. Το πρόβλημα των διατεταγμένων κατηγοριών

Το πρόβλημα των διατεταγμένων κατηγοριών είναι ένα σύνολο περιπτώσεων του $n\,|\,1\,|\,seq-indep\,|\,f(\pi)$ στο οποίο υπάρχει μία τουλάχιστον βέλτιστη λύση για κάθε πρόβλημα, όπου η σχετική θέση των ομοειδών εργασιών (εργασιών που ανήκουν στην ίδια ομάδα) είναι γνωστή από την αρχή. Η σχετική θέση των ομοειδών εργασιών εξαρτάται άμεσα από το είδος του κριτηρίου επίδοσης $f(\pi)$ και βρίσκεται πολύ εύκολα (σε χρόνο O(nlogn)) για τις περιπτώσεις της μέγιστης καθυστέρησης, του συνολικού χρόνου περάτωσης των εργασιών, και του αριθμού των καθυστερημένων εργασιών.

Στη γενική του μορφή το πρόβλημα των διατεταγμένων κατηγοριών μπορεί να λυθεί σε δύο βήματα. Στο πρώτο βήμα γίνεται διάταξη των εργασιών μέσα στις ομάδες, ενώ στο δεύτερο γίνεται συγχώνευση των διατεταγμένων ομάδων σε μία, βέλτιστη διάταξη. Όμως ενώ το πρώτο βήμα για πολλές περιπτώσεις συναρτήσεων κόστους εκτελείται γρήγορα, το δεύτερο απαιτεί στις περισσότερες των περιπτώσεων εκθετικό χρόνο ως προς τον αριθμό των ομάδων.

Έτσι ένας επιθυμητός στόχος είναι η μείωση της πολυπλοκότητας του βήματος αυτού, και προς αυτή την κατεύθυνση κινούμαστε στην ανάλυση που θα ακολουθήσει για το πρόβλημα $TCTS_j$, που ορίζεται στο αμέσως επόμενο κεφάλαιο, ως περίπτωση προβλήματος διατεταγμένων κατηγοριών.

1.4. Το πρόβλημα $TCTS_j$

Αν στο πρόβλημα $n\,|\,1\,|\,seq-indep\,|\,f(\pi)$ θεωρήσουμε ως κριτήριο επίδοσης τη συνάρτηση $\sum C_{ib}$, όπου με C_{ib} συμβολίζουμε το χρόνο περάτωσης της i-οστής εργασίας της ομάδας b, προκύπτει το πρόβλημα ελαχιστοποίησης του συνολικού χρόνου περάτωσης n εργασιών σε μια μηχανή, με χρόνους εξάρμωσης ανεξάρτητους ακολουθίας το οποίο συμβολίζουμε με $TCTS_j$.

Το $TCTS_j$ ανήκει στην οικογένεια των προβλημάτων διατεταγμένων κατηγοριών, και μάλιστα το πρώτο βήμα της διάταξης των εργασιών μέσα στις ομάδες γίνεται σε χρόνο πολυωνυμικό με την εφαρμογή του απλού κανόνα SPT ([MoPo89] - ανάλυση στην ενότητα 3). Έτσι η λύση του, έγκειται στην εύρεση ενός αποτελεσματικού τρόπου συγχώνευσης των διατεταγμένων ομάδων.

Η συνάρτηση κόστους $\sum C_{ib}$, η ελαχιστοποίηση της οποίας αποτελεί στόχο του προβλήματος $TCTS_j$, αποτελεί κριτήριο πολύ σπουδαίο, τόσο στον προγραμματισμό παραγωγής, όσο και σε εφαρμογές σε υπολογιστικά συστήματα.

Στον προγραμματισμό παραγωγής ο συνολικός χρόνος περάτωσης εργασιών, αποτελεί κριτήριο καλής διαχείρισης των παραγγελιών, και άρα χρησιμοποιείται ως δείκτης απόκρισης στους πελάτες. Επίσης αν ο χρόνος που παραμένουν οι εργασίες στο σύστημα που τις επεξεργάζεται είναι μικρός, ισοδύναμα μικρές είναι και οι απαιτήσεις σε αποθηκευτικό χώρο για τυχόν ενδιάμεσα προϊόντα και πρώτες ύλες. Παράλληλα ελαχιστοποιείται το απόθεμα του υπό κατεργασία υλικού, καθώς και το κεφάλαιο που δεσμεύεται σε αυτό.

Βέβαια θα μπορούσαμε να ισχυριστούμε ότι ένα τέτοιο κριτήριο δεν οδηγεί το σύστημα σε μέγιστους ρυθμούς παραγωγής (λειτουργίας μηχανών), όπως επιδιώκεται για παράδειγμα με το κριτήριο ελαχιστοποίησης του μέγιστου χρόνου περάτωσης των εργασιών. Όμως από την άλλη είναι προφανές ότι αντιμετωπίζει με πιο δίκαιο τρόπο τους πελάτες που υποβάλλουν εργασίες προς εκτέλεση, εφόσον ο χρόνος εξυπηρέτησής τους θα είναι ανάλογος της απαίτησής τους για χρήση του συστήματος. (Μικρές εργασίες εκτελούνται κατά κανόνα στην αρχή και άρα έχουν μικρό χρόνο συμπλήρωσης, εν αντιθέσει με τις μεγάλες εργασίες που έχουν κατά κανόνα ανάλογα μεγάλο χρόνο συμπλήρωσης.)

Στην περίπτωση των υπολογιστικών συστημάτων, το κριτήριο του συνολικού χρόνου περάτωσης ισοδυναμεί με το μέσο χρόνο απόκρισης του συστήματος, στοιχείο ιδιαίτερα σημαντικό για τους χρήστες του. Επίσης, η ελαχιστοποίηση του χρόνου που δαπανάται σε ενδιάμεσα στάδια για την εκτέλεση διεργασιών και των υπολογιστικών πόρων που χρησιμοποιούνται σ' αυτά την οποία συνεπάγεται, ελαττώνει σημαντικά τις απαιτήσεις σε αποθηκευτικό χώρο που απαιτείται (μνήμη). Μια σχετική μελέτη που έγινε από τους Abdel-Wahab και Kameda [Ab-WaKa87], ασχολείται με κάτι ανάλογο: γίνεται χρήση του κριτηρίου $\min\sum C_{ib}$ για την αποδοτική χρήση του χώρου που χρησιμοποιείται από ενδιάμεσα αρχεία ή προγράμματα κατά την κατασκευή ή μετατροπή Βάσεων Δεδομένων.

Συνοψίζοντας μπορούμε να πούμε ότι το $\min\sum C_{ib}$ ισοδυναμεί με ελαχιστοποίηση:

(i) του χρόνου απόκρισης ενός συστήματος στους πελάτες του, και

(ii) του κόστους ενδιάμεσων βημάτων κατά τη διαδικασία εξυπηρέτησης.

1.5. Εφαρμογές των προβλημάτων χρονικού προγραμματισμού στον προγραμματισμό παραγωγής

Ο προγραμματισμός παραγωγής ενός εργοστασίου είναι μια σύνθετη διαδικασία με δομή ιεραρχική. Ο χωρισμός της διαδικασίας αυτής σε επίπεδα, επιβάλλεται λόγω της άμεσης σχέσης της πλοκής των αποφάσεων με το χρονικό ορίζοντα προγραμματισμού, αλλά και της ευκολότερης και αποτελεσματικότερης αντιμετώπισης της αβεβαιότητας σχετικά με τα αναμενόμενα γεγονότα.

Τρία επίπεδα διακρίνουμε: το στρατηγικό σχεδιασμό, το μεσοπρόθεσμο προγραμματισμό, και το βραχυπρόθεσμο προγραμματισμό παραγωγής. Ο χρονικός ορίζοντας στον οποίο εκτείνεται καθένα από τα παραπάνω επίπεδα, εξαρτάται από τον τύπο του εργοστασίου που μελετάται κάθε φορά, και είναι ένα από τα κρισιμότερα μεγέθη στον προγραμματισμό παραγωγής.

Είναι προφανές ότι η ομαλή λειτουργία της παραγωγικής διαδικασίας είναι κρίσιμη για την ομαλή λειτουργία ολόκληρου του εργοστασίου, εφόσον τυχόν παρεκκλίσεις από αυτήν έχουν άμεσες επιδράσεις σε άλλα τμήματα.

Σημαντικό στοιχείο για την αποτελεσματική λειτουργία του προγραμματισμού παραγωγής αποτελεί ένας μηχανισμός ανάδρασης που περιέχει. Έτσι η αδυναμία σωστής λειτουργίας σε κάποιο από τα επίπεδά του, διαβιβάζεται ως πληροφορία στα υψηλότερα επίπεδα της ιεραρχίας, όπου γίνεται αναθεώρηση αποφάσεων που θα οδηγήσει σε εξάλειψη του προβλήματος.

Η εργασία αυτή εντάσσεται στην ανάπτυξη ενός ενδεικτικού διαλογικού πληροφοριακού συστήματος για τον προγραμματισμό και έλεγχο παραγωγής σε βιομηχανίες πρώτων υλών, ονόματι ΗΦΑΙΣΤΟΣ, από την ομάδα Συστημάτων Στήριξης Αποφάσεων του Ινστιτούτου Πληροφορικής του ΙΤΕ.

Το ΗΦΑΙΣΤΟΣ αποτελείται από τα κάτωθι έξι υποσυστήματα:

-- υποσύστημα βραχυπρόθεσμου προγραμματισμού,

-- υποσύστημα λεπτομερειακού προγραμματισμού,

-- υποσύστημα υλοποίησης προγραμματισμού,

-- υποσύστημα αποθήκης,

-- υποσύστημα πωλήσεων, και

-- υποσύστημα διαχείρισης αρχείων.

Βασικότερα είναι τα συστήματα που αναφέρονται στην παραγωγική διαδικασία και προς την υλοποίηση αυτών γίνεται και η πλέον ουσιαστική δουλειά. Η συγκεκριμένη εργασία εντάσσεται στο υποσύστημα λεπτομερειακού προγραμματισμού παραγωγής.

Στο υποσύστημα αυτό, δίνεται η δυνατότητα στο χρήστη να διαμορφώνει πολλαπλά υποψήφια προγράμματα παραγωγής, ενδεχομένως βάσει διαφορετικών κριτηρίων, να ελέγχει μέσα από συγκεντρωτικά στοιχεία την αποτελεσματικότητά τους, και να επιλέγει το καταλληλότερο από αυτά. Τα προγράμματα αυτά προκύπτουν από αλγορίθμους χρονικού προγραμματισμού για διάφορα κριτήρια επίδοσης. Όμως οι λύσεις που προτείνονται δεν είναι δεσμευτικές. Ο διαλογικός χαρακτήρας του συστήματος, επιτρέπει στο χρήστη να τροποποιεί τις λύσεις προκειμένου να φτάσει σε κάποια επιθυμητή, την οποία τελικά και εκδίδει υπό μορφή εντολών παραγωγής.

Το πρόβλημα με το οποίο ασχολούμαστε, εμφανίζεται στο επίπεδο λεπτομερειακού προγραμματισμού σε πολλές βιομηχανίες, κυρίως πρώτων υλών (πλαστικών, φαρμάκων κ.ά.). Επίσης σε χαλυβουργεία, υφαντουργεία και γενικότερα σε δραστηριότητες που χρησιμοποιούν μια βασική μηχανή ή συγκρότημα μηχανών για την παραγωγή διαφόρων ειδών (ομάδων) προϊόντων. Βέβαια στη βιβλιογραφία αναφέρονται και άλλες εφαρμογές του, εντελώς διαφορετικού χαρακτήρα, όπως σε υπολογιστικά συστήματα για βέλτιστη χρήση μνήμης σε κατασκευές Βάσεων Δεδομένων [Ab-WaKa87], ή σε αεροδρόμια για βέλτιστη εξυπηρέτηση αεροσκαφών που ανήκουν σε διάφορους τύπους, διαφορετική της FCFS (First Come First Served) [Ps80].

1.6. Ορολογία - Συμβολισμοί

Για τη διευκόλυνση των αναγνωστών της παρούσας εργασίας, παρατίθεται στο σημείο αυτό, ένας κατάλογος συμβόλων που σχετίζονται με τις οντότητες και τους τελεστές που εμφανίζονται στην ανάλυσή μας. Σε όποιο σημείο της εργασίας κρίνεται απαραίτητη η χρήση νέου συμβολισμού, αυτός ορίζεται ρητά στο συγκεκριμένο σημείο και καθορίζεται επίσης ο χαρακτήρας που έχει (τοπικός ή γενικός συμβολισμός).

Πέντε είναι οι βασικές οντότητες που εμφανίζονται στην ανάλυση του προβλήματος :

- οι εργασίες,
- οι ομοειδείς και ισόχρονες εργασίες,
- οι πρωτογενείς παρτίδες,
- οι δευτερογενείς παρτίδες, και
- οι γενικευμένες παρτίδες.

Οι βασικοί τελεστές που εμφανίζονται είναι δύο :

- $C(\cdot)$: ο χρόνος περάτωσης του ορίσματος, και
- $ΧΛ(\cdot,\cdot,\cdot)$: ο χαρακτηριστικός λόγος που προκύπτει από μεγέθη σχετικά με τα ορίσματα.

Ακολουθεί ο κατάλογος συμβόλων :

$\mathbf{n}$: αριθμός εργασιών.
$\mathbf{B}$: αριθμός ομάδων.
$\mathbf{s_{i,j}}$: χρόνος εξάρμωσης της μηχανής για την επεξεργασία εργασιών της ομάδας j μετά από εργασίες της ομάδας i.
$\mathbf{s_j}$: (= $\mathbf{s_{i,j}}$ για κάθε i στην περίπτωση χρόνων εξάρμωσης ανεξάρτητων ακολουθίας) χρόνος εξάρμωσης της μηχανής για εκτέλεση εργασιών της ομάδας j.
$\mathbf{n_b}$: αριθμός εργασιών στην ομάδα b.
$\mathbf{p_{ib}}$: χρόνος επεξεργασίας της i-οστής εργασίας της ομάδας b.
$\mathbf{p_i^j}$: χρόνος επεξεργασίας ισόχρονων εργασιών τάξεως i, βάσει του SPT, της

ομάδας j - χαρακτηριστικός χρόνος της αντίστοιχης πρωτογενούς παρτίδας $ΠΠ_i^j$.

w_i^j : αριθμός ισόχρονων εργασιών τάξεως i, βάσει του SPT, της ομάδας j (ονομάζεται και βαρύτητα).

C_i : χρόνος περάτωσης της εργασίας i.

C(O) : συνολικός χρόνος περάτωσης των εργασιών στη διάταξη O.

$ΠΠ_i^j$: πρωτογενής παρτίδα, τάξεως i ως προς SPT της ομάδας j.

$ΔΠ_i^j$: δευτερογενής παρτίδα, τάξεως i ως προς μη φθίνουσα διάταξη χαρακτηριστικών λόγων της ομάδας j.

ΓΠ : γενικευμένη παρτίδα.

ΜΠ : μέγιστη παρτίδα.

ΧΛ(j,i,i+k−1) : χαρακτηριστικός λόγος για τις k ΠΠ της ομάδας j, με πρώτη την i-οστή, βάσει του SPT, πρωτογενή παρτίδα.

$λ_k^j$: τιμή χαρακτηριστικού λόγου της k τάξεως ΔΠ της ομάδας j.

2. Βιβλιογραφική ανασκόπηση

Λόγω της άμεσης πρακτικής εφαρμογής του, αλλά και της απλούστατης μαθηματικής διατύπωσής του, το πρόβλημα που μας απασχολεί εμφανίστηκε αρκετά νωρίς στη βιβλιογραφία. Βέβαια αρκετές παραλλαγές του μελετήθηκαν, καμμιά όμως μέχρι τις αρχές του 1989, δεν πραγματεύτηκε αυτήν ακριβώς που εμείς μελετούμε. Οι περισσότερες από τις παραλλαγές αυτές, ή αναφέρονται σε πολύ εξειδικευμένες περιπτώσεις του προβλήματος προκειμένου να εξαχθούν απλές λύσεις, ή λύνουν προβλήματα πολύ μικρού μεγέθους με χρήση κλασσικών απαριθμητικών μεθόδων.

Βασική βιβλιογραφική αναφορά για τη μελέτη μας αποτέλεσαν οι εργασίες των Monma και Potts [MoPo89] και Ψαραύτη [Ps80], οι οποίες έδωσαν ένα βασικό απαριθμητικό σχήμα για την επίλυση του προβλήματος. Η εργασία των Ahn και Hyun [AhHy90], που πρόσφατα έπεσε στην αντίληψή μας, προσεγγίζει το πρόβλημα με παρόμοια των δικών μας βήματα. Σ' αυτήν προτείνεται ένας προσεγγιστικός αλγόριθμος που στηρίζεται σε βελτιώσεις με αλλαγή στη θέση εκτέλεσης των εργασιών και ταυτόχρονη διατήρηση ενός ενδοομαδικού κανόνα διάταξης των εργασιών κατά μη φθίνοντα χρόνο επεξεργασίας (intra-group SPT), και ελέγχεται η ποιότητα των λύσεών του με χρήση του σχήματος δυναμικού προγραμματισμού που προτείνει ο Ψαραύτης [Ps80].

Οι παραπάνω εργασίες θα συζητηθούν ξεχωριστά στη συνέχεια, ενώ αρχικά θα παρουσιαστεί μια γενική ανασκόπηση των σχετικών με παραλλαγές του προβλήματος εργασιών που έχουν δημοσιευτεί.

2.1. Γενική ανασκόπηση

Η πρώτη από τις εργασίες που σχετίζονται με το πρόβλημα που μελετούμε, δημοσιεύτηκε από τον Smith το 1956 [Sm56]. Σ' αυτήν γίνεται ανάλυση του προβλήματος ελαχιστοποίησης του συνολικού σταθμισμένου χρόνου περάτωσης n εργασιών σε μια μηχανή ($n\,|1|\,|\sum w_i C_i$). Η λύση που εξάγεται είναι απλή και δίνεται από τον πολύ γνωστό κανόνα WSPT (Weighted Shortest Processing Time). Βάσει αυτού του κανόνα για τη βέλτιστη διάταξη ($J_1,J_2,\ldots,J_n$) n εργασιών για το $n\,|1|\,|\sum w_i C_i$ ισχύει : $\frac{p_1}{w_1} \leq \frac{p_2}{w_2} \leq \cdots \leq \frac{p_n}{w_n}$. Ειδική περίπτωση αποτελεί το πρόβλημα ελαχιστοποίησης του συνολικού χρόνου περάτωσης n εργασιών σε μιά μηχανή ($n\,|1|\,|\sum C_i$), το οποίο λύνεται με τον κανόνα SPT (διάταξη εργασιών κατά μη φθίνουσα σειρά των χρόνων επεξεργασίας τους). Έτσι και τα δυο πολύ σπουδαία αυτά προβλήματα, λύνονται βέλτιστα σε χρόνο O(nlogn).

Όπως γίνεται φανερό, στην εργασία του Smith δεν λαμβάνονται υπόψη χρόνοι εξάρμωσης, πράγμα που οδήγησε στην εύρεση απλών λύσεων.

Μια από τις πρώτες εργασίες στα προβλήματα χρονικού προγραμματισμού με χρόνους εξάρμωσης είναι αυτή του Glassey [Gl68], πάνω στην οποία βασίστηκε αργότερα ανάλογη εργασία των Gascon και Leachman [GaLe88]. Στην πρώτη από τις δύο αυτές συγγενείς εργασίες, αναλύεται το πρόβλημα ελαχιστοποίησης του πλήθους των χρόνων εξάρμωσης σε μια μηχανή, η οποία παράγει άνισες ποσότητες διαφορετικών προϊόντων σε μεταβλητές χρονικές περιόδους, με επιπλέον στόχο την ικανοποίηση των απαιτήσεων παραγωγής σε δεδομένους χρόνους παράδοσης. Στη δεύτερη εργασία, τίθεται ο επιπλέον στόχος ελαχιστοποίησης του κόστους τήρησης αποθέματος. Οι αλγόριθμοι που προτείνονται κατασκευάζουν δυναμικά ένα δίκτυο που αντιστοιχεί στο χώρο δυνατών (επιτρεπτών) καταστάσεων. Η εύρεση του ελάχιστου δρόμου στο δίκτυο αυτό δίνει και τη λύση στο εκάστοτε πρόβλημα. Αναφέρονται πειραματικά αποτελέσματα για μικρό πλήθος διαφορετικών ειδών και χρονικών περιόδων (10-50 είδη και 5-30 περιόδους).

Ένα άλλο είδος λειτουργικού κόστους, το οποίο οδηγεί σε προβλήματα ανάλογης δυσκολίας με αυτά στα οποία υπεισέρχονται χρόνοι εξάρμωσης, αποτελεί ο χρόνος μεταγωγής (switching time). Σχετικό πρόβλημα αντιμετωπίζει ο Sahney [Sa71] σε εργασία του, όπου ασχολείται με το πρόβλημα ελαχιστοποίησης του μέσου χρόνου συμπλήρωσης n εργασιών σε δύο μηχανές, όταν υπάρχει ένας εργάτης που πρέπει να κινείται ανάμεσα σ’ αυτές επιβαρύνοντας τον προγραμματισμό με κόστος μεταγωγής, και είναι καλά καθορισμένες οι εργασίες που πρέπει να επεξεργαστεί καθεμιά μηχανή. Κατασκευάζεται ένας αλγόριθμος διαμερισμού και φραγής, ο οποίος ενισχύεται με μηχανισμούς ελάττωσης των διακλαδώσεων, που προκύπτουν από προτάσεις που αποδείχτηκαν για το πρόβλημα, δεν αναφέρονται όμως καθόλου πειραματικά αποτελέσματα. Είναι σαφής η αντιστοιχία αυτού του προβλήματος με το $TCTS_j$ όταν έχουμε n εργασίες κατανεμημένες σε δύο ομάδες.

Φαινομενικά μεγάλη είναι και η ομοιότητα με το $TCTS_j$ του προβλήματος που πραγματεύονται οι Barnes και Vanston σε εργασία τους που δημοσίευσαν το 1981 [BaVa81]. Αυτοί ασχολούνται με το πρόβλημα ελαχιστοποίησης του συνολικού χρόνου εξάρμωσης αλλά και του συνολικού χρόνου έναρξης (ισοδύναμα περάτωσης) n εργασιών, όταν οι χρόνοι εξάρμωσης είναι εξαρτημένοι ακολουθίας και ο χρόνος έναρξης κάθε εργασίας ισούται με το άθροισμα των χρόνων επεξεργασίας (μόνο, και όχι και εξάρμωσης) των προηγούμενων στη διάταξη εργασιών. Για τη λύση του προτείνουν ένα υβριδικό αλγόριθμο δυναμικού προγραμματισμού / διαμερισμού και φραγής, όπου κριτήρια ελάττωσης σχετικά με φράγματα τιμών της συνάρτησης εφαρμόζονται κατά την κατασκευή των καταστάσεων του προβλήματος. Με τον τρόπο αυτό ελαττώνονται σημαντικά οι απαιτήσεις σε υπολογιστικό χώρο και χρόνο για την εκτέλεσή του. Αναφέρονται πειράματα με 10, 20 και 30 εργασίες τα οποία λύνονται σε χρόνους με μέσες τιμές 1.78, 9.34 και 32.92 δευτερόλεπτα αντίστοιχα, ενώ αντίστοιχες μέσες τιμές για τον καλύτερο αλγόριθμο διαμερισμού και φραγής είναι 0.27, 25.76 και > 150 δευτερόλεπτα.

Στη συνέχεια θα αναφερθούμε στην εργασία που δημοσίευσαν οι Dessouky και Deogun το 1981 [DeDe81]. Αυτοί ασχολούνται με το πρόβλημα ελαχιστοποίησης του μέσου (συνολικού) χρόνου περάτωσης n εργασιών σε μια μηχανή, όταν δεν είναι όλες οι εργασίες διαθέσιμες για εκτέλεση την ίδια χρονική στιγμή. Για τη λύση του προτείνουν αλγόριθμο διαμερισμού και φραγής, ο οποίος συνδυάζεται με κανόνες για ελάττωση του χώρου έρευνας. Αναφέρονται αποτελέσματα για προβλήματα με 20 και 50 εργασίες για τη λύση των οποίων απαιτούνται μέσοι χρόνοι 0.08 και 0.96 δευτερόλεπτα αντίστοιχα. Από την ανάλυση των αποτελεσμάτων προκύπτει αύξηση της δυσκολίας όταν οι τιμές των μεγεθών προέρχονται από μικρό σύνολο διαφορετικών τιμών.

Γενίκευση του προηγούμενου προβλήματος πραγματεύονται οι Hariri και Potts [HaPo83] σε σχετική εργασία τους, εφόσον λαμβάνουν επιπλέον υπόψη τους την ύπαρξη συντελεστών προτίμησης (βαρών) για καθεμιά εργασία. Προτείνουν και αυτοί με τη σειρά τους αλγόριθμο διαμερισμού και φραγής, ο οποίος χρησιμοποιεί κάτω φράγματα που προκύπτουν από χαλαρώσεις Lagrange των περιορισμών διαθεσιμότητας για εκτέλεση εργασιών στο αρχικό πρόβλημα. Επίσης κριτήρια ελάττωσης του χώρου καταστάσεων αυξάνουν την αποτελεσματικότητα του αλγορίθμου, που εφαρμόζεται σε προβλήματα με 20, 30, 40 και 50 εργασίες. Για όσα από αυτά λύθηκαν σε χρόνο μικρότερο του ενός λεπτού έγινε υπολογισμός μέσων χρόνων εκτέλεσης που έδωσε 0.06, 1.47, 14.89 και 30.58 δευτερόλεπτα για τις παραπάνω κατηγορίες μεγεθών προβλημάτων.

Με ελαχιστοποίηση του συνολικού χρόνου περάτωσης παρτίδων εργασιών ασχολούνται οι Dobson, Karmarkar και Rummel σε εργασία τους που δημοσιεύτηκε το 1987 [DoKaRu87]. Σ' αυτήν ορίζουν τα προβλήματα ροής (συμπλήρωσης και μεταφοράς εκτός του εργοστασίου) ειδών και παρτίδων, και δίνουν λύσεις που βασίζονται για το πρώτο σ' ένα απλό κανόνα διάταξης, και για το δεύτερο σε πρόταση που εξάγεται από ανάλυση με χαλάρωση Lagrange όταν έχουμε ένα είδος, και σε ευρηματικές διαδικασίες που χρησιμοποιούν την πρόταση αυτή, αν έχουμε πολλά είδη. Κατασκευάστηκαν πειράματα για την ανάλυση της επίδοσης των ευρηματικών διαδικασιών, οι λύσεις των οποίων συγκρίθηκαν με τιμές αντίστοιχων κάτω φραγμάτων της βέλτιστης τιμής. Τα προβλήματα που κατασκευάστηκαν περιελάμβαναν σύνολα 90 έως 900 προϊόντων που ανήκαν σε 3 έως 30 διαφορετικές κατηγορίες (είδη). Λύθηκαν όλα σε χρόνους μικρότερους του ενός δευτερολέπτου και οι μέσες αποκλίσεις τους από τα κάτω φράγματα των βέλτιστων τιμών δεν ξεπέρασαν σε καμμιά περίπτωση το 12%, ενώ 1109 από τα 1250 προβλήματα λύθηκαν βέλτιστα.

Την ίδια χρονιά δημοσιεύτηκε από τους Abdel-Wahab και Kameda μια εργασία, που αφορούσε στην εφαρμογή του προβλήματος ελαχιστοποίησης του συνολικού σταθμισμένου χρόνου περάτωσης εργασιών σε μια μηχανή, στη λύση του προβλήματος βέλτιστης χρήσης αποθηκευτικού χώρου (μνήμης) κατά τη διαδικασία ανακατασκευής και αναδιοργάνωσης μεγάλων Βάσεων Δεδομένων [Ab-WaKa87]. Η προσπάθειά τους στράφηκε κυρίως στην επινόηση και υποστήριξη ενός μετασχηματισμού, με τον οποίο, κάθε αποδοτικός αλγόριθμος για τη λύση του $\min\sum w_i C_i$, μπορεί να χρησιμοποιηθεί για τη λύση του προβλήματός τους.

Μέχρι το σημείο αυτό, η ανασκόπησή μας περιλαμβάνει εργασίες, στις οποίες ο συνολικός χρόνος περάτωσης είναι το μοναδικό κριτήριο βελτίστου. Πολλοί όμως ερευνητές ασχολήθηκαν με προβλήματα δύο κριτηρίων στα οποία ζητείται ελαχιστοποίηση του συνολικού χρόνου περάτωσης των εργασιών (σταθμισμένου ή μη), δεδομένου ότι οι χρόνοι περάτωσης ικανοποιούν δεδομένες σχέσεις ως προς τις προθεσμίες παράδοσης.

Έτσι ο Emmons το 1975 [Em75], αποδεικνύει την αστοχία του αλγορίθμου του Smith για το πρόβλημα $\min\sum w_i C_i$ δ.ό. $T_{\max} = 0$ με κατασκευή αντιπαραδείγματος, ενώ ο Burns σε αντίστοιχη εργασία το 1976 [Bu76], αποδεικνύει ότι η αστοχία του αλγορίθμου του Smith οφείλεται στην μη εύρεση τοπικού βελτίστου από τις αντιμεταθέσεις που προτείνει ο Smith. Δίνει επίσης ικανή και αναγκαία συνθήκη τοπικού ελαχίστου, και προτείνει διαδικασία εύρεσής του.

Οι Potts και Van Wassenhove [PoVa-Wa83], προχωρούν στην κατασκευή αλγορίθμου διαμερισμού και φραγής για τη λύση του ίδιου προβλήματος, επηρεασμένοι από την απόδειξη των Lenstra, Rinnooy Kan και Brucker ότι το πρόβλημα για αυθαίρετες τιμές βαρών w_i είναι NP-hard. Χρησιμοποιούν μέθοδο αναπροσαρμογής πολλαπλασιαστών Lagrange για την εξαγωγή κάτω φραγμάτων και ελέγχουν τις επιδόσεις του αλγορίθμου για προβλήματα με 20, 30, 40 και 50 εργασίες. Οι μέσοι χρόνοι που απαιτούνται είναι 0.04, 0.15, 0.67 και 2.94 δευτερόλεπτα αντίστοιχα.

Με ανάλογο τρόπο προσεγγίζει το πρόβλημα και ο Posner σε δύο εργασίες του το 1985 και το 1988 [Po85], [Po88]. Διαφέρει ο τρόπος με τον οποίο εξάγονται τα κάτω φράγματα, εφόσον στηρίζεται σε βασικές σχέσεις διαδοχής. Δίνονται αποτελέσματα για 10, 20 και 30 εργασίες με μέσους χρόνους 0.14, 0.29 και 0.45 δευτερόλεπτα αντίστοιχα.

΄Ενα συναφές δικριτηριακό πρόβλημα απασχόλησε τους Van Wassenhove και Gelders, οι οποίοι δημοσίευσαν σχετική εργασία το 1980 [Va-WaGe80]. Σ' αυτήν γίνεται μελέτη του προβλήματος ελαχιστοποίησης του συνολικού χρόνου περάτωσης των εργασιών και της μέγιστης καθυστέρησης ταυτόχρονα. Τα δυο κριτήρια θεωρούνται ισοδύναμα, γι' αυτό γίνεται απαρίθμηση ενός συνόλου αποτελεσματικών λύσεων με χρήση ψευδοπολυωνυμικού αλγορίθμου τάξης Ο($n^2\bar{p}logn$), όπου $\bar{p}=\frac{1}{n}\sum p_i$, από τις οποίες μπορεί να επιλεγεί η θεωρούμενη ως προτιμώτερη. Αναφέρονται πειραματικά αποτελέσματα με 10, 30 και 50 εργασίες τα οποία λύνονται σε χρόνους μικρότερους των 0.4 δευτερολέπτων.

΄Ομως εκτός από τις εργασίες που προτείνουν λύσεις σε διάφορα προβλήματα, συγγενή του $TCTS_j$, υπάρχουν άλλες οι οποίες ασχολούνται με την υπολογιστική πλοκή σχετικών με αυτό προβλημάτων.

΄Ετσι ο Sahni αποδεικνύει σε εργασία του [Sa76], ότι το $n\,|2|\;|\sum w_i C_i$ είναι NP-πλήρες, με αναγωγή στο γνωστό NP-πλήρες πρόβλημα της k-διαμέρισης, ενώ οι Kawaguchi και Kyan [KaKy88] αναφέρουν ότι το $n\,|2|preemption\,|\sum C_i$ είναι πρόβλημα άγνωστης πλοκής. Σε γενίκευση του συμπεράσματος του Sahni αναφέρονται οι Bruno, Coffman και Sethi σε εργασία τους το 1974 [BrCoSe74] ως εξής : το $n\,|\geq 1|\;|\sum w_i C_i$ είναι NP-πλήρες. Η απόδειξή τους γίνεται με αναφορά στο γνωστό NP-πλήρες πρόβλημα του "εκδρομικού σάκκου" (Knapsack).

Καμιά εργασία από τις παραπάνω δεν αναφέρεται στην πλοκή του $TCTS_j$. Μόνο οι Monma και Potts στην πολύ σπουδαία ανάλυσή τους ισχυρίζονται ότι παραμένει ακόμα ανοικτό πρόβλημα.

Την παραπάνω γενική ανασκόπηση θα ακολουθήσει η αναφορά σε τρεις εργασίες οι οποίες σχετίζονται περισσότερο με το πρόβλημα που αναλύουμε - θεωρητικά και πρακτικά.

2.2. Οι εργασίες των Monma-Potts και Ψαραύτη

Οι βασικότερες εργασίες, που σχετίζονται άμεσα με κατασκευή βέλτιστης λύσεως και θεωρητική μελέτη του $TCTS_j$, καθώς και μιας ειδικής περίπτωσής του, έγιναν από τους Monma και Potts το 1989 [MoPo89], και Ψαραύτη το 1980 [Ps80]. Σ' αυτές τις εργασίες αναπτύσσεται εκτός των άλλων, και το βασικό απαριθμητικό σχήμα που χρησιμοποιούμε στη δική μας μελέτη για την εξαγωγή βέλτιστων λύσεων.

Ειδικότερα οι Monma και Potts αναφέρονται στο πρόβλημα $n\,|\,1\,|\,seq-dep\,|\,f(\pi)$, ανάλογο του $n\,|\,1\,|\,seq-indep\,|\,f(\pi)$ που ορίσαμε στο κεφάλαιο 1.2. Αναφέρονται σε συναρτήσεις κόστους αθροιστικές και μέγιστης τιμής, και ορίζουν το πρόβλημα των διατεταγμένων κατηγοριών. Αποδεικνύουν ότι για το πρόβλημα αυτό, και για συναρτήσεις κόστους $\sum w_{ib} C_{ib}$ και $L_{\max}$ (μέγιστη βραδύτητα), υπάρχουν βέλτιστες διατάξεις στις οποίες οι εργασίες μέσα στις ομάδες είναι διατεταγμένες βάσει των κανόνων WSPT (Weighted Shortest Processing Time) και EDD (Earliest Due Date) αντίστοιχα. Επίσης για συνάρτηση κόστους ίση με το πλήθος των καθυστερημένων εργασιών, αποδεικνύουν ότι υπάρχει βέλτιστη διάταξη στην οποία οι μη καθυστερημένες εργασίες κάθε ομάδας διατάσσονται βάσει του EDD.

Στη συνέχεια προτείνουν έναν αλγόριθμο δυναμικού προγραμματισμού για την επίλυση του γενικού προβλήματος των διατεταγμένων κατηγοριών, καθώς και διάφορες μεταβλητές καταστάσεων προκειμένου να ελαττωθούν οι απαιτήσεις σε μνήμη του αλγορίθμου. Από την ανάλυση του αλγορίθμου προκύπτουν συγκεκριμένες εκφράσεις για την πλοκή των προβλημάτων $n\,|\,1\,|\,seq-dep\,|\sum w_i C_i$, $n\,|\,1\,|\,seq-dep\,|\,L_{\max}$ και $n\,|\,1\,|\,seq-dep\,|\,n_T$. Τα δύο πρώτα μπορεί να λυθούν σε χρόνο $O(B^2 n^B \min\{n^S, T\})$, όπου :

$$T = \sum_{b=1}^{B}\sum_{i=1}^{n_b} p_{ib} + \sum_{b=1}^{B} n_b \max_{0\le a\le B}\{s_{a,b}\}$$

και S ο αριθμός των διαφορετικών χρόνων εξάρμωσης (για seq-indep S≤B). Το τρίτο μπορεί να λυθεί σε χρόνο $O(B^2 n^B \min\{W, D, T\})$, όπου

$$W = \sum_{b=1}^{B}\sum_{i=1}^{n_b} w_{ib}$$

$$D = \max_{1 \le b \le B} \{d_{n_b b}\}$$

και T ορισμένο όπως προηγουμένως.

Εύκολα διαπιστώνουμε ότι η πλοκή του αλγορίθμου αυτού είναι εκθετική ως προς τον αριθμό των ομάδων. Όμως για την περίπτωση του $TCTS_j$ οι συγγραφείς θεωρούν την πλοκή του ως ανοιχτό πρόβλημα.

Ο Ψαραύτης στην εργασία του [Ps80] πραγματεύεται το πρόβλημα $n\,|\,1\,|\,seq-dep, p^j\,|\sum C_i$. Μια ειδική περίπτωση δηλαδή όπου ζητείται η ελαχιστοποίηση του συνολικού χρόνου περάτωσης n εργασιών σε μια μηχανή, δεδομένου ότι υπάρχουν χρόνοι εξάρμωσης εξαρτημένοι ακολουθίας και οι ομοειδείς εργασίες έχουν όλες ίδιο χρόνο επεξεργασίας. Για τη λύση του χρησιμοποιεί ένα παρόμοιο σχήμα δυναμικού προγραμματισμού μ’ αυτό των Monma και Potts, το οποίο για B ομάδες και $\lceil n/B \rceil$ εργασίες ανά ομάδα, απαιτεί χρόνο $B^2(\lceil n/B \rceil+1)^B$ και μνήμη για $B(\lceil n/B \rceil+1)^B$ διαφορετικές καταστάσεις. Έχει δηλαδή χρονική και χωρική πλοκή εκθετική ως προς τον αριθμό των ομάδων.

(Σημείωση : Λεπτομερής αναφορά στο σχήμα δυναμικού προγραμματισμού που χρησιμοποιείται στις εν λόγω εργασίες θα γίνει στο κεφάλαιο 5, όπου θα παρουσιαστεί αναλυτικά ο αλγόριθμος δυναμικού προγραμματισμού που χρησιμοποιήσαμε στη μελέτη μας).

2.3. Η εργασία των Ahn-Hyun

Η εργασία των Ahn και Hyun που δημοσιεύτηκε το 1990 [AhHy90], προσεγγίζει το πρόβλημα $n\,|\,1\,|\,seq-dep\,|\sum C_i$ με τρόπο ακριβώς ανάλογο αυτού που εμείς χρησιμοποιούμε για την ανάλυση του $TCTS_j$.

Δυστυχώς αγνοούσαμε μέχρι πρόσφατα την ύπαρξη αυτής της εργασίας, κι έτσι χρειάστηκε να δουλέψουμε από την αρχή για την κατασκευή της διαδικασίας επίλυσης που χρησιμοποιούμε. Όμως η από τη βάση ενασχόλησή μας με το πρόβλημα, και η άγνοια της ανάλογης αυτής εργασίας, είχε ένα καλό αποτέλεσμα : την εξαγωγή προτάσεων οι οποίες αν και δεν είναι αποδεδειγμένο θεωρητικά ή πειραματικά ότι οδηγούν σε καλύτερο ως προς την ποιότητα της λύσης προσεγγιστικό αλγόριθμο, βοηθούν ωστόσο, εφόσον ληφθούν υπόψη στην εξέλιξη του ακριβούς αλγορίθμου, στη βέλτιστη λύση προβλημάτων αρκετά μεγαλύτερων διαστάσεων. Κάτι τέτοιο συνεπάγεται και εγκυρότερη ένδειξη της ευστοχίας του προσεγγιστικού αλγορίθμου, εφόσον αυτός ελέγχεται για μεγαλύτερο εύρος προβλημάτων ως προς τις διαστάσεις.

Οι Ahn και Hyun χρησιμοποιούν το δυναμικό αλγόριθμο που προτείνει ο Ψαραύτης στην εργασία του [Ps80], προσαρμοσμένο ώστε να λαμβάνεται υπόψη η παρατήρηση ότι υπάρχει βέλτιστη διάταξη, στην οποία οι εργασίες κάθε ομάδας ακολουθούν τον κανόνα SPT (αναφέρουν αυτήν την ιδιότητα ως ενδοομαδικό κανόνα SPT). ΄Ετσι καταφέρνουν να λύσουν βέλτιστα προβλήματα μέχρι 50 εργασιών κατανεμημένων σε 5 ομάδες.

Παράλληλα αναπτύσσουν προσεγγιστικό αλγόριθμο για τη λύση προβλημάτων μεγαλύτερων διαστάσεων ο οποίος :

- ξεκινάει από αρχική λύση που ικανοποιεί τον ενδοομαδικό SPT,
- βελτιώνει τη λύση επιδιώκοντας μετατοπίσεις εργασιών προς τα εμπρός και προς τα πίσω, και
- σταματάει όταν δεν είναι δυνατό άλλες μετατοπίσεις να οδηγήσουν σε περαιτέρω βελτιώσεις.

Ο αλγόριθμος αυτός χρειάζεται χρόνο O($\lceil \frac{n}{B} \rceil^3 B^2$) σε κάθε επανάληψή του στη χειρότερη περίπτωση, και άρα μπορεί να χρησιμοποιηθεί για τη λύση πολύ μεγάλων προβλημάτων.

Από τα πειράματα δε, έχει εξαχθεί ότι η μέση απόκλισή του από το βέλτιστο είναι μικρότερη του 0.07% και η χείριστη περίπου 3%.

(Κατά τη διάρκεια της μελέτης μας κρίθηκε απαραίτητη η υλοποίηση του προσεγγιστικού αυτού αλγορίθμου και η πειραματική μελέτη της συμπεριφοράς του. Αυτή όμως η ενέργεια καθώς και οι κινήσεις που έγιναν μετά τα συμπεράσματα από τη μελέτη του αλγορίθμου παρουσιάζονται στην 6η ενότητα.)

3. Ανάλυση του προβλήματος $TCTS_j$

Στο κεφάλαιο αυτό θα δείξουμε ότι το $TCTS_j$ ανήκει στην κατηγορία προβλημάτων διατεταγμένων κατηγοριών και θα εξετάσουμε διεξοδικότερα την πλοκή του. Επίσης θα αποδείξουμε θεωρήματα που οδηγούν στη δημιουργία παρτίδων, οι οποίες αποτελούν βασικές οντότητές του. Ως παρτίδες θεωρούμε ακολουθίες ομοειδών εργασιών που εκτελούνται διαδοχικά σε βέλτιστες διατάξεις.

3.1. Το $TCTS_j$ ως πρόβλημα διατεταγμένων κατηγοριών

Έχουμε ήδη ορίσει στην παράγραφο 1.3 το γενικό πρόβλημα των διατεταγμένων κατηγοριών για διάφορες συναρτήσεις κόστους. Στο σημείο αυτό θα διατυπώσουμε και θα αποδείξουμε ένα θεώρημα που εντάσσει το $TCTS_j$ στην κατηγορία αυτή. Μάλιστα το θεώρημα αυτό είναι αναγκαία συνθήκη βελτίστου - επιβάλλει δηλαδή τη διάταξη ομοειδών εργασιών βάσει συγκεκριμένου κανόνα (του SPT) σε κάθε βέλτιστη λύση του $TCTS_j$.

Θεώρημα 3.1 Στη βέλτιστη λύση του προβλήματος $n\,|\,1\,|\,seq-indep\,|\sum C_i$, οι ομοειδείς εργασίες είναι διατεταγμένες βάσει του κανόνα SPT.

Απόδειξη: Ας είναι $O=(A,i,B,j,C)$ μια βέλτιστη διάταξη για το πρόβλημα $n\,|\,1\,|\,seq-indep\,|\sum C_i$. Αν οι i και j εργασίες ανήκουν στην ίδια ομάδα, θα δείξουμε ότι $p_i \le p_j$. Ορίζουμε ως:

a,b,c τα πλήθη των εργασιών στις υπακολουθίες A,B,C αντίστοιχα $(a+1+b+1+c=n)$.

$C_{1A}, C_{2A}, \cdots, C_{aA}$ τους χρόνους περάτωσης των εργασιών της υπακολουθίας A στη διάταξη O. Ανάλογα μεγέθη ορίζονται για τις υπακολουθίες B και C, ενώ C_i και C_j είναι οι χρόνοι περάτωσης των εργασιών i και j. (Στη διάταξη O' η οποία θα οριστεί στη συνέχεια τα αντίστοιχα μεγέθη συμβολίζονται τονούμενα π.χ C'_{1A}, C'_j.)

$s_{aA,i}$ το χρόνο εξάρμωσης για την εκτέλεση της εργασίας i αμέσως μετά την α τάξης εργασία της υπακολουθίας A. Ανάλογα ορίζονται τα μεγέθη $s_{i,1B}$, $s_{bB,j}$, $s_{j,1C}$ (καθώς επίσης και τα $s_{1B,2B}$ κ.ο.κ).

Έστω $p_i > p_j$, δηλαδή οι ομοειδείς εργασίες i και j παραβαίνουν τον κανόνα SPT στη διάταξη O. Με την αντιμετάθεση των εργασιών i και j προκύπτει μια νέα διάταξη η $O' = (A, j, B, i, C)$. Ας συμβολίσουμε με $C(O), C(O')$ τα αθροίσματα των χρόνων περάτωσης των εργασιών στις διατάξεις O και O' αντίστοιχα. Γι' αυτά έχουμε :

$$C(O) = C_{1A} + \cdots + C_{aA} + C_i + C_{1B} + \cdots + C_{bB} + C_j + C_{1C} + \cdots + C_{cC}$$

$$C(O') = C'_{1A} + \cdots + C'_{aA} + C'_j + C'_{1B} + \cdots + C'_{bB} + C'_i + C'_{1C} + \cdots + C'_{cC}$$

Όμως:

$$C_{1A} = C'_{1A}, \cdots, C_{aA} = C'_{aA} \qquad \text{(i)}$$

εφόσον η υπακολουθία A είναι ίδια και στις δυο διατάξεις.

$$C_i > C'_j \qquad \text{(ii)}$$

εφόσον $C_i = C_{aA} + s_{aA,i} + p_i$, $C'_j = C'_{aA} + s_{aA,j} + p_j$ και $C_{aA} = C'_{aA}$ από (i), $s_{aA,i} = s_{aA,j}$ από την υπόθεση ότι οι i και j είναι ομοειδείς καθώς και από την υπόθεση $p_i > p_j$.

$$C_{1B} > C'_{1B} \qquad \text{(iii)}$$

εφόσον $C_{1B} = C_i + s_{i,1B} + p_{1B} > C'_j + s_{j,1B} + p_{1B} = C'_{1B}$, λόγω της (ii) και του ότι $s_{i,1B} = s_{j,1B}$ εφόσον οι i και j είναι ομοειδείς.

$$C_{2B} > C'_{2B}, \cdots, C_{bB} > C'_{bB} \qquad \text{(iv)}$$

λόγω της (iii) και του γεγονότος ότι οι υπακολουθία B είναι ίδια και στις δυο διατάξεις.

$$C_j = C'_i \qquad \text{(v)}$$

εφόσον η περάτωση της εργασίας j στη διάταξη O γίνεται εφόσον περατωθούν ακριβώς ίδιες εργασίες και μετά από ίδιους χρόνους εξάρμωσης μ' αυτούς που απαιτούνται για την περάτωση της εργασίας i στη διάταξη O'.

$$C_{1C} = C'_{1C} \qquad \text{(vi)}$$

εφόσον $C_{1C} = C_j + s_{j,1C} + p_{1C} = C'_i + s_{i,1C} + p_{1C} = C'_{1C}$, λόγω της (v) και του ότι $s_{j,1C} = s_{i,1C}$ εφόσον οι i και j είναι ομοειδείς.

$$C_{2C} = C'_{2C}, \cdots, C_{cC} = C'_{cC} \qquad \text{(vii)}$$

εφόσον η υπακολουθία C είναι ίδια και στις δυο διατάξεις.

Από τις σχέσεις ορισμού των $C(O), C(O')$ καθώς και τις σχέσεις (i)-(vii) προκύπτει ότι

$$C(O) > C(O')$$

δηλαδή η διάταξη O που αρχικά υποθέσαμε βέλτιστη για το πρόβλημα $n\,|\,1\,|\,seq-indep\,|\sum C_i$ δεν είναι τέτοια, εφόσον οι ομοειδείς εργασίες i και j παραβαίνουν τον κανόνα SPT. □

3.2. Υπολογιστική πλοκή του $TCTS_j$

Στο κεφάλαιο αυτό γίνεται λόγος για την πλοκή του $TCTS_j$ η οποία εμφανίζεται ως ≪ανοιχτό προς έρευνα πρόβλημα≫ ([MoPo89]). Το θεώρημα που ακολουθεί δείχνει την ισοδυναμία του $TCTS_j$ με το TRP (Travelling Repairman Problem) του οποίου επίσης η πλοκή δεν έχει ακόμη προσδιοριστεί.

Θεώρημα 3.2 Το $TCTS_j$ είναι ισοδύναμο με το TRP

Απόδειξη: Έστω $S=[s_j]_{n\times 1}$ ο πίνακας που περιέχει τους χρόνους εξάρμωσης για την εκτέλεση των n εργασιών και $P=[p_j]_{n\times 1}$ ο πίνακας που περιέχει τους αντίστοιχους χρόνους επεξεργασίας.

Ορίζουμε τον πίνακα $TR=[tr_{ij}]_{n\times n}$, όπου

$$tr_{ij} = \begin{cases} +\infty & \text{αν } i=j \\ s_j+p_j & \text{αν } i\neq j \end{cases}$$

Αν κατασκευάσουμε προσανατολισμένο δίκτυο n+1 κόμβων, οι n πρώτοι από τους οποίους αντιστοιχούν στις n εργασίες και συνδέονται μεταξύ τους με συνδέσμους μήκους tr_{ij}, ενώ ο n+1-οστός αντιστοιχεί στην έναρξη και λήξη της εκτέλεσης των εργασιών και συνδέεται με τους υπόλοιπους κόμβους με συνδέσμους μήκους $tr_{n+1j}=tr_{ij}$ για τυχόν i , ενώ αυτοί συνδέονται μαζί του με συνδέσμους μήκους $tr_{jn+1}=0$, η λύση του TRP σ' αυτό το δίκτυο, δίνει βέλτιστη διάταξη εργασιών για το $TCTS_j$, δεδομένου ότι ο πρώτος και τελευταίος κόμβος στη βέλτιστη διαδρομή είναι ο n+1.

Πράγματι, αν $tr_1^*, \cdots ,tr_n^*$ είναι βέλτιστη διαδρομή ως προς το TRP έχουμε :

$$((n-1)tr_1^* + (n-2)tr_2^* + tr_n^*) = \min \Rightarrow$$

$$tr_1^* + (tr_1^* + tr_2^*) + \cdots + (tr_1^* + tr_2^* + \cdots + tr_n^*) = \min \Rightarrow$$

$$(s_1^* + p_1^*) + (s_1^* + p_1^* + s_2^* + p_2^*) + \cdots + (s_1^* + p_1^* + s_2^* + p_2^* + \cdots + s_n^* + s_n^*) = \min \Rightarrow$$

$$C_1^* + C_2^* + \cdots + C_n^* = \min$$

Δηλαδή, σ' αυτήν αντιστοιχεί βέλτιστη διάταξη εργασιών ως προς το $\sum C_i$ κριτήριο. □

Μετά από την απόδειξη αυτή, κρίνουμε απαραίτητο να αναφέρουμε ξανά, ότι οι περιπτώσεις του $TCTS_j$ όπου $s_j=0$ για κάθε j, καθώς και αυτή στην οποία είναι εξ' αρχής γνωστές οι μέγιστες παρτίδες της βέλτιστης λύσεως, λύνονται σε χρόνο τάξεως O(nlogn). Η πρώτη με εφαρμογή του κανόνα SPT και η δεύτερη με εφαρμογή ενός απλού κανόνα τον οποίο διατυπώνουμε και αποδεικνύουμε στο επόμενο κεφάλαιο.

3.3. Ορισμός παρτίδων στο $TCTS_j$ - Διάταξη μέγιστων παρτίδων στο $TCTS_j$

΄Εχουμε ήδη αναφερθεί σε προηγούμενα κεφάλαια στις βασικές οντότητες του $TCTS_j$, που ονομάζουμε παρτίδες. Στο κεφάλαιο αυτό θα ορίσουμε τις μέγιστες παρτίδες, και θα διατυπώσουμε και αποδείξουμε έναν απλό κανόνα που δίνει βέλτιστη διάταξή τους για το $TCTS_j$.

Ορισμός 3.3 Μέγιστες παρτίδες (ΜΠ) στο $TCTS_j$ είναι ακολουθίες ομοειδών εργασιών, οι οποίες δεν μπορεί να εκτελεστούν διαδοχικά με άλλες ομοειδείς τους εργασίες σε καμμιά βέλτιστη διάταξη για το $TCTS_j$.

Η περίπτωση των ανεξάρτητων ακολουθίας χρόνων εξάρμωσης, διευκολύνει την απόδειξη του ακόλουθου θεωρήματος, που προτείνει ένα βέλτιστο τρόπο (κανόνα) διάταξης ΜΠ.

Θεώρημα 3.3 ΄Εστω γ ΜΠ όχι κατ' ανάγκη ετεροειδείς μεταξύ τους. Αν με $M\Pi_{O(j)}$ συμβολίσουμε την τάξεως j ΜΠ ως προς κάποια διάταξη Ο, η διάταξη αυτή είναι βέλτιστη αν ακολουθεί τον κανόνα:

$$\frac{s_{O(1)} + \sum_{i \in M\Pi_{O(1)}} p_i}{w_{M\Pi_{O(1)}}} \leq \frac{s_{O(2)} + \sum_{i \in M\Pi_{O(2)}} p_i}{w_{M\Pi_{O(2)}}} \leq \cdots \leq \frac{s_{O(\gamma)} + \sum_{i \in M\Pi_{O(\gamma)}} p_i}{w_{M\Pi_{O(\gamma)}}}$$

($s_{O(j)}$ και $w_{M\Pi_{O(j)}}$ είναι ο χρόνος εξάρμωσης που απαιτείται για την έναρξη κατεργασίας της ΜΠ τάξεως j στη διάταξη Ο και η βαρύτητα αυτής αντίστοιχα)

Απόδειξη: Ας είναι $C_k^{M\Pi_{O(j)}}$ ο χρόνος περάτωσης της εργασίας τάξεως k της ΜΠ τάξεως j στη διάταξη O και $C(M\Pi_{O(j)})$ ο χρόνος περάτωσης της εν λόγω ΜΠ. Η διάταξη O θέλουμε να είναι τέτοια ώστε να επιτυγχάνεται η ελαχιστοποίηση του αθροίσματος των χρόνων περάτωσης όλων των εργασιών στις γ ΜΠ. Ο στόχος μας αν λάβουμε υπόψη την ύπαρξη ΜΠ εκφράζεται ως εξής:

$$\min \sum_{j=1}^{\gamma} \sum_{k=1}^{w_{M\Pi_{O(j)}}} C_k^{M\Pi_{O(j)}}$$

Όμως:

$$\sum_{j=1}^{\gamma} \sum_{k=1}^{w_{M\Pi_{O(j)}}} C_k^{M\Pi_{O(j)}} = \sum_{j=1}^{\gamma} \sum_{k=1}^{w_{M\Pi_{O(j)}}} [C(M\Pi_{O(j)}) - (p_{k+1} + \cdots + p_{w_{M\Pi_{O(j)}}})] =$$

$$\sum_{j=1}^{\gamma} \sum_{k=1}^{w_{M\Pi_{O(j)}}} C(M\Pi_{O(j)}) - \sum_{j=1}^{\gamma} \sum_{k=1}^{w_{M\Pi_{O(j)}}} (p_{k+1} + \cdots + p_{w_{M\Pi_{O(j)}}}) =$$

$$\sum_{j=1}^{\gamma} w_{M\Pi_{O(j)}} C(M\Pi_{O(j)}) - \sum_{j=1}^{\gamma} \sum_{k=1}^{w_{M\Pi_{O(j)}}} (p_{k+1} + \cdots + p_{w_{M\Pi_{O(j)}}})$$

Αν λάβουμε υπόψη ότι οι ΜΠ είναι εξ' αρχής καθορισμένες, η δεύτερη ποσότητα είναι σταθερή, ανεξάρτητη της διάταξης O. Έτσι λοιπόν το αρχικό πρόβλημα ανάγεται στην εύρεση διάταξης O η οποία θα πρέπει να ικανοποιεί το στόχο:

$$\min \sum_{j=1}^{\gamma} w_{M\Pi_{O(j)}} C(M\Pi_{O(j)})$$

Στο σημείο αυτό μπορούμε να θεωρήσουμε τις ΜΠ ως «γενικευμένες εργασίες» στις οποίες μπορεί να συνυπολογιστεί και ο απαιτούμενος χρόνος εξάρμωσης για την εκτέλεσή τους (εφόσον αυτός εξαρτάται κάθε φορά μόνο από την προς εκτέλεση ΜΠ κι όχι από τη διάταξη). Έτσι καλούμαστε να λύσουμε ένα πρόβλημα ελαχιστοποίησης του σταθμισμένου αθροίσματος των χρόνων συμπλήρωσης των «γενικευμένων εργασιών», χωρίς χρόνους εξάρμωσης. Όμως είναι ήδη γνωστό ότι μια διάταξη που ακολουθεί τον WSPT είναι βέλτιστη για το πρόβλημα αυτό ([Sm56]). Αν λοιπόν με $p_{M\Pi_{O(j)}}$ συμβολίσουμε το χρόνο επεξεργασίας της «γενικευμένης εργασίας» τάξεως $O(j)$ η βέλτιστη διάταξη για το καινούργιο πρόβλημα και επομένως και για το αρχικό είναι αυτή η οποία ικανοποιεί τη σχέση:

$$\frac{p_{M\Pi_{o(1)}}}{w_{M\Pi_{o(1)}}} \leq \frac{p_{M\Pi_{o(2)}}}{w_{M\Pi_{o(2)}}} \leq \cdots \leq \frac{p_{M\Pi_{o(\gamma)}}}{w_{M\Pi_{o(\gamma)}}} \iff$$

$$\frac{s_{O(1)} + \sum_{i \in M\Pi_{o(1)}} p_i}{w_{M\Pi_{o(1)}}} \leq \frac{s_{O(2)} + \sum_{i \in M\Pi_{o(2)}} p_i}{w_{M\Pi_{o(2)}}} \leq \cdots \leq \frac{s_{O(\gamma)} + \sum_{i \in M\Pi_{o(\gamma)}} p_i}{w_{M\Pi_{o(\gamma)}}} \quad \square .$$

3.4. Πρωτογενείς παρτίδες στο $TCTS_j$

Το πρώτο είδος παρτίδων οι οποίες είναι δυνατόν να προσδιοριστούν με σαφή τρόπο στη διαδικασία επίλυσης του $n\,|\,1\,|\,seq-indep\,|\sum C_i$, είναι οι πρωτογενείς. (Στο εξής θα γίνεται πολλές φορές χρήση της συντομογραφίας ΠΠ για αναφορά σε Πρωτογενείς Παρτίδες).

Το θεώρημα που ακολουθεί ενέπνευσε αλλά και επέβαλε την αναγκαιότητα του ορισμού των ΠΠ. Στη διατύπωσή του εμφανίζονται οι όροι ομοειδείς και ισόχρονες εργασίες, οι οποίοι αναφέρονται σε εργασίες της ίδιας ομάδας με ίσους χρόνους επεξεργασίας.

Θεώρημα 3.4 Υπάρχει βέλτιστη λύση του προβλήματος $n\,|\,1\,|\,seq-indep\,|\sum C_i$, στην οποία οι ομοειδείς και ισόχρονες εργασίες εκτελούνται διαδοχικά.

Απόδειξη: Ας είναι $O=(A,i,B,j,C)$ μια βέλτιστη διάταξη για το πρόβλημα $n\,|\,1\,|\,seq-indep\,|\sum C_i$. Αν οι i και j εργασίες είναι ομοειδείς και ισόχρονες, θα δείξουμε ότι υπάρχει διάταξη στην οποία αυτές εκτελούνται διαδοχικά και είναι καλύτερη ή ισότιμη της O ως προς το κριτήριο $\sum C_i$.

1) Υπόθεση Υ1:

Στην υπακολουθία B δεν υπάρχει εργασία ομοειδής και ισόχρονη των i και j. Με άλλα λόγια η εργασία j είναι η πρώτη ομοειδής και ισόχρονη εργασία της i που εμφανίζεται μετά απ’ αυτήν στη διάταξη O.

2) Υπόθεση Υ2:

Η υπακολουθία B δεν είναι κενή, ειδάλλως οι εργασίες i και j θα ήταν διαδοχικές στη διάταξη O.

Ιδιαίτερα χρήσιμη για την απόδειξη του θεωρήματος είναι η ακόλουθη παρατήρηση η οποία αποδεικνύεται αμέσως:

Παρατήρηση: Στην υπακολουθία B, τηρουμένων των υποθέσεων Y1 και Y2, δεν υπάρχουν εργασίες ομοειδείς των i και j.

Πράγματι έστω ότι η εργασία i ανήκει στην ίδια ομάδα με κάποια εργασία k της υπακολουθίας B. Βάσει της υπόθεσης Y1 οι εν λόγω εργασίες δεν είναι ισόχρονες. Αυτό σημαίνει ότι ή $p_i > p_k$ (i) ή $p_i < p_k$ (ii). Η δεύτερη των περιπτώσεων ισοδυναμεί με την $p_j < p_k$, εφόσον οι i και j είναι ισόχρονες ή αλλιώς $p_k > p_j$ (ii)′. Όμως τόσο η (i) όσο και η (ii)′ είναι περιπτώσεις στις οποίες έχουμε παράβαση του κανόνα SPT για εργασίες της ίδιας ομάδας, επομένως βάσει του θεωρήματος 3.2 η διάταξη O δεν είναι βέλτιστη.
Με ανάλογο τρόπο μπορούμε να δείξουμε την ισχύ της παρατήρησης για την εργασία j.

Προχωρούμε τώρα στην απόδειξη του θεωρήματος. Ορίζουμε ως:

a, b, c τα πλήθη των εργασιών στις υπακολουθίες A,B,C αντίστοιχα ($a+1+b+1+c=n$).

$C_{1A}, C_{2A}, \cdots, C_{aA}$ τους χρόνους περάτωσης των εργασιών της υπακολουθίας A στη διάταξη O. Ανάλογα μεγέθη ορίζονται για τις υπακολουθίες B και C, ενώ C_i και C_j είναι οι χρόνοι περάτωσης των εργασιών i και j. (Στις διατάξεις O' και O'' οι οποίες θα οριστούν στη συνέχεια τα αντίστοιχα μεγέθη συμβολίζονται τονούμενα π.χ $C'_{1A}, C'_j, C''_{1A}, C''_j$.)

$s_{aA,i}$ το χρόνο εξάρμωσης για την εκτέλεση της εργασίας i αμέσως μετά την εργασία τάξης a της υπακολουθίας A. Ανάλογα ορίζονται τα μεγέθη $s_{i,1B}$, $s_{bB,j}$ κ.ο.κ.

Έστω ότι μεταθέτουμε τις εργασίες i και j έτσι ώστε να εκτελούνται διαδοχικά, χωρίς καμμιά άλλη αλλαγή στη διάταξη O. Οι δυο δυνατές τέτοιες μεταθέσεις δίνουν τις ακόλουθες διατάξεις:

$$O' = (A, i, j, B, C) \quad \text{και} \quad O'' = (A, B, i, j, C)$$

Αν $\boldsymbol{C(O), C(O')}$ είναι τα αθροίσματα των χρόνων περάτωσης των εργασιών στις διατάξεις O και O' αντίστοιχα, έχουμε γι' αυτά:

$$\boldsymbol{C(O)} = C_{1A} + \cdots + C_{aA} + C_i + C_{1B} + \cdots + C_{bB} + C_j + C_{1C} + \cdots + C_{cC}$$

$$\boldsymbol{C(O')} = C'_{1A} + \cdots + C'_{aA} + C'_i + C'_j + C'_{1B} + \cdots + C'_{bB} + C'_{1C} + \cdots + C'_{cC}$$

Όμως:

$$C'_{1A} = C_{1A}, \cdots, C'_{aA} = C_{aA}, \ C'_i = C_i \qquad \text{(i)}$$

εφόσον μέχρι την περάτωση της i οι δυο διατάξεις είναι ίδιες.

$$C'_j = C'_i + p_j = C_i + p_j \qquad \text{(ii)}$$

εφόσον $C_i = C'_i$ και οι i και j είναι ομοειδείς.

$$C'_{1B} = C_{1B} + p_j, \cdots, C'_{bB} = C_{bB} + p_j \qquad \text{(iii)}$$

εφόσον στη διάταξη O' οι εργασίες της υπακολουθίας B καθυστερούν την έναρξή τους κατά p_j σε σχέση με την διάταξη O. (Δεν παρεμβάλλεται επιπλέον χρόνος εξάρμωσης εφόσον οι εργασίες i και j ανήκουν στην ίδια ομάδα.)

$$C_j = C_{bB} + s_{bB,j} + p_j \qquad \text{(iv)}$$

εφόσον η j εκτελείται στη διάταξη O αμέσως μετά τη b στη σειρά εργασία της υπακολουθίας B, με την οποία βάσει προηγούμενης παρατήρησης είναι ετεροειδής.

Για τα $C'_{1C}, \cdots, C'_{cC}$ διακρίνουμε τρεις περιπτώσεις:

(a) Η εργασία j είναι ομοειδής με την πρώτη στη σειρά εργασία της υπακολουθίας C. Τότε, βάσει της παρατήρησης που έγινε παραπάνω, η τελευταία εργασία της υπακολουθίας B δεν ανήκει στην ίδια ομάδα με την πρώτη εργασία της υπακολουθίας C. Αν συμβαίνει κάτι τέτοιο, οι εργασίες της C στη διάταξη O' εκτελούνται με διαφορά έναρξης $(s_{j,1B} + s_{bB,1C}) - (s_{i,1B} + s_{bB,j})$ απ’ ότι στη διάταξη O. Όμως εφόσον οι εργασίες i και j είναι ομοειδείς έχουμε ότι $s_{j,1B} = s_{i,1B}$. Επίσης η j είναι ομοειδής της πρώτης στη σειρά εργασίας της C και επομένως $s_{bB,1C} = s_{bB,j}$. Βάσει των παραπάνω:

$$C'_{1C} = C_{1C}, \ldots, C'_{cC} = C_{cC} \qquad \text{(va)}$$

(b) Τόσο η εργασία j όσο και η τελευταία στη σειρά εργασία της υπακολουθίας B είναι ετεροειδείς της πρώτης στη σειρά εργασίας της υπακολουθίας C. Αν συμβαίνει κάτι τέτοιο, οι εργασίες της C στη διάταξη O' εκτελούνται με διαφορά έναρξης $(s_{j,1B} + s_{bB,1C}) - (s_{i,1B} + s_{bB,j} + s_{j,1C})$ απ’ ότι στη διάταξη O. Όμως εφόσον οι εργασίες i και j είναι ομοειδείς και οι χρόνοι εξάρμωσης ανεξάρτητοι ακολουθίας έχουμε ότι $s_{j,1B} = s_{i,1B}$ και $s_{bB,1C} = s_{j,1C}$. Επομένως:

$$C'_{1C} = C_{1C} - s_{bB,j}, \ldots, C'_{cC} = C_{cC} - s_{bB,j} \qquad \text{(vb)}$$

(c) Η εργασία j είναι ετεροειδής με την πρώτη στη σειρά εργασία της υπακολουθίας C, ενώ η τελευταία στη σειρά εργασία της υπακολουθίας B είναι ομοειδής της πρώτης στη σειρά εργασία της υπακολουθίας C. Αν συμβαίνει κάτι τέτοιο, οι εργασίες της C στη διάταξη O' εκτελούνται με διαφορά έναρξης $s_{j,1B} - (s_{i,1B} + s_{bB,j} + s_{j,1C})$ απ' ότι στη διάταξη O. Όμως εφόσον οι εργασίες i και j είναι ομοειδείς έχουμε ότι $s_{j,1B} = s_{i,1B}$ και επομένως:

$$C'_{1C} = C_{1C} - (s_{bB,j} + s_{j,1C}), \ldots, C'_{cC} = C_{cC} - (s_{bB,j} + s_{j,1C}) \qquad \text{(vc)}$$

Από τις σχέσεις ορισμού των $C(O), C(O')$ καθώς και τις σχέσεις (i)-(iv) έχουμε για καθεμιά από τις προηγούμενες περιπτώσεις:

$$C(O') - C(O) = (C_i + p_j) - (C_{bB} + s_{bB,j} + p_j) + bp_j \qquad \text{(I)}$$

$$C(O') - C(O) = (C_i + p_j) - (C_{bB} + s_{bB,j} + p_j) + bp_j - cs_{bB,j} \qquad \text{(II)}$$

$$C(O') - C(O) = (C_i + p_j) - (C_{bB} + s_{bB,j} + p_j) + bp_j - c(s_{bB,j} + s_{j,1C}) \qquad \text{(III)}$$

Αν λοιπόν $(C_i + p_j) - (C_{bB} + s_{bB,j} + p_j) + bp_j < 0$, τότε και για τις τρεις διαφορετικές περιπτώσεις που εξετάζονται παραπάνω ισχύει ότι:

$$C(O) > C(O')$$

δηλαδή η διάταξη O' στην οποία οι i και j εκτελούνται διαδοχικές είναι καλύτερη της O, η οποία αρχικά υποτέθηκε βέλτιστη.

Έστω ότι η προηγούμενη υπόθεση δεν ισχύει, δηλαδή:

$$(C_i + p_j) - (C_{bB} + s_{bB,j} + p_j) + bp_j \geq 0 \qquad \text{(S)}$$

Τότε, όπως φαίνεται τουλάχιστον από τη σχέση (I), η διάταξη O' δεν είναι πάντα καλύτερη της O. Αυτό που ακολουθεί, είναι η απόδειξη του γεγονότος ότι στην περίπτωση αυτή η διάταξη $O'' = (A, B, i, j, C)$ η οποία ορίστηκε παραπάνω, είναι καλύτερη της O. Αν δηλαδή γίνει στην O μια άλλη δυνατή μετάθεση, έτσι ώστε οι ομοειδείς και ισόχρονες εργασίες i και j να εκτελούνται διαδοχικά, η διάταξη που προκύπτει είναι καλύτερη της αρχικής.

Κατ' αναλογία με τα $C(O), C(O')$ ορίζουμε το $C(O'')$ για το οποίο έχουμε:

$$C(O'') = C''_{1A} + \cdots + C''_{aA} + C''_{1B} + \cdots + C''_{bB} + C''_i + C''_j + C''_{1C} + \cdots + C''_{cC}$$

Όμως:

$$C''_{1A} = C_{1A},\ \cdots,\ C''_{aA} = C_{aA} \tag{iS}$$

εφόσον μέχρι την περάτωση της τελευταίας στη σειρά εργασίας της υπακολουθίας A οι δυο διατάξεις είναι ίδιες.

Για τους χρόνους περάτωσης των υπόλοιπων εργασιών στη διάταξη O'' διακρίνουμε τρεις περιπτώσεις:

(aS) Η τελευταία στη σειρά εργασία της υπακολουθίας A είναι ομοειδής της i. Τότε βάσει προηγούμενης παρατήρησης, η τελευταία εργασία της A είναι ετεροειδής της πρώτης στη σειρά εργασίας της B και επίσης η τελευταία στη σειρά εργασία της B είναι ετεροειδής της i. Αν συμβαίνει κάτι τέτοιο:

$$C''_{1B} = C_{1B} - p_i,\ \cdots,\ C''_{bB} = C_{bB} - p_i \tag{iiaS}$$

εφόσον οι εργασίες της υπακολουθίας B στη διάταξη O'' εκτελούνται με διαφορά έναρξης $s_{aA,1B} - (p_i + s_{i,1B})$ απ' ότι στη διάταξη O και επίσης $s_{aA,1B} = s_{i,1B}$ καθώς η τελευταία εργασία της A και η i είναι ομοειδείς.

$$C''_i = C''_{bB} + s_{bB,i} + p_i = C_{bB} - p_i + s_{bB,i} + p_i = C_{bB} + s_{bB,i} \tag{iiiaS}$$

λόγω της τελευταίας των σχέσεων (iiaS) και της σειράς εκτέλεσης της i στη διάταξη O''.

$$C''_j = C''_i + p_j = C_{bB} + s_{bB,i} + p_j = C_j \tag{ivaS}$$

λόγω της (iiiaS), της σειράς εκτέλεσης της j στις δυο διατάξεις, και του ότι οι i και j είναι ομοειδείς.

$$C''_{1C} = C_{1C},\ \ldots,\ C''_{cC} = C_{cC} \tag{vaS}$$

λόγω της (ivaS) και του ότι μετά την εργασία j η διατάξη O'' είναι ίδια με την O.

(bS) Η τελευταία στη σειρά εργασία της υπακολουθίας A είναι ετεροειδής τόσο της i όσο και της πρώτης στη σειρά εργασίας της υπακολουθίας B. Αν συμβαίνει κάτι τέτοιο:

$$C''_{1B} = C_{1B} - (p_i + s_{\alpha A,i}),\ \cdots,\ C''_{bB} = C_{bB} - (p_i + s_{\alpha A,i}) \tag{iibS}$$

εφόσον οι εργασίες της υπακολουθίας B στη διάταξη O'' εκτελούνται με διαφορά έναρξης $s_{aA,1B} - (s_{aA,i} + p_i + s_{i,1B})$ απ' ότι στη διάταξη O και επίσης $s_{aA,1B} = s_{i,1B}$ καθώς είναι χρόνοι εξάρμωσης ανεξάρτητοι ακολουθίας.

$$C''_i = C''_{bB} + s_{bB,i} + p_i = C_{bB} - (p_i + s_{\alpha A,i}) + s_{bB,i} + p_i$$

$$= C_{bB} + s_{bB,i} - s_{\alpha A,i} = C_{bB} \quad \text{(iiibS)}$$

λόγω της τελευταίας των σχέσεων (iibS) και της σειράς εκτέλεσης της i στη διάταξη O''.

$$C_j'' = C_i'' + p_j = C_{bB} + p_j$$

$$= C_{bB} + s_{bB,j} + p_j - s_{bB,j} = C_j - s_{bB,j} \quad \text{(ivbS)}$$

λόγω της (iiibS) και της σειράς εκτέλεσης της j στις δυο διατάξεις.

$$C''_{1C} = C_{1C} - s_{\alpha A,i}, \cdots, C''_{cC} = C_{cC} - s_{\alpha A,i} \quad \text{(vbS)}$$

εφόσον οι εργασίες της υπακολουθίας C στη διάταξη O'' εκτελούνται με διαφορά έναρξης $(s_{aA,1B} + s_{bB,i} + s_{j,1C}) - (s_{aA,i} + s_{i,1B} + s_{bB,j} + s_{j,1C})$, $s_{aA,1B} = s_{i,1B}$ ως χρόνοι εξάρμωσης ανεξάρτητοι ακολουθίας και $s_{bB,i} = s_{bB,j}$ εφόσον οι εργασίες i και j είναι ομοειδείς.

(cS) Η τελευταία στη σειρά εργασία της υπακολουθίας A είναι ετεροειδής της i αλλά ομοειδής της πρώτης στη σειρά εργασία της υπακολουθίας B. Αν συμβαίνει κάτι τέτοιο:

$$C''_{1B} = C_{1B} - (p_i + s_{aA,i} + s_{i,1B}), \cdots,$$

$$C''_{bB} = C_{bB} - (p_i + s_{aA,i} + s_{i,1B}) \quad \text{(iicS)}$$

εφόσον οι εργασίες της υπακολουθίας B στη διάταξη O'' εκτελούνται με διαφορά έναρξης $0 - (p_i + s_{aA,i} + s_{i,1B})$ απ' ότι στη διάταξη O.

$$C_i'' = C''_{bB} + s_{bB,i} + p_i = C_{bB} - (p_i + s_{aA,i} + s_{i,1B}) + s_{bB,i} + p_i$$

$$= C_{bB} - s_{i,1B} \quad \text{(iiicS)}$$

λόγω της τελευταίας των σχέσεων (iicS), της σειράς εκτέλεσης της i στη διάταξη O'' και του ότι $s_{aA,i} = s_{bB,i}$ εφόσον πρόκειται για χρόνους εξάρμωσης ανεξάρτητους ακολουθίας.

$$C_j'' = C_i'' + p_j = C_{bB} - s_{i,1B} + p_j = (C_{bB} + s_{bB,j} + p_j) - s_{bB,j} - s_{i,1B}$$

$$= C_j - (s_{i,1B} + s_{bB,j}) \quad \text{(ivcS)}$$

λόγω της (iiicS) και της σειράς εκτέλεσης της j στις δυο διατάξεις.

$$C''_{1C} = C_{1C} - (s_{aA,i} + s_{i,1B}), \cdots, C''_{cC} = C_{cC} - (s_{aA,i} + s_{i,1B}) \quad \text{(vcS)}$$

εφόσον οι εργασίες της υπακολουθίας C στη διάταξη O'' εκτελούνται με διαφορά έναρξης $(s_{bB,i} + s_{j,1C}) - (s_{aA,i} + s_{i,1B} + s_{bB,j} + s_{j,1C})$ και $s_{bB,i} = s_{bB,j}$ εφόσον οι εργασίες i και j είναι ομοειδείς.

Από τις σχέσεις ορισμού των $C(O), C(O'')$ καθώς και τις σχέσεις (iS) έχουμε για καθεμιά από τις παραπάνω περιπτώσεις:

$$C(O'') - C(O) = -bp_i + C_{bB} + s_{bB,i} - C_i$$

$$= -\{ (C_i + p_j) - (C_{bB} + s_{bB,j} + p_j) + bp_j \} \leq 0 \qquad \text{(IS)}$$

λόγω της (S) και του $p_i = p_j$.

$$C(O'') - C(O) = -b(p_i + s_{\alpha A,i}) + C_{bB} - C_i - s_{bB,j} - cs_{\alpha A,i}$$

$$= -\{ (C_i + p_j) - (C_{bB} + s_{bB,j} + p_j) + bp_j \}$$

$$-(b + c)s_{\alpha A,i} < 0 \qquad \text{(IIS)}$$

λόγω της (S), του $p_i = p_j$ και του $s_{\alpha A,i} > 0$.

$$C(O'') - C(O) = -b(p_i + s_{aA,i} + s_{i,1B}) + C_{bB} - s_{i,1B} - C_i - s_{i,1B} - s_{bB,j}$$

$$- c(s_{aA,i} + s_{i,1B})$$

$$= -\{ (C_i + p_j) - (C_{bB} + s_{bB,j} + p_j) + bp_j \}$$

$$-(b + c + 2)s_{i,1B} - (b + c)s_{aA,i} < 0 \qquad \text{(IIIS)}$$

λόγω της (S), του $p_i = p_j$, του $s_{i,1B} > 0$, και του $s_{aA,i} > 0$.

Από τις σχέσεις (IS), (IIS), (IIIS) φαίνεται ότι η διάταξη O'' στην οποία οι i και j εκτελούνται διαδοχικές είναι εξίσου καλή ή καλύτερη της O. □

Έχοντας πλέον υπόψη το θεώρημα 3.4 μπορούμε να προχωρήσουμε στον ορισμό των Πρωτογενών Παρτίδων (ΠΠ).

Ορισμός 3.4 Ονομάζουμε ΠΠ μιας ομάδας ομοειδών εργασιών, κάθε μέγιστο υποσύνολο αυτής που περιέχει εργασίες με ίσους χρόνους επεξεργασίας.

Η ΠΠ είναι μια θεμελιώδης οντότητα εφόσον περιγράφει το ελάχιστο σύνολο εργασιών μιας ομάδας οι οποίες εκτελούνται διαδοχικές σε βέλτιστη διάταξη για το πρόβλημα $n \mid 1 \mid seq-indep \mid \sum C_i$.

Στη συνέχεια θα γίνεται χρήση των πρωτογενών παρτίδων ως θεμελιωδών οντοτήτων του προβλήματος. Κατά μία έννοια θα θεωρούνται αυτές πλέον οι βασικές οντότητες μέσα στις ομάδες αντί για τις εργασίες. Για το σκοπό αυτό θα χρειαστεί να ορίσουμε χαρακτηριστικά μεγέθη για τις ΠΠ, ανάλογα του χρόνου επεξεργασίας των εργασιών, που θα βοηθήσουν στην περαιτέρω ανάλυση του προβλήματος. Ορίζουμε λοιπόν ως:

- χαρακτηριστικό χρόνο επεξεργασίας της i στη σειρά ΠΠ της ομάδας j, p_i^j, το χρόνο επεξεργασίας μιας εργασίας που ανήκει σ' αυτήν την ΠΠ. Τον ονομάσαμε δε χαρακτηριστικό γιατί είναι αυτός που στην ουσία καθορίζει τις ΠΠ στις ομάδες.
- βαρύτητα της i στη σειρά ΠΠ της ομάδας j, w_i^j, το πλήθος των εργασιών της ομάδας j που απαρτίζουν την i στη σειρά ΠΠ αυτής.

Μετά τους ορισμούς αυτούς διατυπώνουμε δύο προτάσεις οι οποίες καθορίζουν τον τρόπο διάταξης των ΠΠ. Η πρώτη απ' αυτές έχει ήδη διατυπωθεί έμμεσα στην προηγούμενη παράγραφο, ενώ η δεύτερη είναι συνέπεια προηγουμένων προτάσεων.

Πρόταση 3.4.1 Οι ΠΠ μιας ομάδας είναι τα ελάχιστα υποσύνολα εργασιών της ομάδας (εκτός βεβαίως των ίδιων των εργασιών), που εκτελούνται αδιάσπαστα σε κάποια βέλτιστη διάταξη για τη λύση του $n\,|\,1\,|\,seq-indep\,|\sum C_i$.

Απόδειξη: Βάσει του θεωρήματος 3.4, υπάρχει βέλτιστη διάταξη για τη λύση του $n\,|\,1\,|\,seq-indep\,|\sum C_i$ στην οποία ομοειδείς και ισόχρονες εργασίες εκτελούνται διαδοχικά. Όμως με βάση προηγούμενο ορισμό, αυτές αποτελούν ΠΠ, οι οποίες στην εν λόγω βέλτιστη διάταξη δεν διασπώνται.

Πρόταση 3.4.2 Οι ΠΠ μιας ομάδας ακολουθούν τον κανόνα SPT ως προς τους χαρακτηριστικούς τους χρόνους επεξεργασίας σε κάθε βέλτιστη διάταξη του $n\,|\,1\,|\,seq-indep\,|\sum C_i$.

Απόδειξη: Αν υπήρχε διάταξη στην οποία οι ΠΠ κάποιας ομάδας παρέβαιναν τον SPT ως προς τους χαρακτηριστικούς τους χρόνους, αυτή δεν θα ήταν βέλτιστη βάσει του θεωρήματος 3.1, εφόσον τότε οι εργασίες αυτής της ομάδας θα παρέβαιναν τον SPT.

Θα πρέπει να δοθεί ιδιαίτερη προσοχή στην πρόταση 3.4.2, ως προς το ότι αναφέρεται σε χαρακτηριστικούς και όχι συνολικούς χρόνους ΠΠ. Σε πολλές περιπτώσεις, ΠΠ με μικρό συνολικό χρόνο και μικρή βαρύτητα εκτελείται μετά από ΠΠ με μεγάλο συνολικό χρόνο και μεγάλη βαρύτητα.

3.5. Δευτερογενείς παρτίδες στο $TCTS_j$ - Ορισμός χαρακτηριστικών λόγων ΧΛ

Το δεύτερο είδος παρτίδων που ορίζουμε για την ανάλυση του $n\,|\,1\,|\,seq-indep\,|\sum C_i$, είναι οι δευτερογενείς παρτίδες, ΔΠ.

Όπως εύκολα μπορούμε να φανταστούμε, μια τέτοια οντότητα θα είναι η γενίκευση της έννοιας της ΠΠ αναφορικά πλέον με ΠΠ αντί για εργασίες. Έτσι μια ΔΠ θα απαρτίζεται από ΠΠ μιας ομάδας και θα εκτελείται αδιάσπαστη σε βέλτιστη διάταξη του $n\,|\,1\,|\,seq-indep\,|\sum C_i$ στην οποία έπεται ετεροειδούς ΔΠ. Μια ΔΠ θα περιλαμβάνει ολόκληρες ΠΠ κάποιας ομάδας γιατί έτσι περιορίζουμε το χώρο έρευνάς μας σε διατάξεις που απαρτίζονται από αδιάσπαστες ΠΠ, ανάμεσα στις οποίες ξέρουμε ήδη (πρόταση 3.4.1) ότι υπάρχει κάποια βέλτιστη ως προς το $n\,|\,1\,|\,seq-indep\,|\sum C_i$.

Τα ακόλουθα θεωρήματα είναι αυτά που δείχνουν τον τρόπο με τον οποίο κατασκευάζονται οι ΔΠ.

Θεώρημα 3.5.1 Έστω $\Pi\Pi_i{}^j$ και $\Pi\Pi_{i+1}{}^j$ η i και i+1 στη σειρά, βάσει του SPT, ΠΠ της ομάδας j, με χαρακτηριστικούς χρόνους $p_i{}^j$ και $p_{i+1}{}^j$ και βαρύτητες $w_i{}^j$ και $w_{i+1}{}^j$ αντίστοιχα. Αν

$$\frac{s_{kj} + w_i{}^j p_i{}^j + w_{i+1}{}^j p_{i+1}{}^j}{w_i{}^j + w_{i+1}{}^j} \le \frac{s_{kj} + w_i{}^j p_i{}^j}{w_i{}^j}$$

οι $\Pi\Pi_i{}^j$ και $\Pi\Pi_{i+1}{}^j$ εκτελούνται διαδοχικές σε κάθε βέλτιστη διάταξη του $n\,|\,1\,|\,seq-indep\,|\sum C_i$, που απαρτίζεται από αδιάσπαστες ΠΠ και στην οποία πριν από την $\Pi\Pi_i{}^j$ διατάσσονται ΠΠ κάποιας ομάδας k διαφορετικής της j. (με s_{kj} συμβολίζουμε το χρόνο εξάρμωσης από την αμέσως προηγούμενη στη διάταξη ΠΠ της $\Pi\Pi_i{}^j$.)

Απόδειξη: Ας είναι $O=(A, \Pi\Pi_i{}^j, B, \Pi\Pi_{i+1}{}^j, C)$ μια βέλτιστη διάταξη για το πρόβλημα $n\,|\,1\,|\,seq-indep\,|\sum C_i$. Αν οι $\Pi\Pi_i{}^j$ και $\Pi\Pi_{i+1}{}^j$, ικανοποιούν τη σχέση που εμφανίζεται στη διατύπωση του παρόντος θεωρήματος θα δείξουμε ότι διάταξη στην οποία εκτελούνται διαδοχικές είναι καλύτερη ως προς το $n\,|\,1\,|\,seq-indep\,|\sum C_i$.

Εύκολα μπορούμε να παρατηρήσουμε ότι σε μια διάταξη όπως την Ο στην υπακολουθία Β δεν είναι δυνατό να υπάρχουν ΠΠ ομοειδείς των $\Pi\Pi_i{}^j$ και $\Pi\Pi_{i+1}{}^j$. Αυτό συμβαίνει εφόσον έχουμε υποθέσει τις $\Pi\Pi_i{}^j$ και $\Pi\Pi_{i+1}{}^j$ διαδοχικές βάσει του SPT και ξέρουμε επίσης ότι σε κάθε βέλτιστη διάταξη για το $n\,|\,1\,|\,seq-indep\,|\sum C_i$ οι ΠΠ μιας ομάδας ακολουθούν τον SPT (πρόταση 3.4.2).

Για τη συνέχεια της απόδειξης ορίζουμε ως:

a, b, c τα πλήθη των εργασιών στις υπακολουθίες A,B,C αντίστοιχα ($a + w_i^j + b + w_{i+1}^j + c = n$).

$C_{1A}, C_{2A}, \cdots, C_{aA}$ τους χρόνους περάτωσης των εργασιών της υπακολουθίας A στη διάταξη O. Ανάλογα μεγέθη ορίζονται για τις υπακολουθίες B και C, ενώ $C_{1\Pi\Pi_i^j}, \cdots, C_{w_i^j \Pi\Pi_i^j}$, και $C_{1\Pi\Pi_{i+1}^j}, \cdots, C_{w_{i+1}^j \Pi\Pi_{i+1}^j}$ είναι οι χρόνοι περάτωσης των εργασιών των $\Pi\Pi_i^j$ και $\Pi\Pi_{i+1}^j$. (Στις διατάξεις O' και O'' οι οποίες θα οριστούν στη συνέχεια τα αντίστοιχα μεγέθη συμβολίζονται τονούμενα.)

$s_{aA,j}$ το χρόνο εξάρμωσης για την εκτέλεση ΠΠ της ομάδας j αμέσως μετά την a τάξης εργασία της υπακολουθίας A. Ανάλογα ορίζονται τα μεγέθη $s_{j,1B}$, $s_{bB,j}$ κ.ο.κ

TcB το συνολικό χρόνο επεξεργασίας της υπακολουθίας B, ήτοι το άθροισμα των χρόνων επεξεργασίας των εργασιών που περιλαμβάνει και των χρόνων εξάρμωσης που παρεμβάλλονται κατά την εκτέλεση αυτών των εργασιών.

Έστω ότι μεταθέτουμε τις $\Pi\Pi_i^j$ και $\Pi\Pi_{i+1}^j$ έτσι ώστε να εκτελούνται διαδοχικές, χωρίς άλλη αλλαγή στη μέχρι τώρα υποτιθέμενη βέλτιστη διάταξη O. Οι δυο δυνατές τέτοιες μεταθέσεις δίνουν τις ακόλουθες διατάξεις:

$$O' = (A, \Pi\Pi_i^j, \Pi\Pi_{i+1}^j, B, C) \text{ και } O'' = (A, B, \Pi\Pi_i^j, \Pi\Pi_{i+1}^j, C)$$

Αν $\boldsymbol{C}(\boldsymbol{O}), \boldsymbol{C}(\boldsymbol{O}'), \boldsymbol{C}(\boldsymbol{O}'')$ είναι τα αθροίσματα των χρόνων περάτωσης των εργασιών στις διατάξεις O, O' και O'' αντίστοιχα, θα δείξουμε ότι η πρόταση

$$\boldsymbol{C}(\boldsymbol{O}) \le \boldsymbol{C}(\boldsymbol{O}') \text{ και } \boldsymbol{C}(\boldsymbol{O}) \le \boldsymbol{C}(\boldsymbol{O}'')$$

είναι ψευδής. Αυτό θα έχει ως αποτέλεσμα την ύπαρξη κάποιας διάταξης καλύτερης της O ως προς το $n \,|\, 1 \,|\, seq{-}indep \,|\, \sum C_i$. Για τα $\boldsymbol{C}(\boldsymbol{O}), \boldsymbol{C}(\boldsymbol{O}'), \boldsymbol{C}(\boldsymbol{O}'')$ έχουμε:

$$\begin{aligned}
\boldsymbol{C}(\boldsymbol{O}) = {} & C_{1A} + \cdots + C_{aA} \\
& + C_{1\Pi\Pi_i^j} + \cdots + C_{w_i^j \Pi\Pi_i^j} \\
& + C_{1B} + \cdots + C_{bB} \\
& + C_{1\Pi\Pi_{i+1}^j} + \cdots + C_{w_{i+1}^j \Pi\Pi_{i+1}^j} \\
& + C_{1C} + \cdots + C_{cC}
\end{aligned}$$

$$\begin{aligned}
\boldsymbol{C}(\boldsymbol{O}') = {} & C'_{1A} + \cdots + C'_{aA} \\
& + C'_{1\Pi\Pi_i^j} + \cdots + C'_{w_i^j \Pi\Pi_i^j} \\
& + C'_{1\Pi\Pi_{i+1}^j} + \cdots + C'_{w_{i+1}^j \Pi\Pi_{i+1}^j}
\end{aligned}$$

$$+ C'_{1B} + \cdots + C'_{bB}$$
$$+ C'_{1C} + \cdots + C'_{cC}$$

$$\boldsymbol{C(O'')} = C''_{1A} + \cdots + C''_{aA}$$
$$+ C''_{1B} + \cdots + C''_{bB}$$
$$+ C''_{1\Pi\Pi_i^j} + \cdots + C''_{w_i^j\Pi\Pi_i^j}$$
$$+ C''_{1\Pi\Pi_{i+1}^j} + \cdots + C''_{w_{i+1}^j\Pi\Pi_{i+1}^j}$$
$$+ C''_{1C} + \cdots + C''_{cC}$$

Όμως:

$$C'_{1A} = C_{1A}, \cdots, C'_{aA} = C_{aA},$$
$$C'_{1\Pi\Pi_i^j} = C_{1\Pi\Pi_i^j}, \cdots, C'_{w_i^j\Pi\Pi_i^j} = C_{w_i^j\Pi\Pi_i^j} \qquad \text{(i)'}$$

εφόσον μέχρι και το τέλος της $\Pi\Pi_i{}^j$ οι δυο διατάξεις είναι ίδιες.

$$C'_{1\Pi\Pi_{i+1}^j} = C_{1\Pi\Pi_{i+1}^j} - (TcB + s_{j,1B} + s_{bB,j}), \cdots,$$
$$C'_{w_{i+1}^j\Pi\Pi_{i+1}^j} = C_{w_{i+1}^j\Pi\Pi_{i+1}^j} - (TcB + s_{j,1B} + s_{bB,j}) \qquad \text{(ii)'}$$

εφόσον οι εργασίες της $\Pi\Pi_{i+1}{}^j$ εκτελούνται στην O' με διαφορά έναρξης από την O, $s_{aA,j} - (s_{aA,j} + s_{j,1B} + TcB + s_{bB,j})$.

$$C'_{1B} = C_{1B} + w_{i+1}{}^j p_{i+1}{}^j, \cdots, C'_{bB} = C_{bB} + w_{i+1}{}^j p_{i+1}{}^j \qquad \text{(iii)'}$$

εφόσον οι εργασίες της υπακολουθίας B εκτελούνται στην O' με διαφορά έναρξης από την O, $(s_{aA,j} + w_{i+1}{}^j p_{i+1}{}^j + s_{j,1B}) - (s_{aA,j} + s_{j,1B})$. ($w_{i+1}{}^j p_{i+1}^j$ είναι ο χρόνος που απαιτείται για την περάτωση όλων των εργασιών που ανήκουν στην $\Pi\Pi_{i+1}{}^j$)

$$C'_{1C} = C_{1C} - s_{bB,j}, \cdots, C'_{cC} = C_{cC} - s_{bB,j} \qquad \text{(iv)'}$$

εφόσον οι εργασίες της υπακολουθίας C στη διάταξη O' εκτελούνται κατά $s_{bB,j}$ ενωρίτερα των αντίστοιχων εργασιών της O.

Από τις παραπάνω σχέσεις (i)’ — (iv)’ έχουμε:

$$\boldsymbol{C(O') - C(O)} = -w_{i+1}{}^j(s_{j,1B} + TcB + s_{bB,j}) + bw_{i+1}{}^j p_{i+1}{}^j - cs_{bB,j} \qquad \text{(R)'}$$

Επίσης:

$$C''_{1A} = C_{1A}, \cdots, C''_{aA} = C_{aA} \qquad \text{(i)''}$$

εφόσον η υπακολουθία A εκτελείται με την ίδια ακριβώς σειρά στις O'' και O.

$$C''_{1B} = C_{1B} - w_i{}^j p_i{}^j - s_{aA,j}, \cdots, C''_{bB} = C_{bB} - w_i{}^j p_i^j - s_{aA,j} \qquad \text{(ii)''}$$

εφόσον οι εργασίες της υπακολουθίας B εκτελούνται στην O'' με διαφορά έναρξης από την O, $s_{aA,1B} - (s_{aA,j} + w_i{}^j p_i^j + s_{j,1B})$ και επίσης $s_{aA,1B} = s_{j,1B}$ ως χρόνοι εξάρμωσης ανεξάρτητοι ακολουθίας.

$$C''_{1\Pi\Pi_i{}^j} = C_{1\Pi\Pi_i{}^j} + s_{aA,1B} + TcB, \cdots, C''_{w_i{}^j\Pi\Pi_i{}^j} = C_{w_i{}^j\Pi\Pi_i{}^j} + s_{aA,1B} + TcB \qquad \text{(iii)''}$$

εφόσον οι εργασίες της $\Pi\Pi_i{}^j$ εκτελούνται στην O'' με διαφορά έναρξης από την O, $(s_{aA,1B} + TcB + s_{bB,j}) - s_{aA,j}$ και επίσης $s_{bB,j} = s_{aA,j}$ ως χρόνοι εξάρμωσης ανεξάρτητοι ακολουθίας.

$$C''_{1\Pi\Pi_{i+1}{}^j} = C_{1\Pi\Pi_{i+1}{}^j} - s_{aA,j}, \cdots, C''_{w_{i+1}{}^j\Pi\Pi_{i+1}{}^j} = C_{w_{i+1}{}^j\Pi\Pi_{i+1}{}^j} - s_{aA,j}$$

$$C''_{1C} = C_{1C} - s_{aA,j}, \cdots, C''_{cC} = C_{cC} - s_{aA,j} \qquad \text{(iv)''}$$

εφόσον οι εργασίες της $\Pi\Pi_{i+1}{}^j$ και της υπακολουθίας C εκτελούνται στη διάταξη O'' κατά $s_{aA,j}$ ενωρίτερα των αντίστοιχων εργασιών στη διάταξη O.

Από τις σχέσεις (i)'' — (iv)'' έχουμε:

$$\boldsymbol{C(O'') - C(O)} = -b(w_i{}^j p_i{}^j + s_{aA,j}) + w_i{}^j(TcB + s_{aA,1B}) - w_{i+1}{}^j s_{aA,j} - cs_{aA,j} \qquad \text{(R)''}$$

Στο σημείο αυτό μπορούμε χωρίς βλάβη της γενικότητας να υποθέσουμε ότι: $\boldsymbol{C(O') - C(O)} \geq 0$. Όμως βάσει προηγουμένων σχέσεων θα έχουμε:

$$\boldsymbol{C(O') - C(O)} \geq 0 \Rightarrow -w_{i+1}{}^j(s_{j,1B} + TcB + s_{bB,j}) + bw_{i+1}{}^j p_{i+1}{}^j - cs_{bB,j} \geq 0 \Rightarrow$$

$$bw_{i+1}{}^j p_{i+1}{}^j \geq w_{i+1}{}^j(s_{j,1B} + TcB + s_{bB,j}) + cs_{bB,j}$$

Πολλαπλασιάζοντας και τα δύο μέλη της παραπάνω ανισότητας με το θετικό αριθμό $w_i{}^j/w_{i+1}{}^j$ έχουμε:

$$bw_i{}^j p_{i+1}{}^j \geq w_i{}^j TcB + w_i{}^j(s_{j,1B} + s_{bB,j}) + c\frac{w_i{}^j}{w_{i+1}{}^j}s_{bB,j}$$

Η ανισότητα που εμφανίζεται στη διατύπωση του θεωρήματος που αποδεικνύουμε με στοιχειώδεις αλγεβρικές πράξεις γίνεται:

$$w_i{}^j p_i{}^j + s_{kj} \geq w_i{}^j p_{i+1}{}^j$$

Από τις δύο αμέσως προηγούμενες σχέσεις έχουμε:

$$bw_i{}^j p_i{}^j + bs_{kj} \geq w_i{}^j TcB + w_i{}^j (s_{j,1B} + s_{bB,j}) + c\frac{w_i{}^j}{w_{i+1}{}^j} s_{bB,j}$$

Προσθέτοντας και στα δύο μέλη της προηγούμενης ανισότητας την ίδια ποσότητα προκύπτει η ακόλουθη σχέση:

$$bw_i{}^j p_i{}^j + bs_{kj} + \{bs_{aA,j} - w_i{}^j (TcB + s_{aA,1B}) + w_{i+1}{}^j s_{aA,j} + cs_{aA,j}\} \geq$$

$$w_i{}^j TcB + w_i{}^j (s_{j,1B} + s_{bB,j}) + c\frac{w_i{}^j}{w_{i+1}{}^j} s_{bB,j} + \{bs_{aA,j} - w_i{}^j (TcB + s_{aA,1B}) + w_{i+1}{}^j s_{aA,j} + cs_{aA,j}\}$$

Η προηγούμενη σχέση, λόγω της (R)'' και του ότι $s_{j,1B} = s_{aA,1B}$ γίνεται μετά και από αλγεβρικές πράξεις:

$$C(O) - C(O'') + bs_{kj} \geq w_i{}^j s_{bB,j} + c\frac{w_i{}^j}{w_{i+1}{}^j} s_{bB,j} + bs_{aA,j} + w_{i+1}{}^j s_{aA,j} + cs_{aA,j} \Rightarrow$$

$$C(O) - C(O'') \geq -bs_{kj} + w_i{}^j s_{bB,j} + c\frac{w_i{}^j}{w_{i+1}{}^j} s_{bB,j} + bs_{aA,j} + w_{i+1}{}^j s_{aA,j} + cs_{aA,j}$$

Όμως $s_{kj} = s_{aA,j}$. Έτσι τελικά έχουμε:

$$C(O) - C(O'') \geq w_i{}^j s_{bB,j} + c\frac{w_i{}^j}{w_{i+1}{}^j} s_{bB,j} + w_{i+1}{}^j s_{aA,j} + cs_{aA,j}$$

Έχουμε ήδη παρατηρήσει στην αρχή της απόδειξης ότι $s_{bB,j} > 0$. Επομένως :

$$w_i{}^j s_{bB,j} + c\frac{w_i{}^j}{w_{i+1}{}^j} s_{bB,j} + w_{i+1}{}^j s_{aA,j} + cs_{aA,j} > 0 \Rightarrow$$

$C(O) - C(O'') > 0 \Rightarrow C(O'') < C(O)$ και άρα η O'' διάταξη είναι καλύτερη της O.

□

Με βάση το παραπάνω θεώρημα λοιπόν, αν δύο ομοειδείς και διαδοχικές βάσει του SPT για τους χαρακτηριστικούς χρόνους τους ΠΠ έχουν μεγέθη (χαρακτηριστικούς χρόνους και βαρύτητες), τέτοια ώστε να ικανοποιείται η υπόθεση του θεωρήματος, αυτές εκτελούνται διαδοχικές σε κάθε βέλτιστη διάταξη που απαρτίζεται από αδιάσπαστες ΠΠ, και στην οποία προηγούνται αυτών ετεροειδείς ΠΠ. Αυτό μας οδηγεί στην ακόλουθη γενίκευση του θεωρήματος :

Θεώρημα 3.5.2 Έστω $\Pi\Pi_i{}^j, \cdots, \Pi\Pi_{i+m}{}^j$, m+1 διαδοχικές στη σειρά, βάσει του SPT, ΠΠ της ομάδας j, με χαρακτηριστικούς χρόνους $p_i{}^j, \cdots, p_{i+m}{}^j$, και βαρύτητες $w_i{}^j, \cdots, w_{i+m}{}^j$ αντίστοιχα. Αν

$$\frac{s_{kj} + w_i{}^j p_i{}^j + \cdots + w_{i+m}{}^j p_{i+m}{}^j}{w_i{}^j + \cdots + w_{i+m}{}^j} \leq \frac{s_{kj} + w_i{}^j p_i{}^j + \cdots + w_{i+m-1}{}^j p_{i+m-1}{}^j}{w_i{}^j + \cdots + w_{i+m-1}{}^j}$$

οι $\Pi\Pi_i{}^j, \cdots, \Pi\Pi_{i+m}{}^j$ εκτελούνται διαδοχικές σε κάθε βέλτιστη διάταξη του $n\,|\,1\,|\,seq-indep\,|\sum C_i$, που απαρτίζεται από αδιάσπαστες ΠΠ. (με s_{kj} συμβολίζουμε το χρόνο εξάρμωσης από την αμέσως προηγούμενη στη διάταξη ΠΠ της $\Pi\Pi_i{}^j$.)

Απόδειξη: Για την απόδειξη του θεωρήματος θα χρειαστεί το ακόλουθο λήμμα:

Λήμμα 3.5.2 Αν $0 < p_i{}^j \leq \ldots \leq p_{i+m}{}^j$ και $w_i{}^j, \ldots, w_{i+m}{}^j > 0$ τότε για κάθε l, $0 < l \leq m$

$$\frac{s_{kj} + w_i{}^j p_i{}^j + \cdots + w_{i+m}{}^j p_{i+m}{}^j}{w_i{}^j + \cdots + w_{i+m}{}^j} \leq \frac{s_{kj} + w_i{}^j p_i{}^j + \cdots + w_{i+m-1}{}^j p_{i+m-1}{}^j}{w_i{}^j + \cdots + w_{i+m-1}{}^j} \Rightarrow$$

$$\frac{s_{kj} + w_i{}^j p_i{}^j + \cdots + w_{i+m-l}{}^j p_{i+m-l}{}^j}{w_i{}^j + \cdots + w_{i+m-l}{}^j} \leq \frac{s_{kj} + w_i{}^j p_i{}^j + \cdots + w_{i+m-l-1}{}^j p_{i+m-l-1}{}^j}{w_i{}^j + \cdots + w_{i+m-l-1}{}^j}$$

Απόδειξη : Θέτουμε :

$$P = s_{kj} + w_i{}^j p_i{}^j + \cdots + w_{i+m-l}{}^j p_{i+m-l}{}^j$$

$$W = w_i{}^j + \cdots + w_{i+m-l}{}^j$$

Έστω ότι ισχύει η υπόθεση της συνεπαγωγής που εμφανίζεται στη διατύπωση του λήμματος αλλά όχι το συμπέρασμα, δηλαδή υπάρχει l τέτοιο ώστε:

$$\frac{s_{kj} + w_i^j p_i^j + \cdots + w_{i+m-l}^j p_{i+m-l}^j}{w_i^j + \cdots + w_{i+m-l}^j} > \frac{s_{kj} + w_i^j p_i^j + \cdots + w_{i+m-l-1}^j p_{i+m-l-1}^j}{w_i^j + \cdots + w_{i+m-l-1}^j}$$

Κάνοντας την αντικατάσταση των P και W στην προηγούμενη ανισότητα και με χιαστί πολλαπλασιασμό των μελών της έχουμε:

$$P(w_i^j + \cdots + w_{i+m-l-1}^j) > W(s_{kj} + w_i^j p_i^j + \cdots + w_{i+m-l-1}^j p_{i+m-l-1}^j)$$

Προσταφαιρώντας το Pw_{i+m-l}^j στο αριστερό μέλος και το $Ww_{i+m-l}^j p_{i+m-l}^j$ στο δεξιό μέλος της παραπάνω ανισότητας παίρνουμε:

$$-w_{i+m-l}^j P + WP > -w_{i+m-l}^j p_{i+m-l}^j W + WP$$

Και επειδή $w_{i+m-l}^j > 0$,

$$P < p_{i+m-l}^j W \Rightarrow \frac{P}{W} < p_{i+m-l}^j$$

εφόσον το W είναι επίσης θετικό. Εφόσον δε έχουμε υποθέσει ότι οι χαρακτηριστικοί χρόνοι των ΠΠ ακολουθούν τον κανόνα SPT, θα ισχύει $p_{i+m-l}^j < p_{i+m-l+1}^j$ και επομένως:

$$\frac{P}{W} < p_{i+m-l+1}^j \tag{I}$$

Αναλύουμε το $\frac{s_{kj} + w_i^j p_i^j + \cdots + w_{i+m-l}^j p_{i+m-l}^j}{w_i^j + \cdots + w_{i+m-l}^j}$ σε δύο κλάσματα (με τρόπο ανάλογο αυτού που χρησιμοποιείται κατά την ολοκλήρωση με ανάλυση σε απλά κλάσματα). Έτσι:

$$\frac{s_{kj} + w_i^j p_i^j + \cdots + w_{i+m-l}^j p_{i+m-l}^j}{w_i^j + \cdots + w_{i+m-l}^j} =$$

$$\frac{s_{kj} + w_i^j p_i^j + \cdots + w_{i+m-l}^j p_{i+m-l}^j}{w_i^j + \cdots + w_{i+m-l+1}^j} + \frac{w_{i+m-l+1}^j (s_{kj} + w_i^j p_i^j + \cdots + w_{i+m-l}^j p_{i+m-l}^j)}{(w_i^j + \cdots + w_{i+m-l+1}^j)(w_i^j + \cdots + w_{i+m-l}^j)} =$$

$$\frac{s_{kj} + w_i{}^j p_i{}^j + \cdots + w_{i+m-l}{}^j p_{i+m-l}{}^j}{w_{i+1}{}^j + \cdots + w_{i+m-l+1}{}^j} + \frac{P}{W}\frac{w_{i+m-l+1}{}^j}{w_i{}^j + \cdots + w_{i+m-l+1}{}^j} <$$

$$\frac{s_{kj} + w_i{}^j p_i{}^j + \cdots + w_{i+m-l}{}^j p_{i+m-l}{}^j}{w_i{}^j + \cdots + w_{i+m-l+1}{}^j} + \frac{w_{i+m-l+1}{}^j p_{i+m-l+1}{}^j}{w_i{}^j + \cdots + w_{i+m-l+1}{}^j} =$$

$$\frac{s_{kj} + w_i{}^j p_i{}^j + \cdots + w_{i+m-l+1}{}^j p_{i+m-l+1}{}^j}{w_i{}^j + \cdots + w_{i+m-l+1}{}^j}$$

Η ανισότητα στην παραπάνω διαδικασία προκύπτει από τη σχέση (Ι) ενώ τα υπόλοιπα βήματα είναι απλές πράξεις.

Η παραπάνω διαδικασία λοιπόν μας έδωσε:

$$\frac{s_{kj} + w_i{}^j p_i{}^j + \cdots + w_{i+m-l+1}{}^j p_{i+m-l+1}{}^j}{w_i{}^j + \cdots + w_{i+m-l+1}{}^j} > \frac{s_{kj} + w_i{}^j p_i{}^j + \cdots + w_{i+m-l}{}^j p_{i+m-l}{}^j}{w_i{}^j + \cdots + w_{i+m-l}{}^j}$$

Ξεκινώντας από την τελευταία ανισότητα και επαναλαμβάνοντας όλη τη μέχρι τώρα διαδικασία $l-1$ φορές φτάνουμε στο:

$$\frac{s_{kj} + w_i{}^j p_i{}^j + \cdots + w_{i+m}{}^j p_{i+m}{}^j}{w_i{}^j + \cdots + w_{i+m}{}^j} > \frac{s_{kj} + w_i{}^j p_i{}^j + \cdots + w_{i+m-1}{}^j p_{i+m-1}{}^j}{w_i{}^j + \cdots + w_{i+m-1}{}^j}$$

που αντιβαίνει την υπόθεση του λήμματος (αριστερό μέλος συνεπαγωγής).

Άρα :

$$\frac{s_{kj} + w_{i+1}{}^j p_{i+1}{}^j + \cdots + w_{i+m-l}{}^j p_{i+m-l}{}^j}{w_{i+1}{}^j + \cdots + w_{i+m-l}{}^j} \le \frac{s_{kj} + w_{i+1}{}^j p_{i+1}{}^j + \cdots + w_{i+m-l-1}{}^j p_{i+m-l-1}{}^j}{w_{i+1}{}^j + \cdots + w_{i+m-l-1}{}^j}$$

για κάθε l, $0<l\le m-1$ □.

Το προηγούμενο λήμμα μας λέει λοιπόν ότι αν για $m+1$ διαδοχικές στη σειρά, βάσει του SPT, ΠΠ κάποιας ομάδας ισχύει η υπόθεση του θεωρήματος 3.5.2, τότε ισχύει και για κάθε υποσύνολο αυτών των ΠΠ το οποίο προκύπτει από αφαίρεση των ΠΠ με τους l μεγαλύτερους χαρακτηριστικούς χρόνους επεξεργασίας ($0<l\le m-1$).

Προχωρούμε τώρα στην απόδειξη του θεωρήματος.

Από την υπόθεση του θεωρήματος και το λήμμα 3.5.2 προκύπτει για $l=m-1$ η ακόλουθη σχέση :

$$\frac{s_{kj}+w_i^{\,j}p_i^{\,j}+w_{i+1}^{\,j}p_{i+1}^{\,j}}{w_i^{\,j}+w_{i+1}^{\,j}} \le \frac{s_{kj}+w_i^{\,j}p_i^{\,j}}{w_i^{\,j}}$$

Όμως με βάση το θεώρημα 3.5.1 και λόγω της προηγούμενης σχέσης θα πρέπει οι $ΠΠ_i^{\,j}$ και $ΠΠ_{i+1}^{\,j}$ να εκτελούνται διαδοχικές σε κάθε βέλτιστη διάταξη του $n\,|\,1\,|\,seq-indep\,|\sum C_i$, που απαρτίζεται από αδιάσπαστες ΠΠ. Επομένως στο εξής μπορούμε να τις θεωρούμε ως μία ΠΠ με χαρακτηριστικό χρόνο επεξεργασίας $\frac{w_i^{\,j}p_i^{\,j}+w_{i+1}^{\,j}p_{i+1}}{w_i^{\,j}+w_{i+1}^{\,j}}$ και βαρύτητα $w_i^{\,j}+w_{i+1}^{\,j}$. Προφανώς μια τέτοια υπόθεση δεν επιφέρει καμμιά αλλαγή στα αρχικά δεδομένα του προβλήματος και χρησιμοποιείται μόνο για την ανάλυσή του. Στη συνέχεια μια τέτοια υπόθεση θα γίνει και για περισσότερες των δύο διαδοχικές παρτίδες. Για διευκόλυνση ως προς το συμβολισμό θα χρησιμοποιούμε το $ΠΠ_{i,i+m}^{\,j}$ για τη σύνθετη υποθετική παρτίδα που περιέχει m+1 διαδοχικές ως προς SPT ΠΠ, αρχίζει με την $ΠΠ_i^{\,j}$ και τελειώνει με την $ΠΠ_{i+m}^{\,j}$.

Συνεχίζοντας την εφαρμογή του λήμματος 3.5.2 για $l=m-2$ έχουμε τη σχέση :

$$\frac{s_{kj}+w_i^{\,j}p_i^{\,j}+w_{i+1}^{\,j}p_{i+1}^{\,j}+w_{i+2}^{\,j}p_{i+2}^{\,j}}{w_i^{\,j}+w_{i+1}^{\,j}+w_{i+2}^{\,j}} \le \frac{s_{kj}+w_i^{\,j}p_i^{\,j}+w_{i+1}^{\,j}p_{i+1}^{\,j}}{w_i^{\,j}+w_{i+1}^{\,j}}$$

Ισοδύναμα :

$$\frac{s_{kj}+(w_i^{\,j}+w_{i+1}^{\,j})\dfrac{w_i^{\,j}p_i^{\,j}+w_{i+1}^{\,j}p_{i+1}^{\,j}}{w_i^{\,j}+w_{i+1}^{\,j}}+w_{i+2}^{\,j}p_{i+2}^{\,j}}{(w_i^{\,j}+w_{i+1}^{\,j})+w_{i+2}^{\,j}} \le \frac{s_{kj}+(w_i^{\,j}+w_{i+1}^{\,j})\dfrac{w_i^{\,j}p_i^{\,j}+w_{i+1}^{\,j}p_{i+1}^{\,j}}{w_i^{\,j}+w_{i+1}^{\,j}}}{(w_i^{\,j}+w_{i+1}^{\,j})}$$

Όμως και πάλι το θεώρημα 3.5.1 θα δώσει ότι η υποθετική σύνθετη παρτίδα $ΠΠ_{i,i+1}^{\,j}$ και η $ΠΠ_{i+2}^{\,j}$ πρέπει να εκτελούνται διαδοχικές σε κάθε βέλτιστη διάταξη του $n\,|\,1\,|\,seq-indep\,|\sum C_i$, που απαρτίζεται από αδιάσπαστες ΠΠ. Και αυτές λοιπόν μπορούμε στο εξής να τις αντικαταστήσουμε με την υποθετική σύνθετη παρτίδα $ΠΠ_{i,i+2}^{\,j}$ με μεγέθη που ορίζονται ανάλογα.

Συνεχίζοντας τη διαδοχική εφαρμογή του λήμματος 3.5.2 μέχρι $l=1$ η εισαγωγή ανάλογων υποθετικών παρτίδων και το θεώρημα 3.5.1 οδηγούν στην αλήθεια του συμπεράσματος του θεωρήματος 3.5.2 $\square$.

Στο σημείο αυτό, και πριν προχωρήσουμε στον τρόπο με τον οποίο φτιάχνονται οι ΔΠ, θα προβούμε στον ορισμό κάποιων συμβόλων για διευκόλυνση της περαιτέρω ανάλυσής μας.

θα ονομάσουμε χαρακτηριστικούς λόγους, ΧΛ, τα κλάσματα που εμφανίζονται στις υποθέσεις των δύο προηγουμένων θεωρημάτων. Αυτοί προκύπτουν από μεγέθη σχετικά με τις ΠΠ μιας ομάδας κι επομένως μπορεί να θεωρηθούν συναρτήσεις τους. Επειδή δε η τάξη μιας ΠΠ ως προς SPT καθορίζει και τα σχετικά με την ΠΠ μεγέθη, μπορούμε βάσει αυτών να ορίσουμε τους ΧΛ. ΄Ετσι :

Ορισμός 3.5.1 : Χαρακτηριστικός λόγος (ΧΛ) είναι μια συνάρτηση πάνω στα σύνολα των ομάδων και των ΠΠ τους, που έχει ως όρισμα ένα διάνυσμα με τρεις συνιστώσες. Η πρώτη από αυτές σχετίζεται με την ομάδα για την οποία φτιάχνεται ο ΧΛ, και οι άλλες δύο αντιστοιχούν στην τάξη ως προς SPT της πρώτης και τελευταίας ΠΠ για τις οποίες φτιάχνεται ο ΧΛ. Η τιμή του ΧΛ προκύπτει από τον υπολογισμό της τιμής του κλάσματος που αντιπροσωπεύει.

΄Ετσι για παράδειγμα :

$$X\Lambda(j,i+1,i+3) = \frac{s_j + w_{i+1}{}^j p_{i+1}{}^j + w_{i+2}{}^j p_{i+2}{}^j + w_{i+3}{}^j p_{i+3}{}^j}{w_{i+1}{}^j + w_{i+2}{}^j + w_{i+3}{}^j}$$

και

$$X\Lambda(j,i,i) = \frac{s_j + w_i{}^j p_i{}^j}{w_i{}^j}$$

(στα προηγούμενα παραδείγματα παραλείπουμε το δείκτη k από το s_{kj} ο οποίος αναφέρεται στην προηγούμενη ομάδα, εφόσον ασχολούμαστε με χρόνους εξάρμωσης ανεξάρτητους ακολουθίας)

Ο ορισμός των ΧΛ διευκολύνει πολύ στην περιγραφή του τρόπου με τον οποίο κατασκευάζονται οι ΔΠ μιας ομάδας. Όπως έχουμε ήδη πει, αυτές αποτελούνται από ολόκληρες ΠΠ μιας ομάδας, διατεταγμένες κατά SPT. Αν θεωρήσουμε την ομάδα j η οποία περιλαμβάνει ΠΠ(j) πρωτογενείς παρτίδες διατεταγμένες ως $\Pi\Pi_1{}^j, \ldots, \Pi\Pi_i{}^j, \ldots, \Pi\Pi_{\Pi\Pi(j)}{}^j$ με βάση τον SPT, με τις $\Pi\Pi_i{}^j, \ldots, \Pi\Pi_{\Pi\Pi(j)}{}^j$ να μην ανήκουν σε κάποια ΔΠ και να εκτελούνται μετά από ετεροειδείς ΠΠ, μπορούμε από αυτές τις (υπόλοιπες) ΠΠ να φτιάξουμε μια ΔΠ ως εξής :

Υπολογίζουμε τους λόγους :

$$X\Lambda(j,i,i+k),\ X\Lambda(j,i,i+k-1)$$

για κάθε $1 \le k \le \Pi\Pi(j)-i$ μέχρις ότου

$$X\Lambda(j,i,i+k) > X\Lambda(j,i,i+k-1).$$

Ο ΧΛ για την τιμή του οποίου τερματίζεται η παραπάνω διαδικασία, καθορίζει τις ΠΠ που απαρτίζουν την καινούργια ΔΠ μιας ομάδας.

Έτσι λοιπόν, με επαναληπτικό υπολογισμό ΧΛ, βρίσκουμε τη σύνθεση των ΔΠ. Μάλιστα, η διαδικασία αυτή σταματάει στην τιμή εκείνη του k για την οποία οι τιμές των ΧΛ που φτιάχνονται, δεν ικανοποιούν την υπόθεση του θεωρήματος 3.5.2. Εύκολα μπορούμε να δούμε από το λήμμα 3.5.2, ότι δεν υπάρχει περίπτωση να ισχύουν μαζί οι

$$X\Lambda(j,i,i+l) \le X\Lambda(j,i,i+l-1) \qquad \text{για } k < l \le \Pi\Pi(j)-i$$

και

$$X\Lambda(j,i,i+k) > X\Lambda(j,i,i+k-1).$$

Επομένως η λήξη της επαναληπτικής διαδικασίας για την κατασκευή των ΔΠ είναι καλά ορισμένη.

3.6. Γενικευμένες παρτίδες στο $TCTS_j$

Στο σημείο αυτό πρέπει να τονίσουμε ότι οι ΔΠ τις οποίες περιγράψαμε προηγουμένως, γενικά δεν ταυτίζονται με τις ΜΠ (μέγιστες παρτίδες) του κεφαλαίου 3.3, δηλαδή δεν περιέχουν το μέγιστο δυνατό πλήθος ομοειδών ΠΠ που εκτελούνται διαδοχικές σε βέλτιστη διάταξη για το $TCTS_j$. (Μόνο στην περίπτωση που κατά την κατασκευή μιας ΔΠ συμπεριλαμβάνονται σ’ αυτήν όλες οι ΠΠ μιας ομάδας, πχ. όταν ο χρόνος εξάρμωσης είναι πολύ μεγάλος, τότε η ΔΠ είναι μέγιστη.)

Από την άλλη αν οι ΔΠ περιείχαν το μέγιστο δυνατό πλήθος ΠΠ ως προς βέλτιστη διάταξη, αυτές θα ταυτίζονταν με τις ΜΠ του θεωρήματος 3.3, των οποίων η βέλτιστη διάταξη δίνεται από έναν ήδη γνωστό, απλό κανόνα. Στόχος μας λοιπόν στο εξής θα είναι η μετάβαση από τις ΔΠ σε ΜΠ, ή έστω υποσύνολα των ΜΠ μεγαλύτερα από ΔΠ.

Τα υποσύνολα αυτά τα ονομάζουμε γενικευμένες παρτίδες ΓΠ, και είναι οι οντότητες που περιγράφουν την προσπάθεια μετάβασης από ΠΠ και ΔΠ σε ΜΠ. Οι ΓΠ λοιπόν αποτελούνται από ΠΠ και ΔΠ μιας ομάδας και στην περίπτωση που δύο ή περισσότερες ΔΠ συγχωνεύονται για την κατασκευή μιας ΓΠ ισχύει το ακόλουθο θεώρημα:

Θεώρημα 3.6 Υποθέτοντας ότι οι ΔΠ μιας ομάδας βρίσκονται σε κάποια διάταξη μετά από ετεροειδείς ΔΠ ή ΠΠ, και οι ΠΠ εκτελούνται αδιάσπαστες, οι ΓΠ της ομάδας αυτής περιλαμβάνουν ολόκληρες ΔΠ και όχι τμήματα αυτών.

Απόδειξη: Ας είναι $\hat{O} = (\hat{A}, \Delta\Pi_i{}^j, \Delta\Pi_{i+1}{}^j, ..., \Delta\Pi_{i+k}{}^j, \Pi\Pi_l{}^j, \hat{B}, \Pi\Pi_{l+1}{}^j, ..., \Pi\Pi_{l+m}{}^j, \hat{C})$ μια βέλτιστη διάταξη όπου οι ΠΠ εκτελούνται αδιάσπαστες, για το $n\,|\,1\,|\,seq-indep\,|\sum C_i$. Σ' αυτήν, η διαδοχή $\Delta\Pi_i{}^j - \Delta\Pi_{i+1}{}^j - \cdots - \Delta\Pi_{i+k}{}^j - \Pi\Pi_l{}^j$ είναι μια ΓΠ που αποτελείται από k+1 ολόκληρες ΔΠ της ομάδας j με πρώτη στη διαδοχή αυτών την i-οστή, και την πρώτη ΠΠ της i+k+1-οστής ΔΠ της ίδιας ομάδας.

Εφόσον οι $(\Pi\Pi_l{}^j, ..., \Pi\Pi_{l+m}{}^j) \equiv \Delta\Pi_{i+k+1}{}^j$ αποτελούν ΔΠ θα έχουμε βάσει του θεωρήματος 3.5.2 και του ορισμού των ΧΛ :

$$X\Lambda(j, l, l+m) \le X\Lambda(j, l, l+m-1)$$

και από το λήμμα 3.5.2 :

$$X\Lambda(j, l, l+1) \le X\Lambda(j, l, l) \qquad \text{(I)}$$

Η σχέση (I) μας παραπέμπει στο θεώρημα 3.5.1. Αν ακολουθήσουμε κατά βήμα την απόδειξή του όπου τώρα :

$$A = \hat{A} - \Delta\Pi_i{}^j - \Delta\Pi_{i+1}{}^j - \cdots - \Delta\Pi_{i+k}{}^j$$

$$B = \hat{B}$$

$$C = \Pi\Pi_{l+2}{}^j - \cdots - \Pi\Pi_{l+m}{}^j - \hat{C}$$

θα προκύψουν οι διατάξεις

$$O' = (A, \Pi\Pi_l{}^j, \Pi\Pi_{l+1}{}^j, B, C)$$

και

$$O'' = (A, B, \Pi\Pi_l{}^j, \Pi\Pi_{l+1}{}^j, C)$$

μια από τις οποίες είναι καλύτερη της $\hat{O}$.

Αν είναι καλύτερη η O'', η ΓΠ $\Delta\Pi_i{}^j - \Delta\Pi_{i+1}{}^j - \cdots - \Delta\Pi_{i+k}{}^j - \Pi\Pi_l{}^j$ είναι βέλτιστη αν αποκοπεί απ' αυτήν η $\Pi\Pi_l{}^j$, και επομένως αποτελείται από ολόκληρες ΔΠ.

Αν είναι καλύτερη η O', επαναληπτική εφαρμογή της αποδεικτικής διαδικασίας που ακολουθήσαμε ως τώρα για m-l-1 το πολύ φορές, θα δώσει το ζητούμενο : βέλτιστη διάταξη δηλαδή, στην οποία η ΓΠ που μας απασχόλησε αποτελείται από ολόκληρες ΔΠ. □

Βεβαίως για την απόδειξη του προηγουμένου θεωρήματος θα μπορούσαμε να επικαλεστούμε το θεώρημα 3.5.2, βάσει του οποίου δεν είναι δυνατόν να υπάρχει βέλτιστη διάταξη για το $n\,|\,1\,|\,seq-indep\,|\sum C_i$, όπου οι ΠΠ εκτελούνται αδιάσπαστες, και οι ΔΠ διασπώνται. Η επιπλέον παρατήρηση ότι στην αποδεικτική διαδικασία η υπακολουθία A μπορεί να τελειώνει με ΠΠ ομοειδείς της $\Pi\Pi_i{}^j$, ενισχύει την ορθότητα αυτής της επίκλησης (στο θεώρημα 3.5.2).

4. Πρώτος προσεγγιστικός αλγόριθμος για το $TCTS_j$

Στην ενότητα αυτή, θα πραγματευτούμε την κατασκευή ενός προσεγγιστικού αλγορίθμου για το $TCTS_j$, ο οποίος προκύπτει κυρίως από προτάσεις που αποδείχθηκαν στην προηγούμενη ενότητα. Θα παρουσιαστούν αναλυτικά τα βήματα του αλγορίθμου, ευρήματα με τα οποία αντιμετωπίζονται περιπτώσεις στις οποίες ο αλγόριθμος μπορεί να εξελιχθεί προς διαφορετικές κατευθύνσεις, μια σύνοψη του αλγορίθμου, και τέλος εκτίμηση της υπολογιστικής πλοκής του.

4.1. Περιγραφή - Γενική ιδέα

Ο προσεγγιστικός αλγόριθμος που κατασκευάσαμε με σκοπό τη γρήγορη και κατά το δυνατό καλύτερης ποιότητας λύση μεγάλων προβλημάτων, βασίζεται κυρίως στην ιδέα των παρτίδων και στον κανόνα βάσει του οποίου γίνεται η διάταξη ΜΠ σε βέλτιστη λύση του $TCTS_j$.

Έχουμε ήδη αποδείξει, ότι υπάρχει τουλάχιστον μία βέλτιστη διάταξη για το $TCTS_j$ στην οποία οι ΠΠ εκτελούνται αδιάσπαστες (Θεώρημα 3.4). Θα περιορίσουμε στο εξής την προσπάθειά μας στην κατασκευή μιας τέτοιας βέλτιστης διάταξης, την οποία ας ονομάσουμε O_1.

Από τον ορισμό των ΔΠ και τα θεωρήματα 3.5.1 και 3.5.2 , γνωρίζουμε ότι στη διάταξη O_1, οι ΔΠ εκτελούνται αδιάσπαστες.

Αν λοιπόν μπορέσουμε να κατασκευάσουμε μια διάταξη O_2 η οποία αποτελείται από αδιάσπαστες δευτερογενείς παρτίδες, αυτή είναι δυνατόν να αλλαχθεί ώστε να προκύψει η βέλτιστη διάταξη O_1.

Από την παραπάνω επιχειρηματολογία, έχει οριοθετηθεί η μία κατεύθυνση προς την οποία θα κινηθούμε στην ευρηματική αναζήτηση- κατασκευή βέλτιστης λύσης :

Προσπαθούμε να φτιάξουμε μια διάταξη εργασιών, στην οποία οι ΔΠ εκτελούνται αδιάσπαστες.

Η άλλη κατεύθυνση προς την οποία κινούμαστε προκύπτει από τον κανόνα διάταξης ΜΠ όπως αυτός εκφράζεται στο θεώρημα 3.3. Με βάση το θεώρημα αυτό και τον ορισμό των ΧΛ ξέρουμε ότι κάθε βέλτιστη διάταξη για το $TCTS_j$ αποτελείται από ΜΠ διατεταγμένες κατά μη φθίνουσα σειρά των τιμών των ΧΛ τους. Η παραπάνω πρόταση θα πρέπει λοιπόν να ισχύει και για τη βέλτιστη διάταξη O_1, την οποία θέλουμε να κατασκευάσουμε :

Προσπαθούμε να φτιάξουμε διάταξη ΔΠ, η οποία με όποιο τρόπο κι αν αλλάζει προς την κατεύθυνση κατασκευής ΓΠ ή ΜΠ (και με τελικό στόχο την O_1), να αποτελείται από ΓΠ ή ΜΠ διατεταγμένες κατά μη φθίνουσα σειρά των τιμών των ΧΛ τους.

Στο σημείο αυτό θα πρέπει να διασαφηνίσουμε τον τρόπο με τον οποίο από μια διάταξη ΔΠ προκύπτει άλλη, καλύτερη ως προς το κριτήριο $\sum C_i$, που αποτελείται από ΓΠ ή στην καλύτερη περίπτωση ΜΠ.

Ο τρόπος διαφαίνεται άμεσα από τον ορισμό των ΓΠ (και κατ' επέκταση των ΜΠ) και δεν είναι άλλος από τη συγχώνευση ομοειδών ΔΠ, ή αλλιώς τη διαδοχική εκτέλεση ομοειδών ΔΠ, αν κάτι τέτοιο βελτιώνει την τιμή του κριτηρίου επίδοσης.

Αυτό μπορεί να γίνεται με αναδιάταξη ΔΠ ώστε δύο ή περισσότερες ομοειδείς να εκτελούνται διαδοχικές, φροντίζοντας όμως πάντοτε, να διατηρούμε τη μη φθίνουσα διάταξη των τιμών των ΧΛ στις αναδιατάξεις που γίνονται. (Έτσι εξασφαλίζεται και η ισχύς του SPT για τις εργασίες κάθε ομάδας, που είναι αναγκαία συνθήκη βελτίστου.)

Βεβαίως μια τέτοια αναδιάταξη θα γίνεται μόνο αν έχουμε βελτίωση στην τιμή του κριτηρίου επίδοσης, εν προκειμένω ελάττωση του $\sum C_i$.

Αντί για υπολογισμό της τιμής του κριτηρίου επίδοσης μιας νέας διάταξης, και σύγκρισή του με την τιμή του σε προηγούμενη διάταξη, κρίθηκε καλύτερος ο υπολογισμός της διαφοράς των τιμών του κριτηρίου στις δύο διατάξεις. Αυτή η διαφορά προκύπτει άμεσα από τιμές ενδιάμεσων ΧΛ και ένας τρόπος υπολογισμού της θα δοθεί παρακάτω.

Από τα παραπάνω φαίνεται ότι ο τρόπος με τον οποίο γίνεται η συγχώνευση ΔΠ σε ΓΠ (ή ΜΠ) οδηγεί στην κατασκευή διάταξης με πολλά από τα χαρακτηριστικά μιας κατηγορίας βέλτιστων διατάξεων (διατάξεις με ΠΠ αδιάσπαστες). Αυτή η διαδικασία σταματάει όταν, είτε δεν μπορεί να γίνει άλλη αναδιάταξη λόγω παραβίασης του κανόνα για τους χαρακτηριστικούς λόγους, ή κάθε νέα δυνατή αναδιάταξη δεν επιφέρει βελτίωση στη τιμή του κριτηρίου επίδοσης.

Βεβαίως κατά την εξέλιξη του αλγορίθμου είναι δυνατόν να βρεθούμε σε καταστάσεις με πολλαπλές επιλογές ως προς την πορεία κατασκευής μιας σχεδόν βέλτιστης ή βέλτιστης διάταξης. Για παράδειγμα είναι πιθανό κατά τη συγχώνευση ΔΠ, κάποια να μπορεί να συγχωνευθεί δεξιά με προηγούμενη στη διάταξη ομοειδή ΔΠ ή ΓΠ, αλλά και αριστερά με επόμενη στη διάταξη ΔΠ ή ΓΠ.

Τέτοια προβλήματα, που μπορεί να συναντήσουμε σε κάθε βήμα του αλγόριθμου, αντιμετωπίζονται βάσει βέλτιστων προτάσεων, ή ευρημάτων που στηρίζονται στη διαίσθησή μας.

4.2. Βήματα του πρώτου προσεγγιστικού αλγορίθμου

Στο κεφάλαιο αυτό θα παρουσιάσουμε διεξοδικά τα βήματα του προσεγγιστικού αλγορίθμου που κατασκευάστηκε για τη λύση του $TCTS_j$.

Από την περιγραφή που έγινε στο προηγούμενο κεφάλαιο μπορούμε να διαπιστώσουμε ότι αποτελείται από τρία βασικά βήματα :

1) Κατασκευή των ΔΠ κάθε ομάδας.
2) Αρχική διάταξη ΔΠ κατά μη φθίνουσα σειρά των τιμών των ΧΛ τους.
3) Συγχώνευση ομοειδών ΔΠ σε ΓΠ (ή ΜΠ) με τρόπο ώστε να διατηρείται η μη φθίνουσα σειρά των τιμών αντίστοιχων ΧΛ, και να βελτιώνεται η τιμή του κριτηρίου επίδοσης.

1ο Β Η Μ Α

Το πρώτο βήμα του αλγορίθμου καθορίζεται σαφώς από τον ορισμό των ΔΠ και τη διαδικασία κατασκευής τους που παρουσιάζεται αμέσως μετά. Κρίνουμε απαραίτητο να επαναλάβουμε τη διαδικασία αυτή και εδώ, προκειμένου να έχουμε πληρέστερη εικόνα του αλγορίθμου.

Έστω λοιπόν η j ομάδα εργασιών με ΠΠ(j) ΠΠ, διατεταγμένες βάσει του SPT ως εξής : $\Pi\Pi_1{}^j, \Pi\Pi_2{}^j, \ldots, \Pi\Pi_{\Pi\Pi(j)}{}^j$

Για $1 \le k \le \Pi\Pi(j) - 1$ υπολογίζουμε τις τιμές για κάθε ζεύγος χαρακτηριστικών λόγων $X\Lambda(j,1,k+1)$ $X\Lambda(j,1,k)$ μέχρι να φτάσουμε σε k_1 τέτοιο ώστε : $X\Lambda(j,1,k_1+1) > X\Lambda(j,1,k_1)$. Οι πρώτες k_1 ΠΠ της ομάδας j αποτελούν την πρώτη ΔΠ της.

Επαναλαμβάνουμε την παραπάνω διαδικασία για τις υπόλοιπες ΠΠ της ομάδας φτιάχνοντας ΔΠ, μέχρι $k = \Pi\Pi(j) - 1$ (ώστε να έχουν συμπεριληφθεί σε κάποια ΔΠ, έστω και μονομελή, όλες οι ΠΠ της ομάδας).

Η παραπάνω διαδικασία εφαρμόζεται σε κάθε διαφορετική ομάδα ΠΠ, δίνοντας έτσι όλες τις ΔΠ του συνόλου των εργασιών που πρέπει να διατάξουμε.

Επειδή η ταυτότητα μιας ΔΠ, καθώς και τα χαρακτηριστικά της μεγέθη καθορίζονται σαφώς από τον αντίστοιχο ΧΛ, στην υλοποίηση του αλγορίθμου το βήμα αυτό οδηγεί στην κατασκευή του συνόλου των ΧΛ που αντιστοιχούν στις ΔΠ.

Στο σημείο αυτό πρέπει να αναφερθούμε σ' ένα ατόπημα που έγινε κατά την κατασκευή του αλγορίθμου και είχε ως αποτέλεσμα την απόκλιση από το βέλτιστο. Αυτό έγκειται στην a priori θεώρησή μας ότι πριν από τις ΔΠ μιας ομάδας εκτελούνται πάντοτε ΔΠ άλλης ομάδας (ετεροειδείς). Αν βέβαια κάτι τέτοιο συμβεί (όπως έγινε σε πολλές περιπτώσεις πειραμάτων) το βήμα κατασκευής ΔΠ περιορίζει σωστά το χώρο έρευνας για εύρεση βέλτιστης διάταξης, διαφορετικά οδηγεί σε υποβέλτιστες λύσεις. Η αποτυχία αυτής της αυθαίρετης θεώρησης με βάση τα πειραματικά αποτελέσματα είναι πιθανότερο να συμβεί στις περιπτώσεις που ο μέσος χρόνος εξάρμωσης είναι μεγαλύτερος του μέσου χρόνου επεξεργασίας, η λύση όμως που τότε επιτυγχάνεται έχει μικρότερο μέγιστο χρόνο περάτωσης ($C_{\max}$) από αυτόν της βέλτιστης.

2ο ΒΗΜΑ

Το δεύτερο βήμα του αλγορίθμου είναι μια απλή διάταξη των ΔΠ με βάση τις τιμές των ΧΛ τους, κατά μη φθίνουσα σειρά.

Στο βήμα αυτό όμως, είναι δυνατό λόγω ισότητας των τιμών των ΧΛ να προκύψουν περισσότερες της μιας διατάξεις, από τις οποίες πρέπει να επιλέξουμε μια για να προχωρήσουμε στο επόμενο βήμα.

Βεβαίως θα μπορούσαμε εναλλακτικά να εκτελέσουμε το 3ο βήμα του αλγορίθμου για όλες τις διαφορετικές διατάξεις που προκύπτουν από το βήμα 2, όμως κάτι τέτοιο θα ελεγχθεί σε επόμενη πιθανή ενασχόλησή μας με το πρόβλημα. Στην παρούσα μελέτη αποφεύγουμε κάτι τέτοιο γιατί ο αριθμός των διαφορετικών τέτοιων διατάξεων μπορεί να είναι θεωρητικά πολύ μεγάλος (μεγαλύτερος του B^{Δ}, όπου Β ο αριθμός των ομάδων και Δ ο μικρότερος αριθμός ΔΠ σε όλες τις ομάδες).

Η περίπτωση ετεροειδών ΔΠ με ίσους ΧΛ, αντιμετωπίζεται στην εργασία μας με τη χρήση του ακόλουθου ευρηματικού κανόνα :

Κανόνας 4.2.1 : Κατά την κατασκευή μιας αρχικής διάταξης ΔΠ, αν δύο ή περισσότερες ετεροειδείς ΔΠ έχουν χαρακτηριστικούς λόγους με ίσες τιμές, διατάσσουμε πρώτη αυτήν η οποία μπορεί να συγχωνευθεί με επόμενη ομοειδή της ΔΠ, να δώσει ΓΠ με το μικρότερο ΧΛ.

Ο παραπάνω κανόνας βασίζεται κυρίως στην ακόλουθη παρατήρηση :

Έστω $\lambda_k{}^i, \lambda_{k+1}{}^i, \lambda_k{}^j, \lambda_{k+1}{}^j$ τιμές ΧΛ για τις k και k+1 τάξεως ΔΠ των ομάδων i και j, με $\lambda_k{}^i = \lambda_k{}^j$, και $(\lambda_k{}^i, \lambda_{k+1}{}^i)$, $(\lambda_k{}^j, \lambda_{k+1}{}^j)$ οι τιμές των ΧΛ για τις ΓΠ που προκύπτουν από συγχώνευση των αντίστοιχων ΔΠ με $(\lambda_k{}^i, \lambda_{k+1}{}^i) < (\lambda_k{}^j, \lambda_{k+1}{}^j)$.

Αν θεωρήσουμε την ακόλουθη αρχική διάταξη ΔΠ

$$\cdots < \lambda_k{}^j = \lambda_k{}^i \ (pos\ a) < \cdots < \lambda_{k+1}{}^i < \cdots < \lambda_{k+1}{}^j < \cdots$$

αν η τιμή $(\lambda_k{}^i, \lambda_{k+1}{}^i)$ είναι μικρότερη της τιμής του επόμενου ΧΛ, είναι δυνατή η συγχώνευση των ΔΠ που αντιστοιχούν στο λόγο με αυτήν την τιμή στη θέση pos a, ενώ δεν είναι δυνατή η συγχώνευση των ΔΠ που έχουν λόγους $\lambda_k{}^j$ και $\lambda_{k+1}{}^j$, στην ίδια θέση. Αυτό συμβαίνει γιατί θα παραβιάζονταν η φθίνουσα σειρά των τιμών των ΧΛ.

Εξ' άλλου στην περίπτωση

$$\cdots < \lambda_k{}^i = \lambda_k{}^j \ (pos\ a) < \cdots < \lambda_{k+1}{}^i < \cdots < \lambda_{k+1}{}^j < \cdots$$

αν η τιμή $(\lambda_k{}^j, \lambda_{k+1}{}^j)$ είναι μικρότερη της τιμής του επόμενου ΧΛ, είναι δυνατή μια πιθανή συγχώνευση των ΔΠ που αντιστοιχούν στην τιμή αυτή στη θέση pos a, ενώ παράλληλα είναι δυνατή και η συγχώνευση των παρτίδων που αντιστοιχούν στους λόγους $\lambda_k{}^i$ και $\lambda_{k+1}{}^i$ στην ίδια θέση. Αυτό συμβαίνει γιατί $(\lambda_k{}^i, \lambda_{k+1}{}^i) < (\lambda_k{}^j, \lambda_{k+1}{}^j)$, δηλαδή δεν παραβιάζεται η φθίνουσα σειρά των τιμών των ΧΛ.

Έχουμε δηλαδή τη δυνατότητα για μια επιπλέον συγχώνευση, πράγμα που μπορεί να οδηγήσει πιθανώς σε καλύτερη λύση.

Από την προηγούμενη παρατήρηση, συμπεραίνουμε ότι ο κανόνας 4.2.1 οδηγεί στην κατασκευή αρχικών διατάξεων στις οποίες μπορεί να γίνουν περισσότερες συγχωνεύσεις ΔΠ. Έτσι αυξάνεται και η πιθανότητα βελτίωσης της αρχικής διάταξης, άρα και εύρεσης (κατασκευής) βέλτιστης λύσης.

Όμως επειδή δεν έχει αποδειχθεί ότι ο κανόνας αυτός είναι βέλτιστος ως προς την κατασκευή της αρχικής διάταξης ΔΠ, αποτελεί το πρώτο σημείο του αλγορίθμου στο οποίο η κατασκευή της λύσης μας δεν οδηγεί σίγουρα σε βέλτιστο. Ο έλεγχος της συμπεριφοράς αυτού του κανόνα σε σχέση με την κατασκευή βέλτιστης λύσης, θα μπορούσε να γίνει με εξάλειψή του, και έλεγχο όλων των δυνατών διατάξεων ΔΠ, που μπορεί να προκύψουν από το βήμα 2. Κάτι τέτοιο όμως θα γίνει όπως έχουμε ήδη πει σε περαιτέρω ανάλυση του προβλήματος, που ξεπερνάει τα όρια αυτής της εργασίας.

Ακριβώς όπως και για τον ορισμό των ΔΠ, ουσιαστικά ορίστηκαν οι αντίστοιχοι ΧΛ τους, έτσι και στο βήμα αυτό της διάταξης, δουλεύουμε με ΧΛ τους οποίους και διατάσσουμε βάσει των τιμών τους.

3ο Β Η Μ Α

Περνάμε τώρα στην περιγραφή του τελευταίου βήματος όπου γίνεται, όπως έχουμε πει, συγχώνευση των ΔΠ στη διάταξη που προκύπτει από το βήμα 2, με τρόπο ώστε να διατηρείται η μη φθίνουσα σειρά των τιμών των αντίστοιχων ΧΛ, και στόχο την εύρεση καλύτερης διάταξης ως προς το κριτήριο βελτίστου.

Επομένως το βήμα συγχώνευσης παρτίδων, περιλαμβάνει μια διαδικασία επιλογής των παρτίδων που πρόκειται να συγχωνευτούν, και δυο ειδών ελέγχους : έναν σχετικό με τη διατήρηση της μη φθίνουσας σειράς των τιμών των ΧΛ μετά από κάποια συγχώνευση, και έναν επόμενο, σχετικό με τη μείωση της τιμής του $\sum C_i$ μετά από την επικείμενη συγχώνευση.

Η διαδικασία επιλογής των παρτίδων που θα συγχωνευτούν σε μια νέα παρτίδα, εξελίσσεται προς δύο κατευθύνσεις: από την παρτίδα με το μικρότερο ΧΛ προς την παρτίδα με το μεγαλύτερο ΧΛ οπότε επιχειρούμε συγχωνεύσεις από δεξιά, και αντίθετα οπότε επιχειρούμε συγχωνεύσεις από αριστερά. Πάντως με οποιοδήποτε τρόπο κι αν διατρέχουμε μια προηγούμενη διάταξη, μόλις συναντήσουμε την πρώτη ομοειδή παρτίδα με αυτήν στη θέση της οποίας θα γίνει συγχώνευση, προβαίνουμε στους ελέγχους για τη διατήρηση της μη φθίνουσας σειράς των ΧΛ και βελτίωση της τιμής του κριτηρίου ελαχιστοποίησης, και αν αυτοί είναι επιτυχείς συγχωνεύουμε τις δυο παρτίδες και διατρέχουμε τη νέα πλέον διάταξη προς την ίδια κατεύθυνση από την αρχή της. Αλλιώς αν κάποιος έλεγχος αποτύχει ψάχνουμε για την πρώτη ομοειδή της δεύτερης κατά σειρά επίσκεψης παρτίδας και ενεργούμε με τον ίδιο τρόπο έως ότου ληφθούν υπόψη όλες οι παρτίδες. Στη συνέχεια επιχειρούμε επίσκεψη της διάταξης κατά αντίθετη φορά. Αν προκύψει μια τουλάχιστον συγχώνευση, η διαδικασία επαναλαμβάνεται από την αρχή, αλλιώς σταματάμε εφόσον επιπλέον αλλαγές (συγχωνεύσεις) δεν είναι πραγματοποιήσιμες με βάση τους ελέγχους.

Ο πρώτος από τους δύο ελέγχους, που σχετίζεται με τη διατήρηση της μη φθίνουσας σειράς των ΧΛ, γίνεται πολύ απλά με υπολογισμό της τιμής του ΧΛ που αντιστοιχεί στη ΓΠ που προκύπτει από τη συγχώνευση, και σύγκρισή της με τις τιμές των ΧΛ των γειτονικών της παρτίδων.

Στην πραγματικότητα, σε κάθε περίπτωση εν δυνάμει συγχώνευσης παρτίδων, έλεγχος για διατήρηση της μη φθίνουσας σειράς των τιμών των ΧΛ γίνεται σε σχέση με μόνο μια γειτονική παρτίδα της θέσης στην οποία πρόκειται να γίνει η συγχώνευση: αν πρόκειται για συγχώνευση από αριστερά γίνεται σύγκριση με την τιμή του ΧΛ της αριστερά ευρισκόμενης παρτίδας από τη θέση συγχώνευσης, ενώ αν πρόκειται να γίνει δεξιά γίνεται σύγκριση με την τιμή του ΧΛ της δεξιά ευρισκόμενης παρτίδας από τη θέση της συγχώνευσης. Αυτό δικαιολογείται εύκολα αν λάβουμε υπόψη τις ακόλουθες πολύ απλές προτάσεις:

Πρόταση 4.2.1 : Αν $N_1, P_1, N_2, P_2, C > 0$ και $P_1 < P_2$, τότε $\frac{N_1P_1 + N_2P_2 + C}{N_1 + N_2} < \frac{N_2P_2 + C}{N_2}$

Απόδειξη : Πράγματι

$$
\begin{aligned}
P_1 < P_2 \Rightarrow \frac{N_1P_1 + N_2P_2 + C}{N_1 + N_2} &< \frac{N_1P_2 + N_2P_2 + C}{N_1 + N_2} \\
&= P_2 + \frac{C}{N_1 + N_2} \\
&< P_2 + \frac{C}{N_2} \\
&= \frac{N_2P_2 + C}{N_2}. \ \square
\end{aligned}
$$

Πρόταση 4.2.2 : Αν $N_1, P_1, N_2, P_2, N_3, P_3, C > 0$, $P_1 < P_2 < P_3$ και $\frac{N_1P_1 + N_2P_2 + C}{N_1 + N_2} > \frac{N_1P_1 + C}{N_1}$ τότε $\frac{N_1P_1 + N_2P_2 + N_3P_3 + C}{N_1 + N_2 + N_3} > \frac{N_1P_1 + N_2P_2 + C}{N_1 + N_2}$

Απόδειξη :

$$\frac{N_1P_1 + N_2P_2 + C}{N_1 + N_2} > \frac{N_1P_1 + C}{N_1} \Rightarrow P_2 > P_1 + \frac{C}{N_1} \qquad (I)$$

΄Εστω ότι το συμπέρασμα της πρότασης δεν ισχύει, δηλαδή :

$$
\begin{aligned}
&\frac{N_1P_1 + N_2P_2 + N_3P_3 + C}{N_1 + N_2 + N_3} \le \frac{N_1P_1 + N_2P_2 + C}{N_1 + N_2} \\
&\Rightarrow P_3 \le \frac{N_1P_1 + N_2P_2 + C}{N_1 + N_2} \\
&\qquad < \frac{N_1(P_2 - \frac{C}{N_1}) + N_2P_2 + C}{N_1 + N_2} \qquad \text{(από την (I))} \\
&\qquad = P_2 \\
&\Rightarrow P_3 < P_2
\end{aligned}
$$

που αντιβαίνει την υπόθεση $P_1 < P_2 < P_3$. $\square$

Αν αντιστοιχίσουμε τα N_1, N_2, N_3 στις βαρύτητες τριών διαδοχικών βάσει του SPT πρωτογενών παρτίδων μιας ομάδας με αντίστοιχους χαρακτηριστικούς χρόνους P_1, P_2, P_3, ενώ C είναι ο χρόνος εξάρμωσης για την εκτέλεση των εργασιών της ομάδας αυτής στη μηχανή, τότε η πρόταση 4.2.1 βρίσκει εφαρμογή στη συγχώνευση παρτίδων από αριστερά και μας απαλλάσει από τη σύγκριση του ΧΛ της σύνθετης παρτίδας με αυτόν που βρίσκεται δεξιά από τη θέση στην οποία γίνεται η συγχώνευση. Αυτό συμβαίνει γιατί ο ΧΛ της καινούργιας παρτίδας είναι μικρότερος του ΧΛ αυτής στη θέση της οποίας γίνεται η συγχώνευση, ο οποίος με τη σειρά του είναι μικρότερος του ΧΛ της παρτίδας που βρίσκεται δεξιά του, εκ κατασκευής της αρχικής διάταξης και αντίστοιχου ελέγχου σε ενδιάμεσες διατάξεις. Έτσι πρέπει να γίνει σύγκριση του ΧΛ της εν δυνάμει νέας παρτίδας μόνο με το ΧΛ της παρτίδας που βρίσκεται αριστερά της.

Ανάλογα, η πρόταση 4.2.2 βρίσκει εφαρμογή στη συγχώνευση παρτίδων από δεξιά. Στην περίπτωση που οι παρτίδες που συγχωνεύονται από δεξιά είναι και οι δυο δευτερογενείς (ο τρόπος επιλογής τους επιβάλλει να είναι και διαδοχικές κατά την κατασκευή), η σύνθετη παρτίδα που προκύπτει έχει ΧΛ μεγαλύτερο του ΧΛ αυτής που βρίσκεται αρχικά στη θέση της συγχώνευσης, διαφορετικά οι δυο ΔΠ θα ήταν μία (ανατρέξατε στον τρόπο κατασκευής ομοειδών ΔΠ). Αν η μια απ' αυτές, έστω η πρώτη, είναι γενικευμένη, η πρόταση 4.2.2 (ή γενικεύσεις της) μας λέει ότι ο ΧΛ της νέας σύνθετης παρτίδας είναι και πάλι μεγαλύτερος του ΧΛ της παρτίδας στη θέση της οποίας γίνεται η συγχώνευση, ο οποίος με τη σειρά του είναι μεγαλύτερος του ΧΛ της παρτίδας που βρίσκεται αριστερά του, εκ κατασκευής της αρχικής διάταξης και αντίστοιχου ελέγχου σε ενδιάμεσες διατάξεις. Έτσι πρέπει να γίνεται σύγκριση του ΧΛ της εν δυνάμει νέας σύνθετης παρτίδας μόνο με τον ΧΛ της παρτίδας που βρίσκεται δεξιά της.

Όσον αφορά το δεύτερο έλεγχο, αυτός γίνεται επίσης απλά, είτε με υπολογισμό της τιμής του κριτηρίου βελτίστου για τη διάταξη που θα προκύψει αν γίνει συγχώνευση και σύγκρισή του με την τιμή του κριτηρίου πριν τη συγχώνευση, ή με υπολογισμό της διαφοράς της τιμής του κριτηρίου για την παλιά, εν ισχύ διάταξη, από την τιμή του κριτηρίου για τη νέα, εν δυνάμει διάταξη. Ο δεύτερος τρόπος διεξαγωγής αυτού του ελέγχου, αριθμητικά απαιτεί λιγότερο υπολογιστικό φόρτο. Κάτι τέτοιο όμως δε συμβαίνει απαραίτητα και με τις υλοποιήσεις του ως προγράμματος, παρά μόνο αν χρησιμοποιηθούν σύνθετες και άρα μεγαλύτερες δομές δεδομένων, πράγμα που αποφεύγεται προκειμένου να αποφευχθεί η αύξηση του απαιτούμενου αποθηκευτικού χώρου.

Στο σημείο αυτό παραθέτουμε τους τύπους της διαφοράς των τιμών του κριτηρίου βελτίστου για τις περιπτώσεις συγχωνεύσεων από δεξιά και αριστερά.

Έτσι αν εξετάζουμε την εκτέλεση της $ΔΠ_{i+1}^{j}$ δεξιά, αμέσως μετά την $ΔΠ_{i}^{j}$ (αντί για ΔΠ μπορεί να έχουμε και ΓΠ), η διαφορά της νέας από την παλιά τιμή είναι:

$$C(O')-C(O) =$$

- [(πλήθος εργασιών $ΔΠ_{i+1}^{j}$) ×(συνολικός χρόνος επεξεργασίας και εξάρμωσης στην Ο από το τέλος της $ΔΠ_{i}^{j}$ μέχρι την αρχή της $ΔΠ_{i+1}^{j}$)] - [s_j× (πλήθος εργασιών στην Ο που έπονται της τελευταίας εργασίας της $ΔΠ_{i+1}^{j}$)] + [(συνολικός χρόνος επεξεργασίας για εκτέλεση της $ΔΠ_{i+1}^{j}$)×(πλήθος εργασιών στην Ο που βρίσκονται ανάμεσα στην τελευταία εργασία της $ΔΠ_{i}^{j}$ και την πρώτη της $ΔΠ_{i+1}^{j}$)].

Αν εξετάζουμε την εκτέλεση της $ΔΠ_{i}^{j}$ αριστερά, αμέσως πριν την $ΔΠ_{i+1}^{j}$ (αντί για ΓΠ μπορεί να έχουμε ΠΠ), η διαφορά της νέας από την παλιά τιμή είναι:

$$C(O')-C(O) =$$

- [(συνολικός χρόνος επεξεργασίας και εξάρμωσης για την εκτέλεση της $ΔΠ_{i}^{j}$)×(πλήθος εργασιών στην Ο που βρίσκονται ανάμεσα στην τελευταία εργασία της $ΔΠ_{i}^{j}$ και την πρώτη της $ΔΠ_{i+1}^{j}$)] - [s_j× (πλήθος εργασιών στην Ο που έπονται της τελευταίας εργασίας της $ΔΠ_{i+1}^{j}$)] + [(πλήθος εργασιών της $ΔΠ_{i}^{j}$)×(συνολικός χρόνος επεξεργασίας και εξάρμωσης στην Ο από το τέλος της $ΔΠ_{i}^{j}$ μέχρι την αρχή της $ΔΠ_{i+1}^{j}$)].

Προφανώς και στις δυο περιπτώσεις προβαίνουμε σε συγχωνεύσεις αν οι διαφορές που υπολογίζονται με τους παραπάνω τύπους είναι αρνητικές. Αν έχουμε κάποια μηδενική διαφορά δεν προβαίνουμε σε συγχώνευση, αυξάνοντας έτσι την πιθανότητα βελτίωσης από συγχώνευση με επόμενη στη διάταξη βάσει ΧΛ ομοειδή παρτίδα. (Κρατάμε τη διάταξη με τις περισσότερες παρτίδες - ισοδύναμα με το μεγαλύτερο συνολικό χρόνο εξάρμωσης).

Η περιγραφή και του δεύτερου ελέγχου ολοκληρώνει την παρουσίαση του τρίτου και τελευταίου βήματος του προσεγγιστικού αλγορίθμου, το οποίο όπως είπαμε εκτελείται μέχρι να μην είναι δυνατή περαιτέρω βελτίωση στην τιμή του κριτηρίου επίδοσης.

4.3. Σύνοψη του πρώτου προσεγγιστικού αλγορίθμου

Στο κεφάλαιο αυτό θα παρουσιαστεί σύνοψη του προσεγγιστικού αλγορίθμου που αναπτύχθηκε προηγουμένως.

Για το σκοπό αυτό ας συμβολίσουμε με PL(j,i) έναν πίνακα που αποτελείται από τις i πρωτογενείς παρτίδες κάθε ομάδας j ($0 \le j \le B-1$) και με S μια συνδεδεμένη λίστα τα στοιχεία της οποίας περιέχουν αρχικά τις ΠΠ του PL και τελικά τη διάταξη που προτείνεται από τον αλγόριθμο ως η καλύτερη επιτευχθείσα. Με bestSol συμβολίζουμε μια μεταβλητή ελέγχου η οποία γίνεται 1 αν το πλήθος των ΓΠ που φτιάχνονται γίνει ίσο με το πλήθος των ομάδων του προβλήματος που πραγματευόμαστε, και επομένως δεν είναι δυνατή επιπλέον συγχώνευση, ενώ με merge συμβολίζουμε μια μεταβλητή ελέγχου η οποία γίνεται 1 κάθε φορά που πραγματοποιείται μια συγχώνευση, επιβάλλοντας έτσι να γίνεται έλεγχος για νέες συγχωνεύσεις από ακραίο σημείο της διάταξης που προκύπτει (το αριστερότερο αν πρόκειται για συγχωνεύσεις από δεξιά και το δεξιότερο αν πρόκειται για συγχωνεύσεις από αριστερά).

Προσεγγιστικός αλγόριθμος συγχώνευσης παρτίδων για το $TCTS_j$

Βήμα 1:

- Κατασκευή των ΠΠ κάθε ομάδας και συμπλήρωση του πίνακα PL. (Στην κατασκευή βοηθάει η ταξινόμηση του αρχείου που περιέχει τα δεδομένα κατά μη φθίνουσα σειρά αριθμού ταυτότητας ομάδας, και μέσα σε κάθε ομάδα κατά μη φθίνουσα σειρά χρόνων επεξεργασίας των εργασιών. Για ομοειδείς εργασίες με ίσους χρόνους επεξεργασίας έχουμε διάταξη κατά μη φθίνουσα σειρά αύξοντος αριθμού εργασίας.)
- Κατασκευή από τον πίνακα PL της συνδεδεμένης λίστας S.
- bestSol = 0, merge = 1.
- Κατασκευή ΔΠ από τα στοιχεία της λίστας S υπό μορφή νέων στοιχείων με συνολική πληροφορία που αντικαθιστά αυτήν των στοιχείων από τα οποία προέρχονται, βάσει του κανόνα κατασκευής των ΔΠ κάθε ομάδας. (Συρρίκνωση του "μήκους" της S.)

Βήμα 2:

- Ταξινόμηση κατά μη φθίνουσα σειρά ΧΛ των ΔΠ - στοιχείων της S. Οι ΧΛ υπολογίζονται με αναφορά στον πίνακα PL και η ταξινόμηση του βήματος αυτού προφανώς επιφέρει αλλαγές στη σειρά των στοιχείων της S.

- Αν το πλήθος των στοιχείων της S ισούται με των αριθμό των ομάδων B, bestSol = 1.
- Αν bestSol = 0 γίνεται ανακατάταξη των στοιχείων της S ώστε να ληφθεί υπόψη ο κανόνας 4.2.1 για περιπτώσεις ισότητας ΧΛ.

Όσο (bestSol = 0 και merge = 1) εκτελείται το
Βήμα 3:

- merge = 0.
- Μέχρι να φτάσουμε στην προτελευταία από δεξιά παρτίδα της S εκτελείται το

  Βήμα 3.1:

  - Έλεγχοι για δυνατότητα συγχώνευσης παρτίδας με ομοειδή της που βρίσκεται δεξιά της.
  - Αν οι έλεγχοι είναι επιτυχείς γίνεται συγχώνευση και άρα αλλαγή της S, merge = 1, και ο έλεγχος για δυνατότητα συγχώνευσης επανέρχεται στην πρώτη από αριστερά παρτίδα.
  - Αλλιώς, ο έλεγχος για δυνατότητα συγχώνευσης περνάει στην αμέσως δεξιότερη παρτίδα.
- Ανακατάταξη των στοιχείων της S ώστε να ληφθεί υπόψη ο κανόνας 4.2.1 για περιπτώσεις ισότητας ΧΛ.
- Αν το πλήθος των στοιχείων της S ισούται με των αριθμό των ομάδων B, bestSol = 1.
- Αν (bestSol = 0), μέχρι να φτάσουμε στην δεύτερη από αριστερά παρτίδα της S εκτελείται το

  Βήμα 3.2:

  - Έλεγχοι για δυνατότητα συγχώνευσης παρτίδας με ομοειδή της που βρίσκεται αριστερά της.
  - Αν οι έλεγχοι είναι επιτυχείς γίνεται συγχώνευση και άρα αλλαγή της S, merge = 1, και ο έλεγχος για δυνατότητα συγχώνευσης επανέρχεται στην τελευταία δεξιά παρίδα.
  - Αλλιώς, ο έλεγχος για δυνατότητα συγχώνευσης περνάει στην αμέσως αριστερότερη παρτίδα.

- Ανακατάταξη των στοιχείων της S ώστε να ληφθεί υπόψη ο κανόνας 4.2.1 για περιπτώσεις ισότητας ΧΛ.

- Αν το πλήθος των στοιχείων της S ισούται με των αριθμό των ομάδων B, bestSol = 1.

4.4. Υπολογιστική πλοκή του πρώτου προσεγγιστικού αλγορίθμου

Έχουμε ήδη αναφερθεί στο γεγονός ότι ο αλγόριθμος που παρουσιάζουμε στην ενότητα αυτή είναι πολυωνυμικός. Σαφέστερος προσδιορισμός της υπολογιστικής πλοκής του θα γίνει στο σημείο αυτό.

Έτσι όσον αφορά τη χρονική πλοκή του αλγορίθμου απαιτείται:

- χρόνος τάξης O(n) για την κατασκευή του πίνακα PL (θεωρούμε ότι η ταξινόμηση του αρχείου δεδομένων, τάξης O(nlogn), δεν αποτελεί συνιστώσα διαδικασία του αλγορίθμου), O(n) για την κατασκευή της λίστας S από τον πίνακα PL, και O(B max{0<=b<B}n_b) για τη συγχώνευση σε δευτερογενείς παρτίδες των στοιχείων της S, άρα συνολικά χρόνος τάξης O(2n + B max{0<=b<B}n_b) για το βήμα 1.
- χρόνος τάξης O(nlogn) για τη διάταξη ΔΠ βάσει των τιμών των ΧΛ στη χειρότερη περίπτωση που κάθε ΔΠ έχει μια εργασία, και O(n) για την ανακατάταξη στην περίπτωση ίσων ΧΛ, άρα συνολικά χρόνος τάξης O(n(logn+1)) για το βήμα 2.
- χρόνος τάξης O(n(n-1)/2) για τη συγχώνευση παρτίδων από δεξιά, O(n) για ανακατάταξη στην περίπτωση ίσων ΧΛ και O(n) για έλεγχο bestSol, και ομοίως από αριστερά, άρα συνολικά για μια επανάληψη του βήματος 3 απαιτείται χρόνος τάξης O(n(n-1) + 4n). Όμως αυτό εκτελείται συνολικά το πολύ n φορές. Επομένως απαιτείται χρόνος τάξης O(n^2(n+3)).

Συνολικά λοιπόν η χρονική πλοκή του αλγορίθμου είναι τάξης O(n^3).

Από την άλλη ο απαιτούμενος αποθηκευτικός χώρος για την εκτέλεση του αλγορίθμου είναι αρκετά μικρός και καθορίζεται σαφώς από το πλήθος των εργασιών n και των ομάδων B. Είναι δε : ένας πίνακας B ακεραίων για τους χρόνους εξάρμωσης (Bx4 bytes), ένας πίνακας n στοιχείων για τα χαρακτηριστικά μεγέθη των εργασιών (nx12 bytes), ένας πίνακας n το πολύ στοιχείων για τις ΠΠ (nx12 bytes), και μια λίστα που περιέχει n το πολύ στοιχεία για την αποθήκευση της διάταξης των εργασιών (nx20 bytes), καθώς και μερικές μεταβλητές ελέγχου.

Οι παραπάνω αναφορές τόσο στη χρονική πλοκή όσο και στον αποθηκευτικό χώρο που απαιτείται από τον αλγόριθμο, αποδεικνύουν την ευχρηστία του για τον προσδιορισμό κάποιας λύσης σε περιπτώσεις προβλημάτων μεγάλων διαστάσεων.

5. Ακριβής αλγόριθμος για το $TCTS_j$

Η αποτελεσματικότητα του προσεγγιστικού αλγορίθμου που παρουσιάστηκε στην προηγούμενη ενότητα είναι άμεση συνέπεια της πολυωνυμικής πλοκής του καθώς και του πολύ μικρού χώρου που απαιτεί για τη φύλαξη δεδομένων κατά τη διάρκεια εκτέλεσής του. Όμως επειδή στάθηκε αδύνατη η εξαγωγή θεωρητικού φράγματος για το σφάλμα στην τιμή του κριτηρίου επίδοσης που εμφανίζουν οι λύσεις του, κρίθηκε απαραίτητη η κατασκευή ακριβούς αλγορίθμου οι λύσεις του οποίου συγκρίνονται με αυτές του προσεγγιστικού. Ο αλγόριθμος αυτός ακολουθεί το σχήμα δυναμικού προγραμματισμού που προτείνουν οι Monma και Potts και ο Ψαραύτης σε σχετικές εργασίες τους [MoPo89], [Ps80], βελτιωμένο όμως χάρη στις προτάσεις που σχετίζονται με τις παρτίδες, και στο άνω φράγμα της τιμής του κριτηρίου βελτίστου που υπολογίζεται με χρήση του προσεγγιστικού αλγορίθμου της προηγούμενης ενότητας.

5.1. Περιγραφή

Προκειμένου να χρησιμοποιηθεί σωστά το σχήμα δυναμικού προγραμματισμού που περιγράφουμε στο κεφάλαιο αυτό, είναι απαραίτητη η υπόθεση της ύπαρξης καλά καθορισμένης διάταξης εργασιών μέσα στις ομάδες. Στην περίπτωση του προβλήματος που μας απασχολεί η διάταξη αυτή καθορίζεται καλά βάσει του κανόνα SPT.

Η παραπάνω υπόθεση επάγει την ισοδυναμία του προβλήματος εύρεσης βέλτιστης διάταξης εργασιών με το πρόβλημα εύρεσης βέλτιστης διάταξης ενός συνόλου δεικτών σε ομάδες. Το σύνολο αυτό έχει n στοιχεία τα οποία αντιστοιχούν σε διαδοχή ομάδων στη βέλτιστη λύση. Επειδή δε είναι γνωστή η διάταξη με την οποία εκτελούνται οι εργασίες μιας ομάδας, το σύνολο δεικτών σε ομάδες μετασχηματίζεται άμεσα σε διάταξη εργασιών.

Τυπικότερα μπορούμε να πούμε ότι:

Το πρόβλημα διάταξης n εργασιών κατανεμημένων σε B ομάδες με στόχο την ελαχιστοποίηση του συνολικού χρόνου περάτωσης των εργασιών ισοδυναμεί με το ακόλουθο πρόβλημα:
Να ευρεθεί σύνολο δεικτών στις ομάδες $(b_1,b_2,\ldots,b_n)$, $1 \le b_i \le B$ για $i=1,2,\ldots,n$, έτσι ώστε να έχουμε $\min\sum_{i=0}^{B-1} f(b_i, b_{i+1}, g_1{}^i, \ldots, g_B{}^i)$ με $\sum_{b=1}^{B} g_b{}^0 = n$.

Με $g_j{}^i$ συμβολίζουμε τον αριθμό των εργασιών της ομάδας j που απομένει να διαταχθούν μετά τη θέση i, ενώ n είναι το πλήθος των εργασιών. Τέλος f είναι η συνάρτηση προσθετικού κόστους, η τιμή της οποίας για το $TCTS_j$ ορίζεται ως εξής:

$$f(i,j,g_1,\ldots,g_B) = \begin{cases} (n-t+1)p_{g_j}{}^i & \text{αν } i=j \\ (n-t+1)(s_j+p_{g_j}{}^i) & \text{αν} i\neq j \end{cases}$$

όπου $t=n-\sum_{b=1}^{B} g_b$ και $f(0,j,g_1,\ldots,g_B)=0$

Είναι άμεση η λύση του παραπάνω προβλήματος με δυναμικό προγραμματισμό.

Ορίζουμε ως συνάρτηση βέλτιστης τιμής $V(b,g_1,\ldots,g_B)$ το συνολικό κόστος επεξεργασίας g_i εργασιών της ομάδας i (i=1,...,B), που πρέπει ακόμα να διαταχθούν, δοθέντος ότι η τελευταία στην τρέχουσα διάταξη εργασία προέρχεται από την ομάδα b. Για τη V ισχύει ο ακόλουθος αναδρομικός ορισμός:

$$V(b,g_1,\ldots,g_B) = \begin{cases} 0 & \text{αν } g_1=\cdots=g_B=0 \\ \min_{x\in X} \{f(b,x,g_1,\ldots,g_B) + V(x,g_1',\ldots,g_B')\} & \text{αλλιως} \end{cases} \tag{1}$$

όπου $X=\{x : g_x>0\}$

και $g_i' = \begin{cases} g_i-1 & \text{αν } i=x \\ g_i & \text{αλλιως} \end{cases}$

Η διαδικασία επίλυσης συνεχίζεται ως εξής:

Αρχίζουμε με $g_1=\cdots=g_B=0$. Τότε V=0 και για κάθε b από 1 έως B πηγαίνουμε σε λεξικογραφικά μεγαλύτερες τιμές του διανύσματος $(g_1,\ldots,g_B)$.

Για κάθε τέτοιο διάνυσμα εφαρμόζουμε την (1) για όλες τις πιθανές τιμές του b ($1\leq b\leq B$) και αποθηκεύουμε το x^*, βέλτιστο επόμενο δείκτη ομάδας προς θεώρηση. Αυτή η πληροφορία μπορεί να κρατηθεί σε πίνακα, έστω NEXT($b,g_1,\ldots,g_B$). Αυθαίρετα θέτουμε NEXT(b,0,...,0) = 0 για κάθε b=1,...,B.

Όταν ο πίνακας NEXT γεμίσει με την απαιτούμενη πληροφορία, από την (1) μπορούμε να βρούμε το βέλτιστο σύνολο δεικτών, ξεκινώντας κάθε φορά από δείκτη διαφορετικής ομάδας και χρησιμοποιώντας την πληροφορία του NEXT.

5.2. Πλοκή - Αναποτελεσματικότητα χρήσης - Βελτίωση

Η υπολογιστική πλοκή του αλγορίθμου που σχετίζεται με την εκτέλεση της αναδρομικής διαδικασίας (1) είναι τάξεως $B^2\prod_{b=1}^{B}(1+g_b)$ εφόσον υπάρχουν $B\prod_{b=1}^{B}(1+g_b)$ πιθανές καταστάσεις και σε καθεμιά απ' αυτές εξετάζονται B καταστάσεις.

Οι απαιτήσεις μνήμης που είναι και ο κύριος περιοριστικός παράγοντας για τη χρήση του αλγορίθμου είναι τάξεως $B\prod_{b=1}^{B}(1+g_b)$. Προφανώς για προβλήματα με μεγάλο πλήθος ομάδων η διατήρηση πινάκων τέτοιας τάξης μεγέθους οδηγεί συχνά σε προβλήματα ανεπάρκειας αποθηκευτικού χώρου.

Όμως κάνοντας χρήση του ορισμού των ΠΠ και της πρότασης ότι υπάρχει βέλτιστη λύση του $TCTS_j$ στην οποία εκτελούνται αδιάσπαστες, είναι δυνατόν να περιοριστούν οι απαιτήσεις μνήμης κατά τον παράγοντα ελάττωσης των εργασιών σε ΠΠ.

Επίσης μπορούμε, χρησιμοποιώντας την τιμή του κριτηρίου επίδοσης που επιτυγχάνει ο πρώτος προσεγγιστικός αλγόριθμος ως άνω φράγμα, να απορρίπτουμε καταστάσεις στις οποίες η τιμή της συνάρτησης προσθετικού κόστους f υπερβαίνει το άνω αυτό όριο.

Οι παραπάνω βελτιώσεις αν και μοιάζουν ιδιαίτερα απλές στη σύλληψη μείωσαν αποτελεσματικά τις απαιτήσεις μνήμης του αλγορίθμου, με αποτέλεσμα να είναι δυνατή η χρήση του σε περιπτώσεις προβλημάτων πρακτικών διαστάσεων.

6. Δεύτερος προσεγγιστικός αλγόριθμος για το $TCTS_j$

Την υλοποίηση του ακριβούς αλγορίθμου για τον έλεγχο της ποιότητας των λύσεων του πρώτου προσεγγιστικού, ακολούθησε μια δοκιμαστική υλοποίηση του προσεγγιστικού αλγορίθμου που πρότειναν οι Ahn και Hyun [AhHy90], προκειμένου να συγκριθούν ποιοτικά οι δυο αλγόριθμοι, αλλά και να ελεγχθεί αν αποκλίνουν από το βέλτιστο σε ανάλογες περιπτώσεις προβλημάτων δοκιμών. Η διαπίστωση που προέκυψε από παράλληλη πειραματική μελέτη τους, έδειξε ότι δεν συμπεριφέρονται το ίδιο στις περιπτώσεις αστοχίας τους, αλλά με τρόπο μάλλον συμπληρωματικό. Έτσι αποφασίσαμε να τους συνδυάσουμε και να φτιάξουμε ένα νέο προσεγγιστικό αλγόριθμο με τις ιδιότητες και των δύο. Η ιδέα αυτή αποδείχτηκε καλή, εφόσον ο αλγόριθμος που προέκυψε (δεύτερος προσεγγιστικός) έδωσε βέλτιστη λύση σ’ όλα τα προβλήματα δοκιμών που εξετάσαμε.

6.1. Αστοχία του πρώτου προσεγγιστικού αλγορίθμου

Έχουμε ήδη αναφερθεί προηγουμένως (κεφάλαιο 4.2) στην υπέρβαση, η οποία έγινε κατά την κατασκευή του πρώτου προσεγγιστικού αλγορίθμου.

Η αρχική μας υπόθεση για την κατασκευή ΔΠ και τη σύνθεση ΓΠ από ολόκληρες ΔΠ απαιτεί, προκειμένου να χαρακτηρίζει μια βέλτιστη διάταξη, την εκτέλεση ΔΠ διαδοχικά, <u>μόνο εάν αυτές είναι ετεροειδείς.</u> Όμως προφανώς κάτι τέτοιο δεν είναι πάντα δυνατό. Ιδιαίτερα στις περιπτώσεις προβλημάτων, όπου οι ομάδες έχουν διαφορετικό πλήθος ΔΠ, σίγουρα κάποιες ομοειδείς ΔΠ θα πρέπει να εκτελεστούν διαδοχικά. Τότε όμως ο τρόπος με τον οποίο κατασκευάζονται και πολύ περισσότερο ο τρόπος με τον οποίο συντίθενται σε ΓΠ δεν είναι βέλτιστος, εφόσον δεν ισχύουν όλες οι υποθέσεις των σχετικών κατασκευαστικών θεωρημάτων της ενότητας 3.

Στη διαπίστωση αυτή καταλήξαμε μετά από εξέταση των προβλημάτων δοκιμών στα οποία η λύση του πρώτου προσεγγιστικού αλγορίθμου δεν ήταν βέλτιστη: σε όλα αυτά η τελική διάταξη που πρότεινε ο αλγόριθμος αποτελούνταν από πολύ μεγάλες ΓΠ (στις περισσότερες των περιπτώσεων ταυτίζονταν με τις ομάδες).

Στα περισσότερα από τα προβλήματα αυτά ο μέσος χρόνος εξάρμωσης είχε τιμές παραπλήσιες του μέσου χαρακτηριστικού χρόνου επεξεργασίας των ΠΠ.

Στις περισσότερες των περιπτώσεων που ο μέσος χρόνος εξάρμωσης ήταν πολύ μεγαλύτερος του μέσου χαρακτηριστικού χρόνου επεξεργασίας των ΠΠ, οι πρώτες κατά σειρά κατασκευής ΔΠ κάθε ομάδας (οι οποίες πάντα φτιάχνονται σωστά!) περιείχαν όλες, ή σχεδόν όλες, τις ΠΠ των ομάδων με αποτέλεσμα να μην απαιτούνται βήματα συγχώνευσης και ο αλγόριθμος να συμπεριφέρεται βέλτιστα.

Βέλτιστη συμπεριφορά είχαμε και σε περιπτώσεις πολύ μικρού μέσου χρόνου εξάρμωσης σε σχέση με το μέσο χαρακτηριστικό χρόνο επεξεργασίας των ΠΠ, εφόσον τότε οι ΔΠ που φτιάχνονταν κατά κανόνα ταυτίζονταν με τις ΠΠ, με αποτέλεσμα να αποφεύγεται το ατόπημα που αρχικά συζητήθηκε.

6.2. Αστοχία του προσεγγιστικού αλγορίθμου των Ahn και Hyun

Η αστοχία του αλγορίθμου των Ahn και Hyun, δεν μπορεί να αποδοθεί σε παράβαση κάποιου κανόνα που χαρακτηρίζει βέλτιστες λύσεις και σχετικές οντότητες όπως προηγουμένως, γιατί τέτοιους κανόνες αυτός δεν επικαλείται (εκτός του ενδοομαδικού SPT ο οποίος φροντίζουμε πάντα να ισχύει), ούτε χρησιμοποιεί οντότητες αντίστοιχες των παρτίδων.

Έτσι τα προβλήματα δοκιμών στα οποία δεν έδωσε βέλτιστες λύσεις ανήκουν σε διάφορες κατηγορίες με βάση τις σχέσεις των μεγεθών των βασικών οντοτήτων τους.

Ερευνώντας τα προσεκτικά, διαπιστώσαμε ότι η απόκλιση από το βέλτιστο οφείλονταν σε πολλές περιπτώσεις σε μεταθέσεις που δεν πραγματοποιήθηκαν λόγω ισότητας των τιμών του κριτηρίου βελτίστου στις δυο διαφορετικές διατάξεις. Τέτοιες καταστάσεις σπανίζουν στον πρώτο προσεγγιστικό αλγόριθμο που κατασκευάσαμε λόγω σωστής σύνθεσης των ΔΠ, ή αντιμετωπίζονται επιτυχώς με τη χρήση του κανόνα 4.2.1.

Επίσης σε όχι λιγότερες περιπτώσεις, επειδή οι μεταθέσεις γίνονται σε σχέση μόνο με ετεροειδείς εργασίες, χωρίς να λαμβάνονται υπόψη οι εναπομένουσες ομοειδείς είναι πιθανό να προβούμε σε μεταθέσεις καλές τοπικά αλλά όχι ολικά. (Προς την κατεύθυνση του "ολικά καλού" οδηγούμαστε με κατασκευές όπως τις παρτίδες όταν αυτές φτιάχνονται σωστά, ή καλύτερα ισχύουν οι προϋποθέσεις για να φτιαχτούν σωστά.)

6.3. Η ιδέα του δεύτερου προσεγγιστικού αλγόριθμου

Στα δυο προηγούμενα κεφάλαια αναλύθηκε η αστοχία του πρώτου προσεγγιστικού αλγορίθμου της ενότητας 4 και του προσεγγιστικού αλγορίθμου των Ahn και Hyun. Από την ανάλυση αυτή διαπιστώνουμε ότι οι δυο αυτοί προσεγγιστικοί αλγόριθμοι δεν αποτυγχάνουν για τους ίδιους λόγους σε περιπτώσεις αστοχίας. Μάλιστα θα μπορούσαμε να πούμε ότι ο πρώτος αποτυγχάνει επειδή δεν χρησιμοποιεί την τακτική της θέσει-βελτίωσης του δεύτερου, και ο δεύτερος επειδή δεν χρησιμοποιεί την τακτική της με επαύξηση βελτίωσης του πρώτου.

Σκεφτήκαμε λοιπόν να βάλουμε τους δυο αυτούς αλγορίθμους να δουλέψουν διαδοχικά:

> Πρώτος εκτελείται μέχρι περατώσεως ο προσεγγιστικός αλγόριθμος που παρουσιάσαμε στην ενότητα 4. Αυτός παράγει μια διάταξη παρτίδων οι οποίες διασπώνται στα συστατικά τους δίνοντας έτσι μια αρχική διάταξη η οποία είναι απαραίτητη για την έναρξη της διαδικασίας θέσει-βελτίωσης του αλγορίθμου των Ahn και Hyun. Αυτή η αρχική διάταξη ικανοποιεί τον κανόνα διάταξης ομοειδών εργασιών κατά μη φθίνουσα σειρά των χρόνων επεξεργασίας τους (ή ΠΠ κατά μη φθίνουσα σειρά των χαρακτηριστικών χρόνων επεξεργασίας τους), εφόσον αποτελεί μέλημα και του πρώτου προσεγγιστικού αλγορίθμου το οποίο τηρείται πάντοτε, τόσο στην κατασκευή όσο και στις συγχωνεύσεις παρτίδων (Θεώρημα 3.1). Επιπλέον στη διάταξη αυτή ομοειδείς εργασίες που ανήκουν σε καλά κατασκευασμένες ΔΠ είναι διαδοχικές, πράγμα που βοηθάει να ξεπεραστεί η τοπικότητα της θέσει-βελτίωσης. Κατόπιν εκτελείται μέχρι περάτωσης η διαδικασία θέσει-βελτίωσης των Ahn και Hyun.

Η διάταξη που δίνει με την περάτωσή του ο πρώτος προσεγγιστικός αλγόριθμος θα μπορούσε δυνητικά να αναλυθεί όχι μόνο στα στοιχειώδη συστατικά της (εργασίες ή ΠΠ), αλλά και σε συνθετότερα όπως ΔΠ καλά ορισμένες (τέτοιες που γειτονεύουν με ετεροειδείς ΔΠ). Όμως τέτοια υλοποίηση δεν έγινε εφόσον η διαδικασία των θέσει μεταθέσεων είναι πολυωνυμική ως προς το πλήθος των εργασιών n και των ομάδων B - και άρα αποδοτική ακόμα και χωρίς τον περιορισμό του συνόλου των δυνατών θέσει-μεταθέσεων που επιφέρει η διατήρηση των καλά ορισμένων ΔΠ ως στοιχειωδών συστατικών στην αρχική διάταξη στην οποία επιχειρούνται θέσει-μεταθέσεις.

Στο επόμενο υποκεφάλαιο παρουσιάζονται τα βήματα του δεύτερου προσεγγιστικού αλγόριθμου, την ιδέα του οποίου παρουσιάσαμε εδώ.

6.4. Βήματα του δεύτερου προσεγγιστικού αλγορίθμου

Στο κεφάλαιο αυτό θα παρουσιάσουμε τα βήματα του δεύτερου προσεγγιστικού αλγορίθμου που κατασκευάστηκε για τη λύση του $TCTS_j$, με ιδιαίτερη έμφαση στο τρίτο βήμα εφόσον αυτό αποτελείται από τη διαδικασία θέσει-συγχωνεύσεως η οποία δεν έχει περιγραφεί αναλυτικά μέχρι τώρα.

1ο Β Η Μ Α

Το πρώτο βήμα του δεύτερου προσεγγιστικού αλγορίθμου αποτελείται από όλα τα βήματα του πρώτου προσεγγιστικού αλγορίθμου, όπως αυτά περιγράφονται στο κεφάλαιο 4.2.

2ο Β Η Μ Α

Στο δεύτερο βήμα γίνεται γίνεται διάσπαση της διάταξης που προκύπτει από το βήμα 1 σε στοιχειώδη συστατικά της. Η δική μας προσέγγιση θεωρεί ως θεμελιώδη συστατικά τις ΠΠ και όχι τις εργασίες εφόσον το θεώρημα 3.4 που αναφέρεται στην ύπαρξη βέλτιστης λύσης στην οποία οι ΠΠ εκτελούνται αδιάσπαστες ισχύει, και περιορίζει κάπως το σύνολο των δυνατών θέσει-αντιμεταθέσεων.

Το βήμα αυτό θα μπορούσε να γίνει πιο σύνθετο ώστε να μην διασπά τις ΔΠ που γειτονεύουν με ετεροειδείς ΔΠ στην τελική διάταξη του βήματος 1, όμως κάτι τέτοιο αποφεύχθηκε για λόγους ευκολότερης υλοποίησης.

3ο Β Η Μ Α

Με τη συμπλήρωση των δύο πρώτων βημάτων έχει φτιαχτεί μια αρχική διάταξη στην οποία ισχύει ο ενδοομαδικός SPT και επομένως μπορεί να εφαρμοστεί η διαδικασία θέσει-συγχωνεύσεων του αλγορίθμου των Ahn και Hyun.

Η διαδικασία αυτή εξελίσσεται προς δυο κατευθύνσεις, έμπροσθεν και όπισθεν. Όταν γίνονται έμπροσθεν μεταθέσεις ξεκινούμε από την πρώτη αριστερά θέση της διάταξης και ψάχνουμε για ετεροειδή ΠΠ αυτής που βρίσκεται στην εν λόγω θέση (ή σύνολο ομοειδών ΠΠ, ετεροειδών αυτής που βρίσκεται στην εν λόγω θέση και διαδοχικών στη διάταξη), η οποία αν μεταφερθεί εκεί (μετατοπίζοντας όλες τις άλλες δεξιότερα) θα δώσει διάταξη με μικρότερη τιμή του κριτηρίου βελτίστου. Είναι φανερό ότι προκειμένου να διατηρείται πάντοτε σε ισχύ ο ενδοομαδικός SPT, αν η πρώτη ΠΠ μιας ομάδας δεν μπορεί να μεταφερθεί σε κάποια θέση καμμιά άλλη ομοειδής δεν γίνεται να μεταφερθεί στη θέση αυτή κατά την τρέχουσα επανάληψη της διαδικασίας μετάθεσης. Αν γίνει μετάθεση σε κάποια θέση ελέγχουμε και για αλλη πιθανή μετάθεση στη θέση αυτή, αλλιώς προχωρούμε σε επόμενη θέση. Αυτό γίνεται μέχρις ότου φτάσουμε μια θέση πριν τη δεξιότερη κατά τις έμπροσθεν μεταθέσεις. Ανάλογα λειτουργεί η διαδικασία των όπισθεν μεταθέσεων, μόνο που η εκκίνηση γίνεται από την τελευταία δεξιά στη διάταξη θέση και επαναλαμβάνεται μέχρις ότου φτάσουμε στην δεύτερη αριστερά θέση της διάταξης.

6.5. Υπολογιστική πλοκή του δεύτερου προσεγγιστικού αλγορίθμου

Ο δεύτερος προσεγγιστικός αλγόριθμος που κατασκευάσαμε είναι πολυωνυμικός εφόσον προκύπτει από παράθεση δυο πολυωνυμικών αλγορίθμων και μιας πολυωνυμικής διαδικασίας.

Ειδικότερα απαιτείται:

- για το πρώτο βήμα χρόνος τάξης O(n^3), όπου n πλήθος των εργασιών.
- για το δεύτερο βήμα χρόνος τάξης O(n).
- και για το τρίτο βήμα χρόνος τάξης O($\lceil \frac{n}{B} \rceil^3 B^2 n$).

Επίσης ο απαιτούμενος αποθηκευτικός χώρος είναι ίδιος μ' αυτόν που χρησιμοποιείται στον πρώτο προσεγγιστικό αλγόριθμο, εφόσον το βήμα 3 χρησιμοποιεί τις δομές του βήματος 1, με έναν επιπλέον πίνακα ακεραίων B θέσεων που χρησιμοποιείται για ελέγχους προκειμένου να διατηρείται σε ισχύ ο ενδοομαδικός κανόνας SPT.

7. Υλοποίηση

Στην ενότητα αυτή θα αναφερθούμε στο περιβάλλον στο οποίο έγινε η υλοποίηση των αλγορίθμων και τα πειράματα δοκιμών, καθώς επίσης και σε βασικές προγραμματιστικές δομές που χρησιμοποιήθηκαν στις διάφορες υλοποιήσεις.

7.1. Περιβάλλον υλοποίησης

Η υλοποίηση των αλγορίθμων έγινε σε γλώσσα C και λειτουργικό σύστημα UNIX SunOS 4.1.2. Η επιλογή της C ως γλώσσας προγραμματισμού οφείλεται κυρίως στην περιγραφική της δύναμη και τις δυνατότητες που παρέχει για ορισμό σύνθετων προγραμματιστικών δομών, χρήσιμων για την παράσταση των αντίστοιχων μαθηματικών δομών των αλγορίθμων. Επίσης σοβαρό κριτήριο για την επιλογή της αποτέλεσε η ποιότητα του κώδικα που παράγει σε σχετικά πολύ μικρό χρόνο μεταγλώτισσης. Η μετάφραση έγινε με χρήση της έκδοσης 4.1.2 του μεταφραστού της C, χωρίς επιλογή βελτιστοποίησης.

Τα μηχανήματα που χρησιμοποιήθηκαν ήταν σταθμοί εργασίας SUN και SRARC, ενώ τα αποτελέσματα προέρχονται στο σύνολό τους από SPARCstations 2, που έχουν χρονισμό 40 MHz, απόδοση της τάξης των 28 MIPS και μνήμη 48 Mbytes.

Συγκριτικά χρησιμοποιήθηκαν και PCs 386 στα οποία βέβαια οι χρόνοι εκτέλεσης αυξάνονταν σημαντικά. Επίσης αύξηση παρατηρήθηκε και στο πλήθος των προβλημάτων που δε λύθηκαν από τον ακριβή αλγόριθμο, λόγω ανεπάρκειας της διαθέσιμης μνήμης.

Όμως το γεγονός ότι παράγονται συνεχώς γρηγορότερα μηχανήματα, με ολοένα και πιο μεγάλες μνήμες, καθιστά την επιλογή των SPARCstations 2 για την πειραματική αξιολόγηση των αλγορίθμων, αν μη τι άλλο ρεαλιστική.

7.2. Υλοποίηση των αλγορίθμων

Η υλοποίηση των διάφορων προσεγγιστικών αλγορίθμων επειδή στόχευε εκτός των άλλων και στη μεταξύ τους σύγκριση, έγινε με τη χρήση όμοιων δομών δεδομένων τις οποίες εξετάζουμε στο επόμενο υποκεφάλαιο. Η υλοποίηση του ακριβούς αλγορίθμου παρουσιάζεται ακολούθως σε χωριστό υποκεφάλαιο.

7.2.1. Υλοποίηση των προσεγγιστικών αλγορίθμων

Κατά την υλοποίηση των προσεγγιστικών αλγορίθμων χρησιμοποιούνται τρεις βασικές δομές "αποθήκευσης" και μια "λειτουργική". Η παραπάνω διάκριση γίνεται προκειμένου να επισημανθεί το γεγονός ότι όλες οι αλλαγές που επιφέρουν οι αλγόριθμοι σε ενδιάμεσες διατάξεις μέχρις ότου τερματίσουν γίνονται σε μιά μόνο δομή, η οποία λαμβάνει τις απαραίτητες πληροφορίες για την αναδιάρθρωσή της από τις βασικές δομές αποθήκευσης όπου αναφέρεται έμμεσα, με τιμές βασικών πεδίων της.

Αναλυτικότερα χρησιμοποιούνται:

- ένα διάνυσμα SetUp ακεραίων, διάστασης ίσης με τον αριθμό των ομάδων B.
- ένας πίνακας JOB διάστασης ίσης με το πλήθος των εργασιών n, στοιχεία του οποίου είναι δομές που περιγράφουν τις εργασίες, δηλαδή τριάδες της μορφής (αύξων αριθμός εργασίας, χρόνος επεξεργασίας εργασίας, ομάδα στην οποία ανήκει).
- ένας πίνακας PL διάστασης B, στοιχεία του οποίου είναι δείκτες σε πίνακες PL[i] που αποτελούνται από δομές με πληροφορία σχετική με τις ΠΠ κάθε ομάδας και έχουν ανάλογες διαστάσεις. Οι δομές που περιγράφουν τις ΠΠ είναι τριάδες της μορφής (χαρακτηριστικός χρόνος επεξεργασίας της ΠΠ, δείκτης στον πίνακα JOB στη θέση που αρχίζει η αποθήκευση της ΠΠ, πλήθος εργασιών της ΠΠ, και βοηθούν εκτός των άλλων στην παραγωγή μιας διάταξης εργασιών από μια διάταξη παρτίδων.
- μια απλά συνδεδεμένη λίστα στοιχείων στην οποία αποθηκεύονται οι παρτίδες που κατασκευάζονται στις διάφορες φάσεις εκτέλεσης και απεικονίζεται η σειρά με την οποία διατάσσονται. Σ' αυτήν αναφερόμαστε μέσω του δείκτη ORDER. Τα στοιχεία της λίστας είναι δομές που αποτελούνται από τρεις ακεραίους (αριθμός ταυτότητας ομάδας, αρχική ΠΠ, τελική ΠΠ) κι ένα δείκτη σε επόμενη τέτοια δομή.

Οι συγχωνεύσεις αλλά και οι μεταθέσεις που παράγουν οι αλγόριθμοι γίνονται στα στοιχεία της παραπάνω λίστας, η οποία φτιάχνεται πριν από την έναρξη των βημάτων βελτίωσης.

Οι απαιτούμενοι έλεγχοι γίνονται με αναφορά στους πίνακες PL, JOB και SetUp μέσω των δομών της λίστας ORDER.

Επειδή είναι απαραίτητο να διατρέχουμε τη λίστα και προς την αντίθετη κατεύθυνση και αυτή είναι απλά συνδεδεμένη, σε αρκετές περιπτώσεις χρειάζεται η δημιουργία αντιγράφου της όπου οι δομές συνδέονται κατ' αντίθετη φορά, το οποίο "ελευθερώνεται" μόλις σταματήσει να είναι χρήσιμο.

Στο τέλος της εκτέλεσης κάθε αλγορίθμου, η λίστα ORDER περιέχει την καλύτερη δυνατή διάταξη παρτίδων που μπορεί να δώσει ο αλγόριθμος, η οποία με αναφορά στον PL και ακολούθως στον JOB δίνει την τελική διάταξη εργασιών. Πρέπει να σημειωθεί ότι είναι απαραίτητη η αποθήκευση δεδομένων στον JOB με τρόπο ώστε να διατηρείται η μη φθίνουσα σειρά στις ταυτότητες των ομάδων, μέσα σ' αυτές η μη φθίνουσα σειρά των χρόνων επεξεργασίας, και για ίσους χρόνους επεξεργασίας ομοειδών εργασιών η μη φθίνουσα σειρά ως προς τον αύξοντα αριθμό τους (ταυτότητα εργασίας). Διαφορετικά δε γίνεται σωστή αναφορά στις δομές αποθήκευσης, ούτε ανακατασκευή της τελικής ακολουθίας εργασιών, από την τελική λίστα παρτίδων.

7.2.2. Υλοποίηση του ακριβούς αλγορίθμου

Κατά την υλοποίηση του ακριβούς αλγορίθμου δεν χρησιμοποιήθηκαν ιδιαίτερα σύνθετες δομές αποθήκευσης, προκειμένου να αποφύγουμε περιττή κατανάλωση του αποθηκευτικού χώρου.

Έτσι χρησιμοποιήθηκαν κατ' αποκλειστικότητα πίνακες ακεραίων.

8. Πειραματική αξιολόγηση των αλγορίθμων

Ακολουθεί αναφορά στην πειραματική αξιολόγηση των αλγορίθμων. Ειδικότερα παρουσιάζονται τα κριτήρια αξιολόγησης, ο τρόπος με τον οποίο κατασκευάζονται τα προβλήματα δοκιμών και τέλος παρατίθενται πίνακες αποτελεσμάτων και σχόλια πάνω σ' αυτά.

8.1. Κριτήρια αξιολόγησης

Βασικά κριτήρια αξιολόγησης για τους προσεγγιστικούς αλγορίθμους αποτελούν το μέσο σχετικό σφάλμα ως προς τη βέλτιστη λύση, και η συχνότητα με την οποία επιτυγχάνεται βέλτιστο.

Η μέση τιμή του σχετικού σφάλματος υπολογίζεται για προβλήματα τα οποία δεν λύνονται βέλτιστα και προφανώς η τιμή αυτή ελαττώνεται σημαντικά όταν υπολογίζεται αναφορικά με το σύνολο των προβλημάτων τα οποία λύνονται (βέλτιστα και μη).

Το μέσο απόλυτο σφάλμα είναι ένα κριτήριο που δεν μπορεί να εκτιμηθεί σωστά σε περιπτώσεις αθροιστικών συναρτήσεων κόστους. Αν μάλιστα τα μεγέθη τιμών των οντοτήτων είναι μεγάλα, αυξάνεται πολύ περισσότερο η δυσκολία σωστής εκτίμησής του. Για το λόγο αυτό δε λαμβάνεται υπόψη στην μελέτη μας.

Προκειμένου να εκτιμηθεί η βελτίωση που επιφέρουν οι σχετικές με τις παρτίδες προτάσεις αλλά και η χρήση άνω φραγμάτων στο σχήμα του δυναμικού προγραμματισμού των Monma και Potts, καλό είναι να υπολογιστεί το μέσο μέγεθος των προβλημάτων που λύνονται (βέλτιστα) από αυτόν χωρίς διακοπή λόγω ανεπάρκειας αποθηκευτικού χώρου.

Τέλος κριτήριο αξιολόγησης σπουδαίο για όλους τους αλγορίθμους αποτελεί ο μέσος χρόνος εκτέλεσής τους.

8.2. Κατασκευή προβλημάτων δοκιμών

Στη διαδικασία κατασκευής των προβλημάτων δοκιμών λαμβάνονται υπόψη πέντε παράμετροι:

- το πλήθος εργασιών,

- το πλήθος ομάδων,
- ο παράγοντας ελάττωσης εργασιών μιας ομάδας σε ΠΠ,
- το κάτω όριο στις τιμές των χρόνων εξάρμωσης, και
- το άνω όριο στις τιμές των χρόνων εξάρμωσης.

Οι δυο πρώτες παράμετροι δεν χρειάζονται καμμιά ερμηνεία, ο λόγος τους όμως καθορίζει το μέσο πλήθος εργασιών ανά ομάδα.

Ο παράγοντας ελάττωσης ομοειδών εργασιών σε ΠΠ είναι ένας αριθμός με τιμές στο διάστημα [0,1], ο οποίος καθορίζει το κλάσμα των εργασιών της ομάδας οι οποίες αποτελούν ΠΠ. Για παράδειγμα παράγοντας ελάττωσης 0.6 σε ομάδα 15 εργασιών, σημαίνει ότι αυτές ανήκουν σε 9 το πολύ ΠΠ (ισοδύναμα έχουμε 9 το πολύ διαφορετικές τιμές χρόνων επεξεργασίας για τις εργασίες της ομάδας).

Τα άνω και κάτω όρια στις τιμές των χρόνων εξάρμωσης καθορίζουν το διάστημα τιμών στο οποίο αυτοί κατανέμονται ομοιόμορφα.

Οι χρόνοι επεξεργασίας προέρχονται από ομοιόμορφη κατανομή στο διάστημα [1,100].

Κρίνεται χρήσιμο στην περίπτωση που ο χρόνος εξάρμωσης είναι σταθερός να λαμβάνεται ως ποσοστό του μέσου χρόνου επεξεργασίας. Ιδιαίτερα μελετούμε τις περιπτώσεις που ο σταθερός χρόνος εξάρμωσης είναι 5%, 50%, 100% και 150% του μέσου χρόνου επεξεργασίας.

8.3. Παράθεση και σχολιασμός αποτελεσμάτων

Στο κεφάλαιο αυτό θα δώσουμε πίνακες αποτελεσμάτων σχετικούς με τα κριτήρια αξιολόγησης για όλους τους αλγόριθμους που κατασκευάστηκαν και για διάφορες κατηγορίες προβλημάτων δοκιμών.

Οι κατηγορίες αυτές είναι : 200 εργασίες σε 20 ομάδες με σταθερό χρόνο εξάρμωσης ίσο με 5, 50, 100 και 150, 200 εργασίες σε 20 ομάδες με χρόνο εξάρμωσης ομοιόμορφα κατανεμημένο στο διάστημα [10, 150], 300 εργασίες σε 20 ομάδες με σταθερό χρόνο εξάρμωσης ίσο με 5, 50, 100 και 150, 300 εργασίες σε 20 ομάδες με χρόνο εξάρμωσης ομοιόμορφα κατανεμημένο στο διάστημα [10, 150], 100 εργασίες σε 8 ομάδες με με σταθερό χρόνο εξάρμωσης ίσο με 5, 50, 100 και 150, 100 εργασίες σε 8 ομάδες με χρόνο εξάρμωσης ομοιόμορφα κατανεμημένο στο διάστημα [10, 150], 200 εργασίες σε 5 ομάδες με σταθερό χρόνο εξάρμωσης ίσο με 5, 50, 100 και 150, 200 εργασίες σε 5 ομάδες με χρόνο εξάρμωσης ομοιόμορφα κατανεμημένο στο διάστημα [10, 150], 150 εργασίες σε 4 ομάδες με χρόνο εξάρμωσης ίσο με 5, 50, 100 και 150 και τέλος 150 εργασίες σε 4 ομάδες με χρόνο εξάρμωσης ομοιόμορφα κατανεμημένο στο διάστημα [10, 150]. (Πίνακας 8.3.1)

Κατηγ. Προβλημ.	Αρ. Εργασιών	Αρ. Ομάδων	Χρόνοι Εξάρμωσης
1	200	20	5 50 100 150 ˜[10,150]
2	300	20	5 50 100 150 ˜[10,150]
3	100	8	5 50 100 150 ˜[10,150]
4	200	5	5 50 100 150 ˜[10,150]
5	150	4	5 50 100 150 ˜[10,150]

Πίνακας 8.3.1 : Κατηγορίες προβλημάτων δοκιμών.

Σε όλες τις παραπάνω κατηγορίες προβλημάτων θεωρήσαμε παράγοντα ελάττωσης ίσο με 1, με αποτέλεσμα να είναι πολύ μεγάλος ο αριθμός των ΠΠ που σχηματίζονταν (περίπου ίσος με τον αριθμό των εργασιών). Αυτό έγινε επειδή θεωρήσαμε προβλήματα διαφόρων διαστάσεων ως προς το πλήθος των εργασιών σε κάθε ομάδα και διαπιστώσαμε ότι αυτό δεν είναι παράγοντας κρίσιμος για την ποιότητα των λύσεων που δίνουν οι αλγόριθμοι. Αντίθετα αν πρόκειται να ελέγξουμε τη συμπεριφορά του ακριβούς αλγορίθμου ως προς τον τερματισμό σε βέλτιστη λύση, ο παράγοντας ελάττωσης είναι μέγεθος ιδιαίτερα κρίσιμο γιατί συνεπάγεται σημαντική ελάττωση του χώρου των δυνατών καταστάσεων (οι ΠΠ των ομάδων είναι μεγάλες και λίγες). Θεωρώντας ότι ο ακριβής αλγόριθμος τερματίζει σε κάθε περίπτωση εκτέλεσής του στην οποία δεν δημιουργείται πρόβλημα υπερχείλισης μνήμης, ανεξαρτήτως χρόνου εκτέλεσης, διαπιστώσαμε ότι ο ακριβής αλγόριθμος τερματίζει σε βέλτιστη λύση, όταν ο αριθμός των εργασιών κυμαίνεται σε μερικές εκατοντάδες (100-300) και των ομάδων σε μερικές δεκάδες (10-30), ενώ ο παράγοντας ελάττωσης είναι από 0.1 έως 0.6.

Για καθεμιά από τις παραπάνω 25 διαφορετικές κατηγορίες προβλημάτων λύθηκαν 20 διαφορετικά προβλήματα (σύνολο 500) από κάθε αλγόριθμο. Τα αποτελέσματα παρουσιάζονται χωριστά για κάθε αλγόριθμο. Φυσικά όλα αυτά τα προβλήματα λύθηκαν βέλτιστα και από τον ακριβή αλγόριθμο προκειμένου να μπορούμε να υπολογίσουμε την τιμή του μέσου σχετικού σφάλματος. Προβλήματα των παραπάνω κατηγοριών στα οποία ο ακριβής αλγόριθμος οδηγούνταν σε υπεχείλιση μνήμης πριν φτάσει σε βέλτιστη λύση αντικαταστάθηκαν με άλλα ίδιας κατηγορίας. Έτσι για το σύνολο των 500 αυτών προβλημάτων δεν θα παρουσιάσουμε αποτελέσματα με βάση τα κριτήρια επίδοσης για τον ακριβή αλγόριθμο.

Πρώτος προσεγγιστικός αλγόριθμος

Ο πρώτος προσεγγιστικός αλγόριθμος με βάσει τα αποτελέσματα του πίνακα 8.3.2 εμφανίζει πολύ καλή συμπεριφορά ως προς το χρόνο εκτέλεσης σε όλα τα προβλήματα στα οποία εφαρμόστηκε (μέση τιμή 0.124 δευτερόλεπτα CPU). Όσον αφορά τα προβλήματα τα οποία λύνει βέλτιστα αυτά χαρακτηρίζονται από σταθερό χρόνο εξάρμωσης με τιμή 5, 100 και 150 (πολύ μικρό ή πολύ μεγάλο σε σχέση με το μέσο χρόνο επεξεργασίας). Η μέση τιμή δε του σχετικού σφάλματος της τιμής του κριτηρίου επίδοσης που παρατηρήθηκε είναι 0.064%.

Ο προσεγγιστικός αλγόριθμος των Ahn και Hyun

Ο προσεγγιστικός αλγόριθμος των Ahn και Hyun με βάση τα αποτελέσματα που παρουσιάζονται στον πίνακα 8.3.3 απαιτεί σημαντικά περισσότερο χρόνο CPU από τον πρώτο προσεγγιστικό αλγόριθμο (μέση τιμή 45.44 δευτερόλεπτα CPU). Τα προβλήματα στα οποία δίνει βέλτιστη λύση δεν μοιάζει να ανήκουν σε συγκεκριμένες κατηγορίες, ενώ το μέσο σχετικό σφάλμα της τιμής του κριτηρίου επίδοσης είναι 0.096%

Δεύτερος προσεγγιστικός αλγόριθμος

Ο δεύτερος προσεγγιστικός αλγόριθμος που κατασκευάσαμε, όπως ήδη έχουμε πει λύνει βέλτιστα και τα 500 διαφορετικά προβλήματα που μελετούμε. Επομένως σχόλιο μπορούμε να κάνουμε μόνο για τον χρόνο εκτέλεσης του ο οποίος έχει μέση τιμή 23.29 δευτερόλεπτα CPU. Είναι δηλαδή αρκετά μεγαλύτερος από το μέσο χρόνο εκτέλεσης του πρώτου προσεγγιστικού αλγορίθμου και περίπου μισός από τον χρόνο εκτέλεσης του προσεγγιστικού αλγορίθμου των Ahn και Hyun, πράγμα αναμενόμενο εφόσον η αρχική διάταξη με την οποία ξεκινάει η διαδικασία θέσει-συγχώνευσης στο δεύτερο προσεγγιστικό αλγόριθμο είναι καλύτερη αυτής που προτείνεται από τους Ahn και Hyun.

Κατηγ. Προβλημ.	Λύθ. Βέλτιστα	Μ. Σχ. Σφάλμα	Μ. Χρόν. Εκτ.(CPU sec)
200 20 5	17	0.19	0.1
200 20 50	11	0.27	0.14
200 20 100	19	0.012	0.1
200 20 150	20	-	0.0
200 20 ˜[10,150]	13	0.09	0.1
300 20 5	18	0.046	0.2
300 20 50	7	0.86	0.16
300 20 100	16	0.032	0.2
300 20 150	20	-	0.0
300 20 ˜[10,150]	16	0.11	0.12
100 8 5	19	0.02	0.1
100 8 50	8	0.09	0.1
100 8 100	16	0.04	0.1
100 8 150	20	-	0.0
100 8 ˜[10,150]	12	0.07	0.1
200 5 5	13	0.12	0.3
200 5 50	8	0.88	0.34
200 5 100	12	0.32	0.2
200 5 150	19	0.04	0.1
200 5 ˜[10,150]	10	0.21	0.2
150 4 5	13	0.09	0.2
150 4 50	12	0.14	0.2
150 4 100	15	0.08	0.2
150 4 150	18	0.02	0.2
150 4 ˜[10,150]	11	0.21	0.3

Πίνακας 8.3.2 : Αποτελέσματα από τον πρώτο προσεγγιστικό αλγόριθμο.

Κατηγ. Προβλημ.	Λύθ. Βέλτιστα	Σχ. Σφάλμα	Χρόνος Εκτ.(CPU sec)
200 20 5	12	0.23	56.32
200 20 50	8	0.07	48.52
200 20 100	17	0.092	61.7
200 20 150	18	0.02	64.65
200 20 ˜[10,150]	16	0.05	43.12
300 20 5	8	0.67	153.21
300 20 50	11	0.92	163.87
300 20 100	15	0.072	112.23
300 20 150	12	0.12	126.12
300 20 ˜[10,150]	17	0.05	119.87
100 8 5	11	0.05	14.12
100 8 50	13	0.32	11.08
100 8 100	13	0.11	13.21
100 8 150	9	0.31	14.87
100 8 ˜[10,150]	18	0.03	18.92
200 5 5	18	0.02	4.54
200 5 50	16	0.08	6.11
200 5 100	17	0.07	5.32
200 5 150	15	0.11	6.12
200 5 ˜[10,150]	14	0.2	5.64
150 4 5	14	0.08	18.21
150 4 50	9	0.76	19.11
150 4 100	13	0.092	15.63
150 4 150	10	0.23	16.39
150 4 ˜[10,150]	11	0.3	17.1

Πίνακας 8.3.3 : Αποτελέσματα από τον αλγόριθμο των Ahn και Hyun.

9. Επίλογος

Τελειώνοντας την αναφορά μας στο πρόβλημα $TCTS_j$ θα εκθέσουμε τα συμπεράσματά μας από τη μελέτη του, και θα προτείνουμε μελλοντικές επεκτάσεις και βελτιώσεις καθώς και εφαρμογές του σε συγγενή προβλήματα.

9.1. Συμπεράσματα

Στην εργασία αυτή μελετήσαμε το πρόβλημα ελαχιστοποίησης του συνολικού χρόνου περάτωσης n εργασιών οι οποίες κατανέμονται σε B διαφορετικές ομάδες, και πρέπει να εκτελεστούν σε μια μηχανή, παρουσία χρόνων εξάρμωσης.

Για την επίλυση του προβλήματος λαμβάνουμε υπόψη την κατανομή εργασιών σε ομάδες και προσπαθούμε να φτιάξουμε υποσύνολα αυτών, τα οποία εκτελούνται διαδοχικά σε κάποια βέλτιστη διάταξη. Τέτοια υποσύνολα τα ονομάζουμε παρτίδες και τα διακρίνουμε σε πρωτογενείς (ΠΠ), δευτερογενείς (ΔΠ) και γενικευμένες (ΓΠ). Εξάγουμε προτάσεις σχετικές με τις παραπάνω οντότητες και χαρακτηριστικά των βέλτιστων λύσεων, προκειμένου να μπορέσουμε να κατασκευάσουμε τέτοιες.

Από τη σχετική με τις παρτίδες μελέτη κατασκευάσαμε έναν πολυωνυμικό προσεγγιστικό αλγόριθμο, ο οποίος στηρίζεται σε συγχωνεύσεις παρτίδων με στόχο να βελτιώνεται η τιμή του κριτηρίου επίδοσης και παράλληλα να διατηρείται σε ισχύ ένας κανόνας που διέπει τη βέλτιστη διάταξη όταν είναι γνωστή η σύνθεση των παρτίδων που την αποτελούν. Όμως ο αλγόριθμος αυτός επειδή στηρίζεται σε μια υπόθεση που δεν ισχύει πάντα δεν δίνει βέλτιστη λύση σε όλες τις περιπτώσεις. Η ευστοχία του όμως σε προβλήματα όπου η μέση τιμή των χρόνων εξάρμωσης είναι πολύ μικρότερη ή πολύ μεγαλύτερη της μέσης τιμής των χρόνων επεξεργασίας, αποδείχτηκε πειραματικά. Επίσης χαρακτηριστική είναι η αποδοτικότητά του κατά την εκτέλεση, εφόσον σε σύνολο 500 πειραματικών προβλημάτων με μεγάλο πλήθος εργασιών και ομάδων τερμάτισε σε μέσο χρόνο 0.124 δευτερόλεπτα CPU.

Στην προσπάθειά μας για επίτευξη ποιοτικά καλύτερου αλγορίθμου ελέγξαμε έναν ήδη πολύ καλό προσεγγιστικό αλγόριθμο για το ίδιο πρόβλημα που προτάθηκε από τους Ahn και Hyun και διαπιστώσαμε ότι λειτουργεί καλά στις περιπτώσεις που ο δικός μας αποτυγχάνει. Έτσι αποφασίσαμε να φτιάξουμε ένα νέο προσεγγιστικό πολυωνυμικό αλγόριθμο με παράθεση των δυο προηγούμενων προσεγγιστικών αλγορίθμων. Ο νέος αυτός αλγόριθμος κατάφερε να λύσει βέλτιστα όλα τα προβλήματα δοκιμών και μάλιστα σε χρόνο μισό κατά μέσο όρο από αυτόν που χρειάζεται για τα ίδια προβλήματα ο αλγόριθμος των Ahn και Hyun.

Συγκριτικά με τον αλγόριθμο των Ahn και Hyun ο πρώτος προσεγγιστικός αλγόριθμος που κατασκευάσαμε με βάση τις παρτίδες, έδινε καλύτερες λύσεις και σε κατηγορίες προβλημάτων όπου οι τιμές των μεγεθών των διάφορων οντοτήτων ανήκαν σε μικρά διαστήματα τιμών. Κάτι τέτοιο μπορεί να εξηγηθεί από το γεγονός ότι για δεδομένα τέτοιας μορφής, ισότητες στην τιμή του κριτηρίου βελτίστου παρουσιάζονται συχνά σε επικείμενες μεταθέσεις οι οποίες αστόχως δεν πραγματοποιούνται. Κάτι τέτοιο δεν είναι αληθές όταν οι χρόνοι εξάρμωσης διαφέρουν μεταξύ τους οπότε ελαττώνεται και η συχνότητα με την οποία παρουσιάζονται ισότητες στην τιμή του κριτηρίου βελτίστου κατά τις μεταθέσεις.

Παράλληλα η σχετική με τις παρτίδες ανάλυση αποδείχτηκε ιδιαίτερα χρήσιμη στην τροποποίηση ενός αλγορίθμου δυναμικού προγραμματισμού, ο οποίος λαμβάνοντας υπόψη ιδιότητές τους καταφέρνει να λύσει προβλήματα ικανοποιητικού μεγέθους χωρίς να σταματήσει λόγω ανεπάρκειας διαθέσιμου αποθηκευτικού χώρου πριν φτάσει σε βέλτιστο. Τα προβλήματα αυτά περιλαμβάνουν μερικές εκατοντάδες εργασιών (100-300) που κατανέμονται σε (10-30) διαφορετικές ομάδες.

9.2. Μελλοντικές επεκτάσεις - βελτιώσεις

Η ουσιαστικότερη κατεύθυνση προς την οποία θα μπορούσαμε να κινηθούμε σε περαιτέρω ανάλυση του προβλήματος είναι η προσπάθεια περισσότερο συστηματικής (τυπικής) εξήγησης του γεγονότος που κάνει τον δεύτερο προσεγγιστικό αλγόριθμο να συμπεριφέρεται βέλτιστα.

Όμως εκτός αυτής της θεωρητικής κυρίως πιθανής διερεύνησης θα μπορούσαμε να ασχοληθούμε με βελτιώσεις των αλγορίθμων που έχουμε ήδη κατασκευάσει, ξεκινώντας από διαφοροποίηση του πρώτου προσεγγιστικού αλγορίθμου στην οποία μετά τη διάταξη των ΔΠ κατά φθίνουσα σειρά των ΧΛ τους, θα παραμένουν ΔΠ αυτές που γειτονεύουν με ετεροειδείς ΔΠ, ενώ οι άλλες οι οποίες γειτονεύουν με ομοειδείς ΔΠ θα διασπώνται στις ΠΠ που τις απαρτίζουν.

Από την άλλη θα μπορούσαμε να χρησιμοποιήσουμε στο δεύτερο προσεγγιστικό αλγόριθμο ως ολότητες τις ΔΠ που γειτονεύουν με ετεροειδείς παρτίδες κατά τη φάση διάσπασης παρτίδων που ακολουθεί το πρώτο βήμα του, και να ελέγξουμε την επίδραση μιας τέτοιας αλλαγής τόσο στην ποιότητα της λύσης (πιστεύουμε ότι αυτή δεν θα αλλάξει) όσο και στον χρόνο εκτέλεσης.

Η έρευνα προς τις παραπάνω δύο κατευθύνσεις θα διαφωτίσει την υπόθεσή μας, ότι η αστοχία του πρώτου προσεγγιστικού αλγορίθμου οφείλεται στην υπέρβαση των συνθηκών σχηματισμού βέλτιστων ΔΠ (θεώρημα 3.5.1).

Θα μπορούσαμε επίσης να ασχοληθούμε σε μεγαλύτερο βαθμό με περιπτώσεις, στις οποίες έχουμε ισότητα στα μεγέθη που καθορίζουν την εξέλιξη των αλγορίθμων μετά από ελέγχους. Για παράδειγμα, να επιτρέπουμε συγχωνεύσεις σε περιπτώσεις που δίνουν διατάξεις με τιμή του κριτηρίου βελτίστου ίση με αυτήν της διάταξης πριν τη συγχώνευση.

Ιδιαίτερα για τις συγχωνεύσεις, θα μπορούσαμε να αλλάξουμε και τον τρόπο με τον οποίο επιλέγουμε δυο γειτονικές παρτίδες που θα συγχωνευτούν, ώστε να οδηγούμαστε σε νέες διατάξεις με τη μεγαλύτερη δυνατή βελτίωση από συγχώνευση ομοειδών παρτίδων.

Ενέργειες όπως τα παραπάνω είναι δυνατόν να βελτιώσουν τη συμπεριφορά των υπαρχόντων αλγορίθμων. Από την άλλη όμως, καλό είναι να πρσπαθήσουμε να τους επεκτείνουμε ώστε να είναι χρήσιμοι για τη λύση κάποιου γενικότερου προβλήματος (πχ να λαμβάνονται υπόψη χρόνοι εξάρμωσης εξαρτημένοι ακολουθίας).

Τέλος, για περιπτώσεις προβλημάτων με πολύ μεγάλα μεγέθη θα μπορούσαμε να υιοθετήσουμε ένα διαλογικό τρόπο εκτέλεσης τους. Έτσι η εκτέλεση επαναλήψεων που είναι ιδιαίτερα χρονοβόρες θα μπορεί να ελέγχεται από το χρήστη του αλγορίθμου.

9.3. Εφαρμογή στη λύση συγγενών προβλημάτων

Η ισοδυναμία του $TCTS_j$ με το TRP (travelling repairman problem), η οποία αποδείχτηκε στο κεφάλαιο 3.2 δίνει την πρώτη περίπτωση διαφορετικού προβλήματος για τη λύση του οποίου μπορούν να χρησιμοποιηθούν οι αλγόριθμοι που κατασκευάστηκαν κατά τη διάρκεια της μελέτης μας.

Μια δεύτερη περίπτωση, στην οποία η λύση του προβλήματός αποδεικνύεται ιδιαίτερα χρήσιμη αποτελεί το πρόβλημα ελαχιστοποίησης του αθροίσματος των καθυστερήσεων n εργασιών παρουσία χρόνων εξάρμωσης ανεξάρτητων ακολουθίας. Το πρόβλημα αυτό ακόμα κι όταν δεν έχουμε χρόνους εξάρμωσης είναι NP-hard ([Em69], [PoVa-Wa85], [KaKy88]) και η λύση του κινείται στα πλαίσια της λύσης του $n\,|1|\;|T_{\max}$ και του $n\,|1|\;|\sum C_i$. Είναι δε ένας απλός ευρηματικός κανόνας, ο MDD, (Modified Due Date [Si]), ο οποίος συνδυάζει δυναμικά τις βέλτιστες λύσεις των παραπάνω προβλημάτων. Αντίστοιχα μπορεί να αντιμετωπιστεί η περίπτωση που έχουμε χρόνους εξάρμωσης, όμως τώρα θα χρησιμοποιούνται οι λύσεις του $TCTS_j$.

Οι αλγόριθμοι που κατασκευάστηκαν για τη λύση του $TCTS_j$, είτε είναι μεταθετικοί είτε κάνουν συγχωνεύσεις, μπορεί εύκολα να τροποποιηθούν ώστε να λαμβάνονται υπόψη και νέοι περιορισμοί (περιορισμοί διαδοχής, διαθεσιμότητας εργασιών σε διαφορετικές χρονικές στιγμές κλπ). Έτσι οι συγχωνεύσεις και μεταθέσεις θα επιτρέπονται μόνο για εργασίας που ικανοποιούν τους περιορισμούς διαδοχής ή διατίθενται τις ίδιες χρονικές στιγμές. Κατ' αυτό τον τρόπο θα μπορούσαμε ν' αντιμετωπίσουμε το πρόβλημα που απασχόλησε τους Dessouky και Deogun σε κάποια εργασία τους [DeDe81] καθώς και τη γενίκευση του που μελετάται από τους Hariri και Potts [HaPo83].

Τέλος, αποτελεσματικά θα μπορούσε να αντιμετωπιστούν προβλήματα, στα οποία εμφανίζεται χρόνος μεταγωγής και ταυτόχρονα ζητείται ελαχιστοποίηση του μέσου χρόνου εκτέλεσης, όπως αυτό που παρουσιάζει ο Sahney [Sa71] σε σχετική εργασία του.

Παραπομπές

[Ab-WaKa87]

ABDEL-WAHAB, H.M., and KAMEDA, T., ‘‘File Space Utilization in Database Conversion,’’ in *Comput. Opns Res.*, Vol. 14, No. 2, pp. 107-115, 1987.

[AhHy90]

AHN, B-H., and HYUN, J-H., ‘‘Single Facility Multi-Class Job Scheduling,’’ in *Comput. Opns Res.*, Vol. 17, No. 3, pp. 265-272, 1990.

[BaVa81]

BARNES, J.W., and VANSTON, L.K., ‘‘Scheduling Jobs with Linear Delay Penalties and Sequence Dependent Setup Costs,’’ in *Operations Res.*, Vol. 29, No. 1, pp. 146-160, January-February 1981.

[BrCoSe74]

BRUNO, J., COFFMAN, E.G.Jr., and SETHI, R., ‘‘Scheduling Independent Tasks to Reduce Mean Finishing Time,’’ in *Com. of the ACM*, Vol. 17, No. 7, pp. 382-387, July 1974.

[Bu76]

BURNS, R.N., ‘‘Scheduling to Minimize the Weighted Sum of Completion Times with Secondary Criteria,’’ in *Nav. Res. Log. Quart.*, Vol. 23, No. 1, pp. 125-129, 1976.

[DeDe81]

DESSOUKY, M.I., and DEOGUN, J.S., ‘‘Sequencing Jobs with Unequal Ready Times to Minimize Mean Flow Time,’’ in *SIAM J. Comput.*, Vol. 10, No. 1, pp. 192-202, February 1981.

[DoKaRu87]

DOBSON, G., KARMARKAR, U.S., and RUMMEL, J.L., ‘‘Batching to Minimize Flow Times on One Machine,’’ in *Management Sci.*, Vol. 33, No. 6, pp. 784-799, June

1987.

[Em75]

EMMONS, H., ''A Note on a Scheduling Problem with Dual Criteria,'' in *Nav. Res. Log. Quart.*, Vol. 22, No. 3, pp. 615-616, 1975.

[GaLe88]

GASCON, A., and LEACHMAN, R.C., ''A Dynamic Programming Solution to the Dynamic, Multi-Item, Single-Machine Scheduling Problem,'' in *Operations Res.*, Vol. 36, No. 1, pp. 50-56, January-February 1988.

[Gl68]

GLASSEY, C.R., ''Minimum Change-Over Scheduling of Several Products on One Machine,'' in *Operations Res.*, Vol. 16, pp. 342-352, 1968.

[HaPo83]

HARIRI, A.M.A., and POTTS, C.N., ''An Algorithm for Single Machine Sequencing with Release Dates to Minimize Total Weighted Completion Time,'' in *Discr. Applied Math.*, Vol. 5, pp. 99-109, 1983.

[KaKy88]

KAWAGUCHI, T., and KYAN, S., ''Deterministic Scheduling in Computer Systems: A Survey,'' in *J. of the Opns Res. Soc. of Japan*, Vol. 31, No. 2, pp. 190-216, June 1988.

[MoPo89]

MONMA, C.L., and POTTS, C.N., ''On the Complexity of Scheduling with Batch Setup Times,'' in *Operations Res.*, Vol. 37, No. 5, pp. 798-804, September-October 1989.

[Po85]

POSNER, M.E., ''Minimizing Weighted Completion Times with Deadlines,'' in *Operations Res.*, Vol. 33, No. 3, pp. 562-574, May-June 1985.

[Po88]

POSNER, M.E., ‘‘The Deadline Constrained Weighted Completion Time Problem : Analysis of a Heuristic,’’ in *Operations Res.*, Vol. 36, pp. 742-746, No. 5, September-October 1988.

[PoVa-Wa83]

POTTS, C.N., and VAN WASSENHOVE, L.N., ‘‘An Algorithm for Single Machine Sequencing with Deadlines to Minimize Total Weighted Completion Time,’’ in *Eur. J. Opns Res.*, Vol. 12, pp. 379-387, 1983.

[Ps80]

PARAFTIS, H.N., ‘‘A Dynamic Programming Approach for Sequencing Groups of Identical Jobs,’’ in *Operations Res.*, Vol. 28, No. 6, pp. 1347-1359, November-December 1980.

[Sa71]

SAHNEY, V.K., ‘‘Single-Server, Two-Machine Sequencing with Switching Time,’’ in *Operations Res.*, Vol. 19, pp. 24-36, 1971.

[Si]

SIDNEY, J.B., ‘‘An Extension of Moore’s Due Date Algorithm’’.

[Sm56]

SMITH, W.E., ‘‘Various Optimizers for Single Stage Production,’’ in *Nav. Res. Log. Quart.*, Vol. 3, pp. 59-66, 1956.

[Th89]

THURUTHICKARA, C.J., ‘‘Tradeoff Solutions in Single Machine Production Scheduling for Minimizing Flow Time and Maximum Penalty,’’ in *Computers Opns Res.*, Vol. 16, No. 5, pp. 471-479, 1989.

[Va-WaGe80]

VAN WASSENHOVE, L.N., and GELDERS, L.E., ''Solving a Bicriterion Scheduling Problem,'' in *Europ. J. of Operational Res.*, Vol. 4, pp. 42-48, 1980.

Άλλη Βιβλιογραφία

[As70]

ASHOUR, S., ''An Experimental Investigation and Comparative Evaluation of Flow-Shop Scheduling Techniques,'' in *Operations Res.*, Vol. 18, pp. 541-549, 1970.

[BaAh87]

BAGCHI, U., and AHMADI, R.H., ''An Improved Lower Bound for Minimizing Weighted Completion Times with Deadlines,'' *Technical Note* in *Operations Res.*, Vol. 35, No. 2, pp. 311-313, March-April 1987.

[BaRi84]

BAHL, H.C., and RITZMAN, L.P., ''An Integrated Model for Master Scheduling, Lot Sizing and Capacity Requirements Planning,'' in *J. Opl Res. Soc.*, Vol. 35, No. 5, pp. 389-399, 1984.

[Ba75]

BAKER, K.R., ''A Comparative Study of Flow-Shop Algorithms,'' in *Operations Res.*, Vol. 23, No. 1, pp. 62-73, January-February 1975.

[Ba84]

BAKER, K.R., ''Sequencing Rules and Due-Date Assignments in a Job-Shop,'' in *Management Sci.*, Vol. 30, No. 9, pp. 1093-1104, September 1984.

[BlLeRi-Ka83]

BLAZEWICZ, J., LENSTRA, J.K., and RINNOOY-KAN, A.H.G., ''Scheduling Subject to Resource Constraints: Classification and Complexity,'' in *Discr. Applied Math.*, Vol. 5, pp. 11-24, 1983.

[BrDo78]

BRUNO, J. and DOWNEY, P., ''Complexity of Task Sequencing with Deadlines, Set-up Times and Changeover Costs,'' in *SIAM J. Comput.*, Vol. 7, No. 4, pp. 393-

404, November 1978.

[Ch-OwLa88]

CHARLES-OWABA, O.E., and LAMBERT, B.K., ‘‘Sequence Dependent Machine Set-Up Times and Similarity of Parts: A Mathematical Model,’’ in *IIE Transactions*, Vol. 20, No. 1, pp. 12-21, March 1988.

[DaSa89]

DANIELS, R.L., and SARIN, R.K., ‘‘Single Machine Scheduling with Controllable Processing Times and Number of Jobs Tardy,’’ *Technical Note* in *Operations Res.*, Vol. 37, No. 6, pp. 981-984, November-December 1989.

[DiSe88]

DILEEPAN, P., and SEN, T., ‘‘Bicriterion Static Scheduling Research for a Single Machine,’’ in *OMEGA Int. J. of Mgmt Sci.*, Vol. 16, No. 1, pp. 53-59, 1988.

[DoHoRa74]

DORSEY, R.C., HODGSON, T.J., and RATLIFF, H.D., ‘‘A Production Scheduling Problem with Batch Processing,’’ *Technical Note* in *Operations Res.*, Vol. 22, pp. 1271-1279, 1974.

[Em69]

EMMONS, H., ‘‘One-Machine Sequencing to Minimize Certain Functions of Job Tardiness,’’ in *Operations Res.*, Vol. 19, pp. 701-715, 1969.

[FaSc87]

FAALAND, B., and SCHMITT, T., ‘‘Scheduling Tasks with Due Dates in a Fabrication/Assembly Process,’’ in *Operations Res.*, Vol. 35, No. 3, pp. 378-388, May-June 1987.

[Fi73]

FISHER, M.L., ‘‘Optimal Solution of Scheduling Problems Using Lagrange Multipliers: Part I,’’ in *Operations Res.*, Vol. 21, pp. 1114-1127, 1973.

[FiLaLeRi-Ka83]

FISHER, M.L., LAGEWEG, B.J., LENSTRA, J.K., and RINNOOY-KAN, A.H.G., ''Surrogate Duality Relaxation for Job Shop Scheduling,'' in *Discr. Applied Math.*, Vol. 5, pp. 65-75, 1983.

[GaTaWi88]

GAREY, M.R., TARJAN, R.E., and WILFONG, G.T., ''One-Processor Scheduling with Symmetric Earliness and Tardiness Penalties,'' in *Math. of Operations Res.*, Vol. 13, No. 2, May 1988.

[GiTh60]

GIFFLER, B., and THOMPSON, G.L., ''Algorithms for Solving Production Scheduling Problems,'' in *Operations Res.*, Vol. 8, pp. 487-503, 1960.

[GiWa64]

GIGLIO, R.J., and WAGNER, H.M., ''Approximate Solutions to the Three-Machine Scheduling Problem,'' in *Operations Res.*, Vol. 12, pp. 305-324, 1964.

[Gr81]

GRAVES, S.C., ''A Review of Production Scheduling,'' in *Operations Res.*, Vol. 29, No. 4, pp. 646-675, 1981.

[Gr68]

GREENBERG, H.H., ''A Branch-Bound Solution to the General Scheduling Problem,'' in *Operations Res.*, Vol. 16, pp. 353-361, 1968.

[Gu71]

GUPTA, J.N., ''An Improved Combinatorial Algorithm for the Flowshop- Scheduling Problem,'' *Technical Note* in *Operations Res.*, Vol. 19, pp. 1753-1758, 1971.

[HaPo91]

HALL, N.G., and POSNER, M.E., ''Earliness-Tardiness Scheduling Problems, I: Weighted Deviation of Completion Times about a Common Due-Date,'' in

Operations Res., Vol. 39, No. 5, pp. 836-846, September-October 1991.

[HaKuSe91]

HALL, N.G., KUBIAK, W., and SETHI, S.P., ''Earliness-Tardiness Scheduling Problems, II: Deviation of Completion Times about a Restrictive Common Due-Date,'' in *Operations Res.*, Vol. 39, No. 5, pp. 847-856, September-October 1991.

[Ho73]

HORN, W.A., ''Minimizing Average Flow Time with Parallel Machines,'' in *Operations Res.*, Vol. 21, pp. 846-847, 1973.

[HuKuRu87]

HU, T.C., KUO, Y.S., and RUSKEY, F., ''Some Optimum Algorithms for Scheduling Problems with Changeover Costs,'' in *Operations Res.*, Vol. 35, No. 1, pp. 94-99, January-February 1987.

[IgSch65]

IGNALL, E., and SCHRAGE, L., ''Application of the Branch and Bound Technique to Some Flow-Shop Scheduling Problems,'' in *Operations Res.*, Vol. 13, pp. 400-412, 1965.

[Ka80]

KAO, E.P.C., ''A Multiple Objective Decision Theoretic Approach to One-Machine Scheduling Problems,'' in *Comput. Opns Res.*, Vol. 7, pp. 251-259, 1980.

[LaLaPaRo78]

LAKSHMINARAYAN, S., LAKSHMANAN, R., PAPINEAU, R.L., and ROCHETTE, R., ''Optimal Single-Machine Scheduling with Earliness and Tardiness Penalties,'' in *Operations Res.*, Vol. 26, No. 6, pp. 1079-1082, November-December 1978.

[LaSe77]

LAM, S., and SETHI, R., ''Worst Case Analysis of Two Scheduling Algorithms,'' in *SIAM J. COMPUT.*, Vol. 6, No. 3, pp. 518-536, September 1977.

[LiBu88]

LIU, C.Y., and BULFIN, R.L., ‘‘Scheduling Open Shops with Unit Execution Times to Minimize Functions of Due Dates,’’ in *Operations Res.*, Vol. 36, No. 4, pp. 553-558, July-August 1988.

[LoMu72]

LOCKETT, A.G., and MUHLEMANN, A.P., ‘‘A Scheduling Problem Involving Sequence Dependent Changeover Times,’’ *Technical Note* in *Operations Res.*, Vol. 20, pp. 895-902, 1972.

[MaSo86]

MARCOTTE, O., and SOLAND, R.M., ‘‘An Interactive Branch-and-Bound Algorithm for Multiple Criteria Optimization,’’ in *Management Sci.*, Vol. 32, No. 1, pp. 61-75, January 1986.

[Ma88]

MATSUO, H., ‘‘The Weighted Total Tardiness Problem with Fixed Shipping Times and Overtime Utilization,’’ in *Operations Res.*, Vol. 36, No. 2, pp. 293-307, March-April 1988.

[McMaBu67]

McMAHON, G.B., and BURTON, P.G., ‘‘Flow-Shop Scheduling with the Branch-and-Bound Method,’’ in *Operations Res.*, Vol. 15, pp. 473-481, 1967.

[Mo80]

MONMA, C.L., ‘‘Sequencing to Minimize the Maximum Job Cost,’’ in *Operations Res.*, Vol. 26, No. 4, pp. 942-951, July-August 1980.

[NeSaDa86]

NELSON, R.T., SARIN, R.K., and DANIELS, R.L., ‘‘Scheduling with Multiple Performance Measures : The One Machine Case,’’ in *Management Sci.*, Vol. 32, No. 4, pp. 464-479, April 1986.

[PaIs77]

PANWALKAR, S.S., and IKANDER, W., ''A Survey of Scheduling Rules,'' in *Operations Res.*, Vol. 25, No. 1, pp. 45-61, January-February 1977.

[PoVa-Wa85]

POTTS, C.N., and VAN WASSENHOVE, L.N., ''A Branch and Bound Algorithm for the Total Weighted Tardiness Problem,'' in *Operations Res.*, Vol. 33, No. 3, pp. 363-377, March-April 1985.

[PoVa-Wa88]

POTTS, C.N., and VAN WASSENHOVE, L.N., ''Algorithms for Scheduling a Single Machine to Minimize the Weighted Number of Late Jobs,'' in *Management Sci.*, Vol. 34, No. 7, pp. 843-858, July 1988.

[Ra87]

RACHAMADUGU, R.M.V., ''A Note on the Weighted Tardiness Problem,'' in *Operations Res.*, Vol. 35, No. 3, pp. 450-452, May-June 1987.

[Ri-KaLaLe75]

RINNOOY-KAN, A.H.G., LAGEWEG, B.J., and LENSTRA, J.K., ''Minimizing Total Costs in One-Machine Scheduling,'' in *Operations Res.*, Vol. 23, No. 5, pp. 908-927, September-October 1975.

[Sa76]

SAHNI, S.K., ''Algorithms for Scheduling Independent Tasks,'' in *J. of the ACM*, Vol. 23, No. 1, pp. 116-127, January 1976.

[SeRaDi88]

SEN, T., RAISZADEH, F.M.E, and DILEEPAN, P., ''A Branch-and-Bound Approach to the Bicriterion Scheduling Problem Involving Total Flowtime and Range of Lateness,'' *Note* in *Management Sci.*, Vol. 34, No. 2, pp. 254-259, February 1988.

[ShBu82]

SHANTHIKUMAR, J.G., and BUZACOTT, J.A., ‘‘On the Use of Decomposition Approaches in a Single Machine Scheduling Problem,’’ in *J. of the Opns Res. Soc. of Japan*, Vol. 25, No. 1, pp. 29-47, March 1982.

[Si75]

SIDNEY, J.B., ‘‘Decomposition Algorithms for Single-Machine Sequencing with Precedence Relations and Deferral Costs,’’ in *Operations Res.*, Vol. 23, No. 2, pp. 283-298, March-April 1975.

[Si77]

SIDNEY, J.B., ‘‘Optimal Single-Machine Scheduling with Earliness and Tardiness Penalties,’’ in *Operations Res.*, Vol. 25, No. 1, pp.62-69, January-February 1977.

[Si]

SIDNEY, J.B., ‘‘An Extension of Moore’s Due Date Algorithm’’.

[SiSt86]

SIDNEY, J.B., and STEINER, G., ‘‘Optimal Sequencing by Modular Decomposition: Polynomial Algorithms,’’ in *Operations Res.*, Vol. 34, No. 4, pp. 606-612, July-August 1986.

[SmDu67]

SMITH, R.D., and DUDEK, R.A., ‘‘A General Algorithm for Solution of the n-Job, M-Machine Sequencing Problem of the Flow Shop,’’ in *Operations Res.*, Vol. 15, pp. 71-82, 1967.

[UsSm75]

USKUP, E., and SMITH, S.B., ‘‘A Branch-and-Bound Algorithm for Two-Stage Production-Sequencing Problems,’’ in *Operations Res.*, Vol. 23, No. 1, pp. 118-136, January-February 1975.

[Vi80]

VICKSON, R.G., ‘‘Choosing the Job Sequence and Processing Times to Minimize Total Processing Plus Flow Cost on a Single Machine,’’ in *Operations Res.*, Vol. 28, No. 5, pp. 1155-1167, September-October 1980.