

Εξομάλυνση της Χρήσης Πόρων σε Έργα με Υπερευρετικούς Αλγόριθμους

Κ. Π. ΑΝΑΓΝΩΣΤΟΠΟΥΛΟΣ
Αναπληρωτής Καθηγητής Δ.Π.Θ.

Γ.Κ. ΚΟΥΛΙΝΑΣ
Υποψήφιος Διδάκτορας Δ.Π.Θ.

Περίληψη

Η εξομάλυνση της χρήσης των πόρων συγκαταλέγεται στις κορυφαίες προκλήσεις που καλούνται να αντιμετωπίσουν οι διαχειριστές έργων, διότι οι αυξομειώσεις στην απασχόληση ενός πόρου κατά κανόνα αυξάνουν το κόστος, πλήττουν την ποιότητα του παραδοτέου, και υπονομεύουν την έγκαιρη παράδοση του έργου. Η εργασία επικεντρώνεται στην ανάπτυξη δύο υπερευρετικών αλγορίθμων, οι οποίοι αντιμετωπίζουν το πρόβλημα της εξομάλυνσης πόρων σε χρονοδιαγράμματα έργων. Μια υπερευρετική είναι ένας αλγόριθμος ο οποίος δεν βελτιώνει μια λύση απευθείας, αλλά διαχειρίζεται την επιλογή ευρετικών για τη βελτίωση της τρέχουσας λύσης? «μια ευρετική που επιλέγει ευρετικές». Αυτές οι λεγόμενες «χαμηλού επιπέδου» ευρετικές αναλαμβάνουν την αναζήτηση στον χώρο των λύσεων. Ο πρώτος από τους δύο αλγόριθμους, επιλέγει χαμηλού επιπέδου ευρετικές, με βάση την απόδοσή τους όσον αφορά τη βελτίωση της αντικειμενικής συνάρτησης. Ο δεύτερος είναι ένας πολλαπλής εκκίνησης αλγόριθμος, ο οποίος λειτουργεί σαν «λαιμάργος» στο πεδίο των ευρετικών χαμηλού επιπέδου. Η κωδικοποίηση έγινε με χρήση της γλώσσας VBA στο περιβάλλον του εμπορικού πακέτου διαχείρισης έργων Microsoft Project. Η αποτελεσματικότητα των αλγορίθμων αντιπαραβάλλεται με την ενσωματωμένη διαδικασία εξομάλυνσης των πόρων στο Microsoft Project, σε ένα τυχαίο παράδειγμα έργου.

1. ΕΙΣΑΓΩΓΗ

Μια από τις μεγαλύτερες προκλήσεις της Διαχείρισης Έργων είναι η κατανομή πόρων με τέτοιο τρόπο ώστε να υπάρχει όσο το δυνατόν μικρότερη ανάγκη για «πρόσληψη» νέων πόρων ή/και μικρότερη απασχόληση σε ήδη υπάρχοντες. Η επιδίωξη αυτή προκύπτει από τη διαπίστωση ότι ανομοιομορφη χρήση των πόρων σε ένα έργο επιβαρύνει τον προϋπολογισμό του λόγω αναγκαίων μετακινήσεων προσωπικού και εξοπλισμού, αλλά και ενδεχόμενης ανάγκης για πρόσληψη πρόσθετου προσωπικού [7, 12]. Τα προηγούμενα ισχύουν υποθέτοντας ότι υπάρχει επαρκής διαθεσιμότητα, ώστε να πραγματοποιηθεί η πρόσληψη νέων πόρων. Δεδομένου, όμως, ότι στην πράξη δεν είναι βέβαιο ότι εργαζόμενοι κι εξοπλισμός θα είναι διαθέσιμοι τη στιγμή που απαιτούνται, η ανομοιομορφη ζήτηση για πόρους απειλεί γενικά την έγκαιρη ολοκλήρωση του έργου.

Υποβλήθηκε: 3.7.2008 Έγινε δεκτή: 17.9.2008

Οι διαδικασίες που επιδιώκουν να εξομαλύνουν τη χρήση των πόρων επιχειρούν να τροποποιήσουν την απασχόληση συγκεκριμένων πόρων, ούτως ώστε να υπάρχουν όσο το δυνατόν μικρότερες διακυμάνσεις, ενώ παράλληλα η διάρκεια του έργου να κυμαίνεται σε ανεκτά όρια. Ένα πιο ομοιόμορφο ιστόγραμμα χρήσης των πόρων συμβάλει αποφασιστικά στην ελαχιστοποίηση του κόστους και των κινδύνων, καθώς και στον σχεδιασμό μιας πιο αποτελεσματικής εφοδιαστικής αλυσίδας για το έργο.

Στην παρούσα εργασία παρουσιάζονται δύο υπερευρετικοί αλγόριθμοι για την επίλυση του προβλήματος της εξομάλυνσης της χρήσης ενός πόρου σε χρονοδιάγραμμα έργου. Οι υπερευρετικοί αλγόριθμοι αποτελούν μια νέα οικογένεια αλγορίθμων, οι οποίοι έχουν εφαρμοστεί με επιτυχία σε διάφορα προβλήματα βελτιστοποίησης [2, 3, 5]. Οι υπερευρετικοί διαφοροποιούνται από τις γνωστές υπολογιστικές τεχνικές, καθώς δεν ενεργούν ευθέως στον χώρο των εφικτών λύσεων του προβλήματος, αλλά διαχειρίζονται την εφαρμογή των λεγόμενων «χαμηλού επιπέδου» (low level) ευρετικών.

Επιπλέον, επειδή δεν χρησιμοποιούν γνώση σχετική με τη φύση του προβλήματος, είναι σχετικά ανεξάρτητοι από συγκεκριμένα προβλήματα.

Ο πρώτος αλγόριθμος (HH1) χρησιμοποιεί μια από ένα σύνολο διαθέσιμων ευρετικών χαμηλού επιπέδου, πραγματοποιώντας περιοδικά αξιολόγηση της επίδοσης κάθε ευρετικής όσο προχωρά η υπολογιστική διαδικασία. Ο δεύτερος (HH2) είναι ένας πολλαπλής εκκίνησης αλγόριθμος, ο οποίος λειτουργεί σαν «λαιμάργος» (greedy) στο πεδίο των ευρετικών «χαμηλού» επιπέδου. Εξ όσων γνωρίζουν οι συγγραφείς, πρόκειται για την πρώτη εφαρμογή υπερευρετικού αλγόριθμου σε πρόβλημα χρονοπρογραμματισμού έργων.

1.1. Βιβλιογραφική ανασκόπηση

Η εξομάλυνση πόρων στη διαχείριση έργων έχει απασχολήσει κατά το παρελθόν πολλούς ερευνητές. Για την επίλυση του προβλήματος αυτού έχουν αναπτυχθεί διαδικασίες μαθηματικού προγραμματισμού από τους Bandelloni et

al. [1] και τον S.M. Easa [6], οι οποίες μπορούν να εγγυηθούν την εύρεση της βέλτιστης λύσης. Ωστόσο, σημαντικό μειονέκτημα των μεθόδων αυτών είναι ότι δεν μπορούν να αντιμετωπίσουν μεγάλο μεγέθους προβλήματα, καθώς ο απαιτούμενος υπολογιστικός χρόνος αυξάνεται ταχύτατα, συναρτήσει του αριθμού των δραστηριοτήτων του έργου.

Προκειμένου να αντιμετωπιστεί το ζήτημα του υπολογιστικού χρόνου, έχουν προταθεί αρκετοί ευρετικοί αλγόριθμοι, οι οποίοι εξομαλύνουν τη χρήση των πόρων σε έργα με μεγάλο πλήθος δραστηριοτήτων, παρέχοντας μια λύση κοντά στη βέλτιστη σε αποδεκτό χρόνο [7, 8]. Σχετικά πρόσφατα, έχουν αναπτυχθεί επίσης μεταευρετικοί αλγόριθμοι, όπως γενετικοί αλγόριθμοι [9, 13] και αλγόριθμοι προσομοιωμένης απόπτωσης [11]. Η φιλοσοφία των προσεγγίσεων αυτών συνίσταται στην κατασκευή ενός αρχικού χρονοπρογράμματος και στη συνέχεια μετατόπιση της έναρξης ορισμένων δραστηριοτήτων, σύμφωνα με τον χρησιμοποιούμενο ευρετικό κανόνα.

2. ΤΟ ΠΡΟΒΛΗΜΑ ΤΗΣ ΕΞΟΜΑΛΥΝΣΗΣ ΠΟΡΩΝ

Έστω ένα δίκτυο έργου AON (δραστηριότητες στους κόμβους) το οποίο αποτελείται από n δραστηριότητες (κόμβους) και το σύνολο A των τόξων (προτεραιοτήτων). Συμβολίζουμε με A_j όλες τις δραστηριότητες οι οποίες είναι προαπαιτούμενες της j και με f_j την έναρξη της δραστηριότητας j .

Η έναρξη f_j και η λήξη f_n του έργου απεικονίζονται με δραστηριότητες ορόσημα (milestones). Έστω δ_n η ανεκτή χρονική διάρκεια ολοκλήρωσης του έργου. Το δ_n , το οποίο μπορεί να διαφέρει από τη διάρκεια του έργου f_n που προκύπτει από τον αρχικό προγραμματισμό, καθορίζεται από τον διαχειριστή του έργου.

Η παρούσα εργασία εστιάζεται στην εξομάλυνση ενός πόρου για το έργο. Υποθέτουμε ότι η απαιτούμενη ποσότητα του πόρου από μια δραστηριότητα είναι σταθερή ανά χρονική περίοδο διάρκειάς της και ότι δεν υπάρχει περιορισμός στη διαθέσιμη ποσότητα του πόρου. Στόχος της βελτιστοποίησης είναι οι απαιτήσεις ανά χρονική μονάδα για τον πόρο να μην παρουσιάζουν μεγάλες διακυμάνσεις.

Με u_t συμβολίζεται το άθροισμα των ποσοτήτων του πόρου που απαιτείται από όλες τις δραστηριότητες κατά τη χρονική περίοδο t του έργου. Το ζητούμενο είναι να μετακινηθούν οι δραστηριότητες, ούτως ώστε να ελαχιστοποιηθεί η επόμενη ποσότητα

$$z = \sum_{t=1}^D (u_t - \bar{u}_t)^2$$

όπου \bar{u}_t μια μεταβαλλόμενη ποσότητα που καθορίζεται από τον διαχειριστή του έργου για κάθε χρονική περίοδο και D

= f_n η συνολική διάρκεια του έργου. Εάν, δίχως απώλεια της γενικότητας, θεωρήσουμε ομοιόμορφη κατανομή του πόρου, θα έχουμε

$$\bar{u}_t = \bar{u} = \left(\sum_{i=1}^D u_i \right) / D$$

οπότε η σχέση $z = \sum_{t=1}^D (u_t - \bar{u}_t)^2$ μπορεί να γραφεί

$$\begin{aligned} z &= \sum_{t=1}^D (u_t^2 - 2u_t \times \bar{u} + \bar{u}^2) \\ &= \sum_{t=1}^D u_t^2 - 2\bar{u} \times \sum_{t=1}^D u_t + \sum_{t=1}^D \bar{u}^2 \end{aligned}$$

Επειδή τα \bar{u} και $\sum_{t=1}^D u_t$ είναι σταθερά, η ελαχιστοποίηση του z ισοδυναμεί με ελαχιστοποίηση της επόμενης ποσότητας

$$\sum_{t=1}^D u_t^2$$

Το προς επίλυση πρόβλημα μπορεί να διατυπωθεί ως εξής:

$$\min z = \sum_{t=1}^D u_t^2 \quad 1$$

υπό τους περιορισμούς

$$f_i \leq f_j - d_j \quad \text{για όλα τα } i \in A_j, \text{ για όλα τα } j \in A \quad 2$$

$$f_i = 0 \quad 3$$

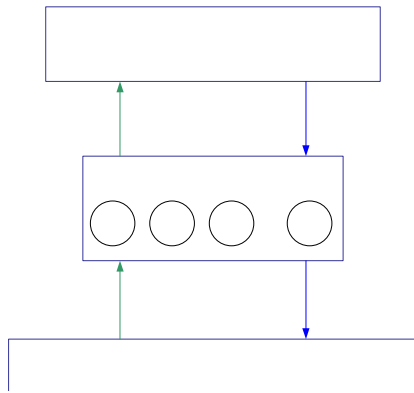
$$f_n \leq \delta_n \quad 4$$

Η βελτιστοποίηση του προβλήματος γίνεται με την υπόθεση ότι θα τηρούνται οι σχέσεις αλληλεξάρτησης μεταξύ των δραστηριοτήτων (Σχέση 2) και ότι η χρονική διάρκεια του έργου δεν θα ξεπεράσει την τιμή της ανεκτής χρονικής διάρκειας δ_n (Σχέση 4). Ως έναρξη του έργου θεωρείται η χρονική στιγμή μηδέν (Σχέση 3).

3. ΠΕΡΙΓΡΑΦΗ ΤΩΝ ΑΛΓΟΡΙΘΜΩΝ

3.1. Οι υπερευρετικοί αλγόριθμοι

Πρόσφατα μία νέα οικογένεια αλγορίθμων, οι υπερευρετικοί, έχουν χρησιμοποιηθεί για την αντιμετώπιση «δύσκολων» υπολογιστικά προβλημάτων. Αντίθετα από τους ευρετικούς ή μεταευρετικούς αλγόριθμους, οι οποίοι ενεργούν απευθείας στον χώρο των λύσεων, οι αλγόριθμοι αυτοί επιλέγουν ευρετικές που θα εφαρμοστούν στον χώρο των λύσεων.



Σχήμα 1: Το πλαίσιο λειτουργίας των υπερερευρητικών αλγορίθμων.
Figure 1: The hyperheuristic framework.

Όπως φαίνεται στο Σχήμα 1, ένας υπερερευρητικός αλγόριθμος ενεργεί σε ανώτερο επίπεδο από αυτό των λύσεων, επιλέγοντας άλλες, απλές σε λειτουργία, ευρετικές κατώτερου επιπέδου, οι οποίες ενεργούν στον χώρο των λύσεων προκειμένου να βελτιώσουν τοπικά την τρέχουσα λύση του προβλήματος.

Το σημαντικό πλεονέκτημα ενός υπερερευρητικού αλγορίθμου είναι το γεγονός ότι εκ κατασκευής μπορεί να χρησιμοποιηθεί απευθείας σε διαφορετικά προβλήματα με ελάχιστες τροποποιήσεις της αντικειμενικής συνάρτησης, καθώς και των χαμηλού επιπέδου ευρετικών.

Η πρόσφατη έρευνα επικεντρώνεται στην ανάπτυξη υπερερευρητικών αλγορίθμων για συγκεκριμένα προβλήματα, οι οποίοι θα βασίζονται σε μεταερευρητικές τεχνικές. Οι Cowling et al. [4] χρησιμοποίησαν έναν γενετικό αλγόριθμο στη θέση του υπερερευρητικού, ενώ έχουν προταθεί επίσης αλγόριθμοι ταμπού αναζήτησης [3], και προσομοιωμένης απόκτησης [5].

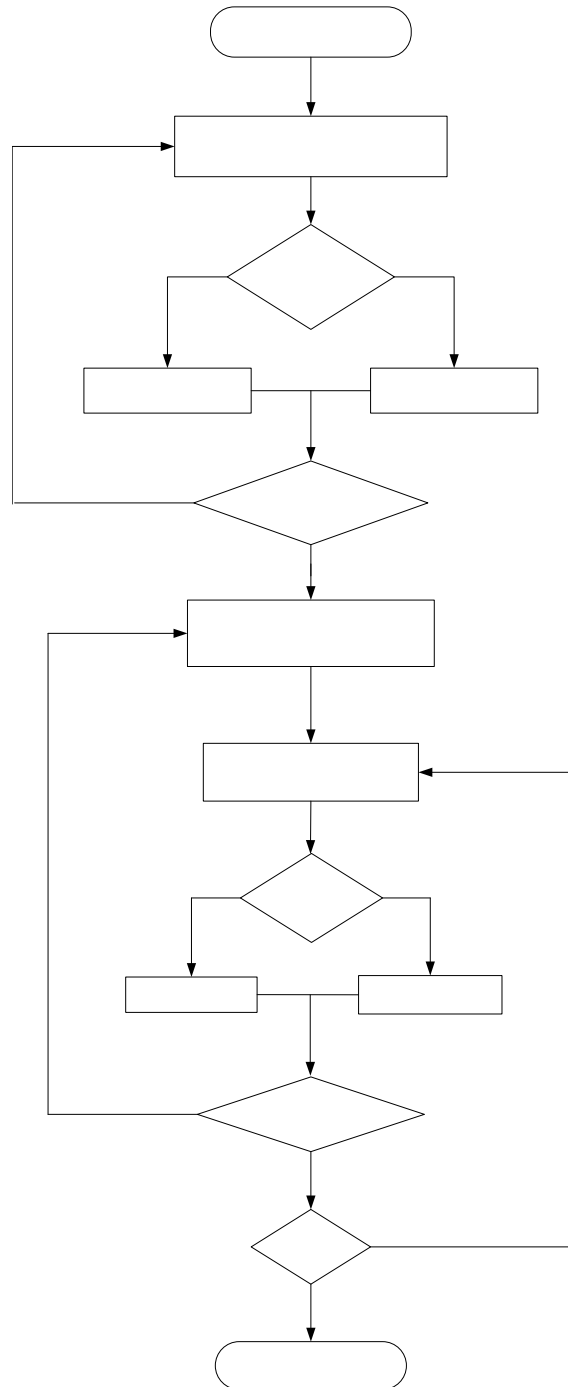
Οι υπερερευρητικοί αλγόριθμοι έχουν εφαρμοστεί με επιτυχία σε προβλήματα προσδιορισμού ωρολόγιου προγράμματος (timetabling) και μεγέθους φορτίων (shipper sizes) [3, 5].

Αναλυτική παρουσίαση των υπερερευρητικών αλγορίθμων περιλαμβάνεται στο [2].

3.2. Περιγραφή των προτεινόμενων αλγορίθμων

Για την επίλυση του προβλήματος της εξομάλυνσης της χρήσης των πόρων, το οποίο ορίζεται από τις Σχέσεις (1)-(4) προτείνονται οι δύο υπερερευρητικοί αλγόριθμοι HH1 και HH2.

Σημειώνεται πως αμφότεροι οι αλγόριθμοι εφαρμόζονται αποκλειστικά στο περιβάλλον του MS-Project, δεδομένου ότι οι χαμηλού επιπέδου ευρετικές ορίζονται με βάση τα επίπεδα προτεραιότητας για τις δραστηριότητες του έργου που παρέχει το συγκεκριμένο λογισμικό.



Σχήμα 2: Ο υπερερευρητικός αλγόριθμος HH1.
Figure 2: The hyperheuristic HH1.

3.2.1. Ο αλγόριθμος HH1

Το διάγραμμα ροής του αλγορίθμου HH1 φαίνεται στο Σχήμα 2. Στην υπολογιστική διαδικασία διευκρινίζονται τα εξής:

1. Κατασκευάζεται το αρχικό χρονοπρόγραμμα και υπολογίζεται η χρήση του πόρου (η τιμή της αντικειμενικής συνάρτησης). Ορίζεται ο μέγιστος αριθμός κλήσεων των ευρετικών σε κάθε κύκλο (max_AP_KΛ), στο τέλος του οποίου περιοδικά θα υπολογίζεται η απόδοση κάθε ευρετικής χαμηλού επιπέδου.
2. Σε κάθε κύκλο επιλέγεται τυχαία μια ευρετική χαμηλού επιπέδου και εφαρμόζεται άπαξ στον χώρο των λύσεων. Αν βελτιώνεται η αντικειμενική συνάρτηση, η νέα λύση γίνεται αποδεκτή. Ειδικά, απορρίπτεται. Στον πρώτο κύκλο όλες οι πιθανότητες κλήσης των ευρετικών χαμηλού επιπέδου είναι ίσες μεταξύ τους.
3. Όταν ολοκληρωθεί ένας κύκλος, υπολογίζεται η απόδοση κάθε ευρετικής χαμηλού επιπέδου (συνολική βελτίωση που επέφερε στην αντικειμενική συνάρτηση δια του αριθμού των κλήσεων της), και με κανονικοποίηση η πιθανότητα με την οποία θα επιλεγεί στον επόμενο κύκλο.
4. Η υπολογιστική διαδικασία ολοκληρώνεται όταν επαληθεύεται η προκαθορισμένη συνθήκη τερματισμού, η οποία στην παρούσα εργασία αφορά έναν μέγιστο αριθμό κλήσεων ευρετικών χαμηλού επιπέδου.

3.2.2. Ο αλγόριθμος ΗΗ2

Ο δεύτερος υπερειρευτικός αλγόριθμος αποτελείται από τα παρακάτω βήματα:

Βήμα 1 Κατασκευάζεται το αρχικό χρονοπρόγραμμα, υπολογίζεται η τιμή της αντικειμενικής συνάρτησης. Αυτή είναι η BELT_ΛΥΣΗ_ΑΡΧ.

$$\text{BELT_ΛΥΣΗ_ΤΡΕΧ} = \text{BELT_ΛΥΣΗ_ΑΡΧ}$$

Επιλέγεται τυχαία μια ευρετική χαμηλού επιπέδου, η οποία εφαρμόζεται στη BELT_ΛΥΣΗ_ΤΡΕΧ επανειλημμένως ενόσω τη βελτιώνει.

Η ευρετική διαγράφεται από τη λίστα των διαθέσιμων ευρετικών χαμηλού επιπέδου.

Βήμα 2 Επιλέγεται τυχαία μια ευρετική χαμηλού επιπέδου, η οποία εφαρμόζεται στη BELT_ΛΥΣΗ_ΤΡΕΧ επανειλημμένως ενόσω τη βελτιώνει. Αν δεν πρόέκυψε βελτίωση, η ευρετική εφαρμόζεται στη BELT_ΛΥΣΗ_ΑΡΧ, και αν η λύση που προκύπτει είναι καλύτερη από την BELT_ΛΥΣΗ_ΤΡΕΧ την αντικαθιστά.

Η ευρετική διαγράφεται από τη λίστα των διαθέσιμων ευρετικών χαμηλού επιπέδου.

Βήμα 3 Εάν έχουν χρησιμοποιηθεί όλες οι ευρετικές χαμηλού επιπέδου, τέλος. Μετάβαση στο Βήμα 2.

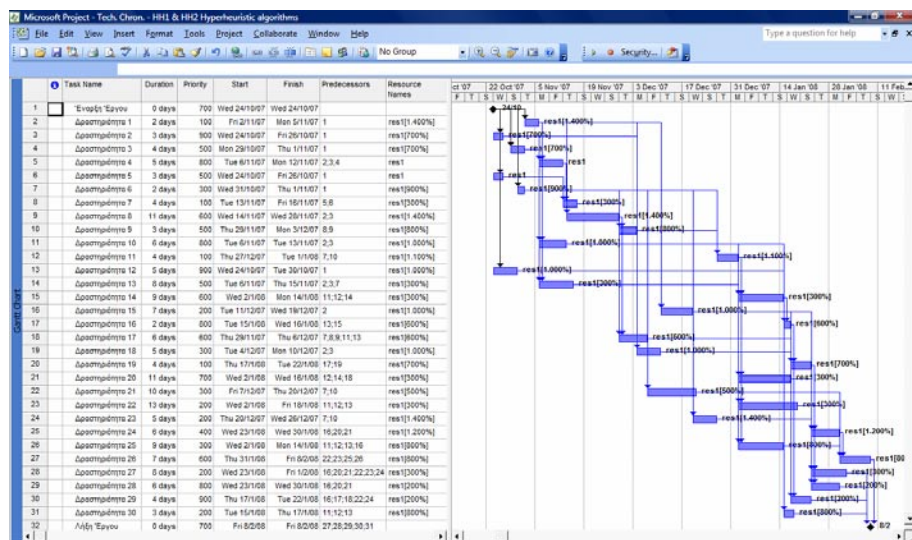
3.3. Ευρετικές χαμηλού επιπέδου

Τα περισσότερα εμπορικά πακέτα διαχείρισης έργων παρέχουν τη δυνατότητα στους χρήστες να καθορίζουν επίπεδα προτεραιότητας για κάθε μία από τις δραστηριότητες

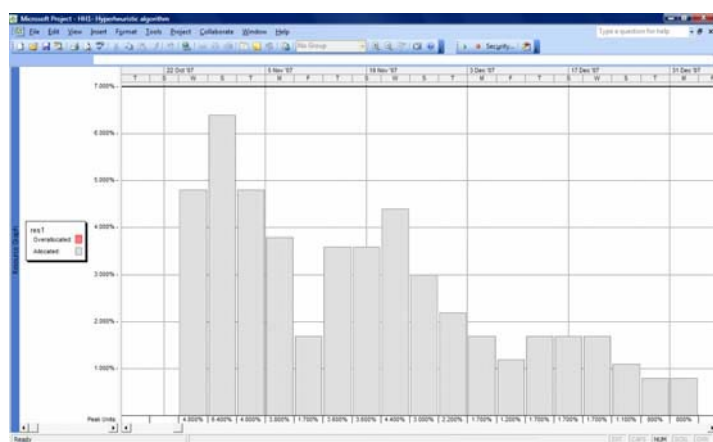
του έργου. Το MS-Project δίνει επίσης τη δυνατότητα να οριστεί επίπεδο προτεραιότητας από μηδέν έως 1000 (πεδίο "Priority"). Πάντως, τα κριτήρια που χρησιμοποιεί ο διαχειριστής του έργου για να αποφασίσει σε ποιο επίπεδο προτεραιότητας θα αντιστοιχίσει κάθε δραστηριότητα του χρονοπρογράμματος είναι σε μεγάλο βαθμό υποκειμενικά.

Οι προτεινόμενες ευρετικές χαμηλού επιπέδου συμβαδίζουν με τη φιλοσοφία της απλότητας που διέπει τους υπερειρευτικούς αλγόριθμους, καθώς η λειτουργία τους έγκειται στην προσθήκη, διαγραφή και αντικατάσταση προτεραιοτήτων. Στην παρούσα εργασία τα χρησιμοποιούμενα επίπεδα προτεραιότητας είναι εννέα (100 - 900). Οι ευρετικές χαμηλού επιπέδου που αναπτύχθηκαν μπορούν να διακριθούν σε δύο κατηγορίες. Πρώτον, όσες επιλέγουν τυχαία δραστηριότητες των οποίων τις προτεραιότητες θα τροποποιήσουν:

1. Ανταλλαγή δύο προτεραιοτήτων (L1): Ο αλγόριθμος επιλέγει τυχαία δύο δραστηριότητες, και ανταλλάσσει τις προτεραιότητές τους.
2. Αντικατάσταση προτεραιότητας (L2): Επιλέγεται τυχαία μία δραστηριότητα και η προτεραιότητά της αντικαθίσταται τυχαία με μια διαφορετική προτεραιότητα.
3. Αντικατάσταση δύο προτεραιοτήτων (L3): Παρόμοια με την L2, αλλά αντικαθιστά ταυτόχρονα τις προτεραιότητες δύο τυχαία επιλεγμένων δραστηριοτήτων.
4. Αντικατάσταση μίας προτεραιότητας και ανταλλαγή δύο άλλων (L4): Η ευρετική αυτή είναι συνδυασμός των L1 και L2. Επιλέγονται τρεις τυχαίες δραστηριότητες. Αντικαθίσταται η προτεραιότητα της πρώτης με μία τυχαία, ενώ ανταλλάσσονται οι προτεραιότητες των δύο άλλων δραστηριοτήτων.
Δεύτερον, ευρετικές οι οποίες επιλέγουν την προτεραιότητα που θα τροποποιηθεί, βάσει κάποιου κριτηρίου:
5. Αντικατάσταση προτεραιότητας (L5): Ο αλγόριθμος εντοπίζει τη δραστηριότητα i με το μεγαλύτερο τρέχον Συνολικό Περιθώριο (ΣΠ) και αντικαθιστά την προτεραιότητά της με την προτεραιότητα μιας τυχαία επιλεγμένης δραστηριότητας j . Η προτεραιότητα της j πρέπει να διαφέρει από την προτεραιότητα της i . Αν δύο ή περισσότερες δραστηριότητες έχουν ίδιο μέγιστο ΣΠ, επιλέγεται η δραστηριότητα με το μεγαλύτερο ID. Είναι κατανοητό ότι κάθε φορά που καλείται η ευρετική μία διαφορετική δραστηριότητα επιλέγεται, καθώς διαφοροποιείται το ΣΠ των δραστηριοτήτων.
6. Αντικατάσταση δύο προτεραιοτήτων (L6): Επιλέγονται οι δραστηριότητες με το μεγαλύτερο και μικρότερο ΣΠ. Σε περίπτωση ισοδυναμίων, επιλέγεται η δραστηριότητα με το μεγαλύτερο ID. Η προτεραιότητα των δύο δραστηριοτήτων αντικαθίσταται αντιστοίχως με "100" και "900", σύμφωνα με τα χρησιμοποιούμενα επίπεδα.
7. Ανταλλαγή δύο προτεραιοτήτων (L7): Εντοπίζονται οι δραστηριότητες με το μεγαλύτερο και μικρότερο ΣΠ, και ανταλλάσσονται οι προτεραιότητές τους.



Εικόνα 1: Τα δεδομένα του παραδείγματος.
 Picture 1: Initial schedule for the example project.



Εικόνα 2: Το ιστόγραμμα κατανομής πόρου στο αρχικό χρονοπρόγραμμα.
 Picture 2: Initial resource usage histogram.

4. ΠΑΡΑΔΕΙΓΜΑ ΕΦΑΡΜΟΓΗΣ ΤΩΝ ΑΛΓΟΡΙΘΜΩΝ

Οι προτεινόμενοι αλγόριθμοι εφαρμόστηκαν για την εξομάλυνση της χρήσης ενός πόρου σε ένα έργο αποτελούμενο από 30 δραστηριότητες.

Τα αποτελέσματα συγκρίνονται με αυτά της εξομάλυνσης του ίδιου δικτύου με χρήση της ενσωματωμένης διαδικασίας στο Ms-Project. Οι αλγόριθμοι προγραμματίστηκαν με τη γλώσσα προγραμματισμού Visual Basic for Applications (VBA).

Τα δεδομένα του παραδείγματος συμπεριλαμβανομένης της διάρκειας, των προαπαιτούμενων και της απαιτούμενης ημερήσιας ζήτησης, κάθε δραστηριότητας, για πόρο διακρίνονται στην Εικόνα 1. Το αρχικό πρόγραμμα έχει διάρκεια 49 ημέρες, αντικειμενική συνάρτηση ίση με 37688 και εύρος χρήσης του πόρου 800% - 6400%. Τα αποτελέσματα από

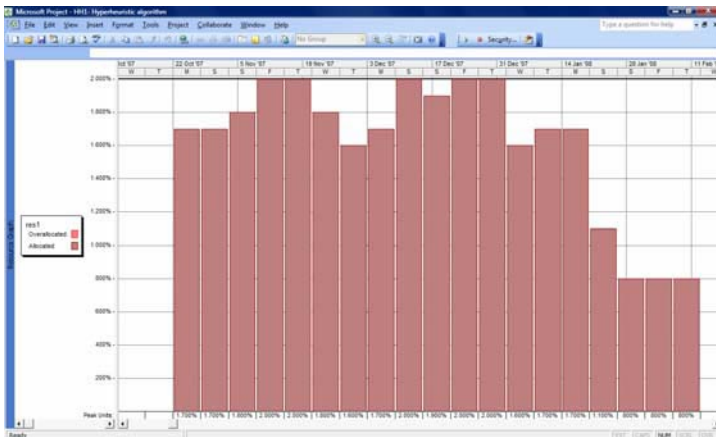
την εφαρμογή των δύο υπερειρευτικών αλγορίθμων στο δίκτυο παρατίθενται στον Πίνακα 1. Επίσης, για λόγους σύγκρισης, εμφανίζονται τα αποτελέσματα που προκύπτουν από την εφαρμογή του ενσωματωμένου στο MS-Project αλγορίθμου εξομάλυνσης.

Η εφαρμογή της εξομάλυνσης του λογισμικού στο δίκτυο έχει ως αποτέλεσμα την επιμήκυνση του χρονοδιαγράμματος στις 78 ημέρες με ταυτόχρονη μείωση της αντικειμενικής συνάρτησης σε 18332.

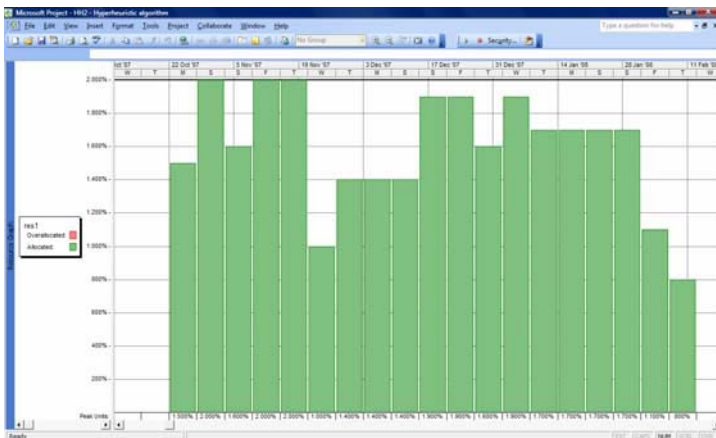
Το σημείο αυτό αποτελεί την αφετηρία για τους υπερειρευτικούς αλγορίθμους (δ_n = 78). Η εφαρμογή τους στο δίκτυο έχει ως αποτέλεσμα τη βελτίωση της αντικειμενικής συνάρτησης, γεγονός που αντανακλά πιο ομοιόμορφη κατανομή του πόρου.

Πίνακας 1: Συγκριτικά αποτελέσματα εξομάλυνσης με το Microsoft Project και τους υπερερευρητικούς αλγορίθμους.
 Table 1: Comparative results for leveling using Microsoft Project and the hyperheuristic algorithms.

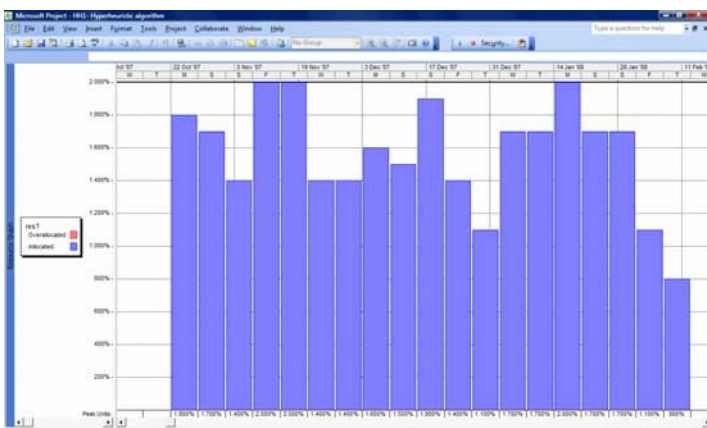
Όνομα Δραστηριότητας	Εξομάλυνση με το Ms-Project	HH1	HH2	HH1	HH2	HH1	HH2
Δραστηριότητα1	100	100	900	700	900	300	100
Δραστηριότητα2	100	900	500	400	200	600	500
Δραστηριότητα3	100	500	900	600	900	500	900
Δραστηριότητα4	100	800	600	700	600	600	600
Δραστηριότητα5	100	500	500	200	500	500	500
Δραστηριότητα6	100	300	500	700	500	800	500
Δραστηριότητα7	100	100	900	100	100	800	100
Δραστηριότητα8	100	600	100	200	100	400	900
Δραστηριότητα9	100	500	700	600	700	500	700
Δραστηριότητα10	100	800	500	200	500	500	500
Δραστηριότητα11	100	100	100	200	100	700	100
Δραστηριότητα12	100	900	100	400	100	500	500
Δραστηριότητα13	100	500	500	800	500	600	100
Δραστηριότητα14	100	600	500	200	500	100	500
Δραστηριότητα15	100	200	500	400	500	100	500
Δραστηριότητα16	100	800	700	400	700	600	700
Δραστηριότητα17	100	600	500	600	500	400	500
Δραστηριότητα18	100	300	200	400	200	400	200
Δραστηριότητα19	100	100	500	800	500	700	500
Δραστηριότητα20	100	700	500	900	500	100	500
Δραστηριότητα21	100	300	500	100	500	100	500
Δραστηριότητα22	100	200	500	200	500	400	500
Δραστηριότητα23	100	200	200	600	200	200	200
Δραστηριότητα24	100	400	500	500	500	400	500
Δραστηριότητα25	100	300	700	400	700	800	700
Δραστηριότητα26	100	600	500	200	500	800	500
Δραστηριότητα27	100	200	500	200	500	400	500
Δραστηριότητα28	100	800	700	200	700	600	700
Δραστηριότητα29	100	900	800	500	800	100	800
Δραστηριότητα30	100	200	900	100	900	300	900
Διάρκεια Έργου	78	78	78	77	77	76	76
z	18332	17760	17874	17948	18000	18252	18324
Εύρος Χρήσης Πόρου	800%-2000%						



Εικόνα 3: Ιστογράμμο κατανομής πόρου μετά την εξομάλνιση με το λογισμικό.
 Picture 3: Resource usage histogram for leveling with Microsoft Project's standard algorithm.



Εικόνα 4: Ιστογράμμο κατανομής πόρου μετά την εξομάλνιση με τον HH2.
 Picture 4: Resource usage histogram for leveling with hyperheuristic HH2.



Εικόνα 5: Ιστογράμμο κατανομής πόρου μετά την εξομάλνιση με τον HH1.
 Picture 5: Resource usage histogram for leveling with hyperheuristic HH1.

Το αρχικό ιστόγραμμα πόρου καταδεικνύει την αναγκαιότητα εφαρμογής διαδικασιών εξομάλυνσης, εξαιτίας της ύπαρξης αρκετά μεγάλων κορυφών και μεγάλης ανομοιογένειας στις απαιτήσεις για πόρο (Εικόνα 2).

Αξίζει να σημειωθεί ότι οι υπερευρετικοί βελτίωσαν την αντικειμενική συνάρτηση ακόμη και σε χρόνο μικρότερο του δη, ενώ ο χρόνος που απαιτήθηκε για την εύρεση των αποτελεσμάτων ήταν της τάξης των δύο λεπτών σε επεξεργαστή Intel P4 (3.0 Ghz). Η χρησιμοποίηση της επιλογής «Level Resources» του MS-Project βελτιώνει την κατανομή του πόρου, παρατείνοντας τη διάρκεια του έργου στις 78 ημέρες (Εικόνα 3). Εφαρμόζοντας τον αλγόριθμο HH2 παρατηρείται εμφανώς πιο ομοιόμορφη χρήση του πόρου κατά τη διάρκεια του έργου, ενώ ο αλγόριθμος HH1 εμφανίζεται πιο αποδοτικός από τον HH2 δεδομένης της επίτευξης ακόμη μικρότερης τιμής για την αντικειμενική συνάρτηση. Η γραφική απεικόνιση των κατανομής που πέτυχαν οι δύο αλγόριθμοι αποδίδεται στα ιστογράμματα των Εικόνων 4 και 5.

5. ΣΥΜΠΕΡΑΣΜΑΤΑ

Στην παρούσα εργασία παρουσιάστηκαν δύο απλοί υπερευρετικοί αλγόριθμοι με τους οποίους επιδιώκεται να βελτιστοποιηθεί η εξομάλυνση της χρήσης ενός πόρου σε έργο. Οι αλγόριθμοι σχεδιάστηκαν για το περιβάλλον του Ms-Project, αλλά σχετικά εύκολα μπορούν να προσαρμοστούν και σε άλλα λογισμικά διαχείρισης έργων. Πρόκειται ίσως για τους απλούστερους υπερευρετικούς αλγόριθμους που θα μπορούσε κανείς να αναπτύξει στο περιβάλλον αυτό. Από μια προκαταρκτική σύγκριση του αλγορίθμου με τη διαδικασία της εξομάλυνσης που υποστηρίζει το Ms-Project προέκυψαν ενθαρρυντικά αποτελέσματα. Παρά ταύτα είναι αυτονόητο ότι απαιτείται μια πιο συστηματική ανάλυση της απόδοσής τους.

Ως εκ τούτου, η ερευνητική μας προσπάθεια εστιάζεται, αφενός στην ανάπτυξη νέων υπερευρετικών βασισμένων σε μεταερευρετικούς αλγορίθμους και, αφετέρου, σε μια συστηματική ανάλυση της υπολογιστικής τους συμπεριφοράς.

Η ανάπτυξη υπερευρετικών αλγορίθμων φιλοδοξεί να βοηθήσει την υλοποίηση μιας νέας γενιάς Συστημάτων Στήριξης Αποφάσεων, τα οποία θα στηρίζονται σε απλούς, προσαρμοσμένους και αποτελεσματικούς αλγορίθμους. Η ανάπτυξη ενός Συστήματος Στήριξης Αποφάσεων για την αντιμετώπιση διαφόρων προβλημάτων του χρονοπρογραμματισμού έργων αποτελεί και τον τελικό στόχο της έρευνάς μας.

6. ΑΝΑΦΟΡΕΣ

1. Bandelloni M., Tucci M., Rinaldi R., "Optimal resource leveling using non-serial dynamic programming", *European Journal of Operational Research* 78, 1994, 162-177.
2. Burke E., Kendall G., Newall J., Hart E., Ross P., Schulenburg S., "Hyper-heuristics: An emerging direction in modern search technology", in F. Glover, G.A. Kochenberger (eds.), *Handbook of Metaheuristics*, Kluwer Academic Publishers, 2003, 457-474.
3. Burke E., Kendall G., Soubeiga E., "A tabu search hyperheuristic for timetabling and rostering", *Journal of Heuristics*, 9, 2004, 451-470.
4. Cowling P., Kendall G., Han L., "An investigation of a hyperheuristic genetic algorithm applied to a trainer scheduling problem", *Proceedings of the Congress on Evolutionary Computation 2002 (CEC 2002)*, 1185-1190.
5. Dowland K.A., Soubeiga E., Burke E., "A simulated annealing based hyperheuristic for determining shipper sizes for storage and transportation", *European Journal of Operational Research* 179, 2007, 759-774.
6. Easa S.M., "Resource leveling in construction by optimization", *Journal of Construction Engineering and Management*, 115(2), 1989, 302 – 316.
7. Harris R.B., *Precedence and Arrow Networking Techniques for Construction*, Wiley, New York, 1978.
8. Harris R.B., "Packing method for resource leveling (PACK)", *Journal of Construction Engineering and Management*, 116(2), 1990, 331 – 350.
9. Hegazy T., "Optimization of resource allocation and levelling using genetic algorithms", *Journal of Construction Engineering and Management*, 1999, 167-175.
10. Leu S.S., Yang C.H., Huang J.C., "Resource leveling in construction by genetic algorithm-based optimization and its decision support system application", *Automation in Construction*, 10, 2000, 27-41.
11. Son J., Skibniewski M.J., "Multiheuristic approach for resource leveling problem in construction engineering: Hybrid approach", *Journal of Construction Engineering and Management*, 1999, 23-31.
12. Shtub A., Bard J.F., Globerson S., *Διαχείριση Έργων: Διεργασίες, Μεθοδολογίες και Τεχνικοοικονομική*, Επιμέλεια-Εισαγωγή: Κ. Π. Αναγνωστόπουλος, Εκδόσεις Επίκεντρο, Θεσσαλονίκη, 2008.
13. Zhao S.L., Liu Y., Zhao H.M., Zhou R.L., "GA-based resource levelling optimization for construction project", *Proceedings of the Fifth International Conference on Machine Learning and Cybernetics*, Dalian, 13-16 August 2006.

Κ. Π. Αναγνωστόπουλος

Πολιτικός Μηχανικός-Οικονομολόγος, Αναπληρωτής Καθηγητής, Τομέας Συστημάτων Διοίκησης, Τμήμα Μηχανικών Παραγωγής και Διοίκησης, Πολυτεχνική Σχολή Δ.Π.Θ., 671 00 Ξάνθη.

Γ. Κ. Κουλίνας

Μηχανικός Παραγωγής και Διοίκησης Δ.Π.Θ., Υποψήφιος Διδάκτορας, Βενιζέλου 70, 671 00 Ξάνθη.

Extended summary

Resource Leveling Using Hyperheuristic Algorithms

K. P. ANAGNOSTOPOULOS

Associate Professor DUTH

G. K. KOULINAS

PhD Candidate

Abstract

In this paper two simple hyperheuristic algorithms for resource leveling are proposed. A hyperheuristic algorithm operates in the heuristics domain rather than in the solutions domain. A set of “low level” heuristics operates in the current solution neighborhood. The first hyperheuristic selects a “low level” heuristic based on its previous average performance. The second hyperheuristic works as a multistart greedy algorithm. Both algorithms have been programmed in Microsoft Project, and a comparison with the software’s standard algorithm has been performed on an example project.

Resource leveling, which is among the top challenges in project management, attempts to reduce the variations among the peaks and valleys in the resource usage histogram so that the project duration does not exceed a specific limit [7, 13].

In this paper, two hyperheuristic algorithms are presented for single resource leveling in a project network. A hyperheuristic algorithm differs from the existing approaches because it operates at a higher level of abstraction and often has no knowledge of the problem domain. It only has access to a set of low level heuristics that it can call upon, but with no knowledge as to the purpose or function of a given low level heuristic [2, 3, 4, 5].

The motivation behind this suggested approach is that once a hyperheuristic algorithm has been developed, then new problem domains can be tackled by only having to replace the set of low level heuristics and the evaluation function, which indicates the quality of a given solution [2].

A diagram of a general hyper-heuristic framework is shown in Figure 1.

The first algorithm (HH1) initially chooses one from a set of different available low level heuristics and applies it once. When a predetermined number of loops are completed the algorithm estimates recursively the performance for each low level heuristic.

Submitted: July 3, 2008 Accepted: Sep. 17, 2008

Then, the probability for a low level heuristic to be chosen depends on its performance. The second hyperheuristic (HH2) is a multistart greedy algorithm that operates on the low level heuristics set.

Both hyperheuristics were developed in Microsoft Project’s environment in an attempt to improve its standard leveling procedure (“Level Resources”). The software allows the user to define a priority level for every single activity, with values from zero to 1000. HH1 and HH2 are coded using Visual Basic for Applications (VBA).

Seven different low level heuristics are used. Some of them are:

1. Swap two priorities (L_1): The algorithm chooses at random two activities and swaps their priorities.
2. Replace priority (L_2): One activity is chosen at random and its priority is replaced with another different random priority.
3. Swap priorities of tasks with the largest and smallest Total Slack (L_7): This low level heuristic finds the tasks having the largest and the smallest current Total Slack and swaps their priorities.

The proposed algorithms were tested in a random network with thirty activities. The input data, including activities’ durations, precedence relations and daily resource demand, are shown in Figure 1.

The initial total project duration for the example project, was 49 days, the objective function value 37688, and the resource usage range between 800% and 6400%. The results of leveling with HH1 and HH2 are shown in Table 1, as well as those achieved by the software’s built-in procedure.

As mentioned above, the standard leveling procedure of MS-Project causes a larger time schedule (78 days) with a smaller objective function value (18332). This is the starting point for the two hyperheuristics, which improved the objective function, that is, they found a more efficient resource usage during the project.

HH2 improves the objective function (17874) for the same project duration (78 days). Algorithm HH1 seems to be more efficient because it further improves the objective function value (17760). The resource allocation achieved by the hyperheuristics is shown in the resource usage histograms (Pictures 4 and 5).

K. P. Anagnostopoulos

Civil Engineer-Economist, Associate Professor, Department of Production Engineering and Management, School of Engineering, Democritus University of Thrace, 671 00 Xanthi.

G. K. Koulinas

Production and Management Engineer, PhD Candidate, Democritus University of Thrace, Venizelou 70, 671 00 Xanthi.