

Συστήματα Επιχειρηματικής Ευφυΐας

Το πρόβλημα Nurse rostering - Ενδεικτική επίλυση με αλγόριθμο Variable Neighborhood Search (VNS)

Έβδομη Διάλεξη - Περιεχόμενα (1)

- Συνοπτική παρουσίαση του προβλήματος **Nurse rostering**
 - ❑ Ορισμός του προβλήματος
 - ❑ Η πολυπλοκότητα του προβλήματος
 - ❑ Οι δύο διεθνείς διαγωνισμοί
 - ❑ Οι μεταβλητές και οντότητες του προβλήματος
 - ❑ Οι ανελαστικοί περιορισμοί
 - ❑ Οι ελαστικοί περιορισμοί

Έβδομη Διάλεξη- Περιεχόμενα (2)

➤ Ενδεικτική επίλυση με αλγόριθμο **Variable Neighborhood Search (VNS)**

- ❑ Η αναπαράσταση της λύσης
- ❑ Το λογικό διάγραμμα του προτεινόμενου αλγορίθμου
- ❑ Οι βασικές λειτουργίες του αλγορίθμου (Neighborhoods)
- ❑ Τελικά συμπεράσματα

Συνοπτική παρουσίαση του προβλήματος Nurse rostering (1)

► Ορισμός του προβλήματος:

Το πρόβλημα αφορά στον ημερήσιο ή εβδομαδιαίο προγραμματισμό των βάρδιών του νοσηλευτικού προσωπικού σε ένα νοσηλευτικό ίδρυμα, με τέτοιο τρόπο ώστε να ικανοποιούνται διάφοροι λειτουργικοί περιορισμοί του ιδρύματος ή περιορισμοί που αφορούν στις προσωπικές ιδιαιτερότητες του προσωπικού.

Συνοπτική παρουσίαση του προβλήματος Nurse rostering (2)

- ▶ Η πολυπλοκότητα του προβλήματος: το πρόβλημα ανήκει στην γενικότερη κλάση των προβλημάτων scheduling. Έχει αποδειχτεί ότι το πρόβλημα στην γενική του μορφή είναι NP-Complete (*). Με απλά λόγια, αυτό σημαίνει ότι η υπολογιστική πολυπλοκότητα (και συνεπώς ο χρόνος επίλυσής του) αυξάνει εκθετικά ή ακόμη και παραγοντικά σε σχέση με την αύξηση του μεγέθους του. Ακόμη σημαίνει, ότι είναι μάλλον απίθανο να βρεθεί ένας ντετερμινιστικός αλγόριθμος πολυωνυμικού χρόνου εκτέλεσης, ο οποίος να το επιλύει. Ως εκ τούτου, η χρήση εναλλακτικών αλγορίθμων (της ΥΝ), οι οποίοι θα δίνουν ικανοποιητική λύση σε αποδεκτό χρόνο, είναι απαραίτητη.

Συνοπτική παρουσίαση του προβλήματος Nurse rostering (3)

- ▶ Οι δύο διεθνείς διαγωνισμοί: ήταν αφιερωμένοι στο πρόβλημα NR.
 - Ο πρώτος διαγωνισμός διεξήχθη το 2010 (INRC-2010 : <https://www.kuleuven-kulak.be/nrprcompetition>) . Σε αυτόν παρουσιάστηκαν κάποια δύσκολα αρχεία δεδομένων από πραγματικές καταστάσεις. (*)

Ο αλγόριθμος VNS που θα παρουσιασθεί στην επόμενη ενότητα, κατάφερε όχι μόνο να πετύχει τα best known αποτελέσματα σχεδόν για όλα τα αρχεία του διαγωνισμού, αλλά σε ορισμένες περιπτώσεις πέτυχε νέα best αποτελέσματα, τα οποία ισχύουν μέχρι σήμερα. (2016)

Η ύλη της παρούσας διάλεξης αναφέρεται, εν πολλοίς, σε αυτόν τον διαγωνισμό.

Συνοπτική παρουσίαση του προβλήματος Nurse rostering (4)

- Ο δεύτερος διαγωνισμός διεξήχθη το 2014.

(INRC II: <http://mobiz.vives.be/inrc2/>) . Εκτός από τις διαφορές στον ορισμό των περιορισμών, μια βασική διαφορά είναι η ύπαρξη της κυκλικότητας, δηλαδή της διατήρησης κάποιου είδους μνήμης από έναν ορίζοντα προγραμματισμού στον επόμενο.

Συνοπτική παρουσίαση του προβλήματος **Nurse rostering (5)**

► Οι μεταβλητές και οντότητες του προβλήματος

- N: σύνολο νοσοκόμων
- D: το σύνολο των ημερών στις οποίες αφορά ο προγραμματισμός
- S: το σύνολο των τύπων βαρδιών (shift types). Κάθε τύπος χαρακτηρίζεται από ένα σύνολο προσόντων (skills).
- C: το σύνολο των συμβάσεων εργασίας (contacts) των νοσοκόμων.
- P: το σύνολο των επιθυμητών ή ανεπιθύμητων προτύπων (patterns). Για παράδειγμα, το πρότυπο Night Shift, Morning Shift είναι ανεπιθύμητο για κάθε νοσοκόμο.

Συνοπτική παρουσίαση του προβλήματος Nurse rostering (6)

► Οι ανελαστικοί περιορισμοί

- H1: Κάλυψη ζήτησης τύπων βαρδιών - σε κάθε ημέρα του ορίζοντα προγραμματισμού πρέπει να παρέχεται ο απαραίτητος αριθμός βαρδιών από κάθε είδος βάρδιας.
- H2: Μοναδικότητα βάρδιας-κάθε νοσοκόμος εργάζεται μια το πολύ βάρδια σε κάθε ημέρα.

Συνοπτική παρουσίαση του προβλήματος Nurse rostering (7)

► Ελαστικοί περιορισμοί

- ❑ S1: Μέγιστο πλήθος αναθέσεων νοσοκόμων-σε κάθε νοσοκόμο ανατίθεται ένας μέγιστος αριθμός βαρδιών, σύμφωνα με την σύμβασή του.
- ❑ S2: Ελάχιστο πλήθος αναθέσεων βαρδιών νοσοκόμων-σε κάθε νοσοκόμο ανατίθεται ένας μέγιστος αριθμός βαρδιών, σύμφωνα με την σύμβασή του.
- ❑ S3: Μέγιστο πλήθος συνεχόμενων ημερών εργασίας νοσοκόμου.
- ❑ S4: Ελάχιστος πλήθος συνεχόμενων ημερών εργασίας νοσοκόμου.

Συνοπτική παρουσίαση του προβλήματος Nurse rostering (8)

- ❑ S5: Μέγιστο πλήθος συνεχόμενων ημερών ρεπό νοσοκόμου.
- ❑ S6: Ελάχιστο πλήθος συνεχόμενων ημερών ρεπό νοσοκόμου.
- ❑ S7: Μέγιστο πλήθος συνεχόμενων week end με τουλάχιστον μια βάρδια σε αυτά.
- ❑ S8: Μέγιστο πλήθος week end με τουλάχιστον μια βάρδια σε αυτά.
- ❑ S9: Δύο ρεπό μετά από κάθε νυχτερινή βάρδια.

Συνοπτική παρουσίαση του προβλήματος Nurse rostering (9)

- ❑ S10: Πλήρη week ends- αν ένας νοσοκόμος εργάζεται τουλάχιστον μια βάρδια σε ένα week end τότε πρέπει να έχει βάρδια σε όλες τις ημέρες του week end.
- ❑ S11: Πανομοιότυπες βάρδιες week end-κάθε νοσοκόμος που εργάζεται σε ένα week end πρέπει να έχει το ίδιο είδος βάρδιας σε όλες τις ημέρες του week end.
- ❑ S12: Ανεπιθύμητα πρότυπα πρέπει να αποφεύγονται.
- ❑ S13: Προτίμηση για ημέρα εργασίας.
- ❑ S14: Προτίμηση για ημέρα ρεπό

Συνοπτική παρουσίαση του προβλήματος Nurse rostering (10)

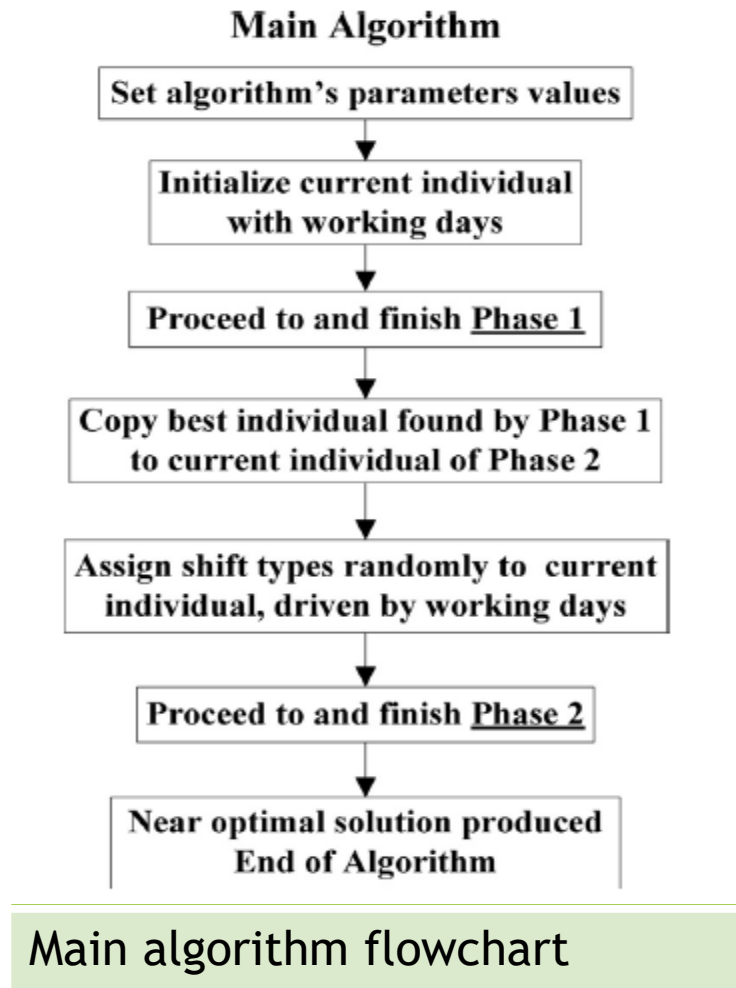
- ❑ S15: Προτίμηση για εργασία συγκεκριμένου τύπου βάρδιας.
- ❑ S16: Προτίμηση για αποφυγή τύπου βάρδιας σε ημέρα εργασίας.
- ❑ S17: Βάρδια με εναλλακτικά προσόντα-αν ένας νοσοκόμος εργάζεται σε βάρδια που απαιτεί προσόντα τα οποία δεν διαθέτει, υπάρχει παραβίαση. Π.χ. Ειδικευόμενος έναντι ειδικευμένου.
- ❑ S18: Όχι νυχτερινή βάρδια πριν από ένα week end με ρεπό.

Ενδεικτική επίλυση με αλγόριθμο Variable Neighborhood Search (VNS) (1)

- Η αναπαράσταση της λύσης
 - Κάθε υποψήφια λύση παριστάνεται ως ένας $|N| \times |D|$ πίνακας R , όπου N το σύνολο των νοσοκόμων και D το σύνολο των ημερών προγραμματισμού. Σε κάθε κελί του πίνακα ανατίθεται είτε ένας αριθμός που είναι ο κωδικός του είδους βάρδιας, είτε -1 που σημαίνει ρεπό.
 - Με βάση την παραπάνω παράσταση και με δεδομένο ότι η τυχαία αρχικοποίηση αναθέτει τον απαιτούμενο αριθμό ειδών βαρδιών σε κάθε ημέρα (στήλη), σύμφωνα με τα δεδομένα εισόδου, ικανοποιείται ο περιορισμός $H1$. Η ικανοποίησή του δεν παύει καθ' όλη την διάρκεια του αλγορίθμου, αφού τα swap εκτελούνται καθέτως.

Ενδεικτική επίλυση με αλγόριθμο Variable Neighborhood Search (VNS) (2)

- Το λογικό διάγραμμα του προτεινόμενου αλγορίθμου



Ενδεικτική επίλυση με αλγόριθμο Variable Neighborhood Search (VNS) (3)

► Οι δύο φάσεις του αλγορίθμου

- ❑ Στην πρώτη γίνεται η αρχικοποίηση της υποψήφιας λύσης με τυχαίο τρόπο και με ικανοποίηση των ενεργών βαρδιών ανά ημέρα. Επίσης, γίνεται η αρχικοποίηση των διάφορων μεταβλητών του αλγορίθμου. Τέλος, με χρήση της συνάρτησης **Swap_Random_Rosters()** προσδιορίζεται το ποιοι νοσοκόμοι θα εργαστούν ανά ημέρα, ανεξαρτήτως shift type. Ο ορισμός και η λειτουργικότητα της **Swap_Random_Rosters()** περιγράφεται στις επόμενες διαφάνειες.
- ❑ Στην δεύτερη φάση, ο αλγόριθμος «βασίζεται» στις αναθέσεις εργασίας των νοσοκόμων στις διάφορες ημέρες και απλώς αναθέτει το είδος της βάρδιας σε κάθε νοσοκόμο.

Ενδεικτική επίλυση με αλγόριθμο Variable Neighborhood Search (VNS) (4)

- Οι βασικές λειτουργίες του αλγορίθμου (Neighborhoods)-οι δύο procedures της 1ης φάσης

1. Create a random list $Days$ of all days of the planning horizon
2. For each day D_1 in $Days$ do
 - 2.1 For each day D_2 in $Days$ with $D_2 \succcurlyeq D_1$ do
 - 2.1.1 Set left bound = D_1 and set right bound = D_2
 - 2.1.1.1 For each day D of $Days$ that is between left and right bounds
 - 2.1.1.1.1 Swap the contents of cells $[R_1, D]$ and $[R_2, D]$ with a predefined probability
 - 2.1.1.2 End for
 - 2.1.2 If fitness after swaps is worse than before then cancel all swaps
 - 2.1.3 Else update fitness and fix the swaps
 - 2.2. End do loop
3. End do loop

1. Create two random lists L_1 and L_2 of all rosters
2. For each roster R_1 of list L_1 do
 - 2.1 For each roster R_2 of list L_2 do
 - 2.1.1 Execute $Group_Swaps(R_1, R_2)$
 - 2.2 End do loop
3. End do loop

Group_Swaps() καλείται από την **Swap_Random_Rosters()**

Swap_Random_Rosters()

Ενδεικτική επίλυση με αλγόριθμο Variable Neighborhood Search (VNS) (4)

► Ο ψευδό-κώδικας της 2^{ης} φάσης

1. Set $K = 1$, $M = 1$ and copy the best individual found by *Phase 1* to current individual of *Phase 2*
2. Assign shift types randomly to the best individual produced by *Phase 1*
3. Set best individual's fitness to worse possible
4. While termination criterion is not met do
 - 4.1. Set *Stagnation_Counter* = 0
 - 4.2. While *Stagnation_Counter* is less than *Stagnation_Limit*
 - 4.2.1 *Stagnation_Counter* = *Stagnation_Counter* + 1
 - 4.2.2 Execute *Swap_Ordered_Rosters()* with probability *Swap_Selection_Probability* or
Execute *Swap_Random_Rosters()* with probability $1 - \text{Swap_Selection_Probability}$, both on current individual
 - 4.2.3 If fitness of current individual is not greater than fitness of best individual then
 - 4.2.3.1 $K = K + 1$ (in case the *Swap_Ordered_Rosters()* was executed at step 4.2.2) or
 $M = M + 1$ (in case the *Swap_Random_Rosters()* was executed at step 4.2.2)
 - 4.2.3.2 Update best individual and its fitness
 - 4.2.3.3 Set *Stagnation_Counter* = 0
 - 4.2.4 End if
 - 4.2.5 Sequentially execute *Swap1()*, *Swap2()*, *Swap3()*, *Swap4()*, *Swap5()* and *Swap6()* on current individual and after any single execution if fitness of current individual is not greater than fitness of best individual then update best individual and its fitness
 - 4.3 End while
 - 4.4 Execute *Swap_Two_Rosters()*
 - 4.5 $\text{Swap_Selection_Probability} = 1 + 0.4 * \text{Density} - K * (1 + 0.8 * \text{Density}) / (K + M)$
5. End while
6. Return best individual and its fitness

Ενδεικτική επίλυση με αλγόριθμο Variable Neighborhood Search (VNS) (5)

► Οι procedures της 2^{ης} φάσης

1. Create a descending ordered list L of all rosters, according to their fitness, i.e. rosters with smaller fitness follow rosters with bigger ones, from left to right.
2. for each roster R_1 of L do
 - 2.1 for each roster R_2 of L , to the right of R_1 , do
 - 2.1.1 Execute *Group_Swaps*(R_1, R_2)
 - 2.2 End do
3. End do

Swap_Ordered_Rosters()

Ενδεικτική επίλυση με αλγόριθμο Variable Neighborhood Search (VNS) (6)

1. For each day (column) D of current individual do

1.1 For each empty cell EC of day D do

1.1.1 For each non empty cell nEC of day D do

1.1.1.1 Swap the contents of cells EC and nEC

1.1.1.2 Record the swap and the fitness of current individual after the swap

1.1.1.3 Restore the swap

1.1.2 End do

1.2 End do

1.3 Select the swap that corresponds to the best fitness among all above recorded swaps (step 1.1.1.2) and apply it to current individual

2. End do

1. For each day (column) D of current individual do

1.1 For each cell C_1 of day D do

1.1.1 For each cell C_2 of day D do

1.1.1.1 Swap the contents of cells C_1 and C_2

1.1.1.2 Record the swap and the fitness of current individual after the swap

1.1.1.3 Restore the swap

1.1.2 End do

1.2 End do

1.3 Select the swap that corresponds to the best fitness among all above recorded swaps (step 1.1.1.2) and apply it to current individual

2. End do

Swap1()

Swap2()

Ενδεικτική επίλυση με αλγόριθμο Variable Neighborhood Search (VNS) (7)

1. For each day (column) D of current individual do
 - 1.1 For each cell C_1 of day D do
 - 1.1.1 For each cell C_2 of day D do
 - 1.1.1.1 Swap the contents of cells C_1 and C_2
 - 1.1.1.2 Record the swap and the fitness of current individual after the swap
 - 1.1.1.3 Restore the swap
 - 1.1.2 End do
 - 1.2 End do
 - 1.3 Select the swap that corresponds to the best fitness among all above recorded swaps (step 1.1.2) and apply it to current individual
2. End do

Swap3()

1. For each day (column) D of current individual do
 - 1.1 For each cell C_1 of day D do
 - 1.1.1 For each cell C_2 of day D do
 - 1.1.1.1 Swap the contents of cells C_1 and C_2
 - 1.1.1.2 Record the swap and the fitness of current individual after the swap
 - 1.1.1.3 Restore the swap
 - 1.1.2 End do
 - 1.2 End do
 2. End do
 3. Select the swap that corresponds to the best fitness among all above recorded swaps (step 1.1.2) and apply it to current individual

Swap4()

Ενδεικτική επίλυση με αλγόριθμο Variable Neighborhood Search (VNS) (8)

1. For each day (column) D of current individual do
 - 1.1 For each empty cell EC of day D do
 - 1.1.1 For each non empty cell nEC of day D do
 - 1.1.1.1 Swap the contents of cells EC and nEC
 - 1.1.1.2 Record the swap and the fitness of current individual after the swap
 - 1.1.1.3 Restore the swap
 - 1.1.2 End do
 - 1.2 End do
2. End do
3. Select the swap that corresponds to the best fitness among all above recorded swaps (step 1.1.1.2) and apply it to current individual

Swap5()

1. For each day (column) D of current individual do
 - 1.1 For each non empty cell nEC_1 of day D do
 - 1.1.1 For each non empty cell nEC_2 of day D do
 - 1.1.1.1 Swap the contents of cells nEC_1 and nEC_2
 - 1.1.1.2 Record the swap and the fitness of current individual after the swap
 - 1.1.1.3 Restore the swap
 - 1.1.2 End do
 - 1.2 End do
2. End do
3. Select the swap that corresponds to the best fitness among all above recorded swaps (step 1.1.1.2) and apply it to current individual

Swap6()

Ενδεικτική επίλυση με αλγόριθμο Variable Neighborhood Search (VNS) (9)

Generic parameters	Value
<i>Number of cycles of Phase 1</i>	1
<i>Initial value of Swap_Selection_Probability</i>	0.3
Parameters which affect the evolution of the algorithm and the value of Swap_Selection_Probability	Value
<i>Density</i>	$\frac{\sum_{d=1}^D \sum_{s=1}^S \text{nurses_required}(d,s)}{\text{Number_of_nurses} \cdot \text{Number_of_days}}$ where <i>nurses_required(d, s)</i> is the total number of nurses required in order to cover all shifts of shift type <i>s</i> in day <i>d</i>
<i>Stagnation_Limit</i>	$\frac{\text{Number_of_nurses} \cdot \text{Number_of_shift_types}}{5}$

Πίνακας: Οι τιμές βασικών παραμέτρων

Τελικά συμπεράσματα

- ❑ Ο προηγούμενος αλγόριθμος δημοσιεύτηκε στο περιοδικό **Computers & Operations Research** το 2015
- ❑ Τα στοιχεία της δημοσίευσης είναι: A two-phase adaptive variable neighborhood approach for nurse rostering, Ioannis X. Tassopoulos, Ioannis P. Solos, Grigorios N. Beligiannis , Computers & Operations Research, 2015.
- ❑ Ο αλγόριθμος κατάφερε να ισοφαρίσει τα καλύτερα μέχρι τότε δημοσιευμένα αποτελέσματα από διάφορους αλγορίθμους.
- ❑ Σε μερικές περιπτώσεις, πέτυχε νέα καλύτερα αποτελέσματα.
- ❑ Σε σύγκριση με τον αλγόριθμο του νικητή του διαγωνισμού ήταν μακράν καλύτερος.
- ❑ Είναι επίσης καλύτερος από κάθε δημοσιευμένο αλγόριθμο που αναφέρεται στα δεδομένα του διαγωνισμού.
- ❑ Τα αποτελέσματα βρίσκονται στο site: <https://www.kuleuven-kulak.be/nrcompetition/instances-results>
- ❑ Επιβεβαιώνεται ότι ο αλγόριθμος VNS και μάλιστα με καθοδήγηση (guided), πετυχαίνει πολύ καλά αποτελέσματα σε πολλά προβλήματα timetabling και scheduling και συγκεκριμένα και στο nurse rostering πρόβλημα.

Σας ευχαριστώ για την προσοχή σας

