

## Εργασία 3

## Απαντήσεις

Ονοματεπώνυμο:  
Αντρέας Παύλου

A.M.: 1115202100223

## Θέμα 1

| Instance | FC-BT   | FC-MAC   | FC-CBJ  | MIN_CONFLICTS                          |
|----------|---|--|---|--|
| 2-f24    | Assignments: 410<br>Checks: 43416<br>Time: 0.16668 sec          | Assignments: 211<br>Checks: 169663<br>Time: 0.3389 sec       | Assignments: 410<br>Checks: 43416<br>Time: 0.1604 sec         | Assignments: -<br>Checks: -<br>Time: - |
| 2-f25    | Assignments: 128489<br>Checks: 22629909<br>Time: 74.2436 sec    | Assignments: 25304<br>Checks: 53784583<br>Time: 149.9657 sec | Assignments: 3495<br>Checks: 608865<br>Time: 2.0099 sec       | Assignments: -<br>Checks: -<br>Time: - |
| 3-f10    | Assignments: 1640808<br>Checks: 168488496<br>Time: 463.3715 sec | Assignments: 772<br>Checks: 1186917<br>Time: 2.7492 sec      | Assignments: 31015<br>Checks: 4208541<br>Time: 11.1466 sec    | Assignments: -<br>Checks: -<br>Time: - |
| 3-f11    | Assignments: 347035<br>Checks: 56386851<br>Time: 133.9254 sec   | Assignments: 8946<br>Checks: 25576285<br>Time: 68.3457 sec   | Assignments: 280720<br>Checks: 45881879<br>Time: 109.0364 sec | Assignments: -<br>Checks: -<br>Time: - |
| 6-w2     | Assignments: 263<br>Checks: 48310<br>Time: 0.0838 sec           | Assignments: 43<br>Checks: 97215<br>Time: 0.117 sec          | Assignments: 263<br>Checks: 48310<br>Time: 0.0826 sec         | Assignments: -<br>Checks: -<br>Time: - |
| 7-w1-f4  | Assignments: 37207<br>Checks: 1428953<br>Time: 2.4260 sec       | Assignments: 525<br>Checks: 356673<br>Time: 0.2864 sec       | Assignments: 19691<br>Checks: 761794<br>Time: 1.4174 sec      | Assignments: -<br>Checks: -<br>Time: - |
| 7-w1-f5  | Assignments: -<br>Checks: -<br>Time: -                          | Assignments: 8476<br>Checks: 34177396<br>Time: 25.4055 sec   | Assignments: 5685<br>Checks: 379211<br>Time: 0.5802 sec       | Assignments: -<br>Checks: -<br>Time: - |
| 8-f10    | Assignments: -<br>Checks: -<br>Time: -                          | Assignments: 17144<br>Checks: 34281675<br>Time: 74.1703 sec  | Assignments: 89600<br>Checks: 11533273<br>Time: 33.3175 sec   | Assignments: -<br>Checks: -<br>Time: - |
| 8-f11    | Assignments: 1541509<br>Checks: 226519475<br>Time: 600.5770 sec | Assignments: 3357<br>Checks: 7500135<br>Time: 16.6046 sec    | Assignments: 15029<br>Checks: 3257124<br>Time: 8.0097 sec     | Assignments: -<br>Checks: -<br>Time: - |
| -11      | Assignments: 3644<br>Checks: 715867<br>Time: 3.4099 sec         | Assignments: 1875<br>Checks: 3895932<br>Time: 9.1482 sec     | Assignments: 3644<br>Checks: 715867<br>Time: 3.2450 sec       | Assignments: -<br>Checks: -<br>Time: - |
| 14-f27   | Assignments: 1008869<br>Checks: 41131131<br>Time: 186.0311 sec  | Assignments: 18161<br>Checks: 6422588<br>Time: 13.1030 sec   | Assignments: 27648<br>Checks: 1506978<br>Time: 9.1750 sec     | Assignments: -<br>Checks: -<br>Time: - |
| 14-f28   | Assignments: 408915<br>Checks: 28573194<br>Time: 123.1860 sec   | Assignments: 6818<br>Checks: 7171206<br>Time: 14.0072 sec    | Assignments: 16502<br>Checks: 1349846<br>Time: 6.1959 sec     | Assignments: -<br>Checks: -<br>Time: - |

## Color Mapping

|                    |             |
|--------------------|-------------|
| Solution Found     | Interrupted |
| Solution Not Found |             |

Σχήμα 1: Αποτελέσματα

- Βάσει τα πιο πάνω αποτελέσματα, εξάγονται τα εξής συμπεράσματα: 1) Η χρήση της ευρετικής DOM/WDEG έχει σταθερό αποτέλεσμα κάθε φορά καθώς δεν εξαρτάται από κάποιο τυχαίο παράγοντα. 2) Όπως και στη θεωρία, τώρα επαληθεύεται ότι οι κόμβοι που έγιναν

assigned και οι έλεγχοι για τον κάθε αλγόριθμο είναι:

$$FC - CBJ \leq FC - BT$$

$$MAC < FC - BT$$

, χωρίς να μπορεί να εξαχθεί κάποιο συμπέρασμα για MAC/FC-CBJ. Η ισότητα μεταξύ FC-CBJ/FC-BT εμφανίζεται σε δύο στιγμιότυπα, στα οποία υπήρξε τελικό αποτέλεσμα μετά από ελάχιστους ελέγχους/ανάθεσεις. Κατά κύριο λόγο η χρήση CBJ αντί για BT συμβάλλει αρκετά στην επίλυση των στιγμιότυπων σε πολύ λιγότερο χρόνο. 3) Min\_conflicts: Δεν φαίνεται να είναι αποδοτική, καθώς τις περισσότερες φορές δεν τερματίζεται καν, με κάποιο αποτέλεσμα. Τα αποτελέσματα διαφέρουν αφού υπάρχει ο τυχαίος παράγοντας (επιλογή μεταβλητής). Συνεπώς αυτή η επιλογή τυχαίας μεταβλητής συμβάλλει στην κακή αποδοτικότητα του αλγορίθμου, λόγω του ότι κατά την επιλογή της, επαναλαμβάνεται πολλές φορές τυχαία μέχρι να βρεθεί η λύση.

Υλοποιήσεις/Αλλαγές: Όλες οι αλλαγές βρίσκονται στη νέα κλάση NewCSP.

- DOM/WDEG: Αποτελείται από τις συναρτήσεις wdeg, dom\_wdeg, find\_dom. Αρχικά γίνεται έλεγχος κάθε μεταβλητής η οποία δεν είναι αρχικοποιημένη με κάποια τιμή. Για κάθε τέτοια μεταβλητή ελέγχουμε τους μη αρχικοποιημένους γείτονες, και προστίθεται, το βάρος του συνδυασμού (var, neighbour) σε ένα νέο counter (αυτό το βάρος οφείλεται στο ότι υπήρξε domain wipe out, κατά τον έλεγχο ενός περιορισμού). Στη συνέχεια βρίσκεται μια νέα μεταβλητή η οποία εξαρτάται από τη διαίρεση του μεγέθους του domain της (στην περίπτωση που το curr\_domain[var] είναι κενό τότε λαμβάνεται υπόψιν το domain[var]) δια του προηγούμενου counter. Τέλος επιστρέφεται η μεταβλητή με τη μικρότερη αναλογία.

Αυξήσεις βαρών

1)Forward\_Check2: Αύξηση βάρους κατά 1 μεταξύ (var, B), (B, var) κατά την στιγμή που το domain της μεταβλητής B γίνεται κενό, από τα prunes που προσέκυψαν, από την ανάθεση τιμής στο var.

2)Mac: Στην συνάρτηση revise2, όμοιος με Forward\_Check2.

-Conflict Directed Backjumping

Κύριες Δομές:

1)Past\_FC(dictionary of sets)

2)Conflict\_setdictionary of sets)

3)no\_good(set)

Βασικά σημεία αλγορίθμου είναι τα εξής: 1) Προσθήκη στο Past\_FC, μίας unassigned μεταβλητής, την τρέχουσα assigned μεταβλητή η οποία προκάλεσε κάποιο prune στο Domain της, προηγούμενης unassigned. 2) Κατά domain wipe out κάποιας unassigned μεταβλητής, γίνεται ενημέρωση του conflict\_set της τρέχουσας μεταβλητής με την ένωση του conflict\_set της τρέχουσας μεταβλητής και του past\_fc της μεταβλητής στην οποία έγινε το domain wipe out. 3) Ενημέρωση no\_good set με την ένωση του past\_fc και conflict\_set της τρέχουσας μεταβλητής αφού έγινε έλεγχος όλων των τιμών της τρέχουσας μεταβλητής. 4) Κατά το backtracking, στο προηγούμενο set βρίσκεται η μεταβλητή που προκάλεσε κάποιο conflict και θα πρέπει να βρεθεί η πιο "βαθιά". 5) Αφού βρεθεί η πιο "βαθιά" μεταβλητή ενημερώνεται το conflict\_set της μεταβλητής ώστε να μην χαθεί πληροφορία όσον αφορά προηγούμενα conflicts. 6) Ενημέρωση όλων των conflict\_set που είχαν αλλάξει κατά την ανάθεση της νέας μεταβλητής.

[Βοηθητικό Link](#)

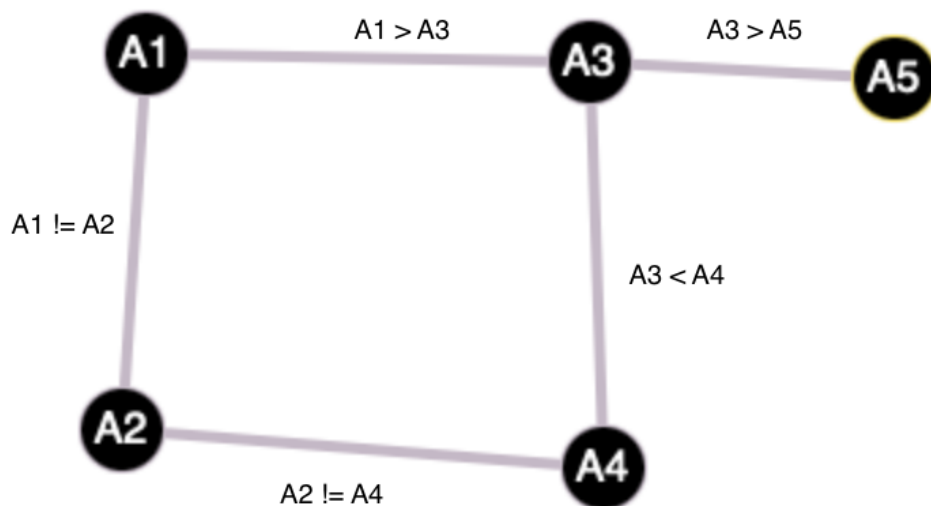
## Θέμα 2

- Variables:  $\{X0, X1, X2, X3\}$   
 Suspects:  $\{X0, X1, X2\}$   
 Domains:  $X0 = \{90\}$ ,  $X1 = \{30, 60\}$ ,  $X2 = \{60, 30\}$   
 Constraints:  $X0 + X2 + X3 = 1$ ,  $X_i = \{0, 1\}$ ,  $\forall i \in \{0, 1, 2\}$   
 $V_i \geq \min\_time = 20 + 45 + 20$ , ( $V = \text{value}$ )
- Αρχικά γίνεται ταξινόμηση μεταβλητών σε φθίνουσα σειρά με βάση την μεγαλύτερη τιμή στο Domain της κάθε μεταβλητής.  
 $\Rightarrow \max(X0) = 90$ ,  $\max(X1) = 60$ ,  $\max(X2) = 60$   
 $\Rightarrow$  Σειρά ελέγχου:  $X0, X1, X2$   
 Για τη σειρά των τιμών, επιλέγεται ταξινόμηση σε φθίνουσα σειρά για το κάθε Domain.  
 Αρχίζοντας τον αλγόριθμο αρχικοποιούμε το  $X0$ , με ένα ( $1 = \text{assigned}$ ,  $0 = \text{unassigned}$ ). Παρατηρούμε ότι τηρούνται όλοι οι περιορισμοί, καθώς μόνο μια μεταβλητή έγινε assigned και ισχύει  $85 < 90$ . Καθώς γνωρίζουμε ότι υπάρχει αποκλειστικά ένας ύποπτος, η ανάθεση κάποιας άλλης μεταβλητής με 1, θα παραβίαζε τον πρώτο περιορισμό. Τέλος, καταλήγουμε ότι ο ύποπτος είναι  $X0$ , ο Γιάννης.
- Ο τρόπος μοντελοποίησης που επιλέχθηκε δεν λαμβάνει υπόψιν γειτονικούς κόμβους, καθώς οι τιμές στα domains απεικονίζουν τα διαστήματα κατά τα οποία οι τρεις ύποπτοι έχουν ελεύθερο χρόνο χωρίς κάποια διακοπή. Συνεπώς, δεν μπορεί να χρησιμοποιηθεί κάποιος αλγόριθμος διάδοσης περιορισμών.

## Θέμα 3

- Variables:  $A1, A2, A3, A4, A5$   
 Domains:  $A1 = \{9, 10, 11\}$ ,  $A2 = \{9, 10, 11\}$ ,  $A3 = \{9, 10, 11\}$ ,  $A4 = \{9, 11\}$ ,  $A5 = \{9, 10, 11\}$   
 Constraints:  $A2 \neq A2$ ,  $A1 > A3$ ,  $A4 \neq 10$ ,  $A3 > A5$ ,  $A3 < A4$ ,  $A2 \neq A4$

## 2. Γράφος



Σχήμα 2: Γράφος

### 3. 3) Σειρά ανάθεσης μεταβλητών: $A1, A2, A3, A4, A5$

Σειρά ανάθεσης τιμών: (Αξουσα)  $Domain_i : \{9, 10, 11\} \forall i \in [1, 5] - 4, Domain_4 = \{9, 11\}$

Assign  $A1 = 9 \Rightarrow D2 = \{10, 11\}, D3 = \emptyset$

Assign  $A1 = 10 \Rightarrow D2 = \{10, 11\}, D3 = \{9\}$

$A3 > A5 \Rightarrow$  Not Consistent

Assign  $A1 = 11 \Rightarrow D2 = \{9, 10\}, D3 = \{9, 10\}$

$(A2 > A4) \Rightarrow$  Consistent

$(A3 < A4) \Rightarrow$  Consistent

$(A3 > A5) \Rightarrow$  Consistent

Assign  $A2 = 9 \Rightarrow D4 = \{11\}$

$(A3 < A4) \Rightarrow$  Consistent

Assign  $A3 = 9 \Rightarrow D4 = \{11\}, D5 = \emptyset$

Assign  $A3 = 10 \Rightarrow D4 = \{11\}, D5 = \{9\}$

Assign  $A4 = 11$

Assign  $A5 = 9$

Αποτέλεσμα:  $A1 = 11, A2 = 2, A3 = 10, A4 = 11, A5 = 9$