

ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ

ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ

ΤΜΗΜΑ ΠΟΛΙΤΙΚΩΝ ΜΗΧΑΝΙΚΩΝ

Διπλωματική Εργασία

**ΑΝΑΠΤΥΞΗ ΛΟΓΙΣΜΙΚΟΥ Η/Υ ΓΙΑ ΤΗΝ ΑΝΑΛΥΣΗ ΚΑΙ ΤΟ  
ΣΧΕΔΙΑΣΜΟ ΚΑΤΑΣΚΕΥΩΝ ΟΠΛΙΣΜΕΝΟΥ ΣΚΥΡΟΔΕΜΑΤΟΣ**

υπό

**ΘΕΟΔΩΡΟ ΑΛΕΞΑΝΔΡΙΔΗ**

Υπεβλήθη για την εκπλήρωση μέρους των

απαιτήσεων για την απόκτηση του

Διπλώματος Πολιτικού Μηχανικού

2021

© 2021 Θεόδωρος Αλεξανδρίδης

Η έγκριση της διπλωματικής εργασίας από το Τμήμα Πολιτικών Μηχανικών της Πολυτεχνικής Σχολής του Πανεπιστημίου Θεσσαλίας δεν υποδηλώνει αποδοχή των απόψεων του συγγραφέα (Ν. 5343/32 αρ. 202 παρ. 2).

## **Εγκρίθηκε από τα Μέλη της Τριμελούς Εξεταστικής Επιτροπής:**

Πρώτος Εξεταστής (Επιβλέπων) Δρ. Λάμπρος Κούτας  
Επίκουρος Καθηγητής, Τμήμα Πολιτικών Μηχανικών, Πανεπιστήμιο  
Θεσσαλίας

Δεύτερος Εξεταστής Δρ. Χρήστος Παπακωνσταντίνου  
Επίκουρος Καθηγητής, Τμήμα Πολιτικών Μηχανικών, Πανεπιστήμιο  
Θεσσαλίας

Τρίτος Εξεταστής Δρ. Ολυμπία Παναγούλη  
Αναπληρώτρια Καθηγήτρια, Τμήμα Πολιτικών Μηχανικών, Πανεπιστήμιο  
Θεσσαλίας

# ΑΝΑΠΤΥΞΗ ΛΟΓΙΣΜΙΚΟΥ Η/Υ ΓΙΑ ΤΗΝ ΑΝΑΛΥΣΗ ΚΑΙ ΤΟ ΣΧΕΔΙΑΣΜΟ ΚΑΤΑΣΚΕΥΩΝ ΟΠΛΙΣΜΕΝΟΥ ΣΚΥΡΟΔΕΜΑΤΟΣ

Θεόδωρος Αλεξανδρίδης

Πανεπιστήμιο Θεσσαλίας, Τμήμα Πολιτικών Μηχανικών, 2021

Επιβλέπων Καθηγητής: Λάμπρος Κούτας, Επίκουρος Καθηγητής, Τμήμα Πολιτικών  
Μηχανικών, Πανεπιστήμιο Θεσσαλίας

## Περίληψη

Στην παρούσα εργασία πραγματοποιήθηκε η ανάπτυξη ενός αυτόνομου λογισμικού στατικής ανάλυσης τρισδιάστατων πλαισιωτών κατασκευών, με δυνατότητα διαστασιολόγησης στοιχείων οπλισμένου σκυροδέματος (Ο.Σ.). Σημαντικό πλεονέκτημα του λογισμικού αποτελούν τα σύγχρονα γραφικά απεικόνισης και το διαδραστικό περιβάλλον για τον χρήστη. Στο παρόν τεύχος παρουσιάζεται αρχικά η μεθοδολογία που ακολουθήθηκε κατά τη διάρκεια ανάπτυξης του λογισμικού. Συγκεκριμένα αναλύεται ο τρόπος δημιουργίας του περιβάλλοντος του προγράμματος, του αλγορίθμου ανάλυσης της κατασκευής καθώς και του αλγορίθμου διαστασιολόγησης των στοιχείων Ο.Σ. Για την ανάλυση χρησιμοποιήθηκε η μητρική στατική μέθοδος και η θεωρία των πεπερασμένων στοιχείων. Ο αλγόριθμος της διαστασιολόγησης στηρίχθηκε στις διατάξεις των Ευρωπαϊκών Κανονισμών (Ευρωκώδικας 2 και 8), καθώς και σε σχετική βιβλιογραφία. Στη συνέχεια παρουσιάζεται το περιβάλλον του προγράμματος που αναπτύχθηκε, και έπειτα παρατίθεται ένα αριθμητικό παράδειγμα ανάλυσης και διαστασιολόγησης. Τέλος, τα αριθμητικά αποτελέσματα της ανάλυσης συγκρίνονται με αυτά ενός εμπορικά διαθέσιμου προγράμματος και επαληθεύεται η αξιοπιστία των αποτελεσμάτων του νέου λογισμικού.

**ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ:** Λογισμικό ανάλυσης κατασκευών, στατική ελαστική ανάλυση, οπλισμένο σκυρόδεμα, διαστασιολόγηση.

## Summary

This thesis studies the development of a stand-alone software for the static analysis of 3D framed structures and design of reinforced concrete (R.C.) frame structures, using modern graphics and an interactive user interface. The present thesis report initially presents the methodology followed for the development of the software. In specific, it presents the fundamentals of creating: a) the program environment, b) the algorithm for the structural analysis and c) the algorithm for the design of R.C. elements (beams and columns). The design algorithm was based on the provisions of the Eurocodes (EC2 and EC8), as well as on relevant literature. The program environment is then presented, followed by a numerical example of analysis and design. The numerical results of the analysis are compared with those of a commercial program and thus the reliability of the results of the new software is verified.

**KEY WORDS:** Analysis and design software, structural analysis, reinforced concrete, design

## Πίνακας περιεχομένων

Περίληψη.....	iv
Summary.....	iv
Πίνακας περιεχομένων.....	v
Κατάλογος Σχημάτων.....	vi
Κατάλογος Πινάκων.....	viii
Ευχαριστίες.....	ix
<b>Κεφάλαιο 1: Εισαγωγή.....</b>	<b>1</b>
1.1 Αντικείμενο της εργασίας, κίνητρο και υπόβαθρο.....	1
1.2 Βιβλιογραφική ανασκόπηση.....	1
1.3 Δομή Διπλωματικής Εργασίας.....	2
<b>Κεφάλαιο 2: Δημιουργία Λογισμικού.....</b>	<b>3</b>
2.1 Στρατηγική σχεδιασμού.....	3
2.1.1 Καθορισμός στόχων.....	3
2.1.2 Επιλογή κατάλληλων προγραμματιστικών εργαλείων.....	4
2.1.3 Γενικό διάγραμμα ροής προγράμματος.....	5
2.2 Στάδια λειτουργίας.....	7
2.2.1 Προεπεξεργαστής - μεθεπεξεργαστής.....	7
2.2.2 Επεξεργαστής.....	12
2.2.3 Σχεδιασμός οπλισμού στοιχείων οπλισμένου σκυροδέματος.....	24
<b>Κεφάλαιο 3: Παρουσίαση Λογισμικού Domisis R.C. ....</b>	<b>36</b>
3.1 Εγκατάσταση.....	36
3.2 Περιβάλλον του χρήστη.....	37
<b>Κεφάλαιο 4: Αριθμητικό παράδειγμα με χρήση του λογισμικού.....</b>	<b>52</b>
4.1 Προσομοίωμα.....	52
4.2 Αποτελέσματα ανάλυσης.....	54
4.3 Αποτελέσματα διαστασιολόγησης.....	58
<b>Κεφάλαιο 5: Σύνοψη και συμπεράσματα.....</b>	<b>61</b>
Προτάσεις για περαιτέρω εξέλιξη.....	61
<b>Βιβλιογραφία.....</b>	<b>63</b>
<b>ΠΑΡΑΡΤΗΜΑΤΑ.....</b>	<b>64</b>
Α. Σύγκριση αποτελεσμάτων του Κεφαλαίου 4 με το λογισμικό SAP 2000.....	64
Β. Κώδικας για τη διαστασιολόγηση των στοιχείων Ο.Σ. σε Visual Basic .NET.....	67

## Κατάλογος Σχημάτων

Σχήμα 2-1 Διάγραμμα ροής λογισμικού.....	6
Σχήμα 2-2. Η τελική λίστα με τις φόρμες που χρησιμοποιήθηκαν για το γραφικό περιβάλλον του προγράμματος.....	7
Σχήμα 2-3. Οι εγκατεστημένες βιβλιοθήκες στο visual studio, πάνω των γραφικών και κάτω της βάσης δεδομένων.....	8
Σχήμα 2-4. Γραφικό πλαίσιο απεικόνισης της βιβλιοθήκης OpenTK, διαθέσιμο στη γραμμή εργαλείων του visual studio.....	8
Σχήμα 2-5. Γραφικό περιβάλλον απεικόνισης κατασκευής. (αριστερά κάτοψη, δεξιά τρισδιάστατη προβολή). ....	9
Σχήμα 2-6. Οι πίνακες που αποτελούν τη βάση δεδομένων του λογισμικού.....	10
Σχήμα 2-7. Η μορφή του πίνακα φορτίσεων στη βάση δεδομένων. ....	11
Σχήμα 2-8. Γραμμές εργαλείων του κύριου παραθύρου του προγράμματος. ....	12
Σχήμα 2-9. Τοπικό σ.α. υποστυλώματος.....	15
Σχήμα 2-10. Τοπικό σ.α. δοκού.....	15
Σχήμα 2-11. Σημεία ελέγχου για τον υπολογισμό $M_{Ed}$ στα άκρα και στο άνοιγμα της δοκού. ....	27
Σχήμα 2-12. Λεπτομέρειες όπλισης δοκών (διαμήκεις οπλισμός), για την Κ.Π.Μ. (Κούτας, 2020).....	28
Σχήμα 2-13. Λεπτομέρειες όπλισης δοκών (διαμήκεις οπλισμός), για την Κ.Π.Υ. (Κούτας, 2020). ....	29
Σχήμα 2-14. Οι ροπές αντοχής σε κόμβο δοκού – υποστυλώματος, για τον ικανοτικό σχεδιασμό (Fardis, Carvalho, Fajfar, & Pecker, 2015). ....	30
Σχήμα 2-15. Διάγραμμα ροής αλγορίθμου διαστασιολόγησης δοκού .....	33
Σχήμα 2-16. Διάγραμμα ροής διαστασιολόγησης υποστυλωμάτων .....	34
Σχήμα 2-17. Διάγραμμα ροής τελικού βήματος διαστασιολόγησης υποστυλωμάτων .....	35
Σχήμα 3-1. Το περιβάλλον εγκατάστασης του λογισμικού Domisis R.C.....	36
Σχήμα 3-2. Συντόμευση του εκτελέσιμου αρχείου στην επιφάνεια εργασίας .....	37
Σχήμα 3-3. Αρχικό παράθυρο του προγράμματος.....	37
Σχήμα 3-4. Η επιλογή File και οι επιλογές διαχείρισης αρχείων. ....	38
Σχήμα 3-5. Η εργαλειοθήκη γρήγορης πρόσβασης λειτουργιών διαχείρισης αρχείων.....	38
Σχήμα 3-6. Γραμμή εργαλείων στο πάνω μέρος του παραθύρου.....	38
Σχήμα 3-7. Καθορισμός κανάβου σχεδίασης. ....	39
Σχήμα 3-8. Φόρμες εισαγωγής και επεξεργασίας διαθέσιμων υλικών. ....	39
Σχήμα 3-9. Φόρμες εισαγωγής και επεξεργασίας διαθέσιμων διατομών.....	40
Σχήμα 3-10. Γραμμή εργαλείων σχεδίασης του φορέα του προσομοιώματος.....	40
Σχήμα 3-11. Σχεδιασμός της γεωμετρίας του φορέα. ....	41
Σχήμα 3-12. Δέσμευση κόμβων κάτοψης με διαφραγματική λειτουργία πλάκας. ....	41
Σχήμα 3-13. Εισαγωγή στηρίξεων.....	42
Σχήμα 3-14. Επιλογές προβολής γραφικών στην καρτέλα View.....	42
Σχήμα 3-15. Τρισδιάστατη προβολή της κατασκευής. ....	42
Σχήμα 3-16. Καρτέλα πληροφοριών επιλεγμένου στοιχείου. ....	43

Σχήμα 3-17. Καρτέλα των προφίλ φόρτισης.....	43
Σχήμα 3-18. Επιβολή φόρτισης στην κατασκευή μέσα από την καρτέλα Assign Loads.....	44
Σχήμα 3-19. Καρτέλα συνδυασμού φορτίσεων.....	45
Σχήμα 3-20. Φόρμα ανάλυσης. ....	45
Σχήμα 3-21. Γραμμή εργαλείων προβολής αποτελεσμάτων ανάλυσης. ....	46
Σχήμα 3-22. Διαγράμματα εντατικών μεγεθών.....	46
Σχήμα 3-23. Μετακινήσεις κόμβων. ....	47
Σχήμα 3-24. Μετακινήσεις και στροφές επιλεγμένου κόμβου.....	47
Σχήμα 3-25. Απόσπασμα του μητρώου δυσκαμψίας της κατασκευής, διαθέσιμο από τους πίνακες ανάλυσης. ....	48
Σχήμα 3-26. Πεδία τιμών για τα δεδομένα του αλγορίθμου διαστασιολόγησης. ....	49
Σχήμα 3-27. Πίνακας αποτελεσμάτων διαστασιολόγησης υποστυλώματος.....	49
Σχήμα 3-28. Πίνακας αποτελεσμάτων διαστασιολόγησης δοκού.....	50
Σχήμα 3-29. Ικανοτικός έλεγχος στον πίνακα αποτελεσμάτων της διαστασιολόγησης. ....	50
Σχήμα 3-30. Γραφική απεικόνιση του τοποθετούμενου οπλισμού. ....	51
Σχήμα 4-1. Διάταξη του φορέα του παραδείγματος σε τρισδιάστατη προβολή. ....	52
Σχήμα 4-2. Μόνιμα φορτία κατανεμημένα στις δοκούς. ....	53
Σχήμα 4-3. Κινητά φορτία κατανεμημένα στις δοκούς.....	53
Σχήμα 4-4. Σεισμικά φορτία στη διεύθυνση X, με τη μορφή συγκεντρωμένων φορτίων. ....	54
Σχήμα 4-5. Μετακινήσεις στον τελευταίο όροφο για τον συνδυασμό 1.35G + 1.5Q + Ex. ....	54
Σχήμα 4-6. Διάγραμμα ροπών για τον συνδυασμό 1.35G + 1.5Q + Ex (μονάδες σε kNm).....	55
Σχήμα 4-7. Διάγραμμα τεμνουσών για τον συνδυασμό 1.35G + 1.5Q + Ex (μονάδες σε kN).....	55
Σχήμα 4-8. Διάγραμμα αξονικών δυνάμεων για τον συνδυασμό 1.35G + 1.5Q + Ex (μονάδες σε kN).....	56
Σχήμα 4-9. Διάγραμμα ροπών στρέψης για τον συνδυασμό 1.35G + 1.5Q + Ex (μονάδες σε kNm).....	56
Σχήμα 4-10. Διάγραμμα περιβάλλουσας των ροπών για τους συνδυασμούς 1.35G + 1.5Q + Ex και 1.35G + 1.5Q - Ex (μονάδες σε kNm). ....	57
Σχήμα 4-11. Το προσομοίωμα του παραδείγματος στο λογισμικό SAP2000.....	57
Σχήμα 4-12. Διατάξεις οπλισμού της κατασκευής στο επίπεδο X-Z για y = -4 m. ....	58
Σχήμα 4-13. Διατάξεις οπλισμού της κατασκευής στο επίπεδο X-Z για y = 0 m.....	59
Σχήμα 4-14. Διάγραμμα αξονικών δυνάμεων στο επίπεδο Y = 0 m, για τον συνδυασμό 1.35G + 1.5Q + Ex (μονάδες σε kN).....	60
Σχήμα 4-15. Διάγραμμα αξονικών δυνάμεων στο επίπεδο Y = -4 m, για τον συνδυασμό 1.35G + 1.5Q + Ex (μονάδες σε kN).....	60
Σχήμα Π-0-1. Διαγράμματα ροπών στο SAP 2000.....	64
Σχήμα Π-0-2. Διαγράμματα ροπών στο Domisis R.C.....	64
Σχήμα Π-0-3. Διαγράμματα ροπών στο SAP 2000.....	65
Σχήμα Π-0-4. Διαγράμματα ροπών στο Domisis R.C.....	65
Σχήμα Π-0-5. Μετακινήσεις κόμβου της οροφής στο SAP 2000. ....	66
Σχήμα Π-0-6. Μετακινήσεις κόμβου της οροφής στο Domisis R.C.....	66

## Κατάλογος Πινάκων

Πίνακας 2-1. Μητρώο δυσκαμψίας γραμμικού στοιχείου στις 3 διαστάσεις. ....	13
Πίνακας 2-2. Μητρώο περιστροφής κατακόρυφων στοιχείων.....	16
Πίνακας 2-3. Μητρώο περιστροφής οριζόντιων στοιχείων. ....	16
Πίνακας 2-4. Μητρώο αντιστοίχισης στοιχείου, στην περίπτωση που οι β.ε. στοιχείου και κατασκευής ταυτίζονται. ....	17
Πίνακας 2-5. Διαδικασία υπολογισμού των μητρώων αντιστοίχισης. ....	17
Πίνακας 2-6. Οι τιμές που προσδίδονται στο μητρώο δυσκαμψίας των εικονικών δοκών, για την προσομοίωση της διαφραγματικής λειτουργίας (kN, m) .....	18
Πίνακας 2-7. Διαδικασία μόρφωσης μητρώου μετάθεσης.....	19
Πίνακας 2-8. Διαδικασία υπολογισμού μητρώου δυσκαμψίας της κατασκευής. ....	20
Πίνακας 2-9. Υπολογισμός μητρώου δράσεων παγίωσης.....	22
Πίνακας 2-10. Αλγόριθμος υπολογισμού εντατικών μεγεθών στοιχείου.....	23
Πίνακας 2-11. Διατάξεις οπλισμού υποστυλωμάτων κατά τον EC2. (Κούτας, 2020).....	31



## Ευχαριστίες

Πρώτα απ' όλα, θέλω να ευχαριστήσω τον επιβλέποντα της διπλωματικής εργασίας μου, Επίκουρο Καθηγητή κ. Λάμπρο Κούτα, που με εμπιστεύτηκε δίνοντάς μου αυτό το θέμα της εργασίας αλλά και για την πολύτιμη βοήθεια και τις υποδείξεις που μου έκανε όλο αυτό το διάστημα. Επίσης, είμαι ευγνώμων και στους υπόλοιπους καθηγητές του τμήματος του δομοστατικού τομέα για το υψηλό επίπεδο σπουδών που παρέχουν σε εμάς τους φοιτητές. Τέλος, ευχαριστώ την οικογένειά μου για τη στήριξη της όλα αυτά τα χρόνια των σπουδών μου.

Αλεξανδρίδης Θεόδωρος

## Κεφάλαιο 1: Εισαγωγή

### 1.1 Αντικείμενο της εργασίας, κίνητρο και υπόβαθρο

Η εκτεταμένη χρήση προγραμμάτων λογισμικού στη σύγχρονη επιστήμη του πολιτικού μηχανικού, και οι τεράστιες δυνατότητες που μπορούν αυτά να προσφέρουν στο επάγγελμα μας αποτέλεσαν το κίνητρο ενασχόλησης με το θέμα αυτής της διπλωματικής εργασίας. Ακόμη, το πολυσύνθετο έργο ανάπτυξης ενός λογισμικού ανάλυσης και διαστασιολόγησης κατασκευών, απαιτεί τον συνδυασμό ενός ευρύτατου συνόλου αντικειμένων της επιστήμης του δομοστατικού μηχανικού (όπως Στατική, Τεχνική Μηχανική, Πεπερασμένα Στοιχεία, Οπλισμένο Σκυρόδεμα, Γραμμική Άλγεβρα, Αριθμητική Ανάλυση, Προγραμματισμός Η/Υ), γεγονός που προάγει τη βαθύτερη κατανόηση της επιστήμης του μηχανικού.

Κύριο αντικείμενο της εργασίας αυτής αποτελεί η ανάπτυξη ενός λογισμικού για την ελαστική στατική ανάλυση πλαίσιακών κατασκευών. Για την ανάλυση η μέθοδος που ακολουθείται είναι αυτή της μητρικής στατικής. Για λόγους απλοποίησης, αγνοείται η αλληλεπίδραση εδάφους – κατασκευής και στις θέσεις των στηρίξεων θεωρείται πάκτωση. Επιπρόσθετα, η εργασία ασχολείται και με την ανάπτυξη κατάλληλου αλγορίθμου για τον σχεδιασμό του οπλισμού των δοκών και των υποστυλωμάτων οπλισμένου σκυροδέματος. Συγκεκριμένα για τα στοιχεία υπολογίζεται μόνο ο διαμήκειος οπλισμός κάμψης για την οριακή κατάσταση αστοχίας Ο.Κ.Α. Ένα επίσης σημαντικό κομμάτι της εργασίας αυτής είναι η δημιουργία ενός γραφικού περιβάλλοντος για την εισαγωγή δεδομένων στο λογισμικό καθώς επίσης και για την παρουσίαση των αποτελεσμάτων.

Μπορεί η συνεισφορά της διπλωματικής αυτής να είναι ελάχιστος ερευνητικής σημασίας, δίνει όμως το κίνητρο σε άλλους φοιτητές να ασχοληθούν με τον προγραμματισμό και τη δημιουργία παρόμοιων εφαρμογών Η/Υ, ενώ ταυτόχρονα αποτελεί έναν σύντομο οδηγό για αυτό το σκοπό. Τέλος, αποτελεί ένα εργαλείο για περαιτέρω έρευνα και ενασχόληση με σημαντικότερα ζητήματα της επιστήμης του δομοστατικού μηχανικού, όπως τη δυναμική ανάλυση των κατασκευών, την ανελαστική συμπεριφορά τους, τη βελτιστοποίηση του οπλισμού του οπλισμένου σκυροδέματος, την εφαρμογή πολύπλοκων πεπερασμένων στοιχείων κλπ.

### 1.2 Βιβλιογραφική ανασκόπηση

Η βιβλιογραφία που χρησιμοποιήθηκε για τη μητρική στατική ανάλυση συνοψίζεται στις ιδίχειρες σημειώσεις του μαθήματος *Στατική ΙΙΙ* (Παναγούλη, 2018), όπου και στηρίχθηκε ο

κορμός του αλγορίθμου ανάλυσης, καθώς και στις πανεπιστημιακές σημειώσεις του μαθήματος *Πεπερασμένα Στοιχεία* (Τζάρος, 2019). Σημειώνεται ότι στη μόρφωση των μητρώων δυσκαμψίας και μητρώων περιστροφής συνέβαλλε το σύγγραμμα *Πεπερασμένα Στοιχεία στην Ανάλυση Κατασκευών* (Προβατίδης, 2016).

Για τη δημιουργία του αλγορίθμου διαστασιολόγησης, ακολουθήθηκε η διαδικασία που προτείνεται στα Κεφάλαια 5 και 7 του συγγράμματος *Seismic Design of Concrete buildings to Eurocode 8* (Fardis, Carvalho, Fajfar, & Pecker, 2015). Ακόμα σημαντική ήταν η συνεισφορά των πανεπιστημιακών σημειώσεων του μαθήματος *Οπλισμένο Σκυρόδεμα I* (Κούτας, 2020). Τέλος, ακολουθήθηκαν οι διατάξεις του Ευρωκώδικα 2 (EN 1992-1-1, 2004) και Ευρωκώδικα 8 (EN 1998-1, 2004).

### 1.3 Δομή Διπλωματικής Εργασίας

Στο Κεφάλαιο 2 παρουσιάζονται τα βήματα και η μεθοδολογία που ακολουθήθηκε για την ανάπτυξη του λογισμικού, δίνοντας ιδιαίτερη έμφαση στους αλγόριθμους ανάλυσης και σχεδιασμού. Όσον αφορά το προγραμματιστικό κομμάτι, στο πλαίσιο της εργασίας αυτής αποφασίστηκε να μην επεκταθεί σε βαθμό που θα ξεφύγει από τα όρια της επιστήμης του δομοστατικού μηχανικού, και διατηρήθηκε στην απολύτως απαραίτητη περιγραφή.

Στο Κεφάλαιο 3 πραγματοποιείται η παρουσίαση του αναπτυχθέντος λογισμικού Domisis R.C. Συγκεκριμένα, παρουσιάζεται το γραφικό περιβάλλον του προγράμματος αλλά και ο τρόπος λειτουργίας του.

Στο Κεφάλαιο 4 παρουσιάζεται μία αριθμητική εφαρμογή με τη χρήση του λογισμικού, όπου επιλύεται ένας τρισδιάστατος φορέας Ο.Σ. και διαστασιολογούνται σε κάμψη οι δοκοί και τα υποστυλώματά του.

Στο Κεφάλαιο 5 παρουσιάζεται η σύνοψη και τα συμπεράσματα της παρούσας εργασίας, καθώς και γίνονται προτάσεις για περαιτέρω εξέλιξη του θέματος.

Τέλος, ακολουθούν δύο παραρτήματα: το Παράρτημα Α στο οποίο συγκεντρώνεται η σύγκριση των αποτελεσμάτων της εφαρμογής του Κεφαλαίου 4 με αυτά ενός εμπορικά διαθέσιμου λογισμικού, και το Παράρτημα Β στο οποίο παρατίθεται ο αλγόριθμος διαστασιολόγησης των στοιχείων Ο.Σ. που χρησιμοποιήθηκε για το λογισμικό Domisis R.C.

## Κεφάλαιο 2: Δημιουργία Λογισμικού

### 2.1 Στρατηγική σχεδιασμού

#### 2.1.1 Καθορισμός στόχων

Το πρώτο στάδιο ανάπτυξης ενός λογισμικού είναι ο καθορισμός των δυνατοτήτων που στοχεύουμε να διαθέτει. Το πρώιμο αυτό στάδιο του αρχικού σχεδιασμού έχει ιδιαίτερη σημασία και κρίθηκε απαραίτητο να σημειωθεί εδώ, καθώς η ιδιαίτερα χρονοβόρα διαδικασία του προγραμματισμού καθιστά τη διαχείριση του χρόνου πρώτιστης σημασίας και οποιαδήποτε απόκλιση από τον αρχικό σχεδιασμό μπορεί να οδηγήσει σε πολλαπλάσια δαπάνη χρόνου και κόπου. Άλλωστε η συγκροτημένη στρατηγική σχεδιασμού μεταφράζεται και σε συγκροτημένο προγραμματισμό, διευκολύνοντας έτσι σημαντικά την ανάγνωση και επέκταση του υπάρχοντος κώδικα. Κατά την ανάπτυξη του λογισμικού αυτού, κατέστη σαφής η σημαντικότητα του σταδίου αυτού.

Οι κύριοι στόχοι που τέθηκαν σε αυτό το πρώτο στάδιο ήταν:

- η ανάπτυξη ενός αυτόνομου λογισμικού περιβάλλοντος (stand-alone software), χωρίς δηλαδή την αξιοποίηση εξωτερικών προγραμμάτων ή βιβλιοθηκών, εκτελέσιμο από έναν κοινό Η/Υ
- η ύπαρξη σύγχρονου και διαδραστικού γραφικού περιβάλλοντος (graphical user interface), ώστε να είναι φιλικό προς τον χρήστη
- η σαφής διάκριση των σταδίων χρήσης του λογισμικού: (α) η εισαγωγή δεδομένων, (β) η στατική ανάλυση του φορέα και (γ) η διαστασιολόγηση των δομικών στοιχείων του φορέα υπό την προϋπόθεση ότι είναι από Ο.Σ.

Δευτερεύοντες στόχους αποτέλεσαν:

- η δυνατότητα αποθήκευσης των δεδομένων της κατασκευής σε ξεχωριστό αρχείο δεδομένων (database file), ώστε να μπορούν να διαβαστούν από το λογισμικό και να φορτωθούν ακόμα και μετά τον τερματισμό του λογισμικού
- η κατά το δυνατόν ποιοτικότερη γραφική παρουσίαση των αποτελεσμάτων της ανάλυσης και της διαστασιολόγησης, ώστε να είναι εύκολα αναγνώσιμα από τον χρήστη

### 2.1.2 Επιλογή κατάλληλων προγραμματιστικών εργαλείων

Αφού έχουν καθοριστεί οι γενικοί στόχοι ενός προγράμματος, μπορούμε να επιλέξουμε το κατάλληλο προγραμματιστικό πλαίσιο στο οποίο αυτό θα αναπτυχθεί. Παρακάτω παρατίθενται τα βασικά εργαλεία με τα οποία αναπτύχθηκε το λογισμικό της παρούσας εργασίας.

#### ➤ Περιβάλλον ανάπτυξης

Για τις ανάγκες του προγράμματος ως περιβάλλον ανάπτυξης επιλέχθηκε το Visual Studio της εταιρείας Microsoft. Οι δυνατότητες του εν λόγω περιβάλλοντος καθιστούν την ανάπτυξη μιας αυτόνομης εφαρμογής με γραφικό περιβάλλον αρκετά φιλική για έναν νέο προγραμματιστή. Στον προγραμματιστή παρέχεται πληθώρα προγραμματιστικών γλωσσών καθώς επίσης και έτοιμων γραφικών υποσυστημάτων με πλήθος εργαλείων. Ο συνδυασμός προγραμματιστικών γλωσσών όπως η Visual Basic και η C#, οι οποίες αποτελούν αντικειμενοστραφείς γλώσσες (δηλαδή οργανώνονται και ανταποκρίνονται με βάση γεγονότα, όπως το πάτημα ενός κουμπιού), με τα γραφικά μοντέλα όπως windows forms (winforms) ή windows presentation foundation (wpf), δίνει τη δυνατότητα στον προγραμματιστή να σχεδιάσει και να προγραμματίσει ένα σύγχρονο διαδραστικό γραφικό περιβάλλον με σχετική ευκολία και οργάνωση.

#### ➤ Γλώσσα προγραμματισμού

Ως γλώσσα προγραμματισμού επιλέχθηκε η Visual Basic, καθότι διδαχθείσα στο πλαίσιο του προπτυχιακού προγράμματος σπουδών του Τμήματος Πολιτικών Μηχανικών του Πανεπιστημίου Θεσσαλίας κρίθηκε ως η καλύτερη επιλογή. Η χρήση της έγινε για τη συγγραφή ολόκληρου του κώδικα του λογισμικού εξαιρουμένου ενός μέρους διαχείρισης της βάσης δεδομένων όπου απαιτείται γλώσσα ερωταποκρίσεων σε SQL (queries). Παρότι η χρήση μιας αντικειμενοστραφούς γλώσσας προγραμματισμού ενδείκνυται για τον προγραμματισμό του γραφικού περιβάλλοντος και της γενικής του ροής, στο κομμάτι της ανάλυσης, όταν απαιτούνται χρονοβόρες υπολογιστικές πράξεις, αυτή υστερεί σε υπολογιστική ταχύτητα σε σχέση με μια διαδικαστική γλώσσα (procedural language) κατάλληλη για υπολογιστικές πράξεις όπως η C, Fortran κ.α.

#### ➤ Γραφικό περιβάλλον

Το γραφικό περιβάλλον του χρήστη χωρίζεται σε δύο μέρη ανάλογα με τη λειτουργία τους. Στο πρώτο μέρος μπορεί κανείς να διακρίνει το περιβάλλον που καθορίζει τη ροή του προγράμματος και δίνει τη δυνατότητα στο χρήστη να αλληλοεπιδρά με αυτό μέσω ενός

παραθύρου γραφικών (Graphical User Interface ή GUI). Συνήθως οργανώνεται από υποφόρμες καθεμία από τις οποίες περιέχει διαφορετικές λειτουργίες, διευκολύνοντας έτσι την οργάνωση και τη ροή του προγράμματος για τον χρήστη. Το κομμάτι αυτό του κώδικα εκτελείται συνήθως από τον επεξεργαστή του υπολογιστή, καθότι οι γραφικές απαιτήσεις είναι αρκετά μειωμένες. Το visual studio δίνει την επιλογή δύο τέτοιων περιβαλλόντων, του winforms και του wpf. Για τις ανάγκες της παρούσας εργασίας επιλέχθηκε το περιβάλλον του winforms.

Ως δεύτερο κομμάτι γραφικών, αναφέρεται αυτό το οποίο διαχειρίζεται τις απαιτήσεις για τρισδιάστατο περιβάλλον απεικόνισης της κατασκευής, των διαγραμμάτων κλπ. με ικανοποιητική ανταπόκριση στο χειρισμό του χρήστη. Η απαιτητική αυτή διαδικασία θα πρέπει να αναλαμβάνεται εξολοκλήρου από την κάρτα γραφικών του υπολογιστή, η οποία είναι σχεδιασμένη άλλωστε για αυτό το σκοπό. Τέτοια γραφικά περιβάλλοντα εκτελούνται σχεδόν αποκλειστικά με χρήση εξωτερικών βιβλιοθηκών γραφικών (Computer Graphics API), οι οποίες εκμεταλλεύονται τις κάρτες γραφικών, με τις πιο γνωστές να είναι οι DirectX, OpenGL, Vulkan. Για την ανάπτυξη των σύνθετων γραφικών αναγκών του λογισμικού επιλέχθηκε η βιβλιοθήκη OpenGL (<https://www.opengl.org/>), για την οποία υπάρχει πλήθος υλικού για μελέτη στο διαδίκτυο, και πιο συγκεκριμένα του πακέτου OpenTK (<https://opentk.net/>) που δίνει πρόσβαση στις εντολές της OpenGL μέσω της γλώσσας Visual Basic.

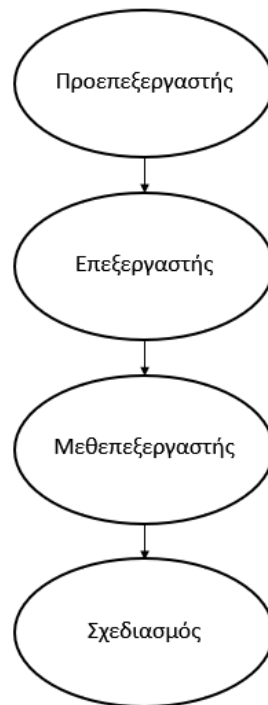
#### ➤ Βιβλιοθήκη δεδομένων

Για τις ανάγκες αποθήκευσης και έπειτα ανάγνωσης δεδομένων, απαιτείται η χρήση κάποιας βιβλιοθήκης (database). Μία από τις πιο γνωστές ελεύθερης για χρήση αποτελεί η SQLite (<https://www.sqlite.org/>), η οποία και επιλέχθηκε.

### 2.1.3 Γενικό διάγραμμα ροής προγράμματος

Ο διαχωρισμός του λογισμικού σε υποπρογράμματα για την καλύτερη οργάνωση γίνεται μέσω της δημιουργία ενός γενικού διαγράμματος ροής (Σχήμα 2-1) στο οποίο απεικονίζεται η αρχιτεκτονική του λογισμικού, χωρίζοντάς το έτσι στα λειτουργικά μέλη που το αποτελούν.

Όπως στα περισσότερα σύγχρονα λογισμικά προγράμματα ανάλυσης και διαστασιολόγησης κατασκευών, έτσι και εδώ γίνεται η διάκριση των εξής τεσσάρων σταδίων λειτουργίας:



Σχήμα 2-1 Διάγραμμα ροής λογισμικού.

➤ Προεπεξεργαστής (preprocessor)

Το πρώτο μέλος ενός προγράμματος κατά το οποίο δίνεται η δυνατότητα εισαγωγής, από τον χρήστη ή κάποιο τρίτο πρόγραμμα (third party program), των απαραίτητων για την ανάλυση δεδομένων που αφορούν τη γεωμετρία της κατασκευής, τα υλικά και τις ιδιότητες τους καθώς και τον προσδιορισμό του τρόπου ανάλυσης. Στα πρώτα λογισμικά που αναπτύχθηκαν στα τέλη του 20<sup>ου</sup> αιώνα, το μέλος αυτό δεν ήταν εμφανές στον χρήστη, καθώς απουσίαζε το φιλικό ως προς τον χρήστη γραφικό περιβάλλον, λόγω της περιορισμένης υπολογιστικής δυνατότητας των τότε ηλεκτρονικών υπολογιστών, και η εισαγωγή των δεδομένων γινόταν προγραμματιστικά. Ακόμα και τότε ωστόσο, ο ρόλος του προεπεξεργαστή ήταν και είναι η ταξινόηση των δεδομένων με τρόπο τέτοιο ώστε να είναι προσβάσιμα για τις διαδικασίες του επεξεργαστή.

➤ Επεξεργαστής (processor)

Αποτελεί τον κύριο κορμό του λογισμικού. Τροφοδοτείται από την έξοδο δεδομένων του προεπεξεργαστή, τα επεξεργάζεται ανάλογα με την επιθυμητή μέθοδο και εν συνεχεία παράγει τα αποτελέσματα που θα παρουσιαστούν στον χρήστη με τη βοήθεια του μεθεπεξεργαστή. Στοιχειοθετείται από την προγραμματιστική προσέγγιση των γνωστών από την επιστήμη του

δομοστατικού μηχανικού μεθοδολογιών ανάλυσης κατασκευών. Η αποτελεσματικότητά του κρίνεται από την τελική ακρίβεια καθώς και τον χρόνο επίλυσης.

➤ Μεθεπεξεργαστής (post processor)

Παρουσιάζει στον χρήστη τα αποτελέσματα της ανάλυσης ή της διαστασιολόγησης με γραφικό ή άλλο τρόπο.

➤ Διαστασιολόγηση (design)

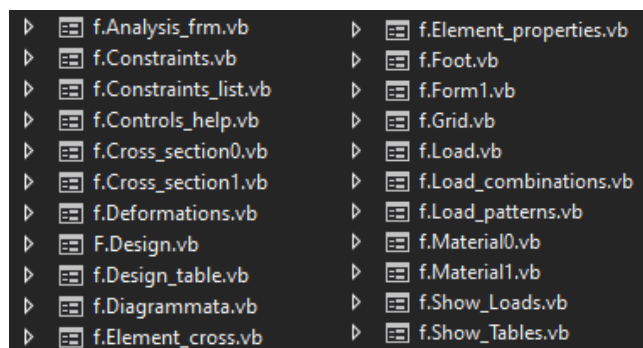
Συνήθως αποτελεί ξεχωριστό κομμάτι από την ανάλυση, καθώς ως δεδομένα απαιτεί τον καθορισμό μεταβλητών που επιλέγει ο μηχανικός με την κρίση του, αφού μελετήσει τα αποτελέσματα της ανάλυσης.

## 2.2 Στάδια λειτουργίας

Στο σημείο αυτό γίνεται αναλυτική παρουσίαση των μεθόδων σχεδιασμού των παραπάνω λειτουργικών σταδίων του προγράμματος.

### 2.2.1 Προεπεξεργαστής - μεθεπεξεργαστής

Η διαδικασία που ακολουθήθηκε για το λογισμικό της εργασίας αυτής ήταν η δημιουργία ενός κύριου παραθύρου/φόρμας (Parent form) καθώς και μίας σειράς υποφορμών (child forms), στα οποία θα συστεγάζεται ολόκληρο το γραφικό περιβάλλον αλληλεπίδρασης με τον χρήστη. Όλες οι φόρμες που δημιουργήθηκαν παρατίθενται στην παρακάτω λίστα (Σχήμα 2-2) και παρουσιάζονται αναλυτικά στο Κεφάλαιο 3.



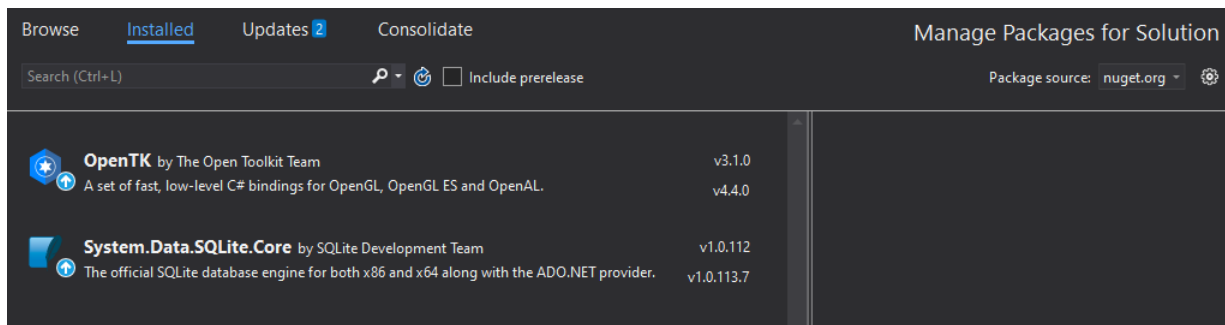
Σχήμα 2-2. Η τελική λίστα με τις φόρμες που χρησιμοποιήθηκαν για το γραφικό περιβάλλον του προγράμματος.

Οι φόρμες αυτές συνδέονται μεταξύ τους με γεγονότα (events), όπως το πάτημα ενός κουμπιού από τον χρήστη, η ολοκλήρωση της ανάλυσης κλπ., τα οποία καθορίζονται κατά το στάδιο του προγραμματισμού. Για τη μόρφωσή τους αξιοποιήθηκαν μία σειρά από εργαλεία που



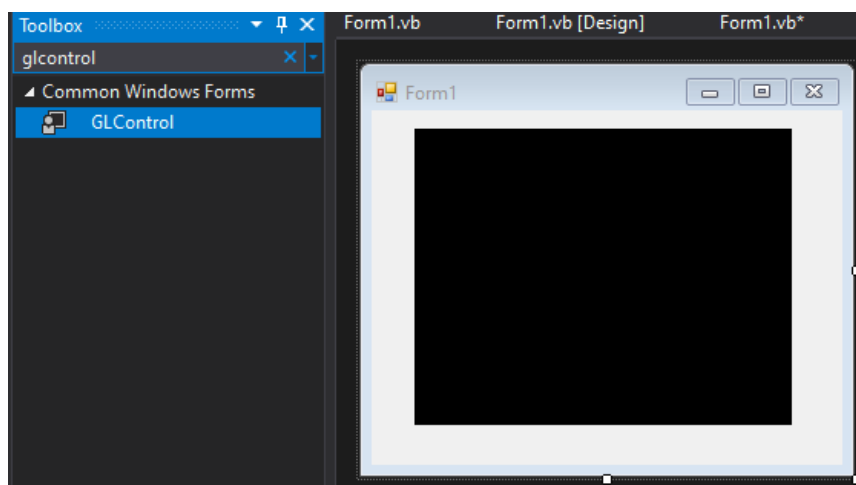
παρέχει η εργαλειοθήκη του visual studio (π.χ. textbox, label, toolstrip, panel, combobox, checkbox, radiobutton, progressbar)

Για την αξιοποίηση της βιβλιοθήκης βάσεων δεδομένων SQLite, καθώς και της βιβλιοθήκης γραφικών OpenGL μέσω του περιβάλλοντος visual studio, έγιναν οι προσθήκες των παρακάτω δύο πακέτων, όπως φαίνεται στο Σχήμα 2-3, αξιοποιώντας το εργαλείο NuGet Package manager.



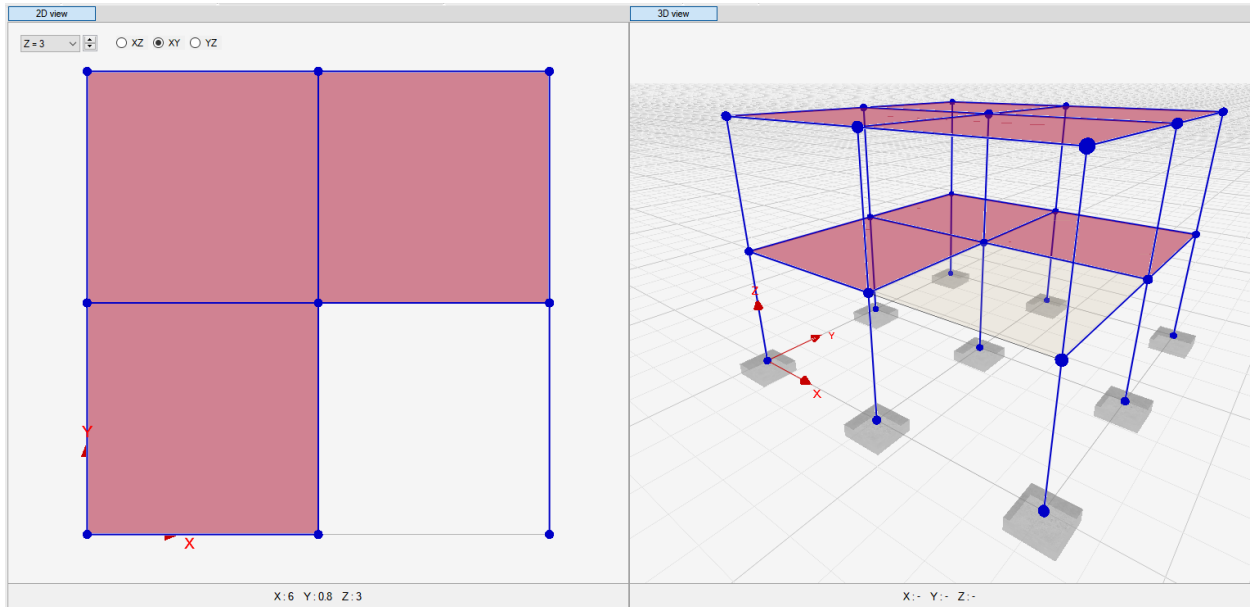
Σχήμα 2-3. Οι εγκατεστημένες βιβλιοθήκες στο visual studio, πάνω των γραφικών και κάτω της βάσης δεδομένων.

Επιπρόσθετα, για την παρουσίαση των γραφικών που θα σχεδιάζονται κατά τη λειτουργία του προγράμματος, προστέθηκε στις φόρμες οι οποίες το απαιτούν το πάνελ GLControl, το οποίο γίνεται διαθέσιμο στην εργαλειοθήκη του visual studio μετά την εγκατάσταση της παραπάνω βιβλιοθήκης γραφικών. Στο πάνελ αυτό θα απεικονίζεται το τρισδιάστατο ή δισδιάστατο περιβάλλον γραφικών, το οποίο περιέχει όλα τα σχέδια που θα του ορίζουμε προγραμματιστικά.



Σχήμα 2-4. Γραφικό πλαίσιο απεικόνισης της βιβλιοθήκης OpenGL, διαθέσιμο στη γραμμή εργαλείων του visual studio.

Για το παρόν λογισμικό χρησιμοποιήθηκαν δύο τέτοια πάνελ στην κύρια φόρμα, ένα για την τρισδιάστατη απεικόνιση της κατασκευής και ένα για τη δισδιάστατη απεικόνιση στα επίπεδα X-Y, X-Z, Y-Z (Σχήμα 2-5).



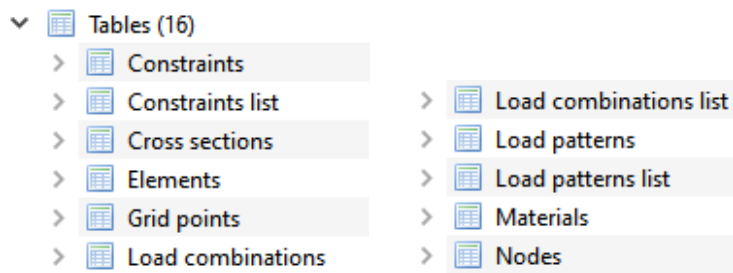
Σχήμα 2-5. Γραφικό περιβάλλον απεικόνισης κατασκευής. (αριστερά κάτοψη, δεξιά τρισδιάστατη προβολή).

Ολόκληρο το γραφικό περιβάλλον που σχεδιάζεται στα δύο αυτά πάνελ, έγινε με την εξής μεθοδολογία:

1. Καθορισμός του τρόπου προβολής στην πλευρά του θεατή, τρισδιάστατη ή δισδιάστατη απεικόνιση, προοπτική, σημείο θέσης του θεατή κλπ.
2. Μετατροπή του συστήματος συντεταγμένων (μετακίνηση, περιστροφή, σμίκρυνση ή μεγέθυνση), δίνοντας έτσι τη δυνατότητα στον χρήστη να κινείται μέσα στο περιβάλλον.
3. Σχεδιασμός των αντικειμένων που σκοπεύουμε να απεικονίσουμε μόνο με τη χρήση γραμμών (Lines) και τρίπλευρων και τετράπλευρων επιφανειών (Triangles και Quads). Τα στοιχεία αυτά σχεδιάζονται στο σύστημα συντεταγμένων διανυσματικά, δηλαδή καθορίζοντας τις συντεταγμένες των κόμβων που τα αποτελούν.
4. Για οποιαδήποτε αλλαγή στην απεικόνιση, ολόκληρο το περιβάλλον ξανασχεδιάζεται από την αρχή.

Για τον σχεδιασμό μιας ομάδας στοιχείων, όπως για παράδειγμα των δοκών και των υποστυλωμάτων, καθώς επίσης και για επαναλαμβανόμενες λειτουργίες σχετικές με τη γραφική απεικόνιση, δημιουργήθηκε μία οργανωμένη βιβλιοθήκη γραφικών σε δομοστοιχείο (module), αποτελούμενη από συναρτήσεις (functions) και διαδικασίες (subs), οι οποίες καλούνται κατά το στάδιο σχεδιασμού των γραφικών.

Αναφορικά με τη δημιουργία της βάσης δεδομένων, αυτή έγινε προγραμματιστικά, οργανώνοντας την σε ξεχωριστούς πίνακες ανάλογα με το είδος των δεδομένων. Στον κώδικα δημιουργείται αντίγραφο του κάθε πίνακα σε στοιχεία datatables, τον οποίων τα δεδομένα είναι άμεσα επεξεργάσιμα με εντολές της visual basic. Οι πίνακες αυτοί περιέχουν δεδομένα για τα υλικά, τις διατομές των μελών, τα γραμμικά στοιχεία και τους κόμβους τους (δοκοί και υποστυλώματα), τις φορτίσεις που ασκούνται στον φορέα και τους συνδυασμούς φορτίσεων, το σύστημα του κανάβου σχεδίασης και τέλος των διαφραγμάτων των κόμβων. Στο Σχήμα 2-6. παρουσιάζονται οι πίνακες που χρησιμοποιήθηκαν για τη βάση δεδομένων, ενώ στο Σχήμα 2-7 φαίνεται η διάταξη του πίνακα φορτίσεων. Κατά την αποθήκευση μιας υπάρχουσας εργασίας σε αρχείο μορφής βάσης δεδομένων (database file), τα δεδομένα αλλά και η μορφή των πινάκων datatables, αντιγράφεται στους πίνακες της βάσης δεδομένων.



Σχήμα 2-6. Οι πίνακες που αποτελούν τη βάση δεδομένων του λογισμικού.

Συγκεκριμένα οι πίνακες του Σχήματος 2-6 περιέχουν:

- Constraints: τα διαφράγματα με τα οποία δεσμεύονται οι κόμβοι της κατασκευής μαζί με τους κόμβους τους οποίους δεσμεύουν
- Cross sections: τη λίστα με τις διαθέσιμες διατομές που έχει καθορίσει ο χρήστης, με τις διαστάσεις τους και το υλικό που χρησιμοποιούν
- Elements: όλα τα γραμμικά στοιχεία της κατασκευής μαζί με τα χαρακτηριστικά τους όπως διατομή, κόμβοι αρχής και τέλους

- Grid points: όλα τα σημεία τομής των γραμμών του κανάβου σχεδίασης
- Load combinations: τη λίστα με τους συνδυασμούς φόρτισης και τα προφίλ φόρτισης που συμμετέχουν στον κάθε συνδυασμό μαζί με τον συντελεστή συμμετοχής τους
- Load patterns: τα διαθέσιμα προφίλ φόρτισης που έχει ορίσει ο χρήστης καθώς και τα φορτία που ασκούνται σε κάθε γραμμικό στοιχείο και κόμβο
- Materials: τη λίστα υλικών και των ιδιοτήτων τους που καθορίζει ο χρήστης
- Nodes: όλους τους κόμβους της κατασκευής μαζί με τις συντεταγμένες και τις συνθήκες στήριξής τους

Table: Load patterns

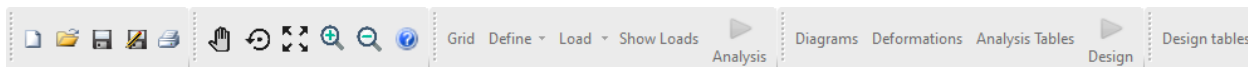
pattern	eidos	ID	X	Y	Z	MX	MY	MZ
Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	0 line	1	0	0	100	0	0	0
2	0 line	5	0	0	100	0	0	0
3	0 line	6	0	0	100	0	0	0
4	0 line	8	0	0	100	0	0	0
5	1 node	1	-70	0	0	0	0	0
6	1 node	6	-20	0	0	0	0	0

Σχήμα 2-7. Η μορφή του πίνακα φορτίσεων στη βάση δεδομένων.

Στον πίνακα των φορτίσεων που φαίνεται στο Σχήμα 2-7 οι στήλες περιέχουν:

- pattern: τον μοναδικό αριθμό του προφίλ φόρτισης στο οποίο αντιστοιχίζεται το φορτίο της γραμμής του πίνακα
- eidos: το είδος του στοιχείου στο οποίο εφαρμόζεται η φόρτιση, γραμμικό στοιχείο ή κόμβος
- ID: τον μοναδικό αριθμό του στοιχείου στο οποίο εφαρμόζεται η φόρτιση
- X-Y-Z: τη φόρτιση που δέχεται το στοιχείο της γραμμής στους αντίστοιχους άξονες, όπου στην περίπτωση που το στοιχείο είναι κόμβος η φόρτιση αναφέρεται σε συγκεντρωμένο φορτίο ενώ στην περίπτωση που είναι γραμμικό η φόρτιση αναφέρεται σε κατανεμημένο φορτίο
- MX-MY-MZ: όμοια με την προηγούμενη στήλη αλλά για την περίπτωση ροπής

Τέλος, για τις ανάγκες του περιβάλλοντος του κύριου παραθύρου του προγράμματος δημιουργήθηκαν γραμμές εργαλείων, στις οποίες οργανώνονται οι λειτουργίες του προγράμματος ανάλογα με τη χρηστική τους λειτουργία και τη ροή του προγράμματος (Σχήμα 2-8).



Σχήμα 2-8. Γραμμές εργαλείων του κύριου παραθύρου του προγράμματος.

### 2.2.2 Επεξεργαστής

Στο κομμάτι αυτό καλείται το λογισμικό να αναλύσει τα δεδομένα που εισάγονται από το προηγούμενο στάδιο και να παράγει αποτελέσματα. Ο τρόπος ανάλυσης καθορίστηκε ως στατική ανάλυση. Για τη δημιουργία ενός αλγορίθμου ικανού να εκτελέσει τη στατική ανάλυση ενός τρισδιάστατου φορέα, αξιοποιήθηκε η μέθοδος της μητρικής στατικής (Παναγούλη, 2018), δηλαδή της μόρφωσης μητρώων και επίλυσης εξισώσεων με Γραμμική Άλγεβρα. Η μέθοδος αυτή άλλωστε αναπτύχθηκε για ακριβώς αυτό το σκοπό.

Για την προσέγγιση της μεθόδου προγραμματιστικά, χρησιμοποιήθηκαν δυναμικοί πίνακες (dynamic arrays), οι οποίοι δίνουν τη δυνατότητα επαναπροσδιορισμού του μέγιστου μεγέθους τους κατά την εκτέλεση του κώδικα, σε αντίθεση με τους στατικούς πίνακες (static arrays), όπου το μέγεθος τους καθορίζεται από τον προγραμματιστή κατά το στάδιο του προγραμματισμού και δεν μπορεί να υπερβληθεί κατά την εκτέλεση του κώδικα. Το γεγονός αυτό επιτρέπει την ελεύθερη χρήση γραμμικών στοιχείων και κόμβων από το χρήστη, ενώ στην αντίθετη περίπτωση ο αριθμός τους θα ήταν περιορισμένος. Ο επανακαθορισμός του μεγέθους ενός πίνακα γίνεται με την εντολή ReDim.

Ακόμα, για τις ανάγκες εκτέλεσης των μαθηματικών πράξεων με μητρώα με χρήση της Γραμμικής Άλγεβρας, δημιουργήθηκε μία βιβλιοθήκη μαθηματικών, με τη μορφή δομοστοιχείου (module), αποτελούμενη από συναρτήσεις (functions). Στις συναρτήσεις αυτές περιλαμβάνονται:

- Πρόσθεση, αφαίρεση και πολλαπλασιασμός πινάκων δύο και τριών διαστάσεων (τριών διαστάσεων γιατί π.χ. ο πίνακας των μετακινήσεων μπορεί να αποτελείται από επιμέρους μητρώα, ένα για κάθε συνδυασμό φόρτισης)
- Υπολογισμός ανάστροφου πίνακα

- Αντιστροφή πίνακα, μέσω της μεθόδου παραγοντοποίησης Cholesky (Καρακασίδης & Σαρρής, 2015)
- Υπολογισμός απόστασης σημείου από ευθεία και ευθείας από ευθεία (για τις ανάγκες των γραφικών του προεπεξεργαστή και του μεθεπεξεργαστή)
- Υπολογισμός ροπών αδρανείας και στρέψης

Τα βήματα τα οποία ακολουθεί ο αλγόριθμος της ανάλυσης που κατασκευάστηκε για τις ανάγκες του προγράμματος παρουσιάζονται παρακάτω (βήμα i έως ix).

**i) Μόρφωση του μητρώου δυσκαμψίας όλων των γραμμικών στοιχείων (δοκών και υποστλωμάτων) στο τοπικό σύστημα αναφοράς.**

Το μητρώο δυσκαμψίας ενός γραμμικού στοιχείου με 12 βαθμούς ελευθερίας στο τοπικό σύστημα αναφοράς, καθώς η ανάλυση αφορά και τις 3 διαστάσεις του χώρου, είναι το παρακάτω 12 x 12 μητρώο (Προβατίδης, 2016):

$$[K_{local}^e] = \begin{bmatrix} \frac{EA}{L} & 0 & 0 & 0 & 0 & 0 & \frac{-EA}{L} & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{12EI_z}{L^3} & 0 & 0 & 0 & \frac{6EI_z}{L^2} & 0 & \frac{-12EI_z}{L^3} & 0 & 0 & 0 & \frac{6EI_z}{L^2} \\ 0 & 0 & \frac{12EI_y}{L^3} & 0 & \frac{-6EI_y}{L^2} & 0 & 0 & 0 & \frac{-12EI_y}{L^3} & 0 & \frac{-6EI_y}{L^2} & 0 \\ 0 & 0 & 0 & \frac{GI_x}{L} & 0 & 0 & 0 & 0 & 0 & \frac{-GI_x}{L} & 0 & 0 \\ 0 & 0 & \frac{6EI_y}{L^2} & 0 & \frac{4EI_y}{L} & 0 & 0 & 0 & \frac{6EI_y}{L^2} & 0 & \frac{2EI_y}{L} & 0 \\ 0 & \frac{6EI_z}{L^2} & 0 & 0 & 0 & \frac{4EI_z}{L} & 0 & \frac{-6EI_z}{L^2} & 0 & 0 & 0 & \frac{2EI_z}{L} \\ \frac{-EA}{L} & 0 & 0 & 0 & 0 & 0 & \frac{EA}{L} & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{-12EI_z}{L^3} & 0 & 0 & 0 & \frac{-6EI_z}{L^2} & 0 & \frac{12EI_z}{L^3} & 0 & 0 & 0 & \frac{-6EI_z}{L^2} \\ 0 & 0 & \frac{-12EI_y}{L^3} & 0 & \frac{6EI_y}{L^2} & 0 & 0 & 0 & \frac{12EI_y}{L^3} & 0 & \frac{6EI_y}{L^2} & 0 \\ 0 & 0 & 0 & \frac{-GI_x}{L} & 0 & 0 & 0 & 0 & 0 & \frac{GI_x}{L} & 0 & 0 \\ 0 & 0 & \frac{-6EI_y}{L^2} & 0 & \frac{2EI_y}{L} & 0 & 0 & 0 & \frac{12EI_y}{L^2} & 0 & \frac{4EI_y}{L} & 0 \\ 0 & \frac{6EI_z}{L^2} & 0 & 0 & 0 & \frac{2EI_z}{L} & 0 & \frac{-6EI_z}{L^2} & 0 & 0 & 0 & \frac{4EI_z}{L} \end{bmatrix}$$

Πίνακας 2-1. Μητρώο δυσκαμψίας γραμμικού στοιχείου στις 3 διαστάσεις.

Τα μόνα μεγέθη του παραπάνω μητρώου που είναι άγνωστα στο στάδιο αυτό είναι οι καμπτικές και στρεπτικές ροπές αδράνειας. Ο υπολογισμός τους γίνεται μέσα από τη βιβλιοθήκη μαθηματικών που δημιουργήθηκε, με χρήση των παρακάτω δύο εξισώσεων (1) και (2). Οι σχέσεις αυτές αφορούν αποκλειστικά ορθογωνικές διατομές και ιδιαίτερα η σχέση (2) έχει τροποποιηθεί για μονάδες μήκους (m).

$$I_{y,z} = b \cdot \frac{h^3}{12} \quad \text{Εξ. (1)}$$

Όπου, b: το πλάτος της διατομής (παράλληλο στον άξονα κάμψης)  
h: το ύψος της διατομής (κάθετο στον άξονα κάμψης)

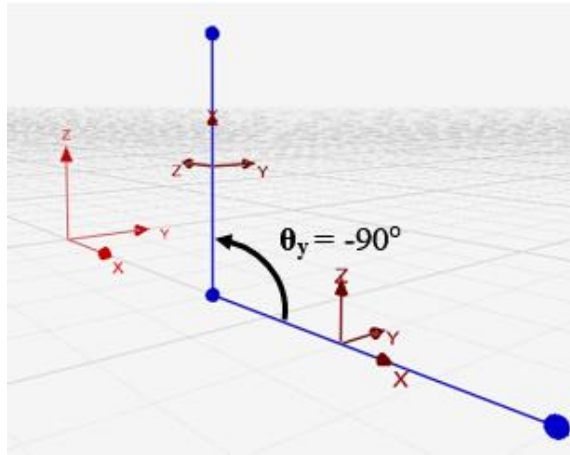
$$I_x = ab^3 \cdot \frac{1}{3} - 0.21 \cdot \frac{b}{a} \left( 1 - \frac{b^4}{12a^4} \right) \quad \text{Εξ. (2)}$$

Όπου, a: το μήκος της μεγάλης πλευράς της διατομής (m)  
b: το μήκος της μικρής πλευράς της διατομής (m)

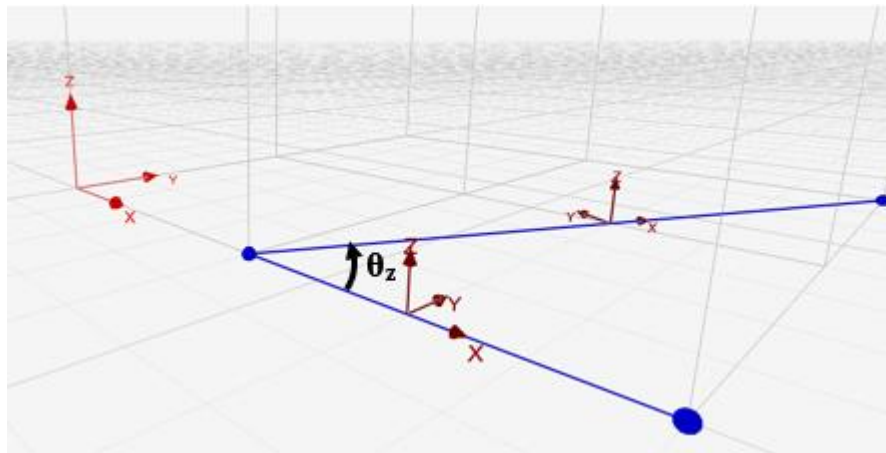
## ii) Μόρφωση του μητρώου μετασχηματισμού όλων των γραμμικών στοιχείων.

Το μητρώο μετασχηματισμού περιέχει την πληροφορία της σχετικής περιστροφής των τοπικών αξόνων ενός στοιχείου ως προς τους καθολικούς άξονες, με σκοπό το μετασχηματισμό των μητρώων που αναφέρονται στο τοπικό σύστημα στα ισοδύναμα μητρώα που αναφέρονται στο καθολικό. Στις τρεις διαστάσεις ο υπολογισμός των σχετικών στρεπτικών των αξόνων τοπικού και καθολικού (Προβατίδης, 2016), είναι μια σύνθετη διαδικασία που απαιτεί ιδιαίτερη προσοχή.

Για τις ανάγκες αυτής της εργασίας όμως, γίνεται η απλούστευση της αποδοχής μονάχα οριζόντιων (δοκοί) και κατακόρυφων (υποστυλώματα) στοιχείων. Έτσι, όλα τα υποστυλώματα θεωρείται ότι έχουν το τοπικό τους σύστημα αναφοράς στραμμένο κατά  $-90^\circ$  περί τον Y άξονα από το καθολικό (Σχήμα 2-9), ενώ όλοι οι δοκοί μπορούν να στραφούν μόνο ως προς τον Z άξονα (Σχήμα 2-10).



Σχήμα 2-9. Τοπικό σ.α. υποστυλώματος



Σχήμα 2-10. Τοπικό σ.α. δοκού

Με τον τρόπο αυτό η διαδικασία μόνρφωσης του μητρώου περιστροφής απλουστεύεται σημαντικά. Πιο συγκεκριμένα, τα μητρώα περιστροφής όλων των στοιχείων μπορούν να περιγραφούν με τα δύο μητρώα του Πίνακα 2-2 και του Πίνακα 2-3, ένα για κάθε κατακόρυφο στοιχείο και ένα για κάθε οριζόντιο στοιχείο αντίστοιχα.



$$[A_{column}^e] = \begin{bmatrix} \cos 90^\circ & 0 & \sin 90^\circ & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -\sin 90^\circ & 0 & \cos 90^\circ & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \cos 90^\circ & 0 & \sin 90^\circ & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -\sin 90^\circ & 0 & \cos 90^\circ & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \cos 90^\circ & 0 & \sin 90^\circ & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -\sin 90^\circ & 0 & \cos 90^\circ & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cos 90^\circ & 0 & \sin 90^\circ \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\sin 90^\circ & 0 & \cos 90^\circ \end{bmatrix}$$

Πίνακας 2-2. Μητρώο περιστροφής κατακόρυφων στοιχείων.

$$[A_{beam}^e] = \begin{bmatrix} \cos \theta z & \sin \theta z & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -\sin \theta z \cos \theta z & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \cos \theta z & \sin \theta z & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -\sin \theta z \cos \theta z & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \cos \theta z & \sin \theta z & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -\sin \theta z \cos \theta z & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cos \theta z & \sin \theta z & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\sin \theta z \cos \theta z & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Πίνακας 2-3. Μητρώο περιστροφής οριζόντιων στοιχείων.

**iii) Μόρφωση του μητρώου αντιστοίχισης όλων των στοιχείων.**

Το μητρώο αυτό περιέχει μόνο τα ψηφία 0 και 1 (Boolean), είναι μεγέθους 12 x (ν συνολικός αριθμός β.ε. της κατασκευής) και αντιστοιχίζει τους β.ε. του τοπικού σ.α. του στοιχείου με τους β.ε. του καθολικού σ.α., περιέχει δηλαδή την πληροφορία για τον τρόπο σύνδεσης των στοιχείων της κατασκευής.

Παρακάτω στον Πίνακα 2-4, δίνεται ένα παράδειγμα μητρώου αντιστοίχισης ενός τυχαίου στοιχείου, στην περίπτωση που οι β.ε. του ταυτίζονται με τους β.ε. της κατασκευής.

$$[L^e] = \begin{matrix} & \begin{matrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & \dots & \nu \end{matrix} \\ \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \\ 9 \\ 10 \\ 11 \\ 12 \end{matrix} & \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & \dots & 0 \end{bmatrix} \end{matrix}$$

Πίνακας 2-4. Μητρώο αντιστοίχισης στοιχείου, στην περίπτωση που οι β.ε. στοιχείου και κατασκευής ταυτίζονται.

Παρακάτω δίνεται η ρουτίνα μόρφωσης του μητρώου αντιστοίχισης όλων των γραμμικών μελών της κατασκευής που χρησιμοποιείται από το λογισμικό (Πίνακας 2-5).

```

Private Sub Le()
    For i = 0 To lines_dt.Rows.Count - 1
        Dim matLe2D(11, (6 * nodes_dt.Rows.Count) - 1) As Double
        For n = 0 To nodes_dt.Rows.Count - 1
            If lines_dt.Rows(i).Item(1) = n + 1 Then 'ΚΟΜΒΟΣ ΑΡΧΗΣ
                matLe2D(0, 6 * n) = 1
                matLe2D(1, 6 * n + 1) = 1
                matLe2D(2, 6 * n + 2) = 1
                matLe2D(3, 6 * n + 3) = 1
                matLe2D(4, 6 * n + 4) = 1
                matLe2D(5, 6 * n + 5) = 1
            Else
                matLe2D(0, 6 * n) = 0
                matLe2D(1, 6 * n + 1) = 0
                matLe2D(2, 6 * n + 2) = 0
                matLe2D(3, 6 * n + 3) = 0
                matLe2D(4, 6 * n + 4) = 0
                matLe2D(5, 6 * n + 5) = 0
            End If
            If lines_dt.Rows(i).Item(2) = n + 1 Then 'ΚΟΜΒΟΣ ΤΕΛΟΥΣ
                matLe2D(6, 6 * n) = 1
                matLe2D(7, 6 * n + 1) = 1
                matLe2D(8, 6 * n + 2) = 1
                matLe2D(9, 6 * n + 3) = 1
                matLe2D(10, 6 * n + 4) = 1
                matLe2D(11, 6 * n + 5) = 1
            Else
                matLe2D(6, 6 * n) = 0
                matLe2D(7, 6 * n + 1) = 0
                matLe2D(8, 6 * n + 2) = 0
                matLe2D(9, 6 * n + 3) = 0
                matLe2D(10, 6 * n + 4) = 0
                matLe2D(11, 6 * n + 5) = 0
            End If
        Next
    Next
End Sub
    
```

Πίνακας 2-5. Διαδικασία υπολογισμού των μητρώων αντιστοίχισης.

**iv) Μόρφωση όλων των παραπάνω μητρώων για την περίπτωση των εικονικών δοκών.**

Για την προσομοίωση της διαφραγματικής λειτουργίας της πλάκας, δηλαδή της δέσμευσης των σχετικών μετακινήσεων των δεσμευμένων κόμβων, ακολουθήθηκε μία προσωπική μέθοδος η οποία σχεδιάστηκε σε συνεργασία με τον επιβλέποντα της εργασίας.

Σύμφωνα με αυτήν τη μέθοδο, η απαραίτητη δυσκαμψία στο διάφραγμα που ορίζουν οι δεσμευμένοι κόμβοι προσδίδεται μέσω της χρήσης εικονικών δοκών, οι οποίες ενώνουν όλους αυτούς τους κόμβους μεταξύ τους ανά δύο. Στις δοκούς αυτές δεν καθορίζονται διαστάσεις διατομής, αλλά απευθείας στον αλγόριθμο της ανάλυσης καθορίζονται οι παρακάτω τιμές που φαίνονται στον Πίνακα 2-6.

$E = 3 * 10^7$ $G = 0$ $A = 10000$ $L = \text{length}(x1, y1, z1, x2, y2, z2)$ $I_y = 0$ $I_z = 10000$ $I_x = 0$
--

Πίνακας 2-6. Οι τιμές που προσδίδονται στο μητρώο δυσκαμψίας των εικονικών δοκών, για την προσομοίωση της διαφραγματικής λειτουργίας (kN, m)

Στην ουσία επιλέχθηκαν αυθαίρετα πολύ μεγάλες τιμές δυσκαμψίας ως προς τον άξονα Z (κάμψη εντός του επιπέδου της πλάκας) και εμβαδού διατομής (δυστένεια δοκού εντός του επιπέδου της πλάκας), στη προσπάθεια να μηδενιστούν οι ανάλογες σχετικές μετακινήσεις των κόμβων του διαφράγματος. Η αποτελεσματικότητα της μεθόδου επαληθεύεται με τη σύγκριση των αποτελεσμάτων του αριθμητικού παραδείγματος του Κεφαλαίου 4, με τα αποτελέσματα ενός εμπορικά διαθέσιμου λογισμικού ανάλυσης.

**v) Μόρφωση του μητρώου μετάθεσης της κατασκευής.**

Σε αυτό το βήμα σχηματίζουμε το μητρώο το οποίο πολλαπλασιάζεται με το μητρώο δυσκαμψίας της κατασκευής, για να ληφθούν υπόψιν οι συνθήκες στήριξης, και να συγκεντρωθούν όλες οι γραμμές των δεσμευμένων β.ε. στο κάτω μέρος του μητρώου δυσκαμψίας και όλες οι αντίστοιχες στήλες στο τέλος δεξιά. Χωρίζεται έτσι σε τέσσερα υπομητρώα, τα οποία θα χρειαστούν στη συνέχεια για τον υπολογισμό των μητρώων μετακινήσεων και αντιδράσεων.

Και σε αυτήν την περίπτωση πρόκειται για ένα μητρώο με τιμές 0 και 1 (Boolean) το οποίο μορφώνεται αντιστοιχώντας τη θέση που καταλαμβάνουν οι β.ε. της κατασκευής στο μητρώο

δυσκαμψίας της κατασκευής, με τις νέες θέσεις τους στο νέο τροποποιημένο μητρώο δυσκαμψίας, έτσι ώστε οι δεσμευμένοι β.ε. λόγω των στηρίξεων να τοποθετηθούν στο τέλος των γραμμών και των στηλών. Ο κώδικας που χρησιμοποιήθηκε για τη μόρφωσή του παρουσιάζεται παρακάτω (Πίνακας 2-7).

```

Private Sub V()
  For i = 0 To (6 * nodes_dt.Rows.Count) - 1 'ΠΡΩΤΑ ΜΗΑΕΝΙΖΩ ΣΕ ΟΛΕΣ ΤΙΣ ΘΕΣΕΙΣ
    For j = 0 To (6 * nodes_dt.Rows.Count) - 1
      matV(i, j) = 0
    Next
  Next
  Dim row_id, c, d As Integer
  row_id = row_ss 'row_ss = ((6 * nodes_dt.Rows.Count) - 1) - 6 * num_paktwsewn + 1
  c = num_paktwsewn
  For i = 0 To nodes_dt.Rows.Count - 1
    If nodes_dt.Rows(i).Item(5) = "YES" Then 'ΑΠΟΤΕΛΕΙ ΚΟΜΒΟ ΣΤΗΡΙΞΗΣ
      row_id = ((6 * nodes_dt.Rows.Count) - 1) - 6 * c + 1
      matV(row_id, 6 * i) = 1
      matV(row_id + 1, 6 * i + 1) = 1
      matV(row_id + 2, 6 * i + 2) = 1
      matV(row_id + 3, 6 * i + 3) = 1
      matV(row_id + 4, 6 * i + 4) = 1
      matV(row_id + 5, 6 * i + 5) = 1
      c -= 1
    Else
      matV(d, 6 * i) = 1
      matV(d + 1, 6 * i + 1) = 1
      matV(d + 2, 6 * i + 2) = 1
      matV(d + 3, 6 * i + 3) = 1
      matV(d + 4, 6 * i + 4) = 1
      matV(d + 5, 6 * i + 5) = 1
      d += 6
    End If
  Next
End Sub

```

Πίνακας 2-7. Διαδικασία μόρφωσης μητρώου μετάθεσης.

#### vi) Μόρφωση του μητρώου δυσκαμψίας της κατασκευής στο καθολικό σ.α.

Στο σημείο αυτό, συνδυάζονται όλα τα παραπάνω μητρώα και με κατάλληλες πράξεις καταλήγουμε στο μητρώο δυσκαμψίας της κατασκευής. Με βάση τις παρακάτω εξισώσεις (3),(4),(5) και (6), μορφώθηκε ο αλγόριθμος του Πίνακα 2-8.

$$[K_e^{global}] = [L_e]^T [\Lambda_e]^T [K_e^{local}] [\Lambda_e] [L_e] \quad \text{Εξ. (3)}$$

$$[K_{structure}^{global}] = \sum [K_e^{global}] \quad \text{Εξ. (4)}$$

$$[K_{structure}^{modified}] = [V] [K_{structure}^{global}] [V]^T \quad \text{Εξ. (5)}$$

$$[K_{structure}^{modified}] = \begin{bmatrix} [K_{ff}] & [K_{fs}] \\ [K_{sf}] & [K_{ss}] \end{bmatrix} \quad \text{Εξ. (6)}$$

```

Public Sub K_KATASKEYHS()
    '[K global] = Σ([LeT] * [AT] * [K] * [Λ] * [Le])
    ' ΦΤΙΑΧΝΩ ΜΗΔΕΝΙΚΟ ΤΟ ΜΗΤΡΩΟ
    For i = 0 To (6 * nodes_dt.Rows.Count) - 1
        For j = 0 To (6 * nodes_dt.Rows.Count) - 1
            matK_KATASKEYHS(i, j) = 0
        Next
    Next
    For i = 0 To lines_dt.Rows.Count - 1
        matKe_GLOBAL(i) =
        MyMATH.mmult(MyMATH.mmult(MyMATH.mmult(MyMATH.mmult(MyMATH.Transpose(matLe(i)),
        MyMATH.Transpose(matAe(i))), matKe(i)), matAe(i)), matLe(i))
        matK_KATASKEYHS = MyMATH.msum(matK_KATASKEYHS, matKe_GLOBAL(i))
    Next
    'ΓΙΑ ΝΑ ΛΑΒΩ ΥΠΟΨΗΝ ΜΟΥ ΤΑ CONSTRAINTS
    For i = 0 To virtual_dt.Rows.Count - 1
        matKe_virtual_GLOBAL(i) =
        MyMATH.mmult(MyMATH.mmult(MyMATH.mmult(MyMATH.Transpose(matLe_virtual(i)),
        MyMATH.Transpose(matAe_virtual(i))), matKe_virtual(i)), matAe_virtual(i)),
        matLe_virtual(i))
        matK_KATASKEYHS = MyMATH.msum(matK_KATASKEYHS, matKe_virtual_GLOBAL(i))
    Next
    matK_KATASKEYHS = MyMATH.mmult(MyMATH.mmult(matV, matK_KATASKEYHS),
    MyMATH.Transpose(matV))

    'ΧΩΡΙΖΩ ΣΕ ΥΠΟΜΗΤΡΩΑ
    For i = 0 To row_ss - 1
        For j = 0 To row_ss - 1
            matKff(i, j) = matK_KATASKEYHS(i, j)
        Next
    Next
    For i = 0 To row_ss - 1
        For j = 0 To 6 * nodes_dt.Rows.Count - 1 - row_ss
            matKfs(i, j) = matK_KATASKEYHS(i, j + row_ss)
        Next
    Next
    For i = 0 To 6 * nodes_dt.Rows.Count - 1 - row_ss
        For j = 0 To row_ss - 1
            matKsf(i, j) = matK_KATASKEYHS(i + row_ss, j)
        Next
    Next
    For i = 0 To 6 * nodes_dt.Rows.Count - 1 - row_ss
        For j = 0 To 6 * nodes_dt.Rows.Count - 1 - row_ss
            matKss(i, j) = matK_KATASKEYHS(i + row_ss, j + row_ss)
        Next
    Next
End Sub

```

Πίνακας 2-8. Διαδικασία υπολογισμού μητρώου δυσκαμψίας της κατασκευής.

**Όλα τα επόμενα βήματα επαναλαμβάνονται για κάθε περίπτωση συνδυασμού φορτίσεων που έχει ορίσει ο χρήστης.**

**vii) Μητρώο φόρτισης του φορέα.**

Στο σημείο αυτό μεταφέρεται η πληροφορία της φόρτισης του κάθε μέλους της κατασκευής στους κόμβους, ως ισοδύναμα κομβικά φορτία παγίωσης. Για να γίνει αυτό, παγώνονται οι κόμβοι αρχής και τέλους κάθε μέλους και στη συνέχεια υπολογίζονται οι αντιδράσεις παγίωσης που προκύπτουν λόγω της κατανεμημένης φόρτισης που δέχεται το κάθε μέλος. Με τις τιμές των αντιδράσεων αυτών συμπληρώνονται οι γραμμές του διανύσματος των δράσεων κάθε στοιχείου (μητρώο με διαστάσεις 12 x 1).

Για τον υπολογισμό των αντιδράσεων στις τρεις διαστάσεις, πρέπει να συνυπολογιστούν τα συνημίτονα διεύθυνσης των διανυσμάτων φόρτισης. Επειδή η φόρτιση που ασκείται καθορίζεται από τον χρήστη στο καθολικό σύστημα αναφοράς, θα πρέπει να προσδιοριστούν τα κατανεμημένα φορτία στο τοπικό σύστημα αναφοράς κάθε στοιχείου. Για τον λόγο αυτό πολλαπλασιάζονται όλα τα φορτία του καθολικού σ.α. με τον κατάλληλο κάθε φορά συντελεστή, έτσι ώστε να μετασηματιστούν στο τοπικό σ.α του στοιχείου. Η διαδικασία που ακολουθήθηκε παρουσιάζεται παρακάτω (Πίνακας 2-9).

```

θz = Math.Asin((z2 - z1) / L)
θx = Math.Asin((x2 - x1) / L)
θy = Math.Asin((y2 - y1) / L)
Dim line_type = lines_dt.Rows(i).Item(4)
Select Case line_type
    Case 0 ' ΔΟΚΟΣ
        matAI(load_pattern, i, 0) = (qx * Math.Sin(θx) + qy * Math.Sin(θy)) * L / 2
        matAI(load_pattern, i, 1) = (-qx * Math.Cos(θx) + qy * Math.Cos(θy)) * L / 2
        matAI(load_pattern, i, 2) = (qz * Math.Cos(θz)) * L / 2
        matAI(load_pattern, i, 3) = 0
        matAI(load_pattern, i, 4) = -(qz * Math.Cos(θz)) * L ^ 2 / 12
        matAI(load_pattern, i, 5) = (-qx * Math.Cos(θx) + qy * Math.Cos(θy)) * L ^ 2 / 12
        matAI(load_pattern, i, 6) = (qx * Math.Sin(θx) + qy * Math.Sin(θy)) * L / 2
        matAI(load_pattern, i, 7) = (-qx * Math.Cos(θx) + qy * Math.Cos(θy)) * L / 2
        matAI(load_pattern, i, 8) = (qz * Math.Cos(θz)) * L / 2
        matAI(load_pattern, i, 9) = 0
        matAI(load_pattern, i, 10) = (qz * Math.Cos(θz)) * L ^ 2 / 12
        matAI(load_pattern, i, 11) = -(-qx * Math.Cos(θx) + qy * Math.Cos(θy)) * L^2/12
    Case 1 ' ΥΠΟΣΤΥΛΩΜΑ
        matAI(load_pattern, i, 0) = (qz * Math.Sin(θz)) * L / 2
        matAI(load_pattern, i, 1) = (qy * Math.Cos(θy)) * L / 2
        matAI(load_pattern, i, 2) = -(qx * Math.Cos(θx)) * L / 2
        matAI(load_pattern, i, 3) = 0
        matAI(load_pattern, i, 4) = (qx * Math.Cos(θx)) * L ^ 2 / 12
        matAI(load_pattern, i, 5) = (qy * Math.Cos(θy)) * L ^ 2 / 12
        matAI(load_pattern, i, 6) = (qz * Math.Sin(θz)) * L / 2
        matAI(load_pattern, i, 7) = (qy * Math.Cos(θy)) * L / 2
        matAI(load_pattern, i, 8) = -(qx * Math.Cos(θx)) * L / 2

```

```

matAI(load_pattern, i, 9) = 0
matAI(load_pattern, i, 10) = -(qx * Math.Cos(theta_x)) * L ^ 2 / 12
matAI(load_pattern, i, 11) = -(qy * Math.Cos(theta_y)) * L ^ 2 / 12
End Select
matAI_global2D = MyMATH.mmult(MyMATH.Transpose(matAe(i)), matAI2D)

```

Πίνακας 2-9. Υπολογισμός μητρώου δράσεων παγίωσης.

Με οδηγό τις τιμές του παραπάνω διανύσματος, εκφρασμένου στο καθολικό σ.α., μορφώνουμε και το διάνυσμα φορτίσεων της κατασκευής. Στο διάνυσμα αυτό έρχονται να προστεθούν και τα επικόμβια φορτία που αποτελούν μέρος του συνδυασμού φόρτισης. Τελικό βήμα είναι η αναδιάταξη του διανύσματος μέσω της εξίσωσης (7), έτσι ώστε να μετακινηθούν στο κάτω μέρος του διανύσματος οι άγνωστες αντιδράσεις των στηρίξεων.

$$[P_{structure}^{modified}] = [V] [P_{structure}] \quad \text{Εξ. (7)}$$

#### viii) Υπολογισμός μητρώου μετακινήσεων

Αφού έχουν μορφωθεί τα μητρώα δυσκαμψίας και το διάνυσμα φόρτισης, μπορούμε να υπολογίσουμε το διάνυσμα των μετακινήσεων μέσω της εξίσωσης (8) και (9).

$$\begin{bmatrix} [P_f] \\ [P_s] \end{bmatrix} = \begin{bmatrix} [K_{ff}] & [K_{fs}] \\ [K_{sf}] & [K_{ss}] \end{bmatrix} \begin{bmatrix} [\Delta_f] \\ [\Delta_s] \end{bmatrix} \quad \text{Εξ. (8)}$$

$$[\Delta_f] = [K_{ff}]^{-1} [P_f] \quad \text{Εξ. (9)}$$

Στο σημείο αυτό, είναι απαραίτητος ο υπολογισμός του αντιστρόφου πίνακα  $[K_{ff}]^{-1}$ . Για τον υπολογισμό αυτό χρησιμοποιήθηκε η μέθοδος παραγοντοποίησης Cholesky, χωρίζοντας έτσι τον πίνακα προς αντιστροφή σε άνω και κάτω τριγωνικό (Καρακασιδής & Σαρρής, 2015).

#### ix) Υπολογισμός ακραίων δράσεων των μελών της κατασκευής, και εντατικών μεγεθών.

Αφού έχουν υπολογιστεί οι μετακινήσεις των ελευθέρων βαθμών ελευθερίας, μπορούμε να υπολογίσουμε τα διανύσματα των ακραίων μετακινήσεων, δηλαδή τις μετακινήσεις στους κόμβους αρχής και τέλους, όλων των μελών της κατασκευής και έπειτα να υπολογιστούν οι ακραίες δράσεις των μελών. Η εξίσωση υπολογισμού των ακραίων δράσεων ενός μέλους της κατασκευής είναι η (10).

$$[A_{local}^e] = [A_I^e] + [K_{local}^e] [A^e] [D_{global}^e] \quad \text{Εξ. (10)}$$

Όπου,  $[A_{local}^e]$ : μητρώο ακραίων δράσεων στοιχείου στο τοπικό σ.α.

$[A_I^e]$ : μητρώο ακραίων δράσεων παγίωσης στοιχείου (βήμα 7)

$[D_{global}^e]$ : μητρώο ακραίων μετακινήσεων στοιχείου στο καθολικό σ.α.

Τελευταίο βήμα αποτελεί ο υπολογισμός των εντατικών μεγεθών όλων των μελών. Με τον όρο εντατικά μεγέθη ενός στοιχείου περιγράφουμε μία συνάρτηση των τιμών των εντατικών μεγεθών και της θέσης κατά μήκος του στοιχείου. Για τις ανάγκες της παρούσας εργασίας η διαδικασία απλουστεύτηκε με διακριτοποίηση του κάθε στοιχείου σε 10 ίσα μέρη, στους κόμβους των οποίων υπολογίστηκαν οι τιμές των εντατικών μεγεθών. Η διαδικασία παρουσιάζεται στον Πίνακα 2-10.

```

θz = Math.Asin((z2 - z1) / L)
θx = Math.Asin((x2 - x1) / L)
θy = Math.Asin((y2 - y1) / L)

Select Case line_type
  Case 0 ' ΔΟΚΟΣ
    qx_topiko = (qx * Math.Sin(θx) + qy * Math.Sin(θy))
    qy_topiko = (-qx * Math.Cos(θx) + qy * Math.Cos(θy))
    qz_topiko = (qz * Math.Cos(θz))
  Case 1 ' ΥΠΟΣΤΥΛΩΜΑ
    qx_topiko = (qz * Math.Sin(θz))
    qy_topiko = (qy * Math.Cos(θy))
    qz_topiko = -(qx * Math.Cos(θx))
End Select
Dim c As Integer = 0

For x = 0 To L Step L / 10
  matN(load_pattern, i, c) = -matA(load_pattern, i, 0) + qx_topiko * x
  matQ2(load_pattern, i, c) = matA(load_pattern, i, 2) - qz_topiko * x
  matQ3(load_pattern, i, c) = matA(load_pattern, i, 1) - qy_topiko * x
  matM1(load_pattern, i, c) = matA(load_pattern, i, 3)
  matM2(load_pattern, i, c) = -matA(load_pattern, i, 5) +
(matA(load_pattern, i, 1) * x - (qy_topiko / 2) * x ^ 2)
  matM3(load_pattern, i, c) = matA(load_pattern, i, 4) +
(matA(load_pattern, i, 2) * x - (qz_topiko / 2) * x ^ 2)
  c += 1
Next

```

Πίνακας 2-10. Αλγόριθμος υπολογισμού εντατικών μεγεθών στοιχείου.

Η παραπάνω μεθοδολογία που ακολουθήθηκε για την ανάπτυξη του αλγορίθμου της ανάλυσης του λογισμικού, επαληθεύεται στο Παράρτημα Α. Εκεί γίνεται σύγκριση των αποτελεσμάτων του λογισμικού που αναπτύχθηκε με τα αποτελέσματα ενός εμπορικά διαθέσιμου λογισμικού ανάλυσης, για το αριθμητικό παράδειγμα του Κεφαλαίου 4.



### 2.2.3 Σχεδιασμός οπλισμού στοιχείων οπλισμένου σκυροδέματος

Όπως αναφέρθηκε και παραπάνω, για τη στρατηγική διαστασιολόγησης του οπλισμού των δοκών και των υποστυλωμάτων ακολουθήθηκε η μέθοδος που προτείνεται στο κεφάλαιο 7 του βιβλίου *Seismic Design of Concrete Buildings to Eurocode 8* (Fardis, Carvalho, Fajfar, & Pecker, 2015). Επιπλέον για τη διαστασιολόγηση έγιναν οι εξής απλουστευτικές παραδοχές:

- Οι δοκοί διαστασιολογούνται μόνο για την Ο.Κ.Α. και μόνο ο διαμήκης οπλισμός τους υπολογίζεται.
- Το συνεργαζόμενο πλάτος πλάκας - δοκού δεν λαμβάνεται υπόψη
- Τα συνεχή καθ' ύψος υποστυλώματα θεωρείται ότι έχουν κοινή διατομή και διαστασιολογούνται όμοια
- Ο οπλισμός των υποστυλωμάτων τοποθετείται συμμετρικά σε κάθε πλευρά
- Οι κόμβοι δοκού υποστυλώματος, τα θεμέλια και οι πλάκες δεν διαστασιολογούνται

Η διαδικασία ξεκινάει με τη διαστασιολόγηση πρώτα των δοκών και τον υπολογισμό των ροπών αντοχής σε κάθε άκρο τους στην άνω και κάτω παρειά. Στη συνέχεια υπολογίζονται σε κάθε κόμβο δοκού - υποστυλώματος οι συνολικές ροπές αντοχής των δοκών που συμβάλλουν σε αυτόν, από τις οποίες προκύπτει και η ελάχιστη απαιτούμενη αντοχή των συμβαλλόμενων υποστυλωμάτων, έτσι ώστε να ικανοποιείται ο ικανοτικός σχεδιασμός κατά τον EC8. Από τις ροπές αντοχής αυτές καθώς και από τις μέγιστες ροπές κάμψης που προκύπτουν από τα αποτελέσματα της ανάλυσης, υπολογίζονται για κάθε υποστυλώμα οι κρίσιμες ροπές σχεδιασμού ανά διεύθυνση για τις οποίες και διαστασιολογείται ο οπλισμός τους.

Η διαδικασία που ακολουθεί ο αλγόριθμος για τη διαστασιολόγηση των δοκών παρουσιάζεται στα παρακάτω βήματα.

#### **i) Υπολογισμός της μέγιστης διαμέτρου των ράβδων που διέρχονται από τον κόμβο αρχής και τέλους κάθε δοκού.**

Σύμφωνα με τον ευρωκώδικα 8, η μέγιστη διάμετρος μίας ράβδου οπλισμού δοκού που μπορεί διέλθει ή να τερματίσει μέσα σε κόμβο είναι αυστηρά καθορισμένη από τις σχέσεις (1) και (2) που παρουσιάζονται παρακάτω (Fardis, Carvalho, Fajfar, & Pecker, 2015). Η σχέση (1) αναφέρεται σε εσωτερικό κόμβο (ως προς τη διεύθυνση της δοκού), ενώ η σχέση (2) σε εξωτερικό.

- Για εσωτερικό κόμβο

$$\frac{d_{bL}}{h_c} \leq \frac{7.5f_{ctm}}{\gamma_{Rd}f_{yd}} \cdot \frac{1 + 0.8v_d}{1 + k \frac{\rho_2}{\rho_{1,max}}} \quad (1)$$

- Για εξωτερικό κόμβο (κατά τη διεύθυνση της δοκού)

$$\frac{d_{bL}}{h_c} \leq \frac{7.5f_{ctm}}{\gamma_{Rd}f_{yd}} \cdot (1 + 0.8v_d) \quad (2)$$

Όπου,  $d_{bL}$  η διάμετρος ράβδου δοκού

$h_c$ : η διάσταση του υποστυλώματος, παράλληλα στη διεύθυνση της δοκού

$f_{ctm}$ : η μέση εφελκυστική αντοχή του σκυροδέματος

$\gamma_{Rd}, k$ : Για ΚΠΜ ( $\gamma_{Rd} = 1.0, k = 0.5$ ), για ΚΠΥ ( $\gamma_{Rd} = 1.2, k = 0.75$ )

$f_{yd}$ : Όριο διαρροής σχεδιασμού του χάλυβα οπλισμού

$v_d$ : Το ανηγμένο αξονικό φορτίο του υποστυλώματος, για το ελάχιστο αξονικό φορτίο από τους σεισμικούς συνδυασμούς δράσεων με τα μόνιμα και κινητά φορτία

$\rho_{1,max}$ : το μέγιστο ποσοστό του θλιβόμενου οπλισμού της δοκού

$\rho_2$ : το μέγιστο ποσοστό του θλιβόμενου οπλισμού της δοκού (στο στάδιο σχεδιασμού λαμβάνεται  $\rho_2 = 0.5 \rho_{1,max}$ )

Ανάλογα με την περίπτωση του κόμβου, υπολογίζεται από τις παραπάνω εξισώσεις η τιμή της μέγιστης τιμής  $d_{bL}$  για τον κόμβο αρχής και τέλους κάθε δοκού.

Σημαντικό είναι να αναφερθεί ότι το λογισμικό που αναπτύχθηκε δεν έχει τη δυνατότητα να αναγνωρίσει τους συνδυασμούς δράσεων που περιέχουν σεισμικά φορτία και συνεπώς στην περίπτωση υπολογισμού του ανηγμένου αξονικού φορτίου στην παραπάνω εξίσωση, όπως και παρακάτω στην περίπτωση υπολογισμού της ροπής σχεδιασμού στις δοκούς, το λογισμικό δεν υπολογίζει με τον ορθό τρόπο, κατά τον EC8, αυτά τα μεγέθη.

## ii) Προσδιορισμός του ελάχιστου απαιτούμενου εφελκυσμένου οπλισμού διατομής για κάθε δοκό.

Από τις σχετικές διατάξεις του ευρωκώδικα 2 και 8 για τις λεπτομέρειες όπλισης των δοκών, καθορίζεται το ελάχιστο απαιτούμενο εμβαδόν εφελκυσμένου οπλισμού διατομής. Συγκεκριμένα

η σχέση για τις ΚΠΜ και ΚΠΥ που εξετάζονται από τον αλγόριθμο είναι η (3) (Fardis, Carvalho, Fajfar, & Pecker, 2015).

$$A_{s,min} = \max \{ \rho_{min} b_w d ; A_s(2\Phi 14) \} \quad (3)$$

$$\text{Όπου,} \quad \rho_{min} = 0.5 \frac{f_{ctm}}{f_{yk}}$$

Το ελάχιστο αυτό όριο αναφέρεται σε όλες τις κρίσιμες διατομές, που λόγω των σεισμικών φορτίων αυτές είναι στα άκρα των στηρίξεων άνω και κάτω, και στο άνοιγμα άνω και κάτω.

### iii) Τοποθέτηση συνεχούς οπλισμού σε όλες τις θέσεις κάθε δοκού

Στο σημείο αυτό, τοποθετείται ενιαίος οπλισμός σε όλο το μήκος της δοκού, στα άκρα των στηρίξεων άνω και κάτω και στο άνοιγμα άνω και κάτω, με διάμετρο ίση με την ελάχιστη τιμή  $\max d_{bL}$  που υπολογίστηκε για τους κόμβους αρχής και τέλους κάθε δοκού, και αριθμό ράβδων τόσο ώστε να ικανοποιείτε η απαίτηση για το ελάχιστο εμβαδό  $A_{s,min}$  σε κάθε διατομή.

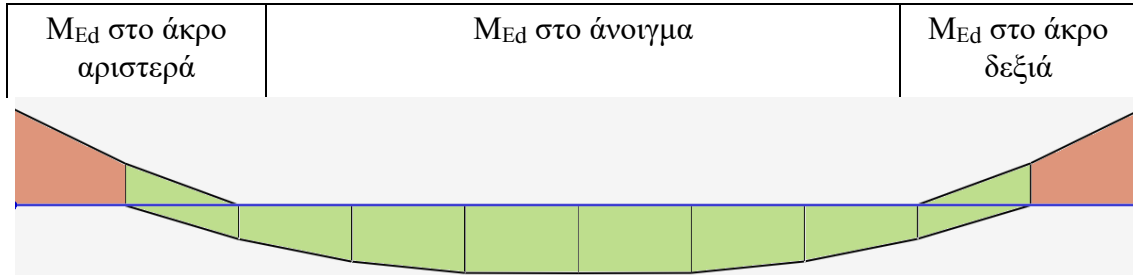
Αξίζει να σημειωθεί πως με τη μέθοδο αυτή, για την περίπτωση της Κ.Π.Μ. όπου το όριο της μέγιστης ράβδου που διέρχεται από κόμβο δοκού υποστυλώματος  $\max d_{bL}$  είναι πιο ελαστικό, παρατηρήθηκε ότι αλγόριθμος επιλέγει μεγάλη διάμετρο για τον συνεχή οπλισμό, αποκλείοντας έτσι την πιθανότητα να διαστασιολογήσει με μικρότερης διαμέτρου ράβδους.

### iv) Υπολογισμός απαιτούμενου εμβαδού οπλισμού $A_{s,req}$ σε κάθε θέση για την Ο.Κ.Α

Η ροπή σχεδιασμού σε αυτό το βήμα προκύπτει από την περιβάλλουσα των ροπών για όλους τους συνδυασμούς φόρτισης, που υπολογίστηκε κατά το στάδιο της ανάλυσης. Στο σημείο αυτό να τονιστεί ότι ο αλγόριθμος δεν έχει τη δυνατότητα να διακρίνει τους συνδυασμούς για τους οποίους υπολογίζεται η ροπή σχεδιασμού κατά τον ευρωκώδικα (συνδυασμοί μόνιμων και κινητών φορτίων και συνδυασμοί  $G + \psi 2Q + E$ ,  $G + \psi 2Q - E$ ), αλλά λαμβάνει υπόψη όλους εκείνους τους συνδυασμούς που θα του ορίσει ο χρήστης. Συγκεκριμένα σε κάθε θέση της δοκού, υπολογίζουμε τον απαιτούμενο οπλισμό ως εφελκόμενο λαμβάνοντας ως ροπή σχεδιασμού, στα άκρα άνω τις μέγιστες αρνητικές τιμές της περιβάλλουσας των ροπών και στο άνοιγμα και τα άκρα κάτω τις μέγιστες θετικές τιμές. Στο άνω άκρο του ανοίγματος δεν υπολογίζεται  $A_{s,req}$ , λόγω της απουσίας εφελκυστικών ροπών.

Επειδή όπως αναφέραμε στο Κεφάλαιο 2, κατά τον αλγόριθμο της ανάλυσης υπολογίζουμε τα εντατικά μεγέθη σε έντεκα διακριτά σημεία του κάθε στοιχείου, για τον προσδιορισμό της

κρίσιμης ροπής γίνεται η παραδοχή ότι η μέγιστη ροπή σχεδιασμού στα άκρα προκύπτει από τις ακραίες τρεις τιμές στα άκρα κάθε δοκού, ενώ στο άνοιγμα προκύπτει από τις υπόλοιπες πέντε τιμές στο εσωτερικό της (Σχήμα 2-11).



Σχήμα 2-11. Σημεία ελέγχου για τον υπολογισμό  $M_{Ed}$  στα άκρα και στο άνοιγμα της δοκού.

Αφού υπολογιστούν οι μέγιστες ροπές σχεδιασμού σε όλες τις θέσεις, υπολογίζονται και τα απαιτούμενα εμβαδά εφελκόμενου οπλισμού για τις τιμές αυτές. Οι σχέσεις που χρησιμοποιήθηκαν είναι οι (4) έως (11), (Fardis, Carvalho, Fajfar, & Pecker, 2015).

Αρχικά υπολογίζονται οι τιμές των  $\mu_{sd}$  και  $\mu_{sd,lim}$  από τις σχέσεις (4) και (5).

$$\mu_d = |M_{Ed}| / (b_{eff} d^2 f_{cd}) \quad \text{Εξ. (4)}$$

$$\mu_{d,lim} = 0.81 \xi_{lim} (1 - 0.416 \xi_{lim}) \quad \text{Εξ. (5)}$$

Έπειτα ξεχωρίζουμε δύο περιπτώσεις:

- $\mu_{sd} < \mu_{sd,lim}$

Δεν απαιτείται θλιβόμενος οπλισμός και το απαιτούμενο εμβαδόν εφελκόμενου οπλισμού προκύπτει από τις σχέσεις (6), (7):

$$\omega = 1 - \sqrt{1 - 2\mu_d} \quad \text{Εξ. (6)}$$

$$\omega = A_s / (b_{eff} d) \cdot (f_{yd} / f_{cd}) \quad \text{Εξ. (7)}$$

- $\mu_{sd} \geq \mu_{sd,lim}$

Απαιτείται θλιβόμενος οπλισμός και το απαιτούμενο εμβαδόν εφελκόμενου οπλισμού προκύπτει από τις σχέσεις (8) - (11):

$$\omega' = \frac{\mu_d - \mu_{d,lim}}{1 - d_2/d} \quad \text{Εξ. (8)}$$

$$\omega_{lim} = 0.973 \left( 1 - \sqrt{1 - \frac{2\mu_{d,lim}}{0.973}} \right) \quad \text{Εξ. (9)}$$

$$\omega = \omega_{lim} + \omega' \quad \text{Εξ. (10)}$$

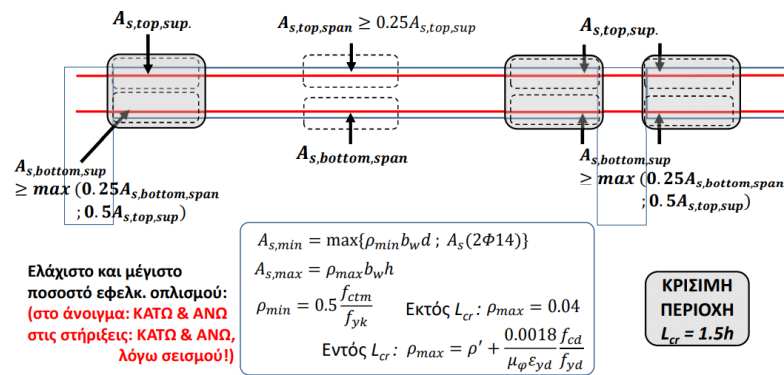
$$\omega = A_s / (b_{eff} d) \cdot (f_{yd} / f_{cd}) \quad \text{Εξ. (11)}$$

**ν) Τοποθέτηση πρόσθετου οπλισμού ώστε να καλύπτεται παντού η απαίτηση  $A_{s,1} > A_{s,req}$**

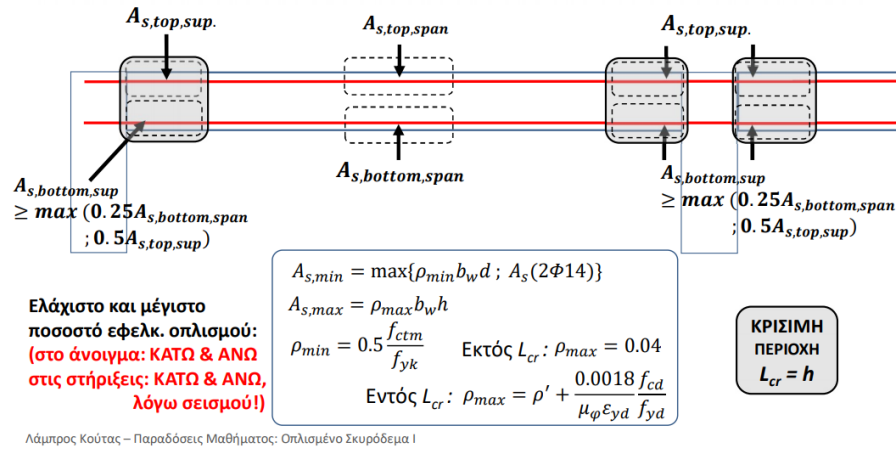
Ο επιπλέον οπλισμός τοποθετείται ξεχωριστά σε κάθε θέση και σε κάθε διατομή επικουρικά με τον συνεχή. Στο στάδιο αυτό, γίνονται επαναληπτικές διαδικασίες δοκιμών με σκοπό να προσδιοριστεί η διάταξη του οπλισμού σε κάθε θέση η οποία να καλύπτει την απαίτηση  $A_{s,1} > A_{s,req}$ . Σε κάθε βήμα αυξάνεται σταδιακά ο οπλισμός και έπειτα υπολογίζεται το νέο συνολικό εμβαδόν. Εάν η απαίτηση καλύπτεται, τότε ο αλγόριθμος ελέγχει την επόμενη θέση της δοκού και αφού ελεγχθούν όλες οι διατομές, μεταβαίνει στην επόμενη δοκό, έως ότου η διαδικασία ολοκληρωθεί για όλες τις δοκούς της κατασκευής. Πρώτα προστίθενται επιπλέον ράβδοι και έπειτα αυξάνεται η διάμετρός τους διαδοχικά, σύμφωνα με τις διαθέσιμες διαμέτρους που έχει καθορίσει ο χρήστης. Εάν απαιτούνται λιγότερες από δύο πρόσθετες ράβδους, ο αλγόριθμος προσπαθεί να επιλέξει την ελάχιστη διάμετρο, ενώ εάν απαιτούνται περισσότερες, προσπαθεί να επιλέξει τον ελάχιστο αριθμό ράβδων.

**vi) Έλεγχος των υπόλοιπων διατάξεων του EC8**

Αφού έχει τοποθετηθεί και ο πρόσθετος οπλισμός, γίνεται έλεγχος των υπόλοιπων περιορισμών που θέτει ο ευρωκώδικας για τις διατάξεις του οπλισμού των δοκών (Σχήμα 2-12 και Σχήμα 2-13).



Σχήμα 2-12. Λεπτομέρειες όπλισης δοκών (διαμήκεις οπλισμός), για την Κ.Π.Μ. (Κούτας, 2020).



Σχήμα 2-13. Λεπτομέρειες όπλισης δοκών (διαμήκεις οπλισμός), για την Κ.Π.Υ. (Κούτας, 2020).

Σε περίπτωση που κάποιος περιορισμός δεν καλύπτεται τότε ο αλγόριθμος προσπαθεί να επέμβει στις θέσεις που απαιτείται και να προσθέσει παραπάνω οπλισμό. Συγκεκριμένα, για να ικανοποιείται το μέγιστο ποσοστό οπλισμού στη θέση των άκρων άνω, ο αλγόριθμος προσθέτει επιπλέον οπλισμό στο άκρο κάτω, έτσι ώστε να αυξηθεί το ποσοστό του θλιβόμενου οπλισμού  $\rho'$ . Στην περίπτωση όπου έχει ξεπεραστεί το μέγιστο ποσοστό του εφελκόμενου οπλισμού στο άνοιγμα, ο αλγόριθμος τερματίζει.

**vii) Υπολογισμός της ροπής αντοχής  $M_{Rd}$  σε κάθε σημείο της δοκού**

Μετά την τοποθέτηση του τελικού οπλισμού, πρέπει να υπολογιστούν οι ροπές αντοχής στα άκρα άνω και κάτω κάθε δοκού, για τον υπολογισμό της απαιτούμενης ροπής σχεδιασμού των υποστυλωμάτων ώστε να ικανοποιείται ο ικανοτικός σχεδιασμός. Οι σχέσεις υπολογισμού των ροπών αντοχής του εφελκόμενου και θλιβόμενου οπλισμού σε κάθε άκρο της δοκού είναι η (12) και (13) (Fardis, Carvalho, Fajfar, & Pecker, 2015).

Ροπή αντοχής εφελκόμενου οπλισμού

$$M_{Rd,b}^+ = A_{s2} f_{yd} \max \left[ \left( h - d_2 - 0.5 A_{s2} f_{yd} / (b_{eff} f_{cd}) \right), (h - d_1 - d_2) \right] \quad (12)$$

Ροπή αντοχής θλιβόμενου οπλισμού

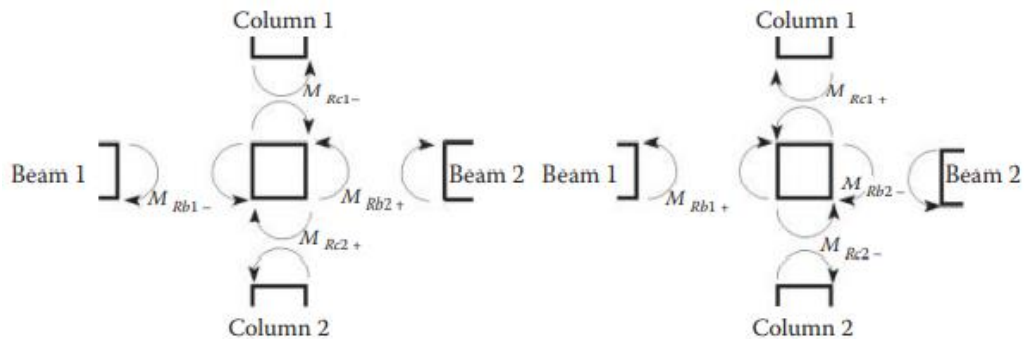
$$M_{Rd,b}^- = \min(A_{s1}, A_{s2}) f_{yd} (h - d_1 - d_2) + \max [0, (A_{s1} - A_{s2})] \cdot f_{yd} \left[ h - d_1 - 0.5 (A_{s1} - A_{s2}) f_{yd} / (b_w f_{cd}) \right] \quad (13)$$

Όπου,  $A_{s1}$ : Το εμβαδόν του εφελκόμενου οπλισμού  
 $A_{s2}$ : Το εμβαδόν του θλιβόμενου οπλισμού

- h: Το ύψος της διατομής
- d1,d2: Οι απόστάσεις των οπλισμών από το την άνω και κάτω ίνα
- b<sub>w</sub>: Το πλάτος του κορμού της δοκού

**viii) Υπολογισμός των ικανοτικών ροπών, σε κάθε κόμβο και σε κάθε διεύθυνση**

Γνωρίζοντας τις ροπές αντοχής στα άκρα κάθε δοκού, μπορούμε να υπολογίσουμε τη συνισταμένη ροπή αντοχής των δοκών σε κάθε κόμβο, για κάθε διεύθυνση ξεχωριστά και για κάθε φορά (αριστερόστροφη και δεξιόστροφη). Έτσι, τελικά καταλήγουμε με τέσσερις συνολικά συνισταμένες ροπών σε κάθε κόμβο, αριστερόστροφη και δεξιόστροφη στη X διεύθυνση, αριστερόστροφη και δεξιόστροφη στη Y διεύθυνση. Οι ροπές αντοχής αθροίζονται για κάθε κόμβο, ανάλογα με την φορά του Σχήματος 2-14.



Σχήμα 2-14. Οι ροπές αντοχής σε κόμβο δοκού – υποστυλώματος, για τον ικανοτικό σχεδιασμό (Fardis, Carvalho, Fajfar, & Pecker, 2015).

Στην περίπτωση που κάποια δοκός τερματίζει σε κόμβο διαγώνια (όχι δηλαδή κατά τη X ή Y διεύθυνση), τότε οι ροπές αντοχής της συμμετέχουν στη συνισταμένη του κόμβου ανάλογα με την κλίση της δοκού ως προς τον X άξονα. Έτσι, στον αλγόριθμό υπολογίζεται για κάθε δοκό η κλίση της διεύθυνσης της και έπειτα με τη βοήθεια των τριγωνομετρικών αριθμών τροποποιείται η συνολική συμμετοχή της στη συνισταμένη ροπή της X και Y διεύθυνσης κατάλληλα.

Έπειτα, παρουσιάζεται η διαδικασία που ακολουθεί ο αλγόριθμος για τη διαστασιολόγηση των υποστυλωμάτων στα παρακάτω βήματα.

**i) Υπολογισμός της κρίσιμης ροπής σχεδιασμού**

Η μέγιστη ροπή σχεδιασμού προκύπτει ως η κρισιμότερη από τη μέγιστη απαιτούμενη ροπή για να ικανοποιείται ο ικανοτικός σχεδιασμός σε κάθε κόμβο και από τη μέγιστη τιμή της περιβάλλουσας των ροπών, που υπολογίστηκε στο στάδιο της ανάλυσης. Τα υποστυλώματα που

συνεχίζουν καθ' ύψος θεωρούνται ενιαία και διαστασιολογούνται όμοια. Έτσι, η μέγιστη ικανοτική ροπή σχεδιασμού προκύπτει από όλους εκείνους τους κόμβους, από τους οποίους διέρχονται τα ενιαία υποστυλώματα. Σε κάθε κόμβο υπολογίσαμε τέσσερις τιμές των συνισταμένων  $M_{Rd}$  των δοκών που συμβάλλουν σε αυτόν. Έτσι, σε κάθε διεύθυνση κρισιμότερη ροπή λαμβάνεται η μεγαλύτερη τιμή από την αριστερόστροφη και δεξιόστροφη φορά. Τελικά, για κάθε ομάδα υποστυλωμάτων που θεωρήθηκαν ενιαία, προκύπτουν δύο ροπές σχεδιασμού  $M_{Ed}$ , μία σε κάθε διεύθυνση.

**ii) Υπολογισμός ελάχιστου αριθμού ράβδων ανά πλευρά**

Ο EC8 για την περίπτωση της Κ.Π.Μ. και Κ.Π.Υ, ορίζει ως ελάχιστο αριθμό ράβδων σε πλευρά υποστυλώματος τις τρεις. Ταυτόχρονα όμως, περιορίζει και τη μέγιστη απόσταση μεταξύ ράβδων που συγκρατούνται σε γωνία συνδετήρα. Για την Κ.Π.Μ, θέτει ως μέγιστη απόσταση τα 200 mm, ενώ για την Κ.Π.Υ. τα 150 mm. Λαμβάνοντας υπόψη τα παραπάνω καθώς και τις διαστάσεις των υποστυλωμάτων, υπολογίζεται ο ελάχιστος αριθμός ράβδων σε κάθε πλευρά ανά διεύθυνση.

**iii) Υπολογισμός της ελάχιστης διαμέτρου ράβδων**

Η ελάχιστη διάμετρος ράβδων προκύπτει από το ελάχιστο ποσοστό οπλισμού που θέτει ο EC2, για τον ελάχιστο αριθμό ράβδων που υπολογίστηκαν παραπάνω, θεωρώντας κοινή διάμετρο ράβδων και στις δυο διευθύνσεις. Στον παρακάτω πίνακα φαίνονται τα όρια που θέτει ο EC2.

Κατηγορία Πλαστιμότητας	ΚΠ Υ	ΚΠ Μ	ΚΠ Χ
$\rho_{min} = A_{s,min}/A_c$	1%		$0.1N_d/A_c f_{yd}$ , 0.2% <sup>(1)</sup>
$\rho_{max} = A_{s,max}/A_c$ *	4%		4% <sup>(1)</sup>
διάμετρος ράβδων, $d_{bL}$	≥8mm		
αριθμός ράβδων ανά πλευρά	≥3		≥2
απόσταση κατά την περίμετρο μεταξύ ράβδων που συγκρατούνται σε γωνία συνδετήρα	≤150mm	≤200mm	-
απόσταση κατά την περίμετρο ράβδου που δεν συγκρατείται σε γωνία συνδετήρα από την πλησιέστερη που συγκρατείται	≤150mm		

(1) Εθνικά Προσδιορίσιμη Παράμετρος (ΕΠΠ) κατά Ευρωκώδικα 2.

Πίνακας 2-11. Διατάξεις οπλισμού υποστυλωμάτων κατά τον EC2. (Κούτας, 2020)



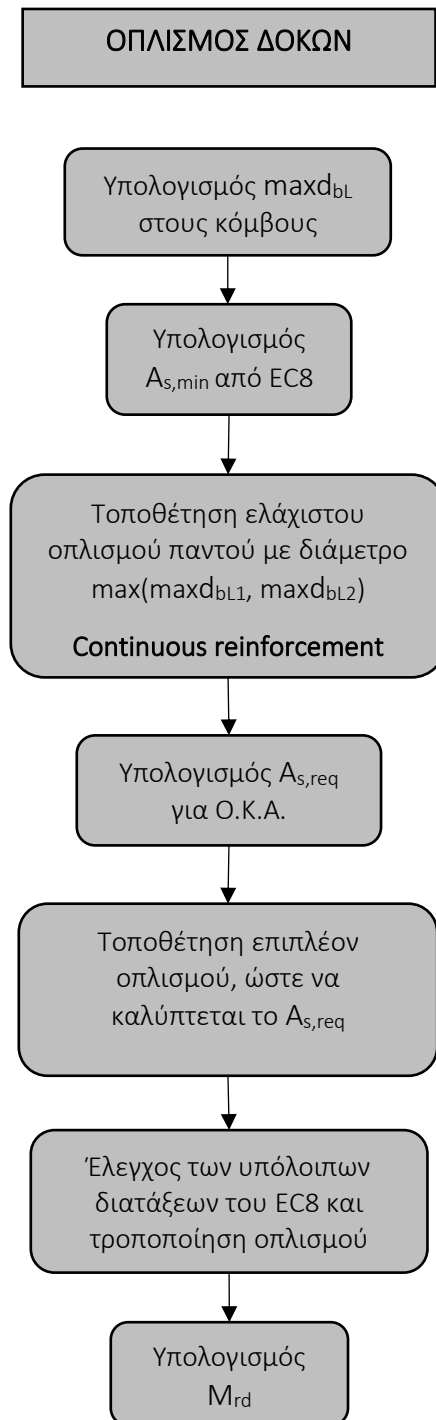
**iv) Τοποθετείται ο ελάχιστος οπλισμός**

Με βάση τα προηγούμενα δύο ελάχιστα μεγέθη που υπολογίστηκαν (ελάχιστος αριθμός ράβδων ανά διεύθυνση και ελάχιστη διάμετρος ράβδων), τοποθετείται ο ελάχιστος οπλισμός του υποστυλώματος, συμμετρικά σε κάθε διεύθυνση.

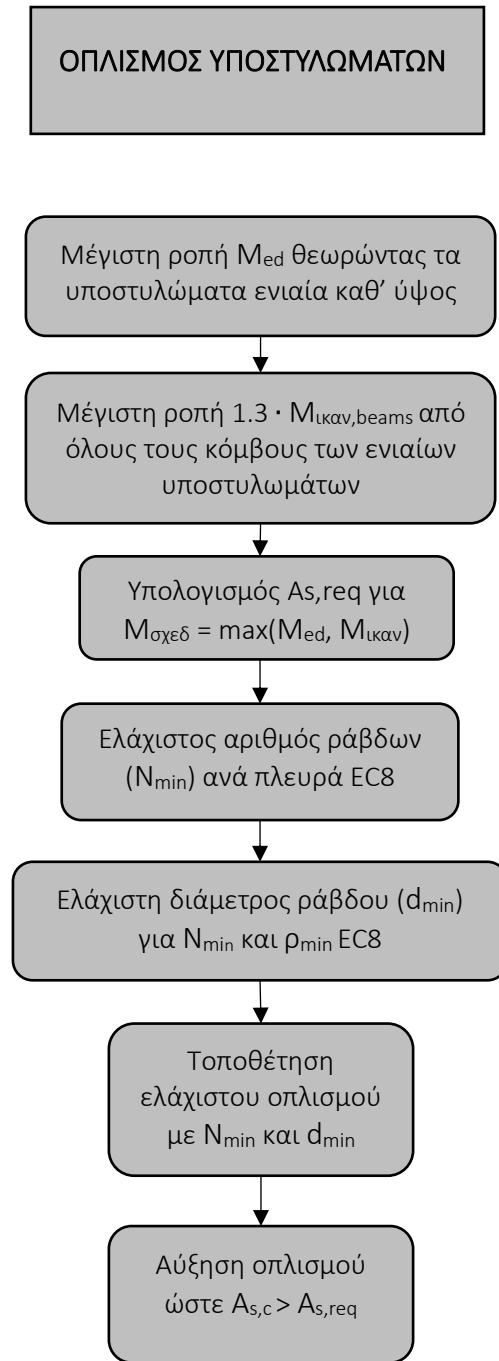
**v) Επιλογή οπλισμού ώστε να ικανοποιείται η σχέση  $A_{s,x,y} > A_{s,req,x,y}$** 

Στο τελευταίο στάδιο οπλισμού των υποστυλωμάτων, αυξάνεται ο υπάρχων οπλισμός κάθε υποστυλώματος, έτσι ώστε να ικανοποιείται το απαιτούμενο εμβαδόν  $A_{s,req}$  σε κάθε διεύθυνση. Πρώτα αυξάνεται η διάμετρος των γωνιακών ράβδων κατά 2 mm και 4 mm παραπάνω από τις εσωτερικές. Εάν η απαίτηση δεν καλύπτεται και πάλι, τότε προστίθενται τέσσερις νέοι ράβδοι, μία σε κάθε παρειά και η διαδικασία επαναλαμβάνεται έως ότου είτε καλυφθεί η απαίτηση σε οπλισμό, είτε ξεπεραστεί το όριο για το μέγιστο ποσοστό οπλισμού. Η παραπάνω διαδικασία παρουσιάζεται στο διάγραμμα ροής του Σχήματος 2-17.

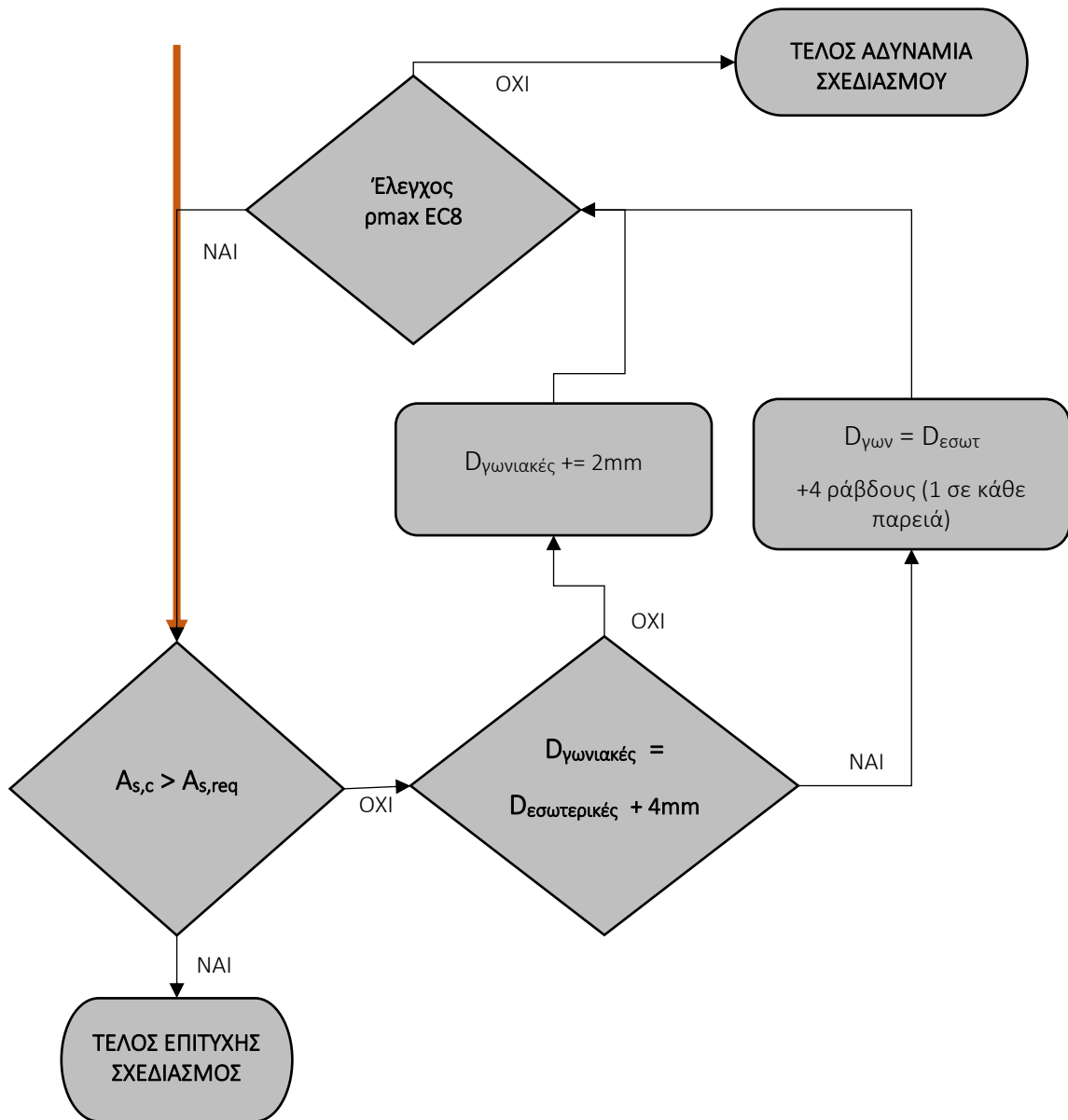
Ο κώδικας που αναπτύχθηκε για όλη την παραπάνω διαδικασία της διαστασιολόγησης που περιεγράφηκε παραπάνω, παρατίθεται στο Παράρτημα Β. Στη συνέχεια, παρουσιάζονται τα διαγράμματα ροής του αλγορίθμου της διαστασιολόγησης. Το Σχήμα 2-15 αναφέρεται στον οπλισμό των δοκών, το Σχήμα 2-16 στον οπλισμό των υποστυλωμάτων ενώ το Σχήμα 2-17 αναλύει περαιτέρω τη διαδικασία όπλισης των υποστυλωμάτων, στο τελευταίο στάδιο του αλγορίθμου.



Σχήμα 2-15. Διάγραμμα ροής αλγορίθμου διαστασιολόγησης δοκού



Σχήμα 2-16. Διάγραμμα ροής διαστασιολόγησης υποστυλωμάτων



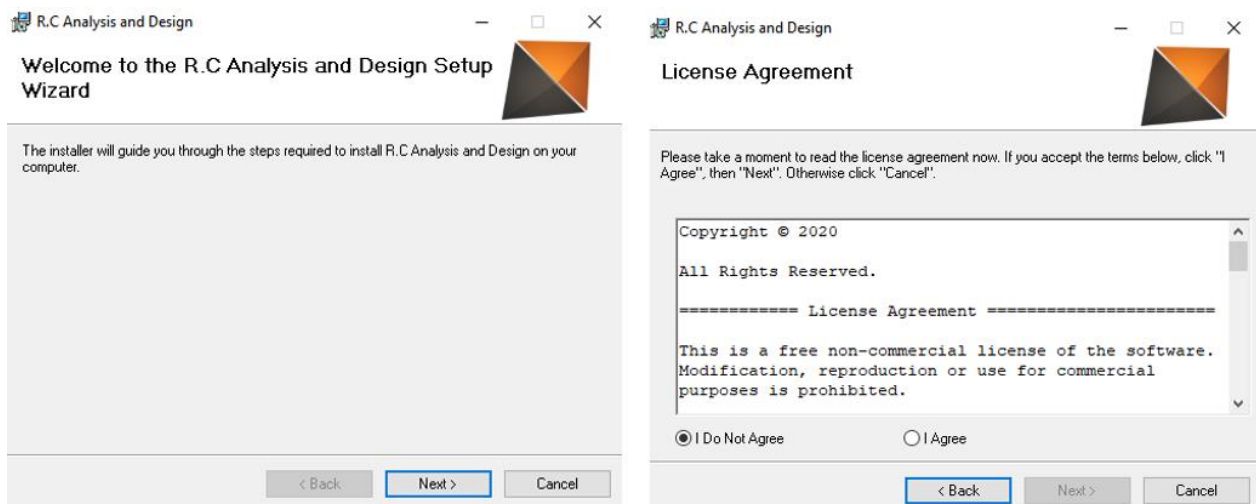
Σχήμα 2-17. Διάγραμμα ροής τελικού βήματος διαστασιολόγησης υποστλωμάτων

## Κεφάλαιο 3: Παρουσίαση Λογισμικού Domisis R.C.

Στο κεφάλαιο αυτό παρουσιάζεται το λογισμικό που δημιουργήθηκε για τις ανάγκες της παρούσας διπλωματικής εργασίας.

### 3.1 Εγκατάσταση

Για τη χρήση του προγράμματος, όπως και κάθε νέου προγράμματος που θέλουμε να χρησιμοποιήσουμε στον υπολογιστή μας, απαιτείται εγκατάσταση μέσω ενός προγράμματος εγκατάστασης (installer, setup wizard). Το πρόγραμμα εγκατάστασης του Domisis R.C. εγκαθιστά όλα τα απαραίτητα αρχεία στο σκληρό δίσκο του υπολογιστή ώστε να μπορεί το λογισμικό να λειτουργήσει. Προϋποθέσεις για την επιτυχή εγκατάσταση είναι οι εξής: α) να υπάρχει αρκετός αποθηκευτικός χώρος στον υπολογιστή, β) το λειτουργικό σύστημα του υπολογιστή να είναι Microsoft Windows 7 ή νεότερο, γ) να υπάρχει εγκατεστημένη η έκδοση .NET framework 4.6 ή νεότερη, και δ) να αποδεχτεί ο χρήστης τους όρους χρήσης του προγράμματος. Στους όρους χρήσης αναφέρεται μεταξύ άλλων ότι δεν είναι εγγυημένη η ορθή λειτουργία του προγράμματος καθώς και η ορθότητα των αποτελεσμάτων, ενώ η χρήση του προορίζεται για αμιγώς ακαδημαϊκούς σκοπούς. Σημαντικό είναι επίσης να σημειωθεί ότι το πρόγραμμα δεν έχει κάποιο ηλεκτρονικό πιστοποιητικό και συνεπώς το λειτουργικό σύστημα κατά την εγκατάσταση θα ενημερώσει τον χρήστη για τους κινδύνους χρήσης του λογισμικού. Το περιβάλλον εγκατάστασης παρουσιάζεται στο Σχήμα 3-1.



Σχήμα 3-1. Το περιβάλλον εγκατάστασης του λογισμικού Domisis R.C.

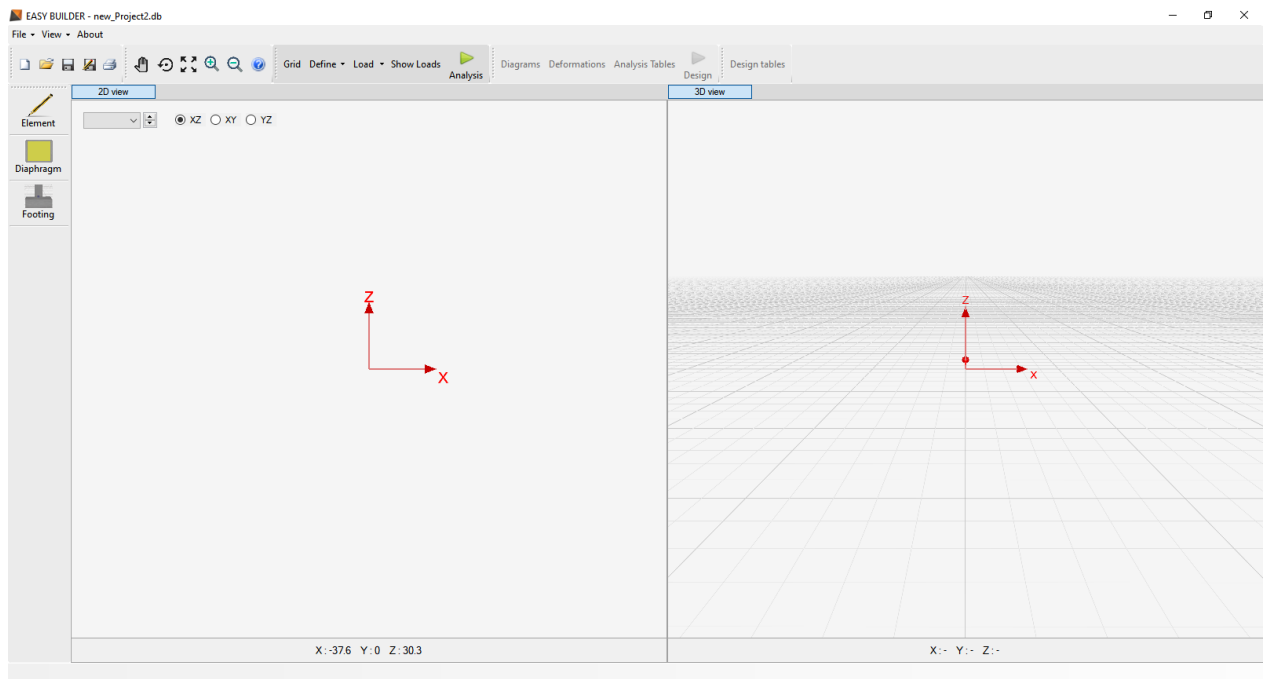
Μετά την επιτυχή εγκατάσταση του λογισμικού, δημιουργείται αυτόματα στην επιφάνεια εργασίας μία συντόμευση του εκτελέσιμου αρχείου (Σχήμα 3-2), από το οποίο εκτελείται το λογισμικό.



Σχήμα 3-2. Συντόμευση του εκτελέσιμου αρχείου στην επιφάνεια εργασίας

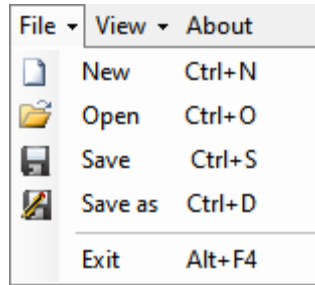
### 3.2 Περιβάλλον του χρήστη

Κατά την εκκίνηση του λογισμικού εμφανίζεται το κύριο παράθυρο στο οποίο περιέχονται τα εργαλεία χειρισμού του καθώς επίσης και τα γραφικά απεικόνιση του προσομοιώματος της κατασκευής (Σχήμα 3-3).



Σχήμα 3-3. Αρχικό παράθυρο του προγράμματος.

Κατά την εκκίνηση, ο χρήστης έχει τη δυνατότητα είτε να φορτώσει προηγούμενα αποθηκευμένα αρχεία που δημιουργήθηκαν με το ίδιο λογισμικό, είτε να δημιουργήσει ένα καινούργιο αρχείο. Οι ενέργειες αυτές είναι δυνατές είτε πάνω από την επιλογή File (Σχήμα 3-4), είτε από τη γραμμή εργαλείων γρήγορης πρόσβασης (Σχήμα 3-5).



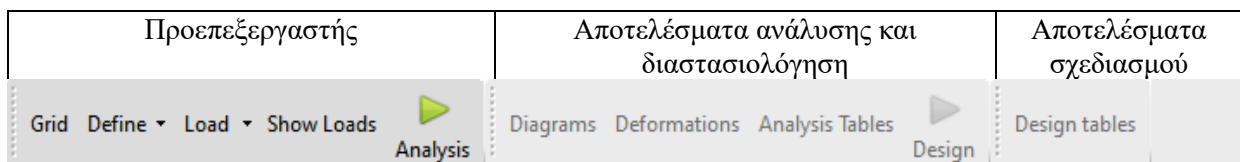
Σχήμα 3-4. Η επιλογή File και οι επιλογές διαχείρισης αρχείων.



Σχήμα 3-5. Η εργαλειοθήκη γρήγορης πρόσβασης λειτουργιών διαχείρισης αρχείων.

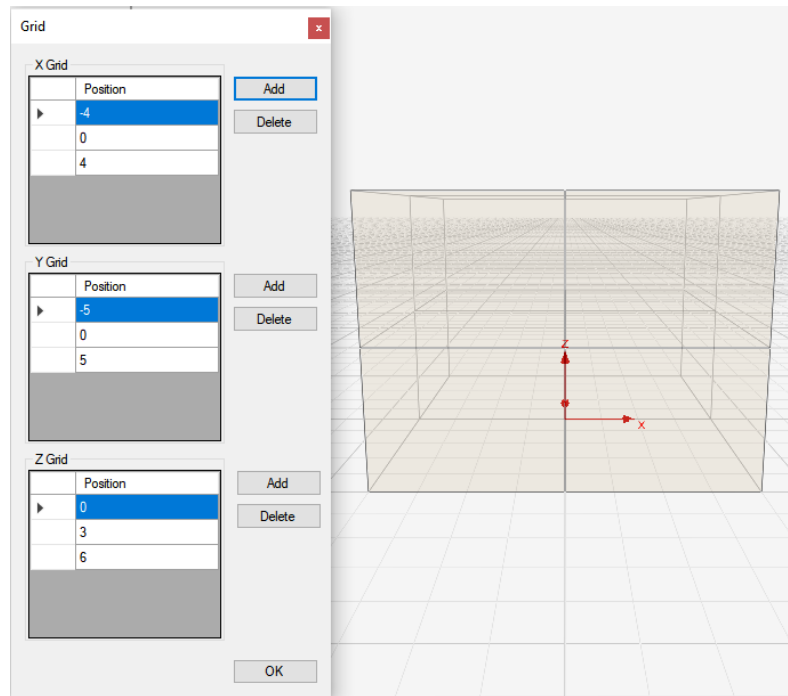
Κατά την αποθήκευση μιας εργασίας, αποθηκεύονται μόνο τα δεδομένα σχετικά με την προσομοίωση της κατασκευής και των συνθηκών φόρτισης και όχι τα αποτελέσματα της ανάλυσης και του σχεδιασμού.

Κατά το στάδιο του προεπεξεργαστή, η κύρια γραμμή εργαλείων στο Σχήμα 3-6 έχει ενεργοποιημένες μόνο τις επιλογές που σχετίζονται με τη δημιουργία και την επεξεργασία του προσομοιώματος, καθώς επίσης και με τις συνθήκες φόρτισης αυτού. Οι υπόλοιπες επιλογές για την προβολή των αποτελεσμάτων της ανάλυσης και του σχεδιασμού ενεργοποιούνται μετά το στάδιο ανάλυσης και σχεδιασμού αντίστοιχα.



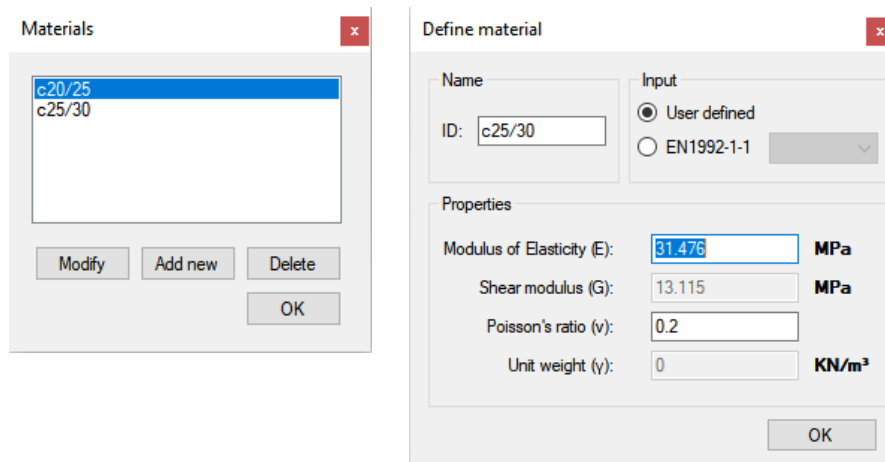
Σχήμα 3-6. Γραμμή εργαλείων στο πάνω μέρος του παραθύρου.

Από τη γραμμή εργαλείων, ο χρήστης θα πρέπει πρώτα να προσδιορίσει τον βοηθητικό κানাβο σχεδίασης από την επιλογή Grid. Με την επιλογή αυτή, εμφανίζεται μία δευτερεύουσα φόρμα καθορισμού της γεωμετρίας του κανάβου σχεδίασης, όπου ο χρήστης καλείται να ορίσει τον αριθμό των γραμμών και τις μεταξύ τους αποστάσεις (Σχήμα 3-7). Με το κλείσιμο της φόρμας, ο κানাβος σχεδίασης εμφανίζεται στο παράθυρο γραφικών.



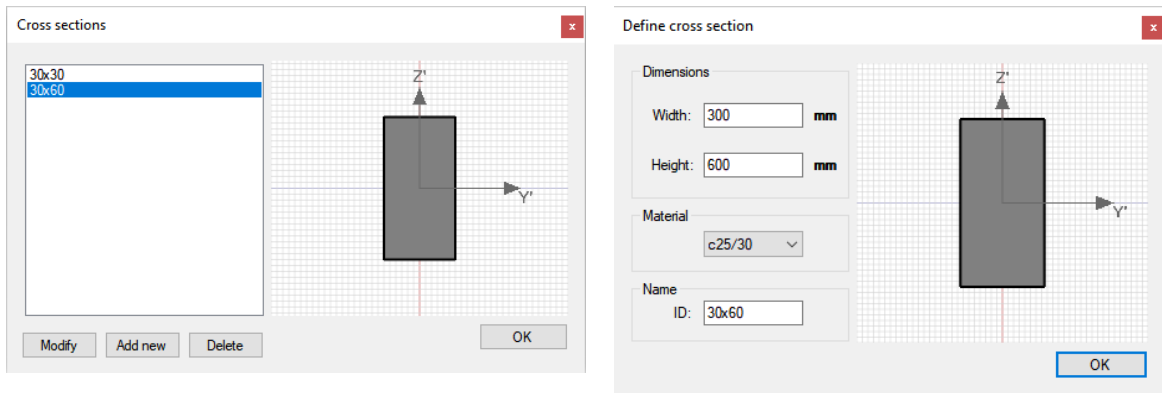
Σχήμα 3-7. Καθορισμός κανάβου σχεδίασης.

Προτού ξεκινήσει η διαδικασία σχεδιασμού του φορέα, ο χρήστης θα πρέπει να ορίσει τα υλικά και τις διατομές των μελών του. Για τη διαδικασία αυτή μεταβαίνουμε στις επιλογές Define → Materials και Define → Cross sections, και έπειτα εισάγουμε τα δεδομένα που ζητούνται στις υποφόρμες που εμφανίζονται (Σχήμα 3-8, Σχήμα 3-9).



Σχήμα 3-8. Φόρμες εισαγωγής και επεξεργασίας διαθέσιμων υλικών.

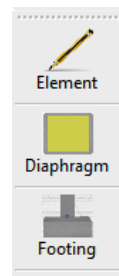




Σχήμα 3-9. Φόρμες εισαγωγής και επεξεργασίας διαθέσιμων διατομών.

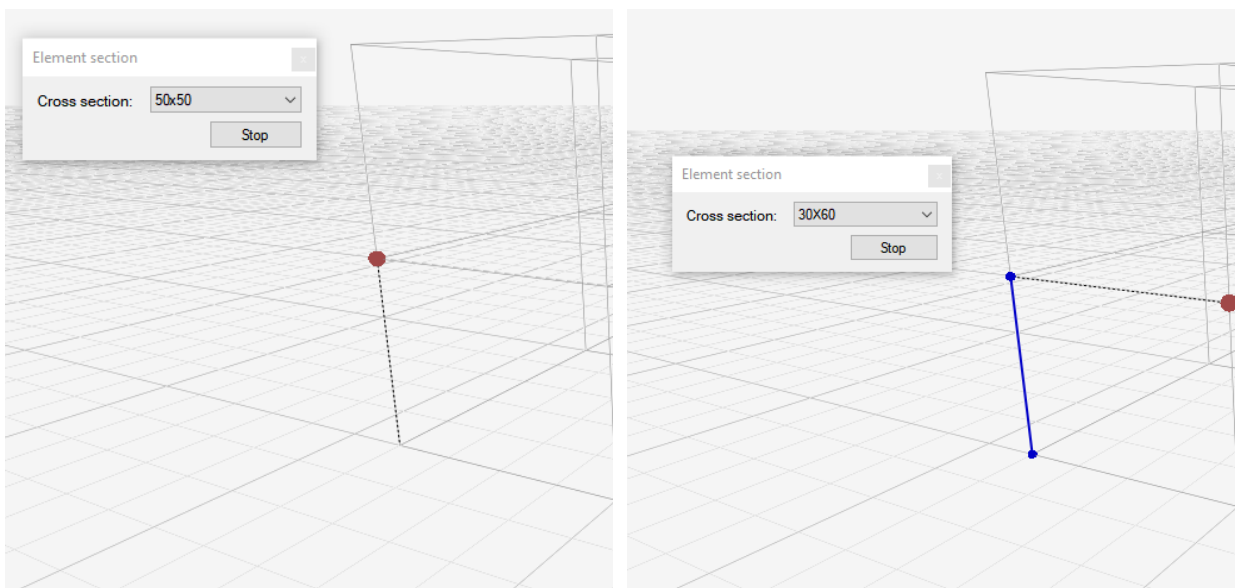
Για τον προσδιορισμό των χαρακτηριστικών μεγεθών ενός υλικού, ο χρήστης μπορεί να εισάγει μόνος του τιμές ή να επιλέξει μεταξύ των έτοιμων τιμών που υπάρχουν για τις ποιότητες σκυροδέματος που ορίζει ο EC2.

Στη συνέχεια, μπορεί να σχεδιαστεί ο φορέας του προσομοιώματος, αξιοποιώντας το γραφικό περιβάλλον σχεδίασης και την αριστερή γραμμή εργαλείων του παραθύρου (Σχήμα 3-10).

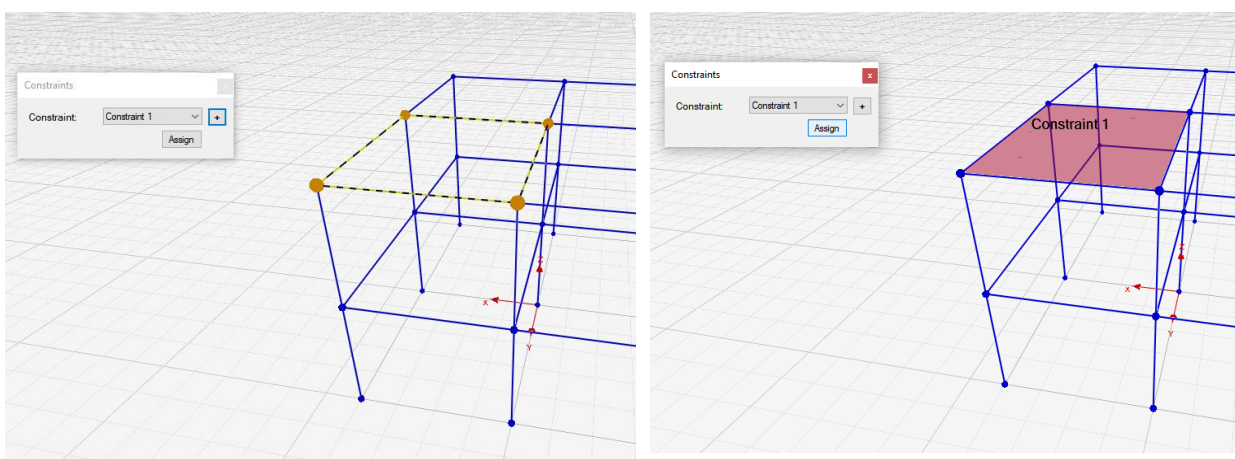


Σχήμα 3-10. Γραμμή εργαλείων σχεδίασης του φορέα του προσομοιώματος

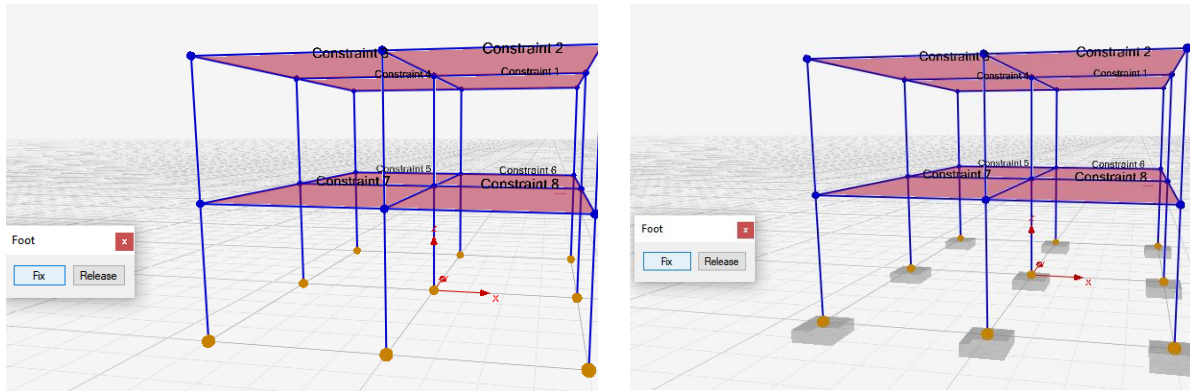
Πιο συγκεκριμένα, με την επιλογή Element σχεδιάζονται τα γραμμικά στοιχεία του φορέα, με την επιλογή Diaphragm δεσμεύονται οι επιλεγμένοι κόμβοι με διαφραγματική λειτουργία πλάκας και με την επιλογή Footing τοποθετούνται στηρίξεις με τη μορφή πάκτωσης (δέσμευση και των 6 β.ε. του κόμβου) στις θέσεις των επιλεγμένων κόμβων. Η επιλογή της πάκτωσης για την προσομοίωση των στηρίξεων αποτελεί μία απλοποιητική παραδοχή, χωρίς να λαμβάνεται υπόψη η αλληλεπίδραση εδάφους - κατασκευής. Όσον αφορά την εισαγωγή διαφραγμάτων, αυτή συνιστάται να μη γίνεται ενιαία για κάθε όροφο αλλά ξεχωριστά για κάθε διάφραγμα που ορίζουν οι δοκοί της κάτοψης (για την ελαχιστοποίηση του αριθμού των εικονικών δοκών που θα δημιουργηθούν). Η παραπάνω διαδικασία παρουσιάζεται στα σχήματα Σχήμα 3-11, Σχήμα 3-12 και Σχήμα 3-13.



Σχήμα 3-11. Σχεδιασμός της γεωμετρίας του φορέα.

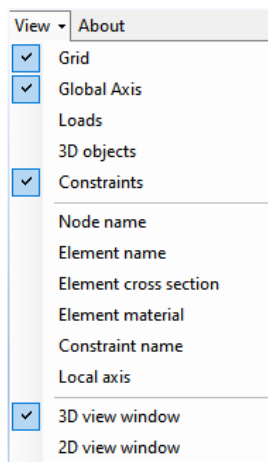


Σχήμα 3-12. Δέσμευση κόμβων κάτοψης με διαφραγματική λειτουργία πλάκας.

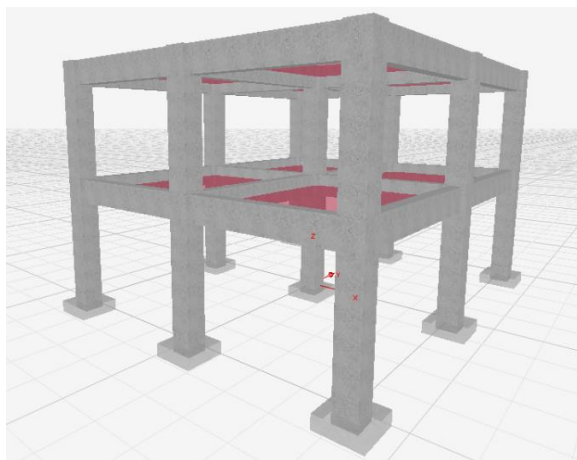


Σχήμα 3-13. Εισαγωγή στηρίξεων.

Στο σημείο αυτό έχουμε ολοκληρώσει τον σχεδιασμό της γεωμετρίας του φορέα. Από την καρτέλα View στο πάνω μέρος του παραθύρου (Σχήμα 3-14) μπορούμε να ενεργοποιήσουμε την προβολή διαφόρων χαρακτηριστικών του φορέα όπως διατομή στοιχείων, τοπικοί άξονες, τρισδιάστατη προβολή μελών κλπ. Στο Σχήμα 3-15 παρακάτω διακρίνεται η τρισδιάστατη προβολή της κατασκευής.

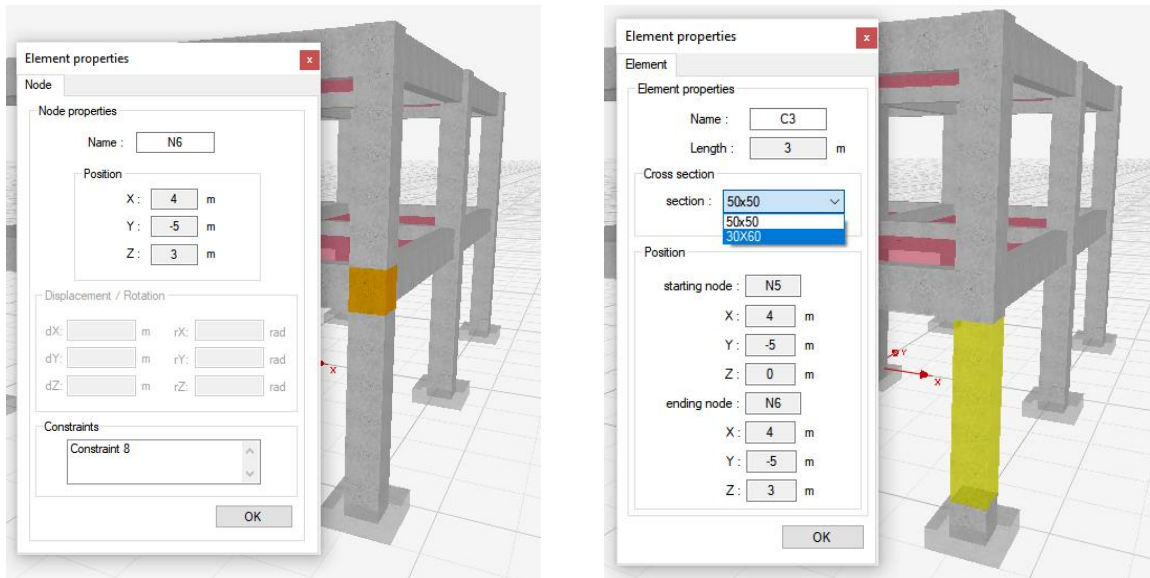


Σχήμα 3-14. Επιλογές προβολής γραφικών στην καρτέλα View.



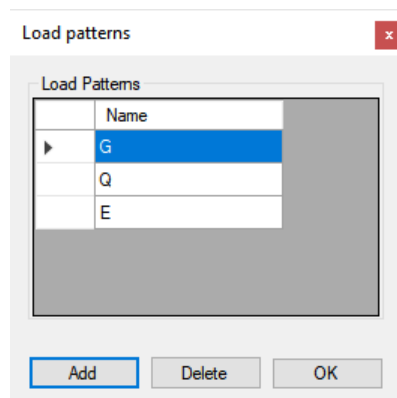
Σχήμα 3-15. Τρισδιάστατη προβολή της κατασκευής.

Ακόμα, επιλέγοντας μέλη και κόμβους, μπορούμε με δεξί κλικ να δούμε πληροφορίες για τη θέση τους, το όνομα τους, τη διατομή του μέλους κλπ. Κάποια πεδία, όπως το όνομα και η διατομή ενός μέλους, μπορούν να επεξεργαστούν (Σχήμα 3-16).



Σχήμα 3-16. Καρτέλα πληροφοριών επιλεγμένου στοιχείου.

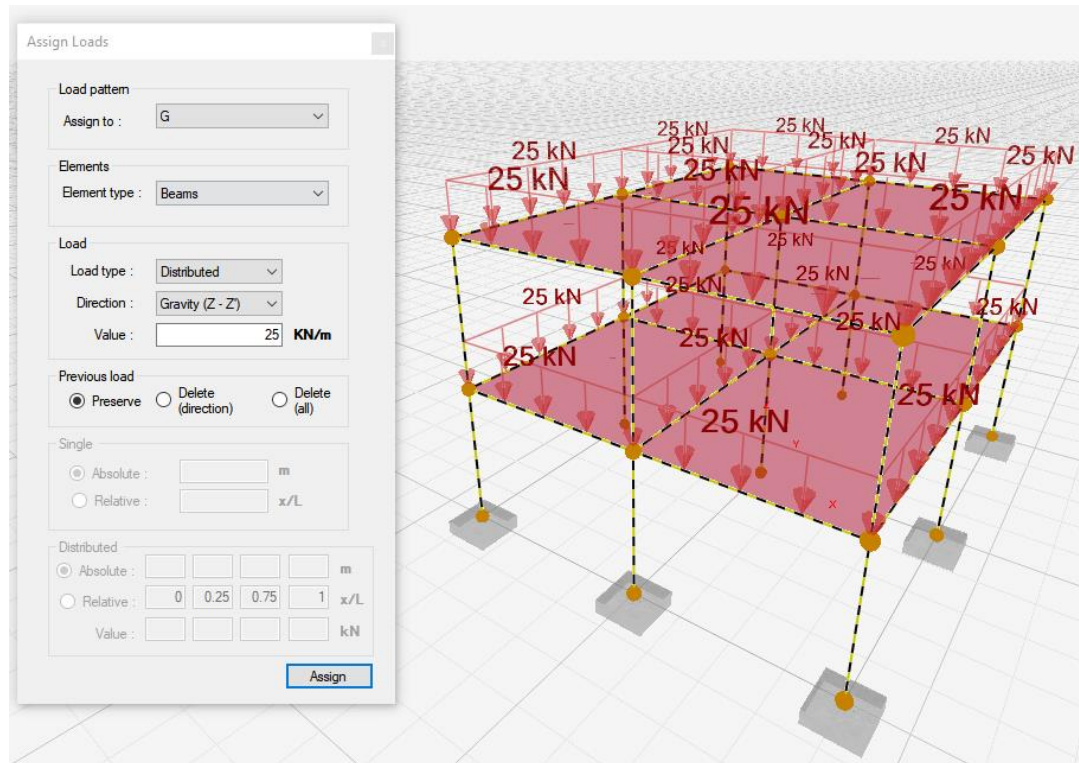
Για την εισαγωγή των συνθηκών φόρτισης της κατασκευής, πρέπει πρώτα να οριστούν τα προφίλ φόρτισης (Load patterns) από την επιλογή Load → Load patterns. Στη φόρμα που εμφανίζεται (Σχήμα 3-17), δημιουργούνται τα απαραίτητα προφίλ στα οποία θα αναθέτει ο χρήστης τα φορτία που εφαρμόζει στην κατασκευή και έπειτα με αυτά τα προφίλ θα δημιουργήσει τους απαραίτητους συνδυασμούς φόρτισης που απαιτούνται για την ανάλυση και αργότερα για τη διαστασιολόγηση.



Σχήμα 3-17. Καρτέλα των προφίλ φόρτισης.

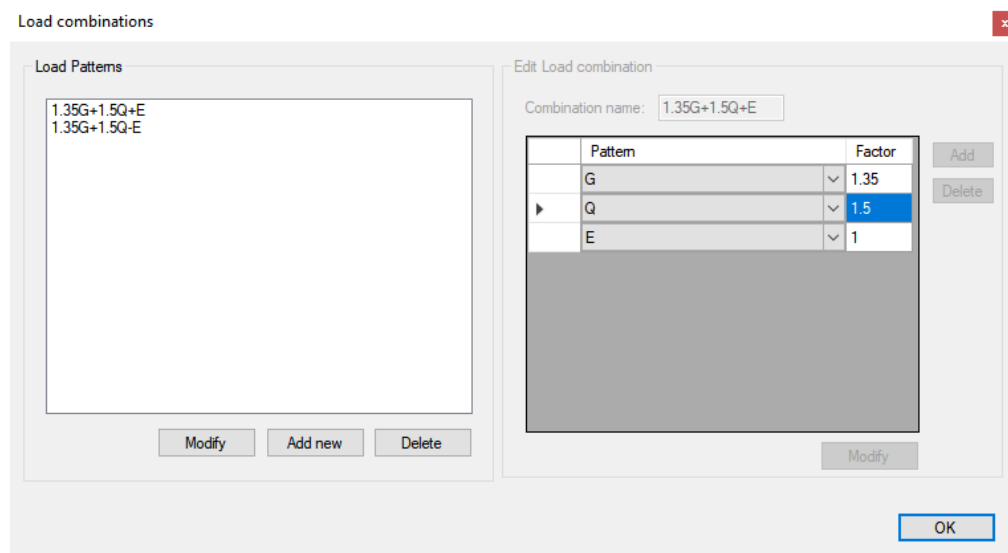
Έπειτα, με την επιλογή Load → Assign loads, ο χρήστης μπορεί να φορτίσει τα στοιχεία που επιλέγει (κόμβους, μέλη) με επικόμβια ή κατανεμημένα φορτία αντίστοιχα. Η κάθε φόρτιση που

εφαρμόζεται εισάγεται στο επιλεγμένο προφίλ που επιλέγεται. Η δυνατότητα προσθήκης συγκεντρωμένου φορτίου κατά μήκος ενός στοιχείου στην παρούσα έκδοση του προγράμματος είναι απενεργοποιημένη. Στο Σχήμα 3-18 φαίνεται η διαδικασία επιβολής φόρτισης στον φορέα της κατασκευής.



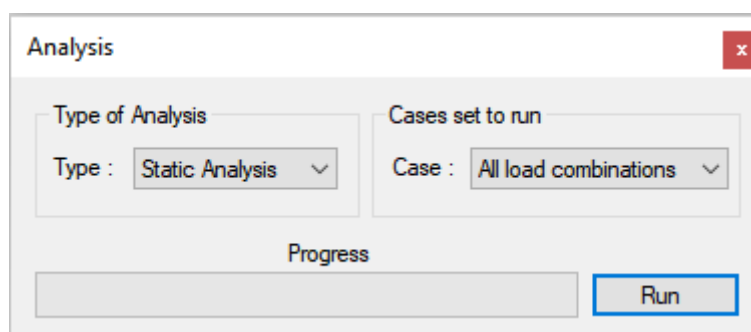
Σχήμα 3-18. Επιβολή φόρτισης στην κατασκευή μέσα από την καρτέλα Assign Loads.

Η διαδικασία καθορισμού των συνθηκών φόρτισης της κατασκευής ολοκληρώνεται με τη δημιουργία των απαραίτητων συνδυασμών φόρτισης. Για τον σκοπό αυτό, ο χρήστης πρέπει να μεταβεί στην καρτέλα Load combinations (Σχήμα 3-19), μέσω της επιλογής Load → Load combinations. Στη φόρμα αυτή καθορίζονται όλοι οι συνδυασμοί, με τον κάθε συνδυασμό να μορφώνεται μέσω της επιλογής των απαραίτητων προφίλ φόρτισης και του αντίστοιχου συντελεστή συμμετοχής τους στον συνδυασμό. Μέσω αυτών των συνδυασμών, προκύπτει η περιβάλλουσα κάθε εντατικού μεγέθους στο στάδιο της ανάλυσης.



Σχήμα 3-19. Καρτέλα συνδυασμού φορτίσεων.

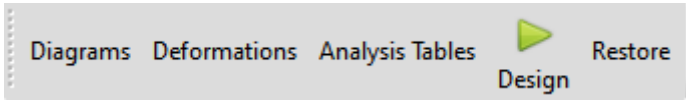
Η στατική ανάλυση γίνεται μέσω της επιλογής Analysis από τη γραμμή εργαλείων, όπου και εμφανίζεται η αντίστοιχη καρτέλα που μας ενημερώνει για την πορεία της διαδικασίας μέσω μίας γραμμής προόδου (progression bar), όπως φαίνεται στο Σχήμα 3-20.



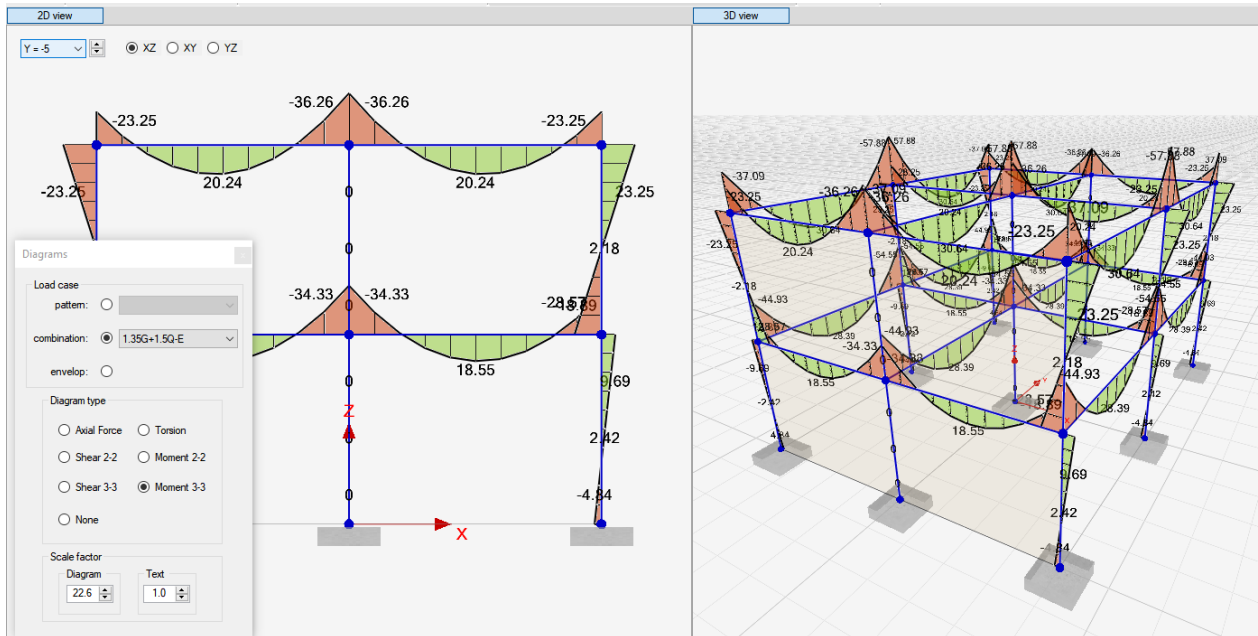
Σχήμα 3-20. Φόρμα ανάλυσης.

Με την επιτυχή ολοκλήρωση της ανάλυσης, γίνονται διαθέσιμα στον χρήστη όλα τα αποτελέσματα αυτής μέσα από τη γραμμή εργαλείων που ενεργοποιείται (Σχήμα 3-21). Ταυτόχρονα, ανοίγει η καρτέλα διαγραμμάτων των εντατικών μεγεθών και εμφανίζονται τα διαγράμματα στην οθόνη μέσω του πάνελ γραφικών (Σχήμα 3-22). Ο χρήστης μπορεί να επιλέξει για προβολή κάθε ένα από τα διαθέσιμα διαγράμματα, καθώς επίσης και να ρυθμίσει την κλίμακα σχεδίασης των διαγραμμάτων και των αναγραφόμενων τιμών τους.



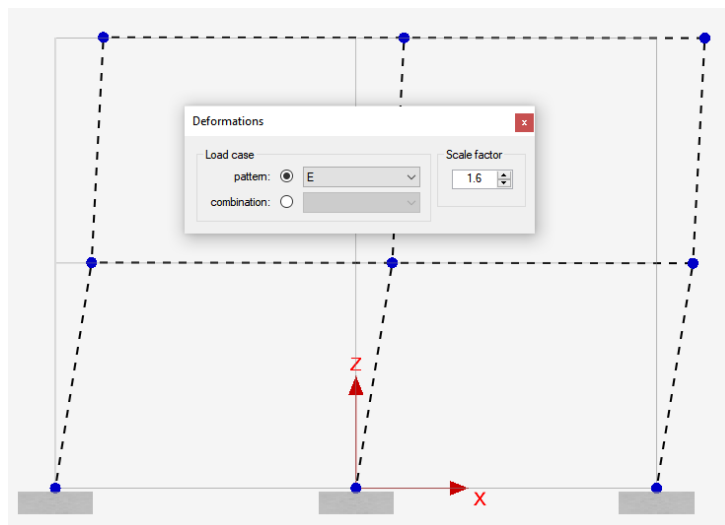


Σχήμα 3-21. Γραμμή εργαλείων προβολής αποτελεσμάτων ανάλυσης.

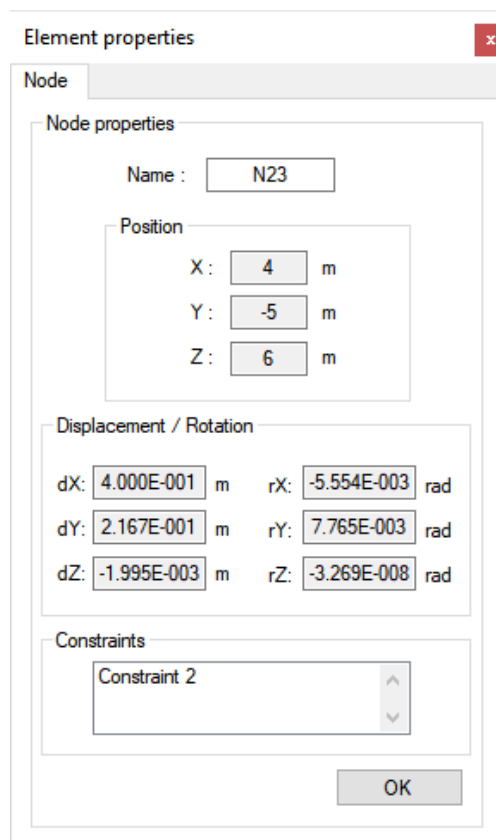


Σχήμα 3-22. Διαγράμματα εντατικών μεγεθών.

Ακόμα, διαθέσιμο στα αποτελέσματα είναι το σχήμα παραμόρφωσης της κατασκευής (Σχήμα 3-23), (χωρίς την ελαστική γραμμή των μελών), μέσω της επιλογής Deformations από τη γραμμή εργαλείων. Με την προβολή του σχήματος παραμόρφωσης του φορέα ενεργοποιημένη, είναι δυνατή η επιλογή των κόμβων και με δεξί κλικ πάνω τους εμφανίζεται η καρτέλα πληροφοριών του κόμβου στην οποία περιέχονται και οι τιμές των μετακινήσεων και των στρεφών, για την προβαλλόμενη πάντα περίπτωση φόρτισης (Σχήμα 3-24).



Σχήμα 3-23. Μετακινήσεις κόμβων.



Σχήμα 3-24. Μετακινήσεις και στροφές επιλεγμένου κόμβου.

Μια ακόμη εξαιρετικά σημαντική δυνατότητα του λογισμικού, είναι η παρουσίαση των πινάκων της ανάλυσης (Analysis tables), οι οποίοι περιέχουν τις τιμές από τα μητρώα που υπολογίστηκαν κατά τη διαδικασία λειτουργίας του αλγορίθμου της ανάλυσης. Με την επιλογή Analysis Tables από τη γραμμή εργαλείων εμφανίζεται η καρτέλα προβολής όλων των διαθέσιμων



πινάκων. Στον πίνακα Σχήματος 3-25 φαίνεται ένα απόσπασμα από το μητρώο δυσκαμψίας της κατασκευής, όπως αυτό υπολογίστηκε από τον αλγόριθμο της ανάλυσης.

Analysis Tables

Matrix: K\_Structure

Description: Stiffness matrix of the whole structure.

	1	2	3	4	5	6	7	8	9
1	2808904...	3233103...	0	0	0	-212584...	-150000...	0	0
2	3233103...	3003394...	0	0	0	2798512...	0	-112500...	0
3	0	0	16819.7...	129.800...	-202.50...	0	0	0	-101.25...
4	0	0	129.800...	1830.15...	0	0	0	0	0
5	0	0	-202.50...	0	1938.30...	0	0	0	202.500...
6	-212584...	2798512...	0	0	0	1454817...	0	-225000...	0
7	-150000...	0	0	0	0	0	5217809...	0	0
8	0	-112500...	0	0	0	-225000...	0	6008788...	0
9	0	0	-101.25...	0	202.500...	0	0	0	16921.0...

Σχήμα 3-25. Απόσπασμα του μητρώου δυσκαμψίας της κατασκευής, διαθέσιμο από τους πίνακες ανάλυσης.

Ο χρήστης μπορεί να επανέλθει στον προεπεξεργαστή για να μετασηματίσει το προσομοίωμα μέσω της επιλογής Restore στη γραμμή εργαλείων, ή να προχωρήσει στο στάδιο του σχεδιασμού μέσω της επιλογής Design. Για τον σχεδιασμό απαιτείται η συμπλήρωση των πεδίων που απεικονίζονται στο Σχήμα 3-26 και έπειτα εκκινείται ο αλγόριθμος της διαστασιολόγησης. Τα απαραίτητα δεδομένα που πρέπει να εισάγει ο χρήστης είναι η κατηγορία πλαστιμότητας, η επικάλυψη του σκυροδέματος, ο δείκτης πλαστιμότητας  $\mu_\phi$ , η θλιπτική αντοχή του σκυροδέματος, η χαρακτηριστική τιμή του ορίου διαρροής και το μέτρο ελαστικότητας του χάλυβα οπλισμού και τέλος οι διαθέσιμες διαμέτροι των ράβδων οπλισμού που θα χρησιμοποιηθούν από τον αλγόριθμο.

Σχήμα 3-26. Πεδία τιμών για τα δεδομένα του αλγορίθμου διαστασιολόγησης.

Με την επιτυχή ολοκλήρωση της διαδικασίας διαστασιολόγησης εμφανίζεται αυτόματα ο πίνακας αποτελεσμάτων όπου ο χρήστης μπορεί να προβάλλει τον οπλισμό που τοποθετήθηκε από τον αλγόριθμο, για κάθε δοκό και υποστύλωμα της κατασκευής. Στο Σχήμα 3-27 παρουσιάζεται ο πίνακας οπλισμού για την περίπτωση υποστυλώματος, ενώ στο Σχήμα 3-28 για την περίπτωση δοκού.

Design table

Type: Element design | Element: C1

	Corners	Mid	Prov. steel area (mm <sup>2</sup> )	Design moment resistance (kNm)
Horizontal (X-X)	2Φ18	2Φ16	911.061884835362	243.7915
Vertical (Y-Y)	2Φ18	2Φ16	911.061884835362	243.7915

Σχήμα 3-27. Πίνακας αποτελεσμάτων διαστασιολόγησης υποστυλώματος.

	Continuous	Added	Prov. steel area (mm <sup>2</sup> )	Design moment resistance (kNm)
▶ Left end, top	2Φ20	1Φ20	942	86.05231
Left end, bottom	2Φ20	None	628	60.88159
Mid-span, top	2Φ20	None	628	60.88159
Mid-span, bottom	2Φ20	None	628	60.88159
Right end, top	2Φ20	1Φ20	942	86.05231
Right end, bottom	2Φ20	None	628	60.88159

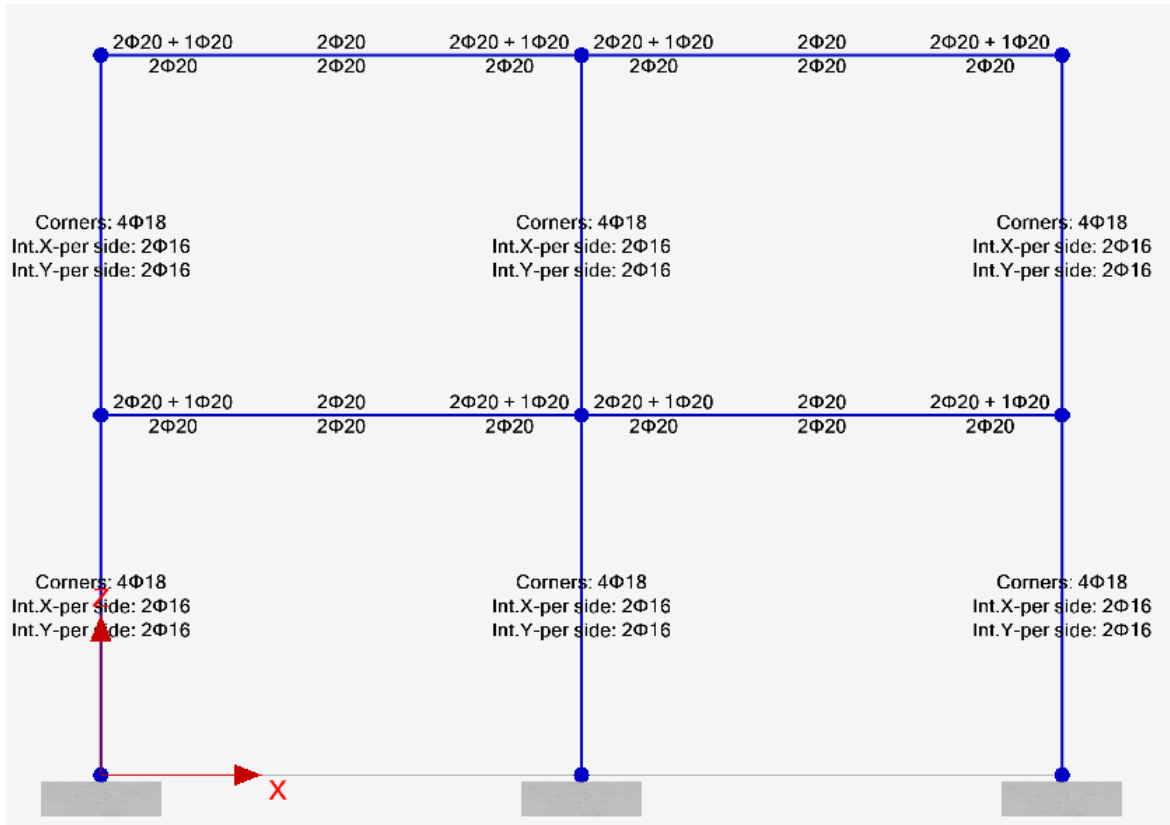
Σχήμα 3-28. Πίνακας αποτελεσμάτων διαστασιολόγησης δοκού.

Επίσης, από τον πίνακα αυτό μπορεί γίνει εποπτεία των αποτελεσμάτων του ικανοτικού σχεδιασμού για κάθε κόμβο της κατασκευής όπως παρουσιάζεται στο Σχήμα 3-29.

	1.3 ΣMrd_b x-x (kNm)	ΣMrd_c x-x (kNm)	1.3 ΣMrd_b y-y (kNm)	ΣMrd_c y-y (kNm)	Check
▶ N2	221.2007	304.6035	221.2007	304.6035	Pass
N3	442.4013	537.165	338.4895	537.165	Pass
N4	221.2007	303.0657	221.2007	303.0657	Pass
N5	338.4895	527.6937	442.4013	527.6937	Pass

Σχήμα 3-29. Ικανοτικός έλεγχος στον πίνακα αποτελεσμάτων της διαστασιολόγησης.

Τέλος, τα αποτελέσματα της διαστασιολόγησης εμφανίζονται και στο γραφικό περιβάλλον της κατασκευής. Για τη γραφική παρουσίαση των αποτελεσμάτων της διαστασιολόγησης, το λογισμικό εμφανίζει πάνω σε κάθε στοιχείο πληροφορίες σχετικά με τον υπολογισμένο οπλισμό. Συγκεκριμένα σε κάθε δοκό εμφανίζεται ο αριθμός και η διάμετρος των ράβδων οπλισμού, σε όλες τις θέσεις που αυτός υπολογίστηκε, δηλαδή στα άκρα άνω και κάτω και στο άνοιγμα άνω και κάτω. Πρώτος αναγράφεται ο συνεχής οπλισμός, ο οποίος συνεχίζει από το ένα άκρο στο άλλο, και δεύτερος ο πρόσθετος, χωρίζοντας τους με το σύμβολο (+). Για τα υποστυλώματα, αναγράφονται στο μέσον τους πρώτα η διάμετρος των τεσσάρων γωνιακών ράβδων (Corners), και έπειτα ο αριθμός και η διάμετρος των εσωτερικών ράβδων ανά πλευρά, στη διεύθυνση X (Int.X-per side) και στη διεύθυνση Y (Int.Y-per side) (Σχήμα 3-30).

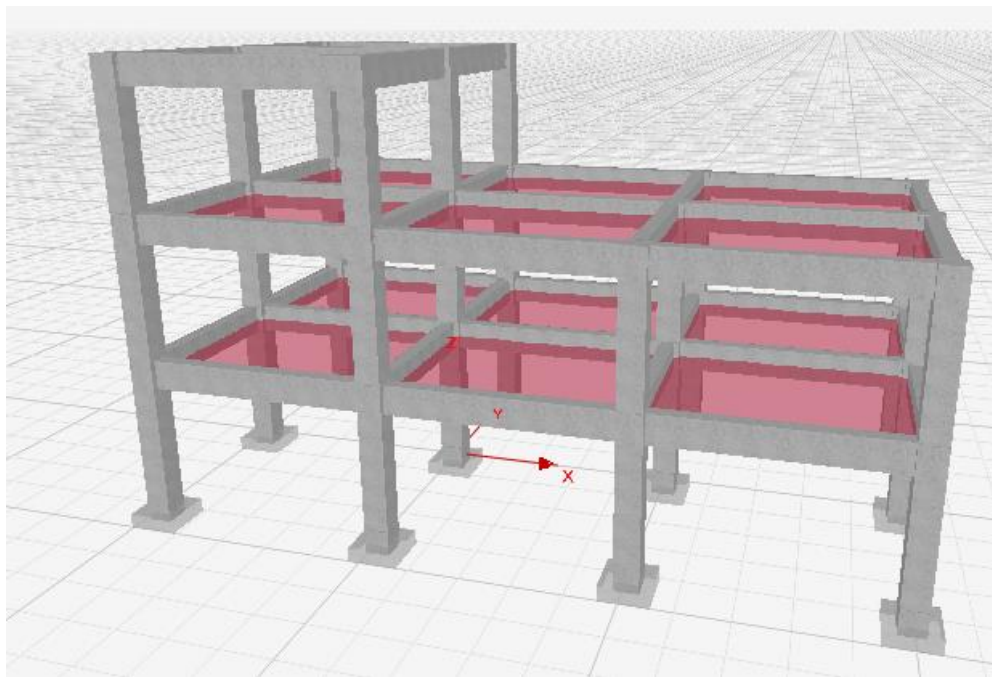


Σχήμα 3-30. Γραφική απεικόνιση του τοποθετούμενου οπλισμού.

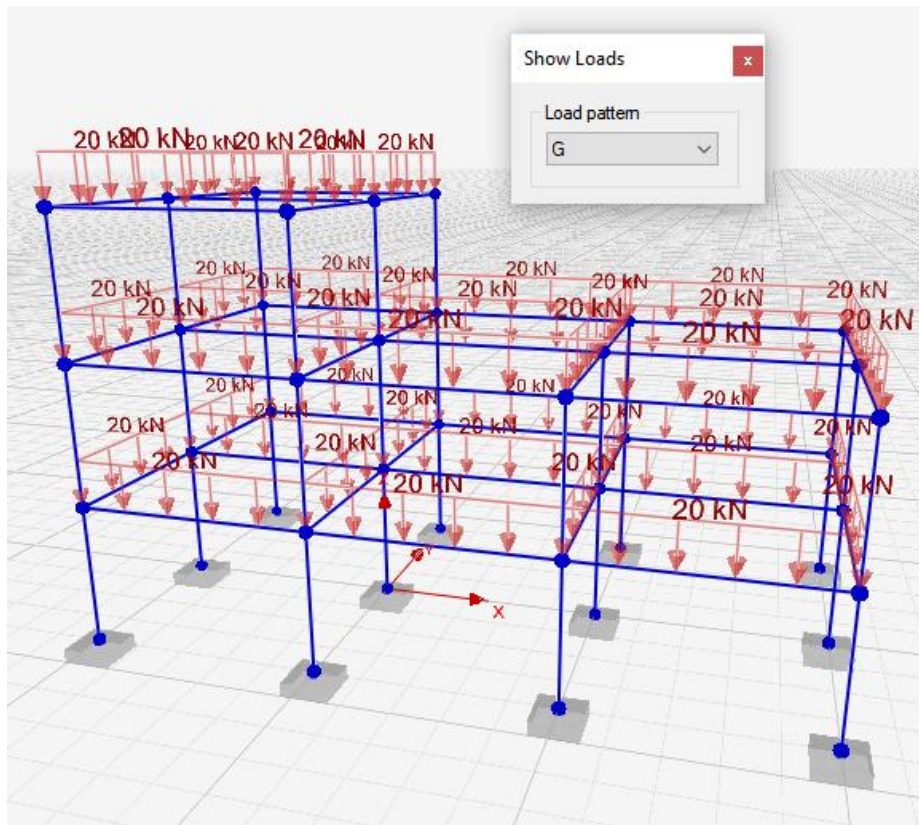
## Κεφάλαιο 4: Αριθμητικό παράδειγμα με χρήση του λογισμικού

### 4.1 Προσομοίωμα

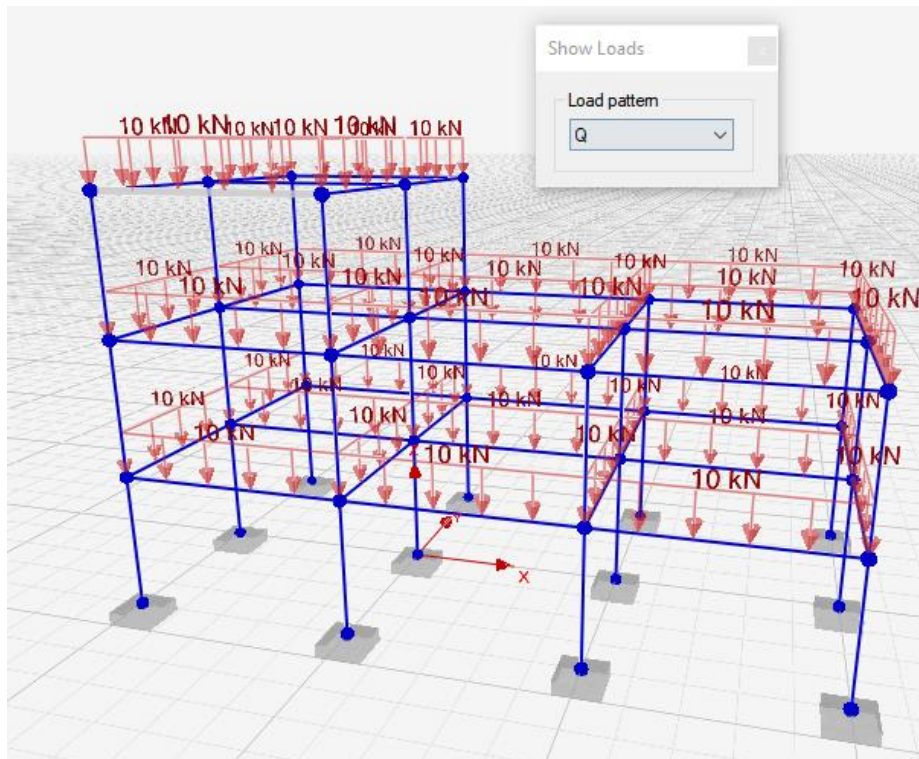
Το προσομοίωμα του παραδείγματος που θα αναλυθεί σε αυτό το κεφάλαιο είναι ένα τριώροφο κτήριο ασύμμετρο καθ' ύψος. Οι δύο πρώτοι όροφοι αποτελούνται από τρία και δύο ανοίγματα στη X και Y διεύθυνση, αντίστοιχα, ενώ ο τελευταίος όροφος αποτελείται από ένα και δύο ανοίγματα στη X και Y διεύθυνση, αντίστοιχα. Κάθε άνοιγμα στη X διεύθυνση έχει μήκος 5 m, ενώ στη Y έχει μήκος 4 m. Το ύψος κάθε ορόφου είναι 3 m. Όλα τα υποστυλώματα έχουν διατομή 50 x 50 cm, ενώ όλες οι δοκοί 60 x 30 cm. Η ποιότητα του σκυροδέματος είναι C20/25 για όλα τα στοιχεία. Σε κάθε όροφο θεωρείται διαφραγματική λειτουργία πλάκας. Όλες οι δοκοί φορτίζονται όμοια με μόνιμα φορτία 20 kN/m και με κινητά 10 kN/m, ενώ για την περίπτωση του σεισμού, αυτός επιβάλλεται με συγκεντρωμένα φορτία μόνο στη X διεύθυνση. Στο διάφραγμα του πρώτου ορόφου ασκείται συνολικό σεισμικό φορτίο 120 kN, του δεύτερου 180 kN και του τρίτου 240 kN. Οι συνδυασμοί φόρτισης που λαμβάνονται υπόψη είναι οι:  $1.35G + 1.5Q + E_x$  και  $1.35G + 1.5Q - E_x$ . Για τη διαστασιολόγηση θεωρείται ενιαία επικάλυψη όλων των στοιχείων 45 mm και κατηγορία πλαστιμότητας υψηλή (ΚΠΥ). Η γεωμετρία του φορέα παρουσιάζεται στο Σχήμα 4-1, ενώ στα Σχήμα 4-2, Σχήμα 4-3 και Σχήμα 4-4 φαίνονται τα προφίλ φορτίσεων.



Σχήμα 4-1. Διάταξη του φορέα του παραδείγματος σε τρισδιάστατη προβολή.

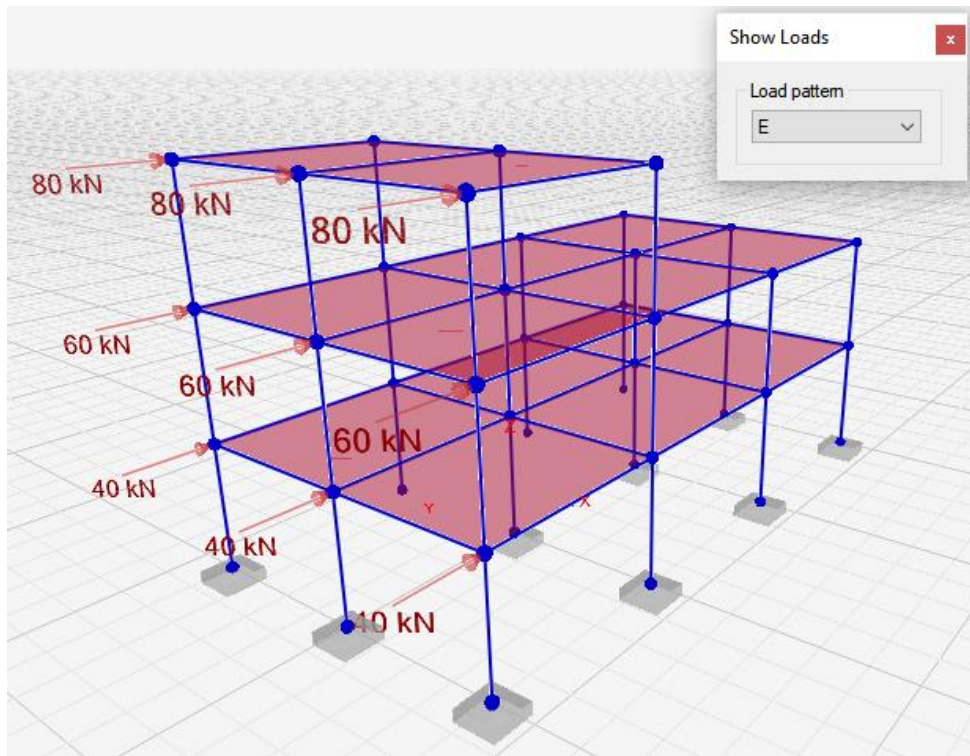


Σχήμα 4-2. Μόνιμα φορτία καταμεμημένα στις δοκούς.



Σχήμα 4-3. Κινητά φορτία καταμεμημένα στις δοκούς.





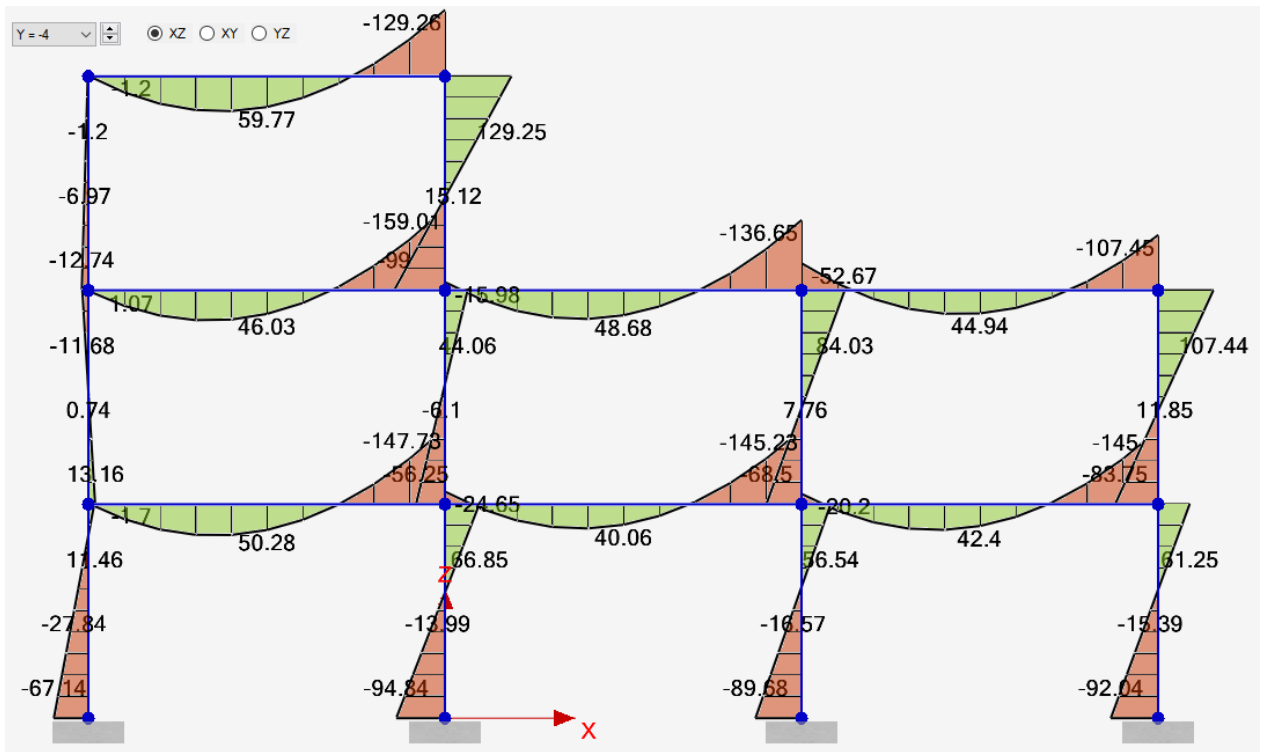
Σχήμα 4-4. Σεισμικά φορτία στη διεύθυνση X, με τη μορφή συγκεντρωμένων φορτίων.

#### 4.2 Αποτελέσματα ανάλυσης

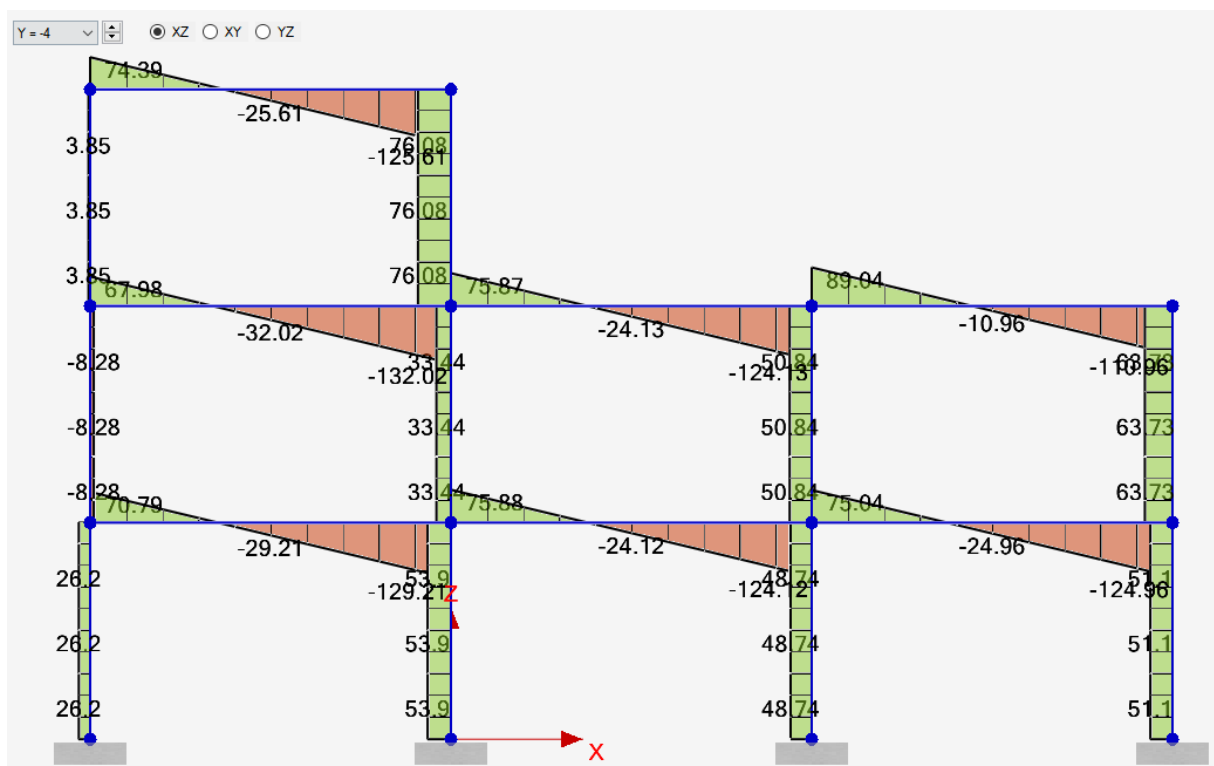
Στα αποτελέσματα παρουσιάζονται για την όψη του κτηρίου στο επίπεδο X-Z, τα διαγράμματα εντατικών μεγεθών για τον συνδυασμό  $1.35G + 1.5Q + E_x$ , η περιβάλλουσα των ροπών από τους δύο συνδυασμούς φόρτισης καθώς και η οριζόντια μετακίνηση της οροφής.



Σχήμα 4-5. Μετακινήσεις στον τελευταίο όροφο για τον συνδυασμό  $1.35G + 1.5Q + E_x$ .

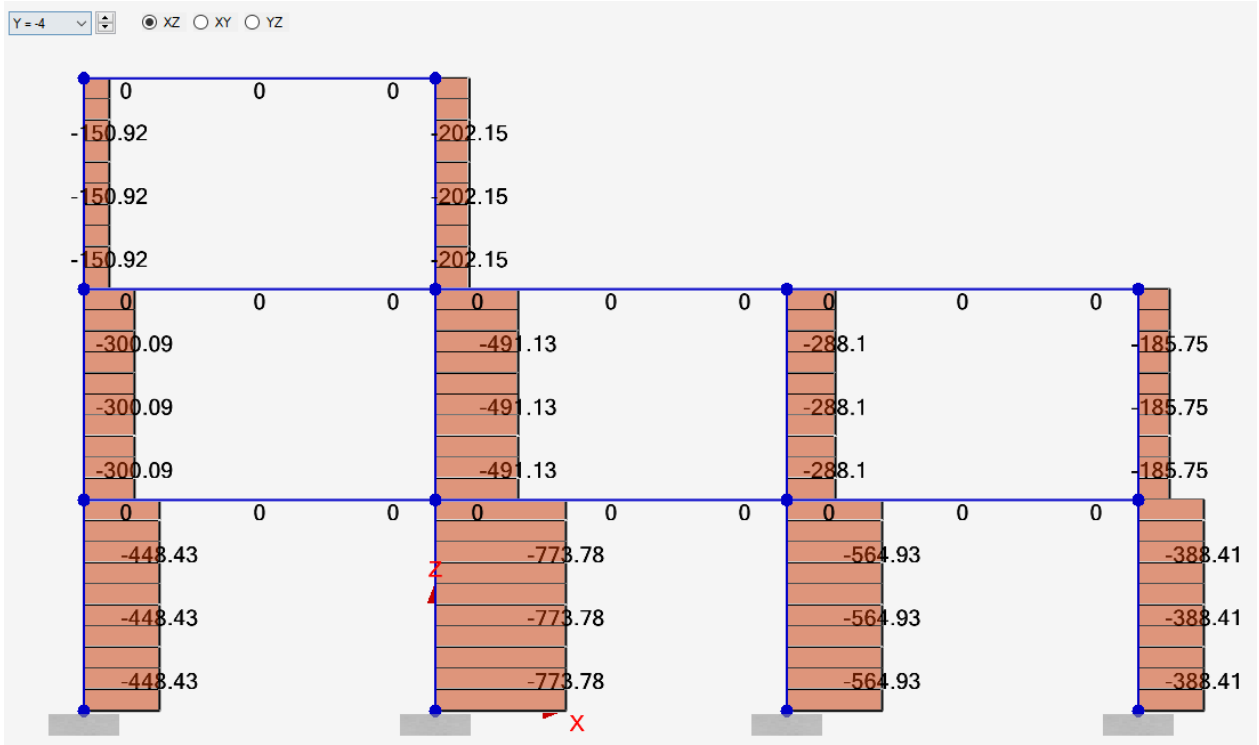


Σχήμα 4-6. Διάγραμμα ροπών για τον συνδυασμό  $1.35G + 1.5Q + Ex$  (μονάδες σε kNm).

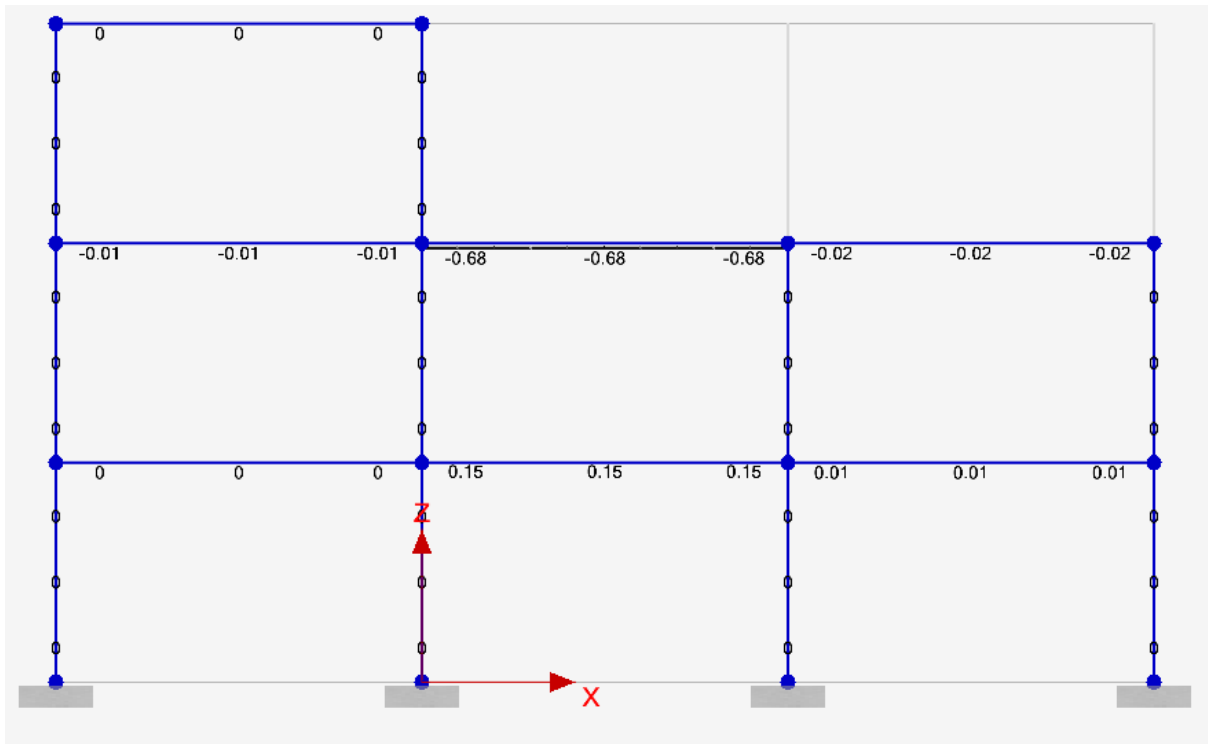


Σχήμα 4-7. Διάγραμμα τεμνουσών για τον συνδυασμό  $1.35G + 1.5Q + Ex$  (μονάδες σε kN).

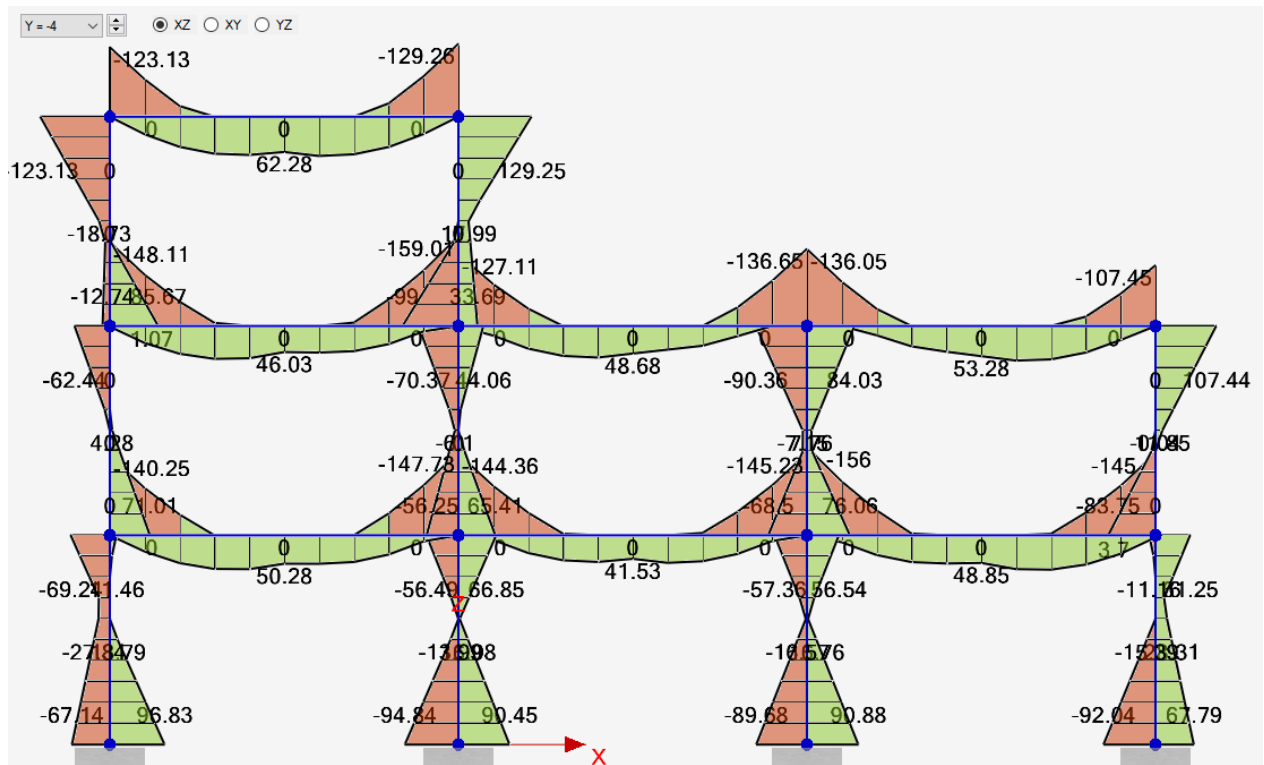




Σχήμα 4-8. Διάγραμμα αξονικών δυνάμεων για τον συνδυασμό  $1.35G + 1.5Q + E_x$  (μονάδες σε kN).

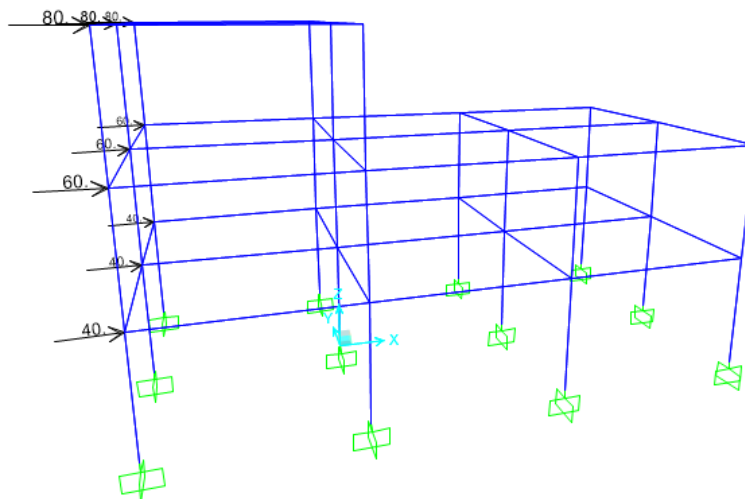


Σχήμα 4-9. Διάγραμμα ροπών στρέψης για τον συνδυασμό  $1.35G + 1.5Q + E_x$  (μονάδες σε kNm)



Σχήμα 4-10. Διάγραμμα περιβάλλουσας των ροπών για τους συνδυασμούς  $1.35G + 1.5Q + E_x$  και  $1.35G + 1.5Q - E_x$  (μονάδες σε kNm).

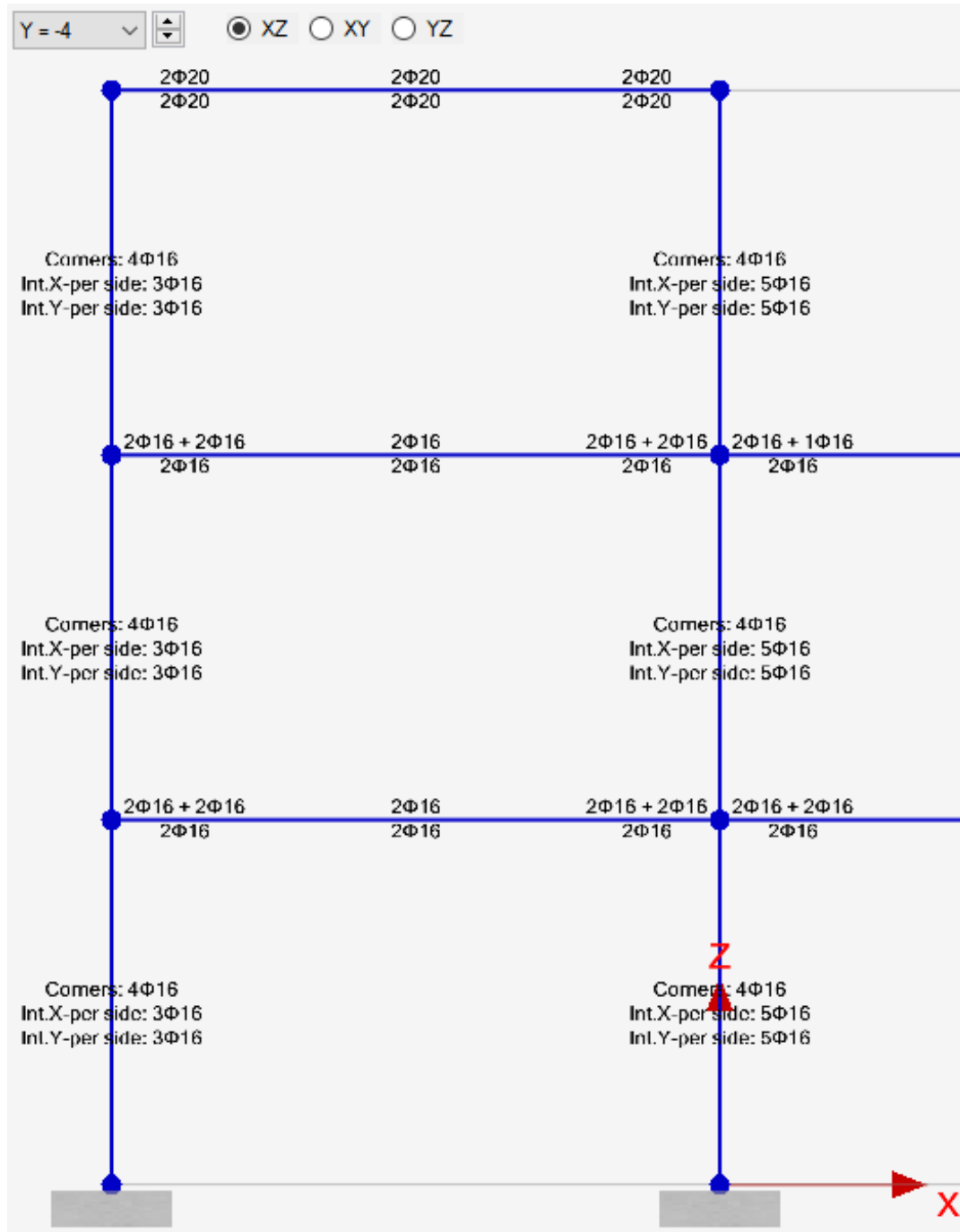
Για την επαλήθευση των παραπάνω αποτελεσμάτων, έγινε η ανάλυση του παραδείγματος του κεφαλαίου και με το λογισμικό SAP2000 (Σχήμα 4-11). Στο Παράρτημα Α, συγκρίνονται τα αποτελέσματα των δύο λογισμικών, όπου και επαληθεύονται.



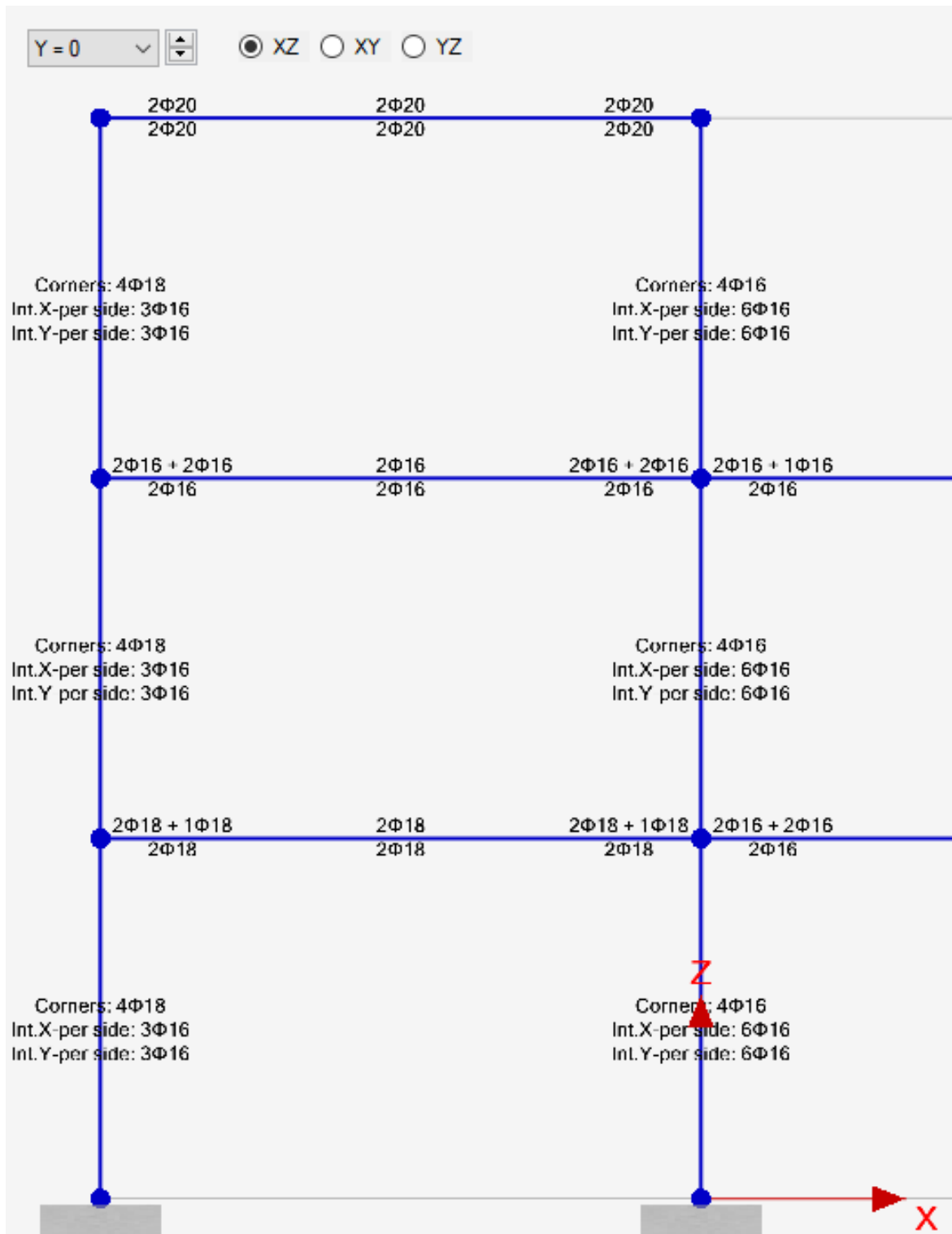
Σχήμα 4-11. Το προσομοίωμα του παραδείγματος στο λογισμικό SAP2000

### 4.3 Αποτελέσματα διαστασιολόγησης

Στα αποτελέσματα της διαστασιολόγησης παρουσιάζονται οι διατάξεις των οπλισμών, όπως αυτές υπολογίστηκαν από το λογισμικό, για το εξωτερικό πλαίσιο στο επίπεδο X-Z και  $Y = -4$  m (Σχήμα 4-12) καθώς και για το εσωτερικό πλαίσιο στο επίπεδο X-Z και  $Y = 0$  m (Σχήμα 4-13).

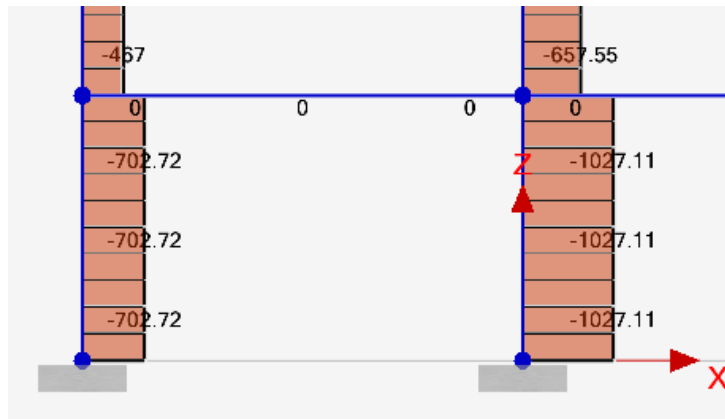


Σχήμα 4-12. Διατάξεις οπλισμού της κατασκευής στο επίπεδο X-Z για  $y = -4$  m.

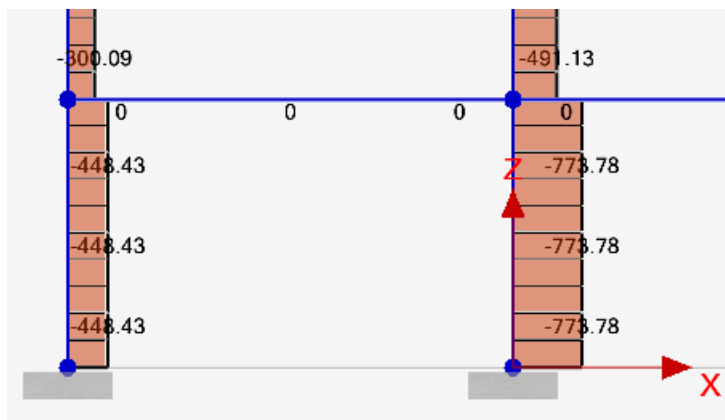


Σχήμα 4-13. Διατάξεις οπλισμού της κατασκευής στο επίπεδο X-Z για  $y = 0$  m.

Παρατηρούμε ότι στα εσωτερικά υποστυλώματα, δηλαδή για  $Y = 0$  m, ο συνολικός οπλισμός είναι μεγαλύτερος απ' ό τι στα εξωτερικά, δηλαδή για  $Y = -4$  m, κάτι το οποίο αναμένουμε καθώς τα αξονικά φορτία στην πρώτη περίπτωση είναι σημαντικά μεγαλύτερα (Σχήμα 4-14 και Σχήμα 4-15). Ακόμα, για τις δοκούς του πρώτου ορόφου, στο εσωτερικό πλαίσιο ( $Y = 0$  m) παρατηρούμε ότι ο συνεχής οπλισμός που τοποθέτησε το λογισμικό είναι διαμέτρου 18 mm, ενώ στη περίπτωση του εξωτερικού πλαισίου ( $Y = -4$  m) ο αντίστοιχος είναι διαμέτρου 16 mm. Και πάλι, η ύπαρξη μεγαλύτερων αξονικών φορτίων στα εσωτερικά υποστυλώματα της διατομής, οδηγεί σε μεγαλύτερη διάμετρο ράβδων που μπορούν να διέλθουν από τους κόμβους (Κεφάλαιο 2.2.3 σχέσεις (1), (2)), και για τους λόγους που εξηγήσαμε στο Κεφάλαιο 2, το λογισμικό επιλέγει μεγαλύτερη διάμετρο για τον συνεχή οπλισμό της δοκού του εσωτερικού πλαισίου.



Σχήμα 4-14. Διάγραμμα αξονικών δυνάμεων στο επίπεδο  $Y = 0$  m, για τον συνδυασμό  $1.35G + 1.5Q + Ex$  (μονάδες σε kN).



Σχήμα 4-15. Διάγραμμα αξονικών δυνάμεων στο επίπεδο  $Y = -4$  m, για τον συνδυασμό  $1.35G + 1.5Q + Ex$  (μονάδες σε kN).

## Κεφάλαιο 5: Σύνοψη και συμπεράσματα

Στο τελευταίο κεφάλαιο αυτής της εργασίας σημειώνονται τα βασικότερα συμπεράσματα που προέκυψαν κατά την ανάπτυξη του λογισμικού, ενώ γίνονται και προτάσεις για περαιτέρω εξέλιξη του θέματος στο μέλλον.

Αρχικά, αξίζει να σημειωθεί πως η ανάπτυξη ενός παρόμοιου λογισμικού δεν απαιτεί εξειδικευμένες γνώσεις προγραμματισμού, όπως διαπιστώθηκε, οι οποίες να αποτρέπουν μηχανικούς του κλάδου μας να ασχοληθούν με τον τομέα αυτό. Ωστόσο, αποδείχθηκε πως η διαδικασία ανάπτυξης ενός τέτοιου λογισμικού αποτελεί μια ιδιαίτερα χρονοβόρα διαδικασία και απαιτεί πολύ μεγάλη προσπάθεια..

Αναφορικά με την αποτελεσματικότητα του λογισμικού που αναπτύχθηκε, αυτή επικεντρώνεται στο κομμάτι της ανάλυσης, καθώς ήταν και ο κύριος στόχος της εργασίας αυτής. Κατά τη σύγκριση των αποτελεσμάτων της ανάλυσης με αυτά ενός λογισμικού του εμπορίου συμπεραίνονται τα εξής για τον αλγόριθμο ανάλυσης:

- Επιτυχής στατική ανάλυση, τα αποτελέσματα ταυτίζονται πλήρως με αυτά ενός εμπορικού λογισμικού, εάν αγνοηθεί η επιρροή του τεκμαρτού εμβαδού διατομής σε διάτμηση στη διεύθυνση 2 και 3 (shear area).
- Η προσέγγιση της διαφραγματικής λειτουργίας με τη μέθοδο προσθήκης εικονικών δοκών δεν δημιουργεί αποκλίσεις στα αποτελέσματα της ανάλυσης, ωστόσο αυξάνει σημαντικά τον όγκο των υπολογισμών και κατ' επέκταση τον απαιτούμενο χρόνο ανάλυσης.
- Ο αλγόριθμος που αναπτύχθηκε έχει μεγάλο χρόνο ανάλυσης για σύνθετες κατασκευές με πολλά γραμμικά στοιχεία και διαφράγματα. Διαπιστώθηκε ότι οι κύριοι παράγοντες που επηρεάζουν τη ταχύτητα της ανάλυσης είναι η προσομοίωση των διαφραγμάτων, με τη μέθοδο εικονικών δοκών που ακολουθήθηκε, καθώς και η διαδικασία υπολογισμού του αντιστρόφου μητρώου δυσκαμψίας.

### Προτάσεις για περαιτέρω εξέλιξη

- Δυνατότητα εισαγωγής πιο σύνθετων γεωμετρικά διατομών (πλακοδοκών, υποστρωμάτων τύπου L κλπ.) καθώς και επέκταση του κώδικα ανάλυσης και σχεδιασμού για τη διαχείριση τους.

- Αυτόματος υπολογισμός ιδίου βάρους και μάζας μελών με βάση τη γεωμετρία και το υλικό, για δυναμική ανάλυση, υπολογισμό ποσότητας υλικών, υπολογισμό κόστους κατασκευής κλπ.
- Προσθήκη και άλλων πιο σύνθετων μεθόδων ανάλυσης, όπως δυναμική φασματική, ανάλυση με χρονοϊστορία, ανάλυση με πεπερασμένα στοιχεία κλπ.
- Δυνατότητα διαστασιολόγησης θεμελίων, κόμβων, τοιχιών, πλακών.
- Δυνατότητα να λαμβάνεται υπόψη η αλληλεπίδραση εδάφους-κατασκευής.
- Υπολογισμός μηκών αγκύρωσης του οπλισμού.
- Χρήση της μεθόδου μητρικής συμπύκνωσης για την προσομοίωση της διαφραγματικής λειτουργίας της πλάκας, αλλά και για την απλοποίηση του μητρώου δυσκαμψίας με σκοπό τη ταχύτερη διαδικασία ανάλυσης.
- Δυνατότητα διαχωρισμού των σεισμικών συνδυασμών δράσεων, έτσι ώστε να γίνεται ορθότερος υπολογισμός των εντατικών μεγεθών σχεδιασμού, όπως ορίζει και ο ευρωκώδικας.

## Βιβλιογραφία

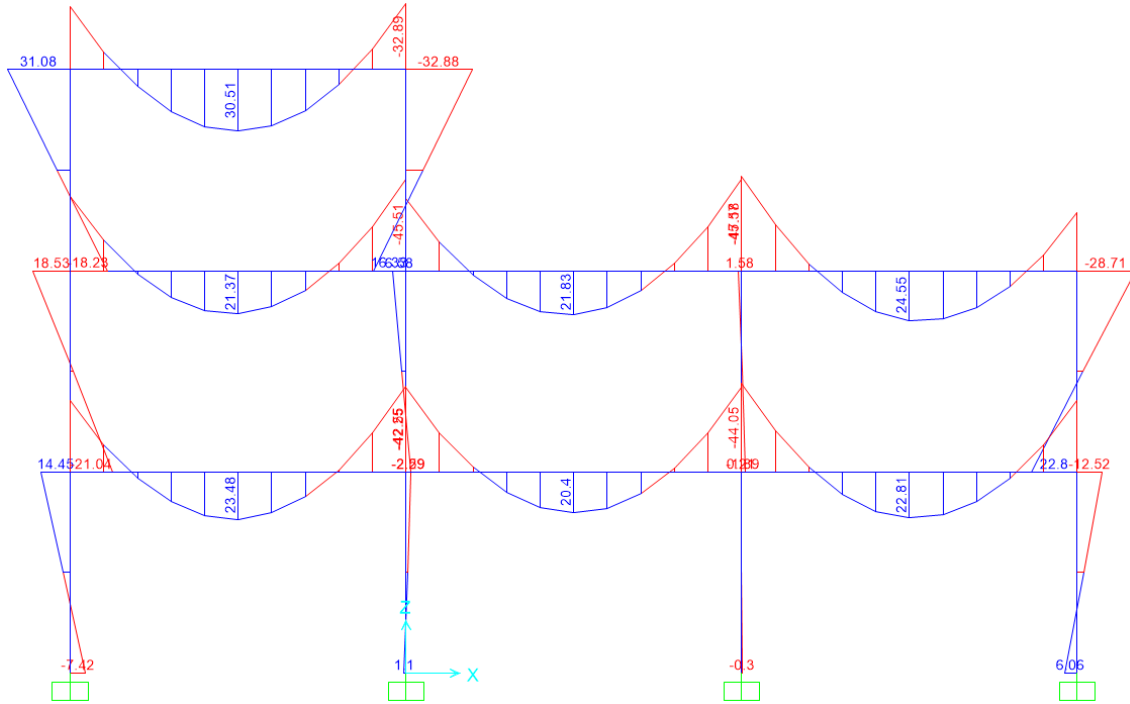
- Καρακασίδης, Θ., & Σαρρής, Ι. (2015). Αριθμητικές Μέθοδοι και Εφαρμογές για Μηχανικούς. Στο Θ. Καρακασίδης, & Ι. Σαρρής, *Αριθμητικές Μέθοδοι και Εφαρμογές για Μηχανικούς* (σσ. 92-96). Εκδόσεις Τζιόλα.
- Κούτας, Λ. (2020). *ΟΠΛΙΣΜΕΝΟ ΣΚΥΡΟΔΕΜΑ Ι, Πανεπιστημιακές Παραδόσεις & Χρήσιμοι Πίνακες [Πανεπιστημιακές Σημειώσεις]*. Βόλος: Πανεπιστήμιο Θεσσαλίας, Τμήμα Πολιτικών Μηχανικών.
- Παναγούλη, Ο. (2018). Στατική 3 [Πανεπιστημιακές Σημειώσεις]. Βόλος: Πανεπιστήμιο Θεσσαλίας, Τμήμα Πολιτικών Μηχανικών.
- Προβατίδης, Χ. Γ. (2016). *Πεπερασμένα Στοιχεία στην Ανάλυση Κατασκευών*. Εκδόσεις Τζιόλα.
- Τζάρος, Κ. (2019). *Εισαγωγή στη Μέθοδο των Πεπερασμένων Στοιχείων, Εφαρμογές της Μεθόδου στην Επιστήμη του Πολιτικού Μηχανικού [Πανεπιστημιακές Σημειώσεις]*. Βόλος: Πανεπιστήμιο Θεσσαλίας, Τμήμα Πολιτικών Μηχανικών.
- EN 1992-1-1. (2004). *Eurocode 2: Design of Concrete Structures, Part 1-1: General Rules and Rules for Buildings*. Brussels: Comite Europeen de Normalisation.
- EN 1998-1. (2004). *Eurocode 8: Design of Structures for Earthquake Resistance, Part 1: General Rules, Seismic Actions and Rules for Buildings*. Brussels: Comite Europeen de Normalisation.
- Fardis, M., Carvalho, E., Fajfar, P., & Pecker, A. (2015). *Seismic Design of Concrete Buildings to Eurocode 8*. Crc Press.



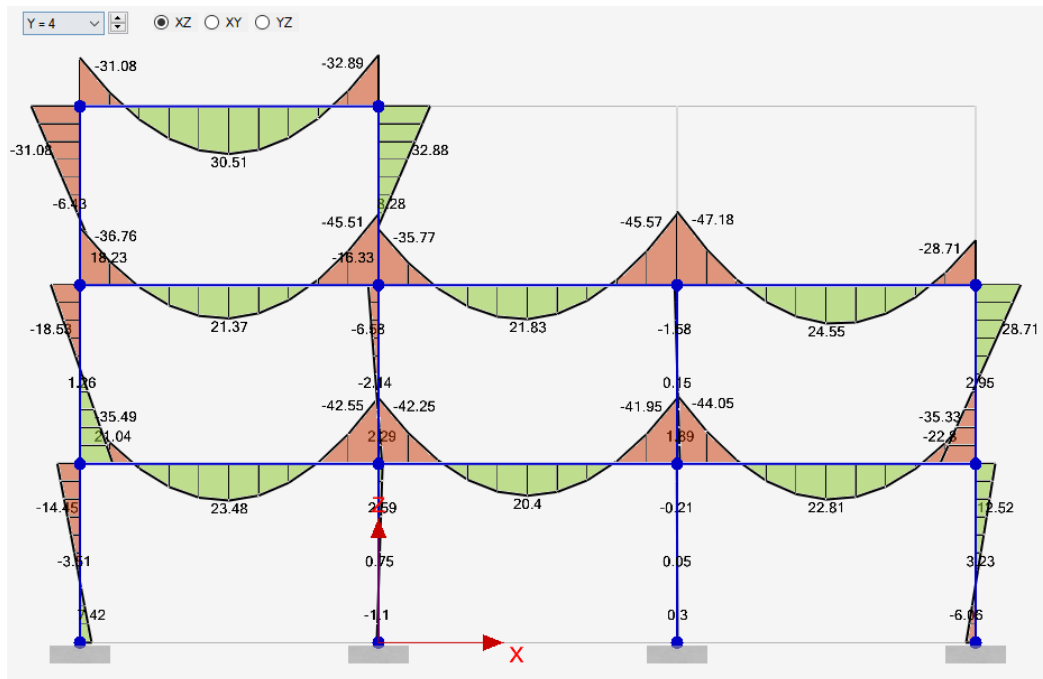
## ΠΑΡΑΡΤΗΜΑΤΑ

### Α. Σύγκριση αποτελεσμάτων του Κεφαλαίου 4 με το λογισμικό SAP 2000

- Διαγράμματα ροπών για τα μόνιμα φορτία στο επίπεδο X-Z

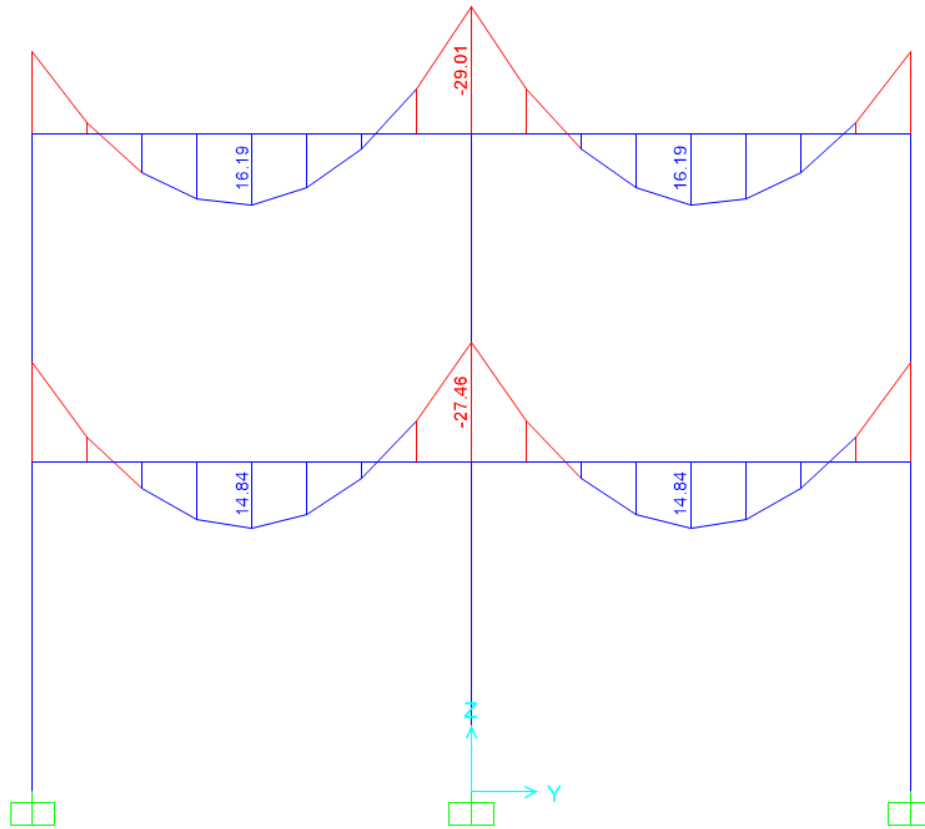


Σχήμα Π-0-1. Διαγράμματα ροπών στο SAP 2000.

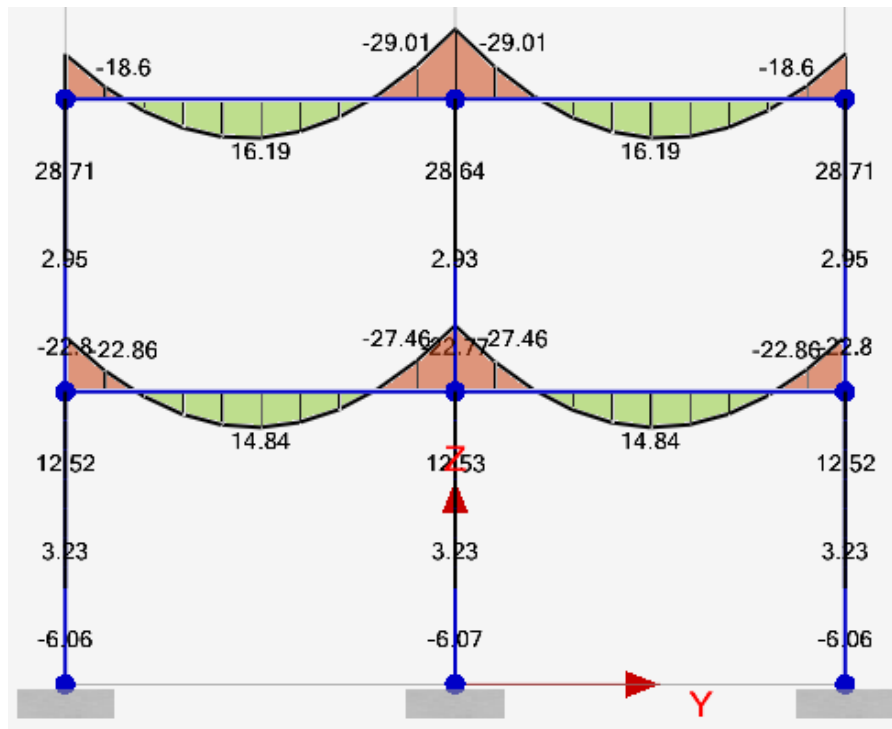


Σχήμα Π-0-2. Διαγράμματα ροπών στο Domisis R.C.

- Διαγράμματα ροπών για τα μόνιμα φορτία στο επίπεδο Y-Z

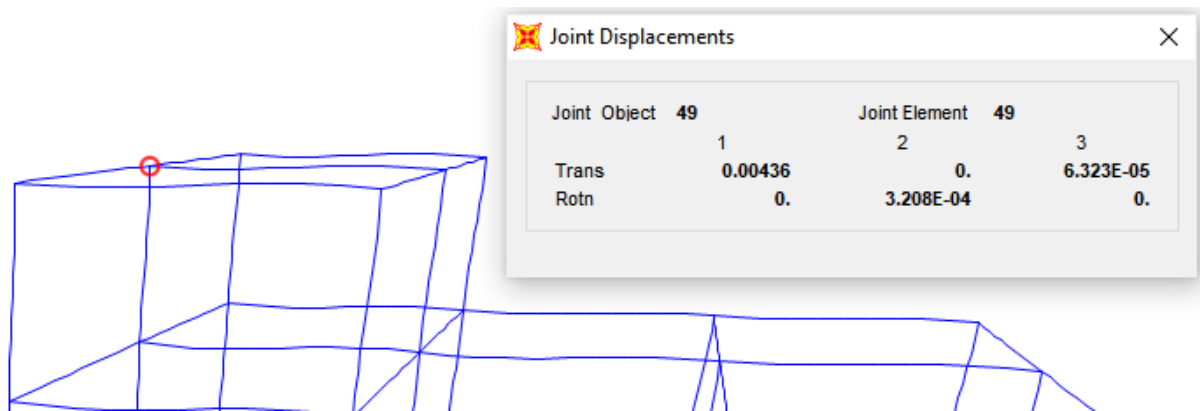


Σχήμα Π-0-3. Διαγράμματα ροπών στο SAP 2000.

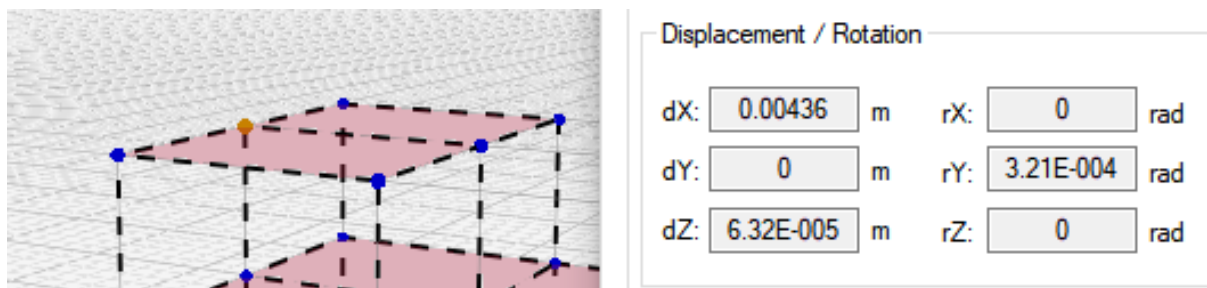


Σχήμα Π-0-4. Διαγράμματα ροπών στο Domisis R.C.

- Μετακινήσεις τελευταίου ορόφου για τον σεισμό Ex.



Σχήμα Π-0-5. Μετακινήσεις κόμβου της οροφής στο SAP 2000.



Σχήμα Π-0-6. Μετακινήσεις κόμβου της οροφής στο Domisis R.C.

Συγκρίνοντας όλα τα παραπάνω σχήματα, παρατηρήθηκε πλήρη σύγκλιση των αποτελεσμάτων της ανάλυσης μεταξύ των δύο λογισμικών, Domisis R.C. και SAP2000. Το γεγονός αυτό επαληθεύει την αποτελεσματικότητα του αλγορίθμου της ανάλυσης που χρησιμοποιείται από το λογισμικό και κατά συνέπεια επαληθεύεται και η εγκυρότητα των μεθοδολογιών που περιεγράφηκαν σε αυτή την εργασία για την ανάπτυξη του συγκεκριμένου αλγορίθμου.

## B. Κώδικας για τη διαστασιολόγηση των στοιχείων Ο.Σ. σε Visual Basic .NET

### Module M\_Design

'ΣΧΟΛΙΟ: ΟΛΑ ΤΑ ΓΡΑΜΜΙΚΑ ΣΤΟΙΧΕΙΑ ΠΕΡΙΕΧΟΝΤΑΙ ΣΤΟ lines\_dt.datatable ΟΠΟΥ ΚΑΘΕ ΣΤΟΙΧΕΙΟ ΕΧΕΙ ΜΟΝΑΔΙΚΟ ID. ΕΔΩ ΟΜΩΣ ΕΧΩ ΦΤΙΑΞΕΙ ΞΕΧΩΡΙΣΤΑ ARRAY ΓΙΑ BEAMS ΚΑΙ COLUMNS ΜΕ ΑΡΙΘΜΗΣΗ ID ΞΕΧΩΡΙΣΤΗ.

```

        'Beams(i, 0) = line_id
        'Beams(i, 1) = b (ΔΙΑΣΤΑΣΗ ΣΤΟΝ Y
AΞΟΝΑ)
        'Beams(i, 2) = h (ΔΙΑΣΤΑΣΗ ΣΤΟΝ Z
AΞΟΝΑ)
        'Beams(i, 3) = node1_id
        'Beams(i, 4) = node2_id

        'COLUMNS(i, 0) = line_id
        'COLUMNS(i, 1) = b (ΔΙΑΣΤΑΣΗ ΣΤΟΝ Y
AΞΟΝΑ)
        'COLUMNS(i, 2) = h (ΔΙΑΣΤΑΣΗ ΣΤΟΝ X
AΞΟΝΑ)
        'COLUMNS(i, 3) = node1_id
        'COLUMNS(i, 4) = node2_id

    Public Sub Design_strategy(beams(,) As Single, columns(,) As Single, N_combo(,) As
Single, M_env(,,) As Single, M_env2(,,) As Single)

        ReDim M_ikanot(nodes_dt.Rows.Count - 1, 1, 1, 1) '(0 gia ΣMrd_beams / 1 gia
ΣMrd_columns , 0 X-X' / 1 Y-Y' , 0 aristerostrofa / 1 de3iostrofa )

        ReDim oplismos_B(beams.GetLength(0) - 1, 5, 1, 2) 'h proteleutaia sthlh (0 c
ontinuous, 1 added) / h teleutaia sthlh (0 ari8mos rabdwn, 1 diametros , 2 embadon)
        '-----
        '===== 0             ===== 2             ===== 4
        '===== 1             ===== 3             ===== 5
        '-----
        '      AKPO1             ANOIFMA             AKPO2
        ReDim Mrd_b(beams.GetLength(0) - 1, 5)
        ReDim oplismos_c(columns.GetLength(0) - 1, 2, 2) '(0 corners /1 horizontal(b
) /2 vertical(h)) h teleutaia sthlh (0 ari8mos rabdwn, 1 diametros , 2 embadon mias
strwshs)
        ReDim Mrd_c(columns.GetLength(0) - 1, 1)

#Region "ΒΗΜΑ 1: ΔΟΚΟΙ ΔΙΑΜΗΚΕΣ ΟΠΛΙΣΜΟΣ"
        Dim hc(beams.GetLength(0) - 1, 1) As Single 'hc(beam_id,0 aristera/1 de3ia)
        Dim dbl_max_b(beams.GetLength(0) - 1, 1)

        For b = 0 To beams.GetLength(0) - 1
            Dim beam_length As Single = MyMATH.length(nodes_dt.Rows(beams(b, 3)).Item
(1), nodes_dt.Rows(beams(b, 3)).Item(2), 0, nodes_dt.Rows(beams(b, 4)).Item(1), nodes
_dt.Rows(beams(b, 4)).Item(2), 0)

#Region "A)dbl_max se ka8e akro"

            Dim dbl_max(1) As Single
            Dim node_id As Integer

            For n = 0 To 1 '->akro1:n=0 akro2:n=1
                node_id = beams(b, 3 + n)
                dbl_max(n) = D_available(D_available.GetLength(0) - 1) 'a8wwos mexri
apodei3h tou anti8etou(se periptwsh pou den uparxei kolwna sto allo akro)
                'eswterikos h e3wterikos kombos

```

```

Dim joint_type As Byte = 1 '0 eswterikos, 1 e3wterikos
For b2 = 0 To beams.GetLength(0) - 1
    If beams(b2, 3) = node_id Or beams(b2, 4) = node_id Then
        Dim klish1, klish2 As Single
        klish1 = (nodes_dt.Rows(beams(b, 4)).Item(2) - nodes_dt.Rows(
beams(b, 3)).Item(2)) / (nodes_dt.Rows(beams(b, 4)).Item(1) - nodes_dt.Rows(beams(b,
3)).Item(1))
        klish2 = (nodes_dt.Rows(beams(b2, 4)).Item(2) - nodes_dt.Rows
(beams(b2, 3)).Item(2)) / (nodes_dt.Rows(beams(b2, 4)).Item(1) - nodes_dt.Rows(beams(
b2, 3)).Item(1))
        If klish1 = klish2 And b <> b2 Then
            joint_type = 0
        End If
    End If
Next
'NEd katw ypostylwmatos (to elaxisto N apo oloys tous syndyasmous dra
sewn)
For c = 0 To columns.GetLength(0) - 1 '8umizw o deyteros kombos einai
pio pshla panta!
    If node_id = columns(c, 4) Then
        Dim NEd_min, Ac As Single
        For combo = 0 To N_combo.GetLength(0) - 1 'to elaxisto Ned gi
a ola ta combo
            If combo = 0 Then
                NEd_min = N_combo(combo, columns(c, 0))
            ElseIf N_combo(combo, columns(c, 0)) < NEd_min Then
                NEd_min = N_combo(combo, columns(c, 0))
            End If
        Next
        If NEd_min > 0 Then NEd_min = 0

        Ac = columns(c, 1) * columns(c, 2) 'b * h

        'hc: b sthn x-x h sthn y-y gia ypostulwmata
        Dim costheta, sintheta As Single
        costheta = (nodes_dt.Rows(beams(b, 4)).Item(1) - nodes_dt.Row
s(beams(b, 3)).Item(1)) / beam_length
        sintheta = (nodes_dt.Rows(beams(b, 4)).Item(2) - nodes_dt.Row
s(beams(b, 3)).Item(2)) / beam_length
        hc(b, n) = Math.Abs(columns(c, 1) * sintheta) + Math.Abs(colu
mns(c, 2) * costheta)
        dbl_max(0 + n) = F_Max_dbL(joint_type, fck, fyk, Math.Abs(NEd
_min), Ac, hc(b, n), KP)

        'prepei na brw se pia dia8esimh diametro 8a to balw
        For i = 0 To D_available.GetLength(0) - 1
            If dbl_max(0 + n) <= D_available(i) Then
                dbl_max(0 + n) = D_available(i)
            Exit For
            End If
            If i = D_available.GetLength(0) - 1 Then
                dbl_max(0 + n) = D_available(i)
            End If
        Next
    End If
Next

```

```

Next

dbl_max_b(b, 0) = dbl_max(0)
dbl_max_b(b, 1) = dbl_max(1)
#End Region

#Region "B)As_required OKA"
'prwta prepei na brw tis MEd sta akra kai sto meso gia ta opoia 8a sxediasw
Dim MEd(5) As Single
Dim As_req(5) As Single
'-----
'===== 0             ===== 2             ===== 4
'===== 1             ===== 3             ===== 5
'-----
'aristera
Dim maxM_syn, maxM_plyn As Single
For i = 0 To 2
  If i = 0 Then
    maxM_syn = M_env(beams(b, 0), i, 0) '0 gia ta syn
    maxM_plyn = M_env(beams(b, 0), i, 1) '1 gia ta plun
  End If
  If maxM_syn < M_env(beams(b, 0), i, 0) Then
    maxM_syn = M_env(beams(b, 0), i, 0) '0 gia ta syn
  End If
  If maxM_plyn < M_env(beams(b, 0), i, 1) Then
    maxM_plyn = M_env(beams(b, 0), i, 1) '1 gia ta plun
  End If
Next
MEd(0) = maxM_plyn
MEd(1) = maxM_syn

For i = 2 To 8
  If i = 2 Then
    maxM_syn = M_env(beams(b, 0), i, 0) '0 gia ta syn
  End If
  If maxM_syn < M_env(beams(b, 0), i, 0) Then
    maxM_syn = M_env(beams(b, 0), i, 0) '0 gia ta syn
  End If
Next
MEd(3) = maxM_syn
For i = 8 To 10
  If i = 8 Then
    maxM_syn = M_env(beams(b, 0), i, 0) '0 gia ta syn
    maxM_plyn = M_env(beams(b, 0), i, 1) '1 gia ta plun
  End If
  If maxM_syn < M_env(beams(b, 0), i, 0) Then
    maxM_syn = M_env(beams(b, 0), i, 0) '0 gia ta syn
  End If
  If maxM_plyn < M_env(beams(b, 0), i, 1) Then
    maxM_plyn = M_env(beams(b, 0), i, 1) '1 gia ta plun
  End If
Next
MEd(4) = maxM_plyn
MEd(5) = maxM_syn

```

```

As_req(0) = F_As_b(MEd(0), beams(b, 1), beams(b, 2), d1_2, d1_2, fck, fyk)(0)
As_req(1) = F_As_b(MEd(1), beams(b, 1), beams(b, 2), d1_2, d1_2, fck, fyk)(0)
As_req(3) = F_As_b(MEd(3), beams(b, 1), beams(b, 2), d1_2, d1_2, fck, fyk)(0)
As_req(4) = F_As_b(MEd(4), beams(b, 1), beams(b, 2), d1_2, d1_2, fck, fyk)(0)
As_req(5) = F_As_b(MEd(5), beams(b, 1), beams(b, 2), d1_2, d1_2, fck, fyk)(0)
#End Region

#Region "Γ)Continuoum Reinforcement(EC8 detailing of beam PART1)"
'-----
'===== 0          ===== 2          ===== 4
'-----
'===== 1          ===== 3          ===== 5
'-----

'prwta ypologizw As_min pantou
Dim As_min As Single = F_As_b_EC8_1(KP, beams(b, 1), beams(b, 2) - d1_2,
fck, fyk, Es)

'epeita ton topo8etw me diametro = min(maxdbl())
Dim A_cont As Single : Dim N_cont As Integer : Dim D_cont As Single
D_cont = Math.Min(dbl_max(0), dbl_max(1))
N_cont = 1

'3ekinaw me duo rabdous D_cont kai pros8etw kai alles ean xreiazetai wste
na ikanopoiiei Asmin
Do
    N_cont += 1
    A_cont = (N_cont) * Math.PI * D_cont ^ 2 / 4
Loop Until A_cont >= As_min

'epeita ton pernew se oles tis 8eseis
For p = 0 To 5
    oplismos_B(b, p, 0, 0) = N_cont
    oplismos_B(b, p, 0, 1) = D_cont
    oplismos_B(b, p, 0, 2) = A_cont
Next

#End Region

#Region "Δ)Added Reinforcement (As_required)"

'TOPO8ETW EXTRA OPLISMO WSTE NA KALYPTETAI PANTOU TO AS_REQ

'briskw thn prwth rabdo pou mporw na dokimasw prepei = D_cont
Dim D_id_arx As Integer
For i = 0 To D_available.GetLength(0) - 1
    If D_available(i) = D_cont Then D_id_arx = i
Next

For p = 0 To 5
    'prwta mhdenizw ton added
    oplismos_B(b, p, 1, 0) = 0
    oplismos_B(b, p, 1, 1) = 0
    oplismos_B(b, p, 1, 2) = 0

    Dim D_id, N_add As Integer
    Dim A_add, D_add As Single

```

```

'ean xreiazontai ews +2 rabdoi (prospa8w na balw thn pio mikrh diamet
ro)
For D_id = D_id_arx To Math.Min(D_id_arx + 2, D_available.GetLength(0
) - 1)
    If oplismos_B(b, p, 0, 2) + oplismos_B(b, p, 1, 2) >= As_req(p) T
hen Exit For 'oso o continuous + added < apaitoumenou
    For N_add = 1 To 2
        If oplismos_B(b, p, 0, 2) + oplismos_B(b, p, 1, 2) >= As_req(
p) Then Exit For 'oso o continuous + added < apaitoumenou
        D_add = D_available(D_id)
        If p = 0 Or p = 1 Then
            If D_add > dbl_max_b(b, 0) Then D_add = dbl_max_b(b, 0)
        ElseIf p = 4 Or p = 5 Then
            If D_add > dbl_max_b(b, 1) Then D_add = dbl_max_b(b, 1)
        End If
        'If p = 1 Then MsgBox(D_add, D_available(D_id), D_id)
        A_add = N_add * Math.PI * D_add ^ 2 / 4
        oplismos_B(b, p, 1, 0) = N_add
        oplismos_B(b, p, 1, 1) = D_add
        oplismos_B(b, p, 1, 2) = A_add
        'If p = 3 Then MsgBox(oplismos_B(b, p, 0, 2) & " " & As_re
q(p))
    Next
Next

'ean xreiazontai parapanw apo +2 rabdoi (prospa8w na balw ton mikrote
ro ari8mo rabdwn)
For N_add = 3 To 100
    If oplismos_B(b, p, 0, 2) + oplismos_B(b, p, 1, 2) >= As_req(p) T
hen Exit For 'oso o continuous + added < apaitoumenou
    For D_id = D_id_arx To Math.Min(D_id_arx + 2, D_available.GetLeng
th(0) - 1)
        If oplismos_B(b, p, 0, 2) + oplismos_B(b, p, 1, 2) >= As_req(
p) Then Exit For 'oso o continuous + added < apaitoumenou
        D_add = D_available(D_id)
        If p = 0 Or p = 1 Then
            If D_add > dbl_max_b(b, 0) Then D_add = dbl_max_b(b, 0)
        ElseIf p = 4 Or p = 5 Then
            If D_add > dbl_max_b(b, 1) Then D_add = dbl_max_b(b, 1)
        End If
        A_add = N_add * Math.PI * D_add ^ 2 / 4
        oplismos_B(b, p, 1, 0) = N_add
        oplismos_B(b, p, 1, 1) = D_add
        oplismos_B(b, p, 1, 2) = A_add
        'If p = 3 Then MsgBox(oplismos_B(b, p, 0, 2) & " " & As_re
q(p))
    Next
Next

Next 'ALL 6 POSITIONS

#End Region

#Region "E)EC8 detailing of beam PART2"

```



```

'E3ETAZW TIS YPOLOIPES DIATA3EIS GIA AS_MIN
Dim As_top_supp1 As Single, As_top_supp2 As Single, As_bot_span As Single
As_top_supp1 = oplismos_B(b, 0, 0, 2) + oplismos_B(b, 0, 1, 2)
As_bot_span = oplismos_B(b, 3, 0, 2) + oplismos_B(b, 3, 1, 2)
As_top_supp2 = oplismos_B(b, 4, 0, 2) + oplismos_B(b, 4, 1, 2)

Dim rtrn() As Single = F_As_b_EC8_2(KP, beams(b, 1), beams(b, 2), fck, fy
k, μφ, Es, As_top_supp1, As_top_supp2, As_bot_span)

Dim Lcr, As_max, As_min_top_span, As_min_bot_supp1, As_min_bot_supp2 As S
ingle
Lcr = rtrn(0)
As_max = rtrn(1)
As_min_top_span = rtrn(2)
As_min_bot_supp1 = rtrn(3)
As_min_bot_supp2 = rtrn(4)

'ΠΡΩΤΑ ΕΛΕΓΧΩ ΤΗΝ ΕΠΑΡΚΕΙΑ ΤΟΥ As_top_span για ( As_min_top_span)
While oplismos_B(b, 2, 0, 2) + oplismos_B(b, 2, 1, 2) < As_min_top_span '
As_top_span < As_min_top_span
'DEN EXEI ADDED OPLISMO KAI ARA D_ADDED, OPOTE BAZW ALLH MIA APO TON
CONTINUOUM
    oplismos_B(b, 2, 1, 1) = oplismos_B(b, 2, 0, 1)
    oplismos_B(b, 2, 1, 0) += 1
    oplismos_B(b, 2, 1, 2) = oplismos_B(b, 2, 1, 0) * Math.PI * oplismos_
B(b, 2, 1, 1) ^ 2 / 4
End While
'ΕΠΕΙΤΑ ΕΛΕΓΧΩ ΤΟΝ As_BOT_span για (As_max)
If As_bot_span > As_max Then
    MsgBox("The dimensions of beam: " & lines_dt.Rows(beams(b, 0)).Item(6
) & ", are not enough. As,max has been reached", vbCritical)
    Exit Sub
End If
'ΕΠΕΙΤΑ ΕΛΕΓΧΩ ΤΟΝ As_bot_supp1,2 > As_min_bot_supp1,2
While oplismos_B(b, 1, 0, 2) + oplismos_B(b, 1, 1, 2) < As_min_bot_supp1
'As_bot_supp1 < As_min_bot_supp1
    If oplismos_B(b, 1, 1, 1) <> 0 Then 'EXEI ADDED OPLISMO KAI ARA D_ADD
ED
        'MsgBox("exw added")
        oplismos_B(b, 1, 1, 0) += 1
        oplismos_B(b, 1, 1, 2) = oplismos_B(b, 1, 1, 0) * Math.PI * oplis
mos_B(b, 1, 1, 1) ^ 2 / 4
    Else
        'DEN EXEI ADDED OPLISMO KAI ARA D
_ADDED, OPOTE BAZW ALLH MIA APO TON CONTINUOUM
        oplismos_B(b, 1, 1, 1) = oplismos_B(b, 1, 0, 1)
        oplismos_B(b, 1, 1, 0) += 1
        oplismos_B(b, 1, 1, 2) = oplismos_B(b, 1, 1, 0) * Math.PI * oplis
mos_B(b, 1, 1, 1) ^ 2 / 4
    End If
End While
While oplismos_B(b, 5, 0, 2) + oplismos_B(b, 5, 1, 2) < As_min_bot_supp1
'As_bot_supp2 < As_min_bot_supp2
    If oplismos_B(b, 5, 1, 1) <> 0 Then 'EXEI ADDED OPLISMO KAI ARA D_ADD
ED
        oplismos_B(b, 5, 1, 0) += 1

```

```

        oplismos_B(b, 5, 1, 2) = oplismos_B(b, 5, 1, 0) * Math.PI * oplis
mos_B(b, 5, 1, 1) ^ 2 / 4
    Else
        'DEN EXEI ADDED OPLISMO KAI ARA D
_ADDED, OPOTE BAZW ALLH MIA APO TON CONTINUOUM
        oplismos_B(b, 5, 1, 1) = oplismos_B(b, 5, 0, 1)
        oplismos_B(b, 5, 1, 0) += 1
        oplismos_B(b, 5, 1, 2) = oplismos_B(b, 5, 1, 0) * Math.PI * oplis
mos_B(b, 5, 1, 1) ^ 2 / 4
    End If
End While

#End Region
Next 'all beams

#Region "Z) ΥΠΟΛΟΓΙΖΩ ΤΙΣ ΡΟΠΕΣ ΑΝΤΟΧΗΣ"

For b = 0 To beams.GetLength(0) - 1
    For p = 0 To 5
        'As1 o panw(8libomenos) As2 o katw(efelkuomenos)
        Mrd_b(b, p) = F_Mrd_b(0, oplismos_B(b, p, 0, 2) + oplismos_B(b, p, 1,
2), beams(b, 1), beams(b, 1), beams(b, 2), d1_2, d1_2, fck, fyk)(1) '(As1, As2, bw,
beff, h, d1, d2, fck, fyk)
    Next
Next 'all beams

'M_rd,beams,kombou gia ikanotiko
For n = 0 To nodes_dt.Rows.Count - 1
    For b = 0 To beams.GetLength(0) - 1

        Dim beam_length As Single = MyMATH.length(nodes_dt.Rows(beams(b, 3)).
Item(1), nodes_dt.Rows(beams(b, 3)).Item(2), 0, nodes_dt.Rows(beams(b, 4)).Item(1), n
odes_dt.Rows(beams(b, 4)).Item(2), 0)
        Dim costheta, sintheta As Single
        costheta = (nodes_dt.Rows(beams(b, 4)).Item(1) - nodes_dt.Rows(beams(
b, 3)).Item(1)) / beam_length
        sintheta = (nodes_dt.Rows(beams(b, 4)).Item(2) - nodes_dt.Rows(beams(
b, 3)).Item(2)) / beam_length
        Dim pc1, pc2, ps1, ps2 As Byte 'na balw th swsth fora se periptwsh po
y exw end node me end node duo beam h to anti8eto
        If costheta >= 0 Then : pc1 = 0 : pc2 = 1 : Else : pc1 = 1 : pc2 = 0
: End If
        If sintheta >= 0 Then : ps1 = 0 : ps2 = 1 : Else : ps1 = 1 : ps2 = 0
: End If

        If beams(b, 4) = n Then
            M_ikanot(n, 0, 0, pc1) += 1.3 * Math.Abs(costheta) * Mrd_b(b, 4)
'X-X' ARISTEROSTROFA
            M_ikanot(n, 0, 0, pc2) += 1.3 * Math.Abs(costheta) * Mrd_b(b, 5)
'X-X' DE3IOSTROFA
            M_ikanot(n, 0, 1, ps1) += 1.3 * Math.Abs(sintheta) * Mrd_b(b, 4)
'Y-Y' ARISTEROSTROFA
            M_ikanot(n, 0, 1, ps2) += 1.3 * Math.Abs(sintheta) * Mrd_b(b, 5)
'Y-Y' DE3IOSTROFA
        ElseIf beams(b, 3) = n Then
            M_ikanot(n, 0, 0, pc1) += 1.3 * Math.Abs(costheta) * Mrd_b(b, 1)
'X-X' ARISTEROSTROFA

```

```

                M_ikanot(n, 0, 0, pc2) += 1.3 * Math.Abs(costheta) * Mrd_b(b, 0)
'X-X' DE3IOSTROFA
                M_ikanot(n, 0, 1, ps1) += 1.3 * Math.Abs(sintheta) * Mrd_b(b, 1)
'Y-Y' ARISTEROSTROFA
                M_ikanot(n, 0, 1, ps2) += 1.3 * Math.Abs(sintheta) * Mrd_b(b, 0)
'Y-Y' DE3IOSTROFA
                End If

                Next 'all beams
            Next 'all nodes

#End Region

#End Region

#Region "ΒΗΜΑ 2: ΥΠΟΣΤΥΛΩΜΑΤΑ ΔΙΑΜΗΚΕΣ ΟΠΛΙΣΜΟΣ"

#Region "Α1) ΜΕΓΙΣΤΗ ΡΟΠΗ ΥΠΟΣΤΥΛΩΜΑΤΩΝ, ΘΕΩΡΩΝΤΑΣ ΕΝΙΑΙΑ ΚΑΘΥΨΟΣ"

    Dim cnt As Integer = 0
    For pt = 0 To gridPt_dt.Rows.Count - 1
        If gridPt_dt.Rows(pt).Item(2) = Zgrid_dt.Rows(0).Item(1) Then
            Dim yparxei As Boolean = False
            For c = 0 To columns.GetLength(0) - 1
                If (nodes_dt.Rows(columns(c, 3)).Item(1) = gridPt_dt.Rows(pt).Item(0) And nodes_dt.Rows(columns(c, 3)).Item(2) = gridPt_dt.Rows(pt).Item(1)) Or
                    (nodes_dt.Rows(columns(c, 4)).Item(1) = gridPt_dt.Rows(pt).Item(0) And nodes_dt.Rows(columns(c, 4)).Item(2) = gridPt_dt.Rows(pt).Item(1)) Then
                    yparxei = True
                    Exit For
                End If
            Next
            If yparxei = True Then
                cnt += 1
            End If
        End If
    Next

    Dim proj_pts(cnt - 1, 1) As Single 'oi syntetagmenes tw n columns

    Dim cnt1 As Integer = 0
    For pt = 0 To gridPt_dt.Rows.Count - 1
        If gridPt_dt.Rows(pt).Item(2) = Zgrid_dt.Rows(0).Item(1) Then
            Dim yparxei As Boolean = False
            For c = 0 To columns.GetLength(0) - 1
                If (nodes_dt.Rows(columns(c, 3)).Item(1) = gridPt_dt.Rows(pt).Item(0) And nodes_dt.Rows(columns(c, 3)).Item(2) = gridPt_dt.Rows(pt).Item(1)) Or
                    (nodes_dt.Rows(columns(c, 4)).Item(1) = gridPt_dt.Rows(pt).Item(0) And nodes_dt.Rows(columns(c, 4)).Item(2) = gridPt_dt.Rows(pt).Item(1)) Then
                    yparxei = True
                    Exit For
                End If
            Next
            If yparxei = True Then

```

```

        proj_pts(cnt1, 0) = gridPt_dt.Rows(pt).Item(0)
        proj_pts(cnt1, 1) = gridPt_dt.Rows(pt).Item(1)
        cnt1 += 1
    End If

End If
Next

Dim col_club(proj_pts.GetLength(0) - 1)() As Integer
For pt = 0 To proj_pts.GetLength(0) - 1
    Dim col_club_2d() As Integer = Nothing
    Dim num_cols As Integer = 0
    For c = 0 To columns.GetLength(0) - 1
        Dim n1(1), n2(1) As Single
        n1(0) = nodes_dt.Rows(columns(c, 3)).Item(1)
        n1(1) = nodes_dt.Rows(columns(c, 3)).Item(2)
        'n2(0) = nodes_dt.Rows(columns(c, 4)).Item(1)
        'n2(1) = nodes_dt.Rows(columns(c, 4)).Item(2)
        If n1(0) = proj_pts(pt, 0) And n1(1) = proj_pts(pt, 1) Then
            ReDim Preserve col_club_2d(num_cols)
            col_club_2d(num_cols) = c
            num_cols += 1
        End If
    Next
    col_club(pt) = col_club_2d
Next
Dim M_col_max(col_club.GetLength(0) - 1, 1) As Double 'Me auth 8a kanw design
Dim M_max_Med(col_club.GetLength(0) - 1, 1) As Double
Dim M_max_ikanot(col_club.GetLength(0) - 1, 1) As Double
Dim N_cc_ed(col_club.GetLength(0) - 1) As Single

Dim N_cm(1), N_cm_ik(1), N_cm_Med(1) As Integer 'gia na mou dwsei to axoniko
gia to column to opoio diastasiologw

For cc = 0 To col_club.GetLength(0) - 1 'ALL COLUMN CLUBS
    M_max_Med(cc, 0) = 0 : M_max_Med(cc, 1) = 0
    M_max_ikanot(cc, 0) = 0 : M_max_ikanot(cc, 1) = 0
    For cm = 0 To col_club(cc).GetLength(0) - 1 ' ALL COLUMN MEMBERS OF THE (
CC) COLUMN CLUB
        Dim c As Integer = col_club(cc)(cm)
        '-----Max_Med-----
        Dim Med_max(1) As Double
        Med_max(0) = M_env(columns(c, 0), 0, 0) 'X-X
        Med_max(1) = M_env2(columns(c, 0), 0, 0) 'Y-Y
        For p = 0 To 10
            If Med_max(0) < Math.Max(M_env(columns(c, 0), p, 0), M_env(column
s(c, 0), p, 1)) Then
                Med_max(0) = Math.Max(M_env(columns(c, 0), p, 0), M_env(column
ns(c, 0), p, 1))
            End If
            If Med_max(1) < Math.Max(M_env2(columns(c, 0), p, 0), M_env2(colu
mns(c, 0), p, 1)) Then
                Med_max(1) = Math.Max(M_env2(columns(c, 0), p, 0), M_env2(col
umns(c, 0), p, 1))
            End If
        Next 'all positions
    Next

```

```

        If M_max_Med(cc, 0) < Med_max(0) Then : M_max_Med(cc, 0) = Med_max(0)
: N_cm_Med(0) = cm : End If
        If M_max_Med(cc, 1) < Med_max(1) Then : M_max_Med(cc, 1) = Med_max(1)
: N_cm_Med(1) = cm : End If
'-----
'-----Max_M_ikanot-----
Dim Mik_max(1) As Double
Dim n1, n2 As Integer
n1 = columns(c, 3)
n2 = columns(c, 4)

Mik_max(0) = Math.Max(Math.Max(M_ikanot(n1, 0, 0, 0), M_ikanot(n1, 0,
0, 1)),
                        Math.Max(M_ikanot(n2, 0, 0, 0), M_ikanot(n2, 0,
0, 1)))

'M_ikanot(n1, 0, 0, 0) 'X-X' ARISTEROSTROFA
'M_ikanot(n1, 0, 0, 1) 'X-X' DE3IOSTROFA
'M_ikanot(n2, 0, 0, 0) 'X-X' ARISTEROSTROFA
'M_ikanot(n2, 0, 0, 1) 'X-X' DE3IOSTROFA
Mik_max(1) = Math.Max(Math.Max(M_ikanot(n1, 0, 1, 0), M_ikanot(n1, 0,
1, 1)),
                        Math.Max(M_ikanot(n2, 0, 1, 0), M_ikano
t(n2, 0, 1, 1)))

'M_ikanot(n1, 0, 1, 0) 'Y-Y' ARISTEROSTROFA
'M_ikanot(n1, 0, 1, 1) 'Y-Y' DE3IOSTROFA
'M_ikanot(n2, 0, 1, 0) 'Y-Y' ARISTEROSTROFA
'M_ikanot(n2, 0, 1, 1) 'Y-Y' DE3IOSTROFA
If M_max_ikanot(cc, 0) < Mik_max(0) Then : M_max_ikanot(cc, 0) = Mik_
max(0) : N_cm_ik(0) = cm : End If
If M_max_ikanot(cc, 1) < Mik_max(1) Then : M_max_ikanot(cc, 1) = Mik_
max(1) : N_cm_ik(1) = cm : End If
'-----
Next
M_col_max(cc, 0) = Math.Max(M_max_ikanot(cc, 0), M_max_Med(cc, 0)) 'me a
uth 8a kanw design
M_col_max(cc, 1) = Math.Max(M_max_ikanot(cc, 1), M_max_Med(cc, 1)) 'me au
th 8a kanw design

If M_max_ikanot(cc, 0) > M_max_Med(cc, 0) Then N_cm(0) = N_cm_ik(0) Else
N_cm(0) = N_cm_Med(0)
If M_max_ikanot(cc, 1) > M_max_Med(cc, 1) Then N_cm(1) = N_cm_ik(1) Else
N_cm(1) = N_cm_Med(1)

'MsgBox(col_club(cc)(0) + 1 & " Mmax-x " & M_col_max(cc, 0) & " Mmax-
y " & M_col_max(cc, 1))

'Nd katw ypostylwmatos (to elaxisto N apo oloys tous syndyasmous drasewn)
Dim Nd_min(1) As Single
For i = 0 To 1
Dim c As Integer = N_cm(i)
For combo = 0 To N_combo.GetLength(0) - 1 'to elaxisto Ned gia ola ta
combo

If combo = 0 Then
Nd_min(i) = N_combo(combo, columns(c, 0))
ElseIf N_combo(combo, columns(c, 0)) < Nd_min(i) Then
Nd_min(i) = N_combo(combo, columns(c, 0))

```

```

        End If
    Next
    If Nd_min(i) > 0 Then Nd_min(i) = 0
Next

N_cc_ed(cc) = Math.Min(Nd_min(0), Nd_min(1))

Next

#End Region

#Region "ΕΛΕΓΧΩ ΕΑΝ ΤΑ ΥΠΟΣΤΥΛΩΜΑΤΑ ΕΙΝΑΙ ΕΝΙΑΙΑ ΚΑΘ΄ΥΨΟΣ"

'prepei na balw elegxo oti ola ta ypostylwmata ka8ypsos exoun idio cross sect
ion kai oti den diakoptontai

For cc = 0 To col_club.GetLength(0) - 1 'ALL COLUMN CLUBS
    Dim diastaseis(1) As Single
    For cm = 0 To col_club(cc).GetLength(0) - 1 ' ALL COLUMN MEMBERS OF THE (
CC) COLUMN CLUB

        If cm = 0 Then
            diastaseis(0) = columns(cm, 1)
            diastaseis(1) = columns(cm, 2)
        Else
            If diastaseis(0) <> columns(cm, 1) Or diastaseis(1) <> columns(cm
, 2) Then
                MsgBox("Continuous columns with different cross sections foun
d. Design couldn't complete", vbCritical)
                Exit Sub
            End If
        End If

    Next
Next

#End Region

For c = 0 To columns.GetLength(0) - 1

#Region "A) ΕΛΑΧΙΣΤΟΣ ΑΡΙΘΜΟΣ ΡΑΒΔΩΝ ΑΝΑ ΠΛΕΥΡΑ "
Dim nmin(1) As Byte 'nmin elax ari8mos rabdwn
Dim result() As Byte = F_nmin_c(columns(c, 1), columns(c, 2), d1_2, KP)
nmin(0) = result(0) 'pleura b
nmin(1) = result(1) 'pleura h
#End Region

#Region "B) ΣΥΓΚΡΙΣΗ ΜΕ ρmin, ρmax "
Dim φmin1, φmin2, φmax1, φmax2 As Single
Dim outcome1, outcome2 As Byte
Dim result2(,) As Single
result2 = F_fmin_max_c(columns(c, 1), columns(c, 2), nmin(0), nmin(1), KP
)

φmin1 = result2(0, 0)
φmin2 = result2(0, 1) 'oi gwniakies rabdoi

```

```

        outcome1 = result2(0, 2) ': If outcome1 = 0 Then MsgBox("The dimensions o
f column: " & lines_dt.Rows(columns(c, 0)).Item(6) & ". are not enough,  $\rho < \rho_{min}$ ", vb
Critical) : Exit Sub
        phi_max1 = result2(1, 0)
        phi_max2 = result2(1, 1) 'oi gwniakies rabdoi
        outcome2 = result2(1, 2) ': If outcome2 = 0 Then MsgBox("The diameter of
available bars are not enough for column: " & lines_dt.Rows(columns(c, 0)).Item(6) &
".  $\rho > \rho_{max}$ ", vbCritical) : Exit Sub
#End Region

#Region "Γ) ΞΕΚΙΝΑΩ ΜΕ ΤΟΝ ΕΛΑΧΙΣΤΟ "
'GWNIAKES
oplismos_c(c, 0, 0) = 2
oplismos_c(c, 0, 1) = phi_min2
oplismos_c(c, 0, 2) = 2 * Math.PI * phi_min2 ^ 2 / 4
'horizontal (b)
oplismos_c(c, 1, 0) = nmin(0) - 2
oplismos_c(c, 1, 1) = phi_min1
oplismos_c(c, 1, 2) = (nmin(0) - 2) * Math.PI * phi_min1 ^ 2 / 4
'vertical (h)
oplismos_c(c, 2, 0) = nmin(1) - 2
oplismos_c(c, 2, 1) = phi_min1
oplismos_c(c, 2, 2) = (nmin(1) - 2) * Math.PI * phi_min1 ^ 2 / 4

#End Region

Next 'all columns

#Region "Δ) ΟΠΛΙΣΜΟΣ "

For cc = 0 To col_club.GetLength(0) - 1 'ALL COLUMN CLUBS

    Dim oplismos_c_club(2, 2) '(0 corners /1 horizontal(b) /2 vertical(h))
h teleutaia sthlih (0 ari8mos rabdwn, 1 diametros , 2 embadon mias strwshs)

    'M_col_max(cc, 0) 'X-X
    'M_col_max(cc, 1) 'Y-Y

    Dim result(1) As Single
    Dim Mx, My, b, h As Single
    Mx = M_col_max(cc, 0)
    My = M_col_max(cc, 1)
    b = columns(col_club(cc)(0), 1) 'pernw tuxaia to cross section ths prwths
sto club afou oles oi alles einai idies kai kala
    h = columns(col_club(cc)(0), 2)

    result = F_As_c2(Mx, My, N_cc_ed(cc), fyk, fck, Es, b, h, d1_2)

    Dim As_c_req(1) As Single
    As_c_req(0) = result(0) 'X-X
    As_c_req(1) = result(1) 'Y-Y

    oplismos_c_club(0, 0) = oplismos_c(col_club(cc)(0), 0, 0)
    oplismos_c_club(0, 1) = oplismos_c(col_club(cc)(0), 0, 1)
    oplismos_c_club(0, 2) = oplismos_c(col_club(cc)(0), 0, 2)

```

```

oplismos_c_club(1, 0) = oplismos_c(col_club(cc)(0), 1, 0)
oplismos_c_club(1, 1) = oplismos_c(col_club(cc)(0), 1, 1)
oplismos_c_club(1, 2) = oplismos_c(col_club(cc)(0), 1, 2)

oplismos_c_club(2, 0) = oplismos_c(col_club(cc)(0), 2, 0)
oplismos_c_club(2, 1) = oplismos_c(col_club(cc)(0), 2, 1)
oplismos_c_club(2, 2) = oplismos_c(col_club(cc)(0), 2, 2)

Dim As_x, As_y As Single
As_x = oplismos_c_club(1, 2) + oplismos_c_club(0, 2) / 2
As_y = oplismos_c_club(2, 2) + oplismos_c_club(0, 2) / 2

Do Until As_x >= As_c_req(0) And
    As_y >= As_c_req(1)

    'briskw to D_id pou exw balei
    Dim D_c_id, D_an_id As Integer
    For i = 0 To D_available.GetLength(0) - 1
        If D_available(i) = oplismos_c(col_club(cc)(0), 0, 1) Then
            D_c_id = D_available(i)
        End If
        If D_available(i) = oplismos_c(col_club(cc)(0), 1, 1) Then
            D_an_id = D_available(i)
        End If
    Next

    'prwta dokimazw na au3hsw tis gwniakes ws duo diametrous pio polu

    For j = D_c_id + 1 To D_an_id + 2

        If j > D_available.GetLength(0) - 1 Then
            Exit For
        End If

        oplismos_c_club(0, 0) = 2
        oplismos_c_club(0, 1) = D_available(j)
        oplismos_c_club(0, 2) = 2 * Math.PI * oplismos_c_club(0, 1) ^ 2 / 4
        As_x = oplismos_c_club(1, 2) + oplismos_c_club(0, 2) / 2
        As_y = oplismos_c_club(2, 2) + oplismos_c_club(0, 2) / 2

        If As_x >= As_c_req(0) Or As_y >= As_c_req(1) Then
            Exit For
        End If
    Next

    'ean den ftanei ay3anw ton ari8mo se ka8e pleyra
    If As_x < As_c_req(0) Or
        As_y < As_c_req(1) Then

        oplismos_c_club(1, 0) += 1
        oplismos_c_club(1, 2) = oplismos_c_club(1, 0) * Math.PI * oplismo
s_c_club(1, 1) ^ 2 / 4
        oplismos_c_club(2, 0) += 1
        oplismos_c_club(2, 2) = oplismos_c_club(2, 0) * Math.PI * oplismo
s_c_club(2, 1) ^ 2 / 4

        As_x = oplismos_c_club(1, 2) + oplismos_c_club(1, 2) / 2

```



```

        As_y = oplismos_c_club(2, 2) + oplismos_c_club(1, 2) / 2
    End If
Loop

'PERNAW TON OPLISMO SE OLA TA COLUMNS POU ANHKOUN STO CLUB
For cm = 0 To col_club(cc).GetLength(0) - 1
    oplismos_c(col_club(cc)(cm), 0, 0) = oplismos_c_club(0, 0)
    oplismos_c(col_club(cc)(cm), 0, 1) = oplismos_c_club(0, 1)
    oplismos_c(col_club(cc)(cm), 0, 2) = oplismos_c_club(0, 2)

    oplismos_c(col_club(cc)(cm), 1, 0) = oplismos_c_club(1, 0)
    oplismos_c(col_club(cc)(cm), 1, 1) = oplismos_c_club(1, 1)
    oplismos_c(col_club(cc)(cm), 1, 2) = oplismos_c_club(1, 2)

    oplismos_c(col_club(cc)(cm), 2, 0) = oplismos_c_club(2, 0)
    oplismos_c(col_club(cc)(cm), 2, 1) = oplismos_c_club(2, 1)
    oplismos_c(col_club(cc)(cm), 2, 2) = oplismos_c_club(2, 2)
Next

Next 'ALL COLUMN CLUBS
#End Region

#Region "Z) ΥΠΟΛΟΓΙΣΤΩ ΤΙΣ ΡΟΠΕΣ ΑΝΤΟΧΗΣ"

For c = 0 To columns.GetLength(0) - 1
    Dim rtrn(1) As Single
    Dim Asv, Ash, Nd As Single

    Asv = oplismos_c(c, 1, 2) + oplismos_c(c, 0, 2) / 2
    Ash = oplismos_c(c, 2, 2) + oplismos_c(c, 0, 2) / 2

    'Nd katw ypostylwmatos (to elaxisto N apo oloys tous syndyasmous drasewn)
    Dim Nd_min As Single
    For combo = 0 To N_combo.GetLength(0) - 1 'to elaxisto Ned gia ola ta com
bo
        If combo = 0 Then
            Nd_min = N_combo(combo, columns(c, 0))
        ElseIf N_combo(combo, columns(c, 0)) < Nd_min Then
            Nd_min = N_combo(combo, columns(c, 0))
        End If
    Next
    If Nd_min > 0 Then Nd_min = 0
    Nd = Nd_min 'sel 198 FARDIS pernw to elaxisto N gia critical

    rtrn = F_Mrd_c2(Asv, Ash, Nd, Es, columns(c, 1), columns(c, 2), d1_2, fck
, fyk)

    Mrd_c(c, 0) = rtrn(0) 'X-X καμψη περι τον χ αξονα
    Mrd_c(c, 1) = rtrn(1) 'Y-Y καμψη περι τον y αξονα
Next 'all columns

'M_rd,beams,kombou gia ikanotiko
For n = 0 To nodes_dt.Rows.Count - 1
    For c = 0 To columns.GetLength(0) - 1
        If columns(c, 4) = n Then
            M_ikanot(n, 1, 0, 0) += Mrd_c(c, 0) 'X-X' ARISTEROSTROFA

```

```

        M_ikanot(n, 1, 0, 1) += Mrd_c(c, 0) 'X-X' DE3IOSTROFA
        M_ikanot(n, 1, 1, 0) += Mrd_c(c, 1) 'Y-Y' ARISTEROSTROFA
        M_ikanot(n, 1, 1, 1) += Mrd_c(c, 1) 'Y-Y' DE3IOSTROFA
    ElseIf columns(c, 3) = n Then
        M_ikanot(n, 1, 0, 0) += Mrd_c(c, 0) 'X-X' ARISTEROSTROFA
        M_ikanot(n, 1, 0, 1) += Mrd_c(c, 0) 'X-X' DE3IOSTROFA
        M_ikanot(n, 1, 1, 0) += Mrd_c(c, 1) 'Y-Y' ARISTEROSTROFA
        M_ikanot(n, 1, 1, 1) += Mrd_c(c, 1) 'Y-Y' DE3IOSTROFA
    End If
Next 'all beams
Next 'all nodes

#End Region 'ΔΙΑΣΤΑΣΙΟΛΟΓΗΣΗ ΔΟΚΩΝ
#End Region 'ΔΙΑΣΤΑΣΙΟΛΟΓΗΣΗ ΥΠΟΣΤΥΛΩΜΑΤΩΝ

End Sub

'=====FUNCTIONS ΠΟΥ ΧΡΗΣΙΜΟΠΟΙΟΥΝΤΑΙ=====

'5.2 MEGISTH DIAMETROS RABDOY POY MPAINEI SE KOMBO
Public Function F_Max_dbL(joint_type As Byte, fck As Single, fyk As Single, Nd As
Single, Ac As Single, hc As Single, KP As Byte) As Single

    'joint_type(0=eswterikos,1=e3wterikos) sthn dieu8ynsh ths dokou!!!
    'vd = the minimum in all combinations
    'ρ1,max megisto pososto oplismou anw oplismou dokou
    'ρ2 pososto oplismou katw oplismou dokou
    'ean o oplismos den einai gnwstos tote ρ2 = 0,5 ρ1,max (etsi antikatesthsa ρ2
/ρ1,max = 0.5)

    Dim dbl_max, fcd, fyd, fctm, k, γRd, vd As Single

    fctm = (0.3 * (fck / 1000) ^ (2 / 3)) * 1000 'se Mpa
    fcd = fck / 1.5
    fyd = fyk / 1.15
    vd = Math.Abs(Nd) / (Ac * fcd)

    Select Case KP
        Case 1
            γRd = 1 : k = 0.5
        Case 2
            γRd = 1.2 : k = 0.75
    End Select

    Select Case joint_type
        Case 0 'eswterikos kombos
            dbl_max = hc * (7.5 * fctm / (γRd * fyd) * (1 + 0.8 * vd) / (1 + k *
0.5))

        Case 1 'e3wterikos kombos
            dbl_max = hc * (7.5 * fctm / (γRd * fyd) * (1 + 0.8 * vd))

```

```

End Select

Return dbl_max

End Function

'5.3.1 KAM4H DOKOU
Public Function F_As_b(Med As Single, beff As Single, h As Single, d1 As Single,
d2 As Single, fck As Single, fyk As Single) As Single()
Dim result(1) As Single
Dim μsd, ω1, ξlim, μsdlim, ω2, ωlim, As1, As2, εcu2, Es, fyd, fcd, d As S
ingle

εcu2 = 0.0035
Es = 200 * 10 ^ 3
fcd = 0.85 * fck / 1.5
fyd = fyk / 1.15
d = h - d1
μsd = Math.Abs(Med) / (beff * d ^ 2 * fcd)

If μsd = 0 Then : result(0) = As1 : result(1) = As2 : Return result : End If

ξlim = 0.45
μsdlim = 0.8 * ξlim * (1 - 0.4 * ξlim)
If μsd < μsdlim Then 'den apaiteitai 8libomenos
ω1 = 1 - Math.Sqrt(1 - 2 * μsd)
As1 = ω1 * beff * d * fcd / fyd
As2 = 0
Else 'apaiteitai 8libomenos
ω2 = (μsd - μsdlim) / (1 - d2 / d)
ωlim = 1 - Math.Sqrt(1 - 2 * μsdlim)
ω1 = ωlim + ω2
As1 = ω1 * beff * d * fcd / fyd
As2 = ω2 * beff * d * fcd / fyd
End If
result(0) = As1
result(1) = As2
Return result
End Function

''5.3.2 EC8 diata3eis
Public Function F_As_b_EC8_1(KP As Byte, bw As Single, d As Single, fck As Single
, fyk As Single, Es As Single) As Single

'As,min PANTOU

Dim fctm, fyd, fcd, εyd As Single
fctm = 0.3 * (fck ^ (2 / 3))
fcd = 0.85 * fck / 1.5
fyd = fyk / 1.15
εyd = fyd / Es

Dim As_min, ρ_min As Single

Select Case KP
Case 1 'KPM
ρ_min = 0.5 * fctm / fyk

```

```

        As_min = Math.Max(ρ_min * bw * d, 2 * Math.PI * 0.014 ^ 2 / 4)
    Case 2 'KPY
        ρ_min = 0.5 * fctm / fyk
        As_min = Math.Max(ρ_min * bw * d, 2 * Math.PI * 0.014 ^ 2 / 4)
    End Select

Dim rtrn As Single
rtrn = As_min
Return rtrn

End Function

Public Function F_As_b_EC8_2(KP As Byte, bw As Single, h As Single, fck As Single
, fyk As Single, μφ As Single, Es As Single, As_top_supp1 As Single,
        As_top_supp2 As Single, As_bot_span As Single) As Si
ngle()

    'μφ sel66 koutas
    'As1 embadon 8libomenou : to 8elw gia na brw ρ'-
    > pmax (sthv arxh gia As_min den to gnwrizw)

    Dim fctm, fyd, fcd, εyd As Single
    fctm = 0.3 * (fck ^ (2 / 3))
    fcd = 0.85 * fck / 1.5
    fyd = fyk / 1.15
    εyd = fyd / Es

    Dim As_max, pmin_tonos1, pmin_tonos2, ρ_cr1, ρ_cr2, ρ_max, Lcr, As_min_to
p_span, As_min_bot_supp1, As_min_bot_supp2 As Single
    ρ_cr1 = As_top_supp1 / (bw * h)
    ρ_cr2 = As_top_supp2 / (bw * h)

    Select Case KP
        Case 1 'KPM
            Lcr = h
            'ektos critical (sto anoigma)
            ρ_max = 0.04
            As_max = ρ_max * bw * h
            'entos critical (stis sthri3eis)
            pmin_tonos1 = ρ_cr1 - 0.0018 / (μφ * εyd) * fcd / fyd
            pmin_tonos2 = ρ_cr2 - 0.0018 / (μφ * εyd) * fcd / fyd

            As_min_top_span = 0
            As_min_bot_supp1 = Math.Max(Math.Max(0.25 * As_bot_span, 0.5 * As
_top_supp1), pmin_tonos1 * (bw * h))
            As_min_bot_supp2 = Math.Max(Math.Max(0.25 * As_bot_span, 0.5 * As
_top_supp2), pmin_tonos2 * (bw * h))

        Case 2 'KPY
            Lcr = 1.5 * h
            'ektos critical (sto anoigma)
            ρ_max = 0.04
            As_max = ρ_max * bw * h
            'entos critical (stis sthri3eis)
            pmin_tonos1 = ρ_cr1 - 0.0018 / (μφ * εyd) * fcd / fyd

```

```

pmin_tonos2 = ρ_cr2 - 0.0018 / (μφ * εyd) * fcd / fyd

As_min_top_span = Math.Max(0.25 * As_top_supp1, 0.25 * As_top_sup
p2)
As_min_bot_supp1 = Math.Max(Math.Max(0.25 * As_bot_span, 0.5 * As
_top_supp1), pmin_tonos1 * (bw * h))
As_min_bot_supp2 = Math.Max(Math.Max(0.25 * As_bot_span, 0.5 * As
_top_supp2), pmin_tonos2 * (bw * h))
End Select

Dim rtrn(4) As Single

rtrn(0) = Lcr
rtrn(1) = As_max
rtrn(2) = As_min_top_span
rtrn(3) = As_min_bot_supp1
rtrn(4) = As_min_bot_supp2

Return rtrn

End Function

'5.3.4 MRD DOKOY SE KAM4H
Public Function F_Mrd_b(As1 As Single, As2 As Single, bw As Single, beff As Singl
e, h As Single,
d1 As Single, d2 As Single, fck As Single, fyk As Single) A
s Single()

'As1 panw , As2 katw
Dim fcd, fyd As Single
fcd = 0.85 * fck / 1.5
fyd = fyk / 1.15
Dim result(1) As Single
result(0) = Math.Min(As1, As2) * fyd * (h - d1 - d2) + Math.Max(0, (As1 - As2
)) * fyd * (h - d1 - 0.5 * (As1 - As2) * fyd / (bw * fcd)) 'Mrd-
result(1) = As2 * fyd * Math.Max((h - d2 - 0.5 * As2 * fyd / (beff * fcd)), (
h - d1 - d2)) 'Mrd+

Return result

End Function

'5.4.2 KAM4H YPOSTYLWMATOS
Public Function F_As_c(Md As Single, Nd As Single, fyk As Single, fck As Single,
Es As Single, b As Single,
h As Single, d1 As Single) As Single
Dim μd, vd, δ1, v1, v2, ξ, w1d, fcd, fyd, d As Single
Const εc2 = 0.002, εcu2 = 0.0035
fcd = fck / 1.5
fyd = fyk / 1.15
d = h - d1
Dim εyd = fyd / Es
'As1 panw , As2 katw

```

```

'Gia or8ogwnia ypostylwmata me symmetriko oplismo
μd = Math.Abs(Md) / (b * d ^ 2 * fcd)
vd = Math.Abs(Nd) / (b * d * fcd)
δ1 = d1 / d

If μd = 0 Then Return 0

v1 = (εcu2 - εc2 / 3) / (εcu2 + εyd)
v2 = δ1 * (εcu2 - εc2 / 3) / (εcu2 - εyd)

If v2 < vd And vd < v1 Then
  'MsgBox("case1 v2 < vd")
  ξ = vd / (1 - (εc2 / 3 * εcu2))

  ω1d = (μd - ξ * ((1 - ξ) / 2 - εc2 / (3 * εcu2)) * (1 / 2 - ξ + εc2 /
(4 * εcu2) * ξ))) / (1 - δ1)

ElseIf vd <= v2 Then
  'MsgBox("case2 vd <= v2")

  Dim ξ1, ξ2, ξ_calc, diff As Single
  Dim c, lc As Integer
  Dim infinityprotection2, infinityprotection1 As Double
  Const tolerance = 0.001
  diff = 100

  c = 0 ' poses fores perasa to 0
  lc = 0 'metrhths loop

  Do Until Math.Abs(diff) < tolerance
    lc = lc + 1
    ξ = ξ + 0.1 * (10 ^ (-c))

    If lc > 1 Then
      infinityprotection1 = diff
    End If

    ω1d = (μd - ξ * ((1 - ξ) / 2 - εc2 / (3 * εcu2)) * (0.5 - ξ + εc2
/ (4 * εcu2) * ξ))) / ((1 - δ1) / 2 * (1 + (ξ - δ1) / ξ * εcu2 / εyd))

    Dim Δ, α, β, γ As Single
    '(1 - εc2 / (3 * εcu2)) * ξ ^ 2 - (vd + ω1d * (1 - εcu2 / εyd)) *
ξ - ω1d * εcu2 * δ1 / εyd = 0
    α = (1 - εc2 / (3 * εcu2))
    β = -(vd + ω1d * (1 - εcu2 / εyd))
    γ = -(ω1d * (εcu2 * δ1) / εyd)

    Δ = β ^ 2 - 4 * α * γ

    ξ1 = (-β + Math.Sqrt(Δ)) / (2 * α)
    ξ2 = (-β - Math.Sqrt(Δ)) / (2 * α)
    ξ_calc = Math.Max(ξ1, ξ2)

    diff = ξ - ξ_calc
    'MsgBox(lc & vbCrLf & " ξ = " & ξ & vbCrLf & " ξcalc = " & ξ_calc
& vbCrLf & " diff = " & diff)

```

```

infinityprotection2 = diff

If lc > 1000 Then
    If infinityprotection2 >= infinityprotection1 Then
        MsgBox("Η διαδικασία τερματίστηκε απρόοπτα")
        Exit Do
    End If
End If

If diff > 0 Then
    c = c + 1
    ξ = ξ - 0.1 * (10 ^ (-1 * (c - 1)))
    diff = infinityprotection1
End If
Loop

ElseIf vd >= v1 Then
    MsgBox("case3  vd >= v1")

    Dim ξ1, ξ2, ξ_calc, diff As Single
    Dim c, lc As Integer
    Dim infinityprotection2, infinityprotection1 As Double
    Const tolerance = 0.001
    diff = 100

    c = 0 ' poses fores perasa to 0
    lc = 0 'metrhths loop

    Do Until Math.Abs(diff) < tolerance
        lc = lc + 1
        ξ = ξ + 0.1 * (10 ^ (-c))

        If lc > 1 Then
            infinityprotection1 = diff
        End If

        ω1d = (μd - ξ * ((1 - ξ) / 2 - εc2 / (3 * εcu2)) * (0.5 - ξ + εc2
/ (4 * εcu2) * ξ)) / ((1 - δ1) / 2 * (1 + (1 - ξ) / ξ * εcu2 / εyd))

        Dim Δ, α, β, γ As Single
        α = (1 - εc2 / (3 * εcu2))
        β = -(vd - ω1d * (1 + εcu2 / εyd))
        γ = -(ω1d * (εcu2) / εyd)

        Δ = β ^ 2 - 4 * α * γ

        ξ1 = (-β + Math.Sqrt(Δ)) / (2 * α)
        ξ2 = (-β - Math.Sqrt(Δ)) / (2 * α)
        ξ_calc = Math.Max(ξ1, ξ2)

        diff = ξ - ξ_calc
        MsgBox(lc & vbCrLf & " ξ = " & ξ & vbCrLf & " ξcalc = " & ξ_calc
& vbCrLf & " diff = " & diff)
        infinityprotection2 = diff

        If lc > 1000 Then

```

```

        If infinityprotection2 >= infinityprotection1 Then
            MsgBox("Η διαδικασία τερματίστηκε απρόοπτα")
        Exit Do
    End If
End If

    If diff > 0 Then
        c = c + 1
        ξ = ξ - 0.1 * (10 ^ (-1 * (c - 1)))
        diff = infinityprotection1
    End If
Loop

End If
MsgBox(ξ)
Dim As1 As Single

As1 = w1d * b * d * (fcd / fyd)

Dim result As Single
result = As1
Return result

End Function

Public Function F_As_c2(Mx As Single, My As Single, Nd As Single, fyk As Single,
fck As Single, Es As Single,
    b As Single, h As Single, d1 As Single) As Single()
    ' by the side length parallel to the vector of My (des sxhma sel 200 fardis)
    Dim result(1) As Single
    Dim Asx, Asy As Single

    Asx = F_As_c(Mx, Nd, fyk, fck, Es, b, h, d1)
    Asy = F_As_c(My, Nd, fyk, fck, Es, h, b, d1)

    result(0) = Asx 'Asv parallhlo ston a3ona x
    result(1) = Asy 'Ash parallhlo ston a3ona y
    Return result

End Function

'5.4.3 Mrd YPOSTYLWMATOS dedomeno oplismo kai a3oniko fortio
Public Function F_Mrd_c(As1 As Single, As2 As Single, Asv As Single, Nd As Single
, Es As Single, b As Single, h As Single, d1 As Single, fck As Single, fyk As Single)
As Single
    Dim Mrd, vd, δ1, v1, v2, ξ, w1d, w2d, wvd, fcd, fyd, d As Single
    Const εc2 = 0.002, εcu2 = 0.0035
    fcd = fck / 1.5 'o FARDIS STO PARADEIGMA XWRIS 0.85
    fyd = fyk / 1.15
    d = h - d1
    Dim eyd = fyd / Es

    'se ka8e gwnia to miso paei horizontal- to allo miso vertical

    w1d = As1 / (b * d) * (fyd / fcd)

```



```

w2d = As2 / (b * d) * (fyd / fcd)
wvd = Asv / (b * d) * (fyd / fcd)

δ1 = d1 / d

vd = Math.Abs(Nd) / (b * d * fcd)
v1 = w2d - w1d + wvd / (1 - δ1) * ((εcu2 - εyd) / (εcu2 + εyd) - δ1) + (εcu2
- εc2 / 3) / (εcu2 + εyd)
v2 = w2d - w1d + wvd / (1 - δ1) * (δ1 * (εcu2 + εyd) / (εcu2 - εyd) - 1) + δ1
* (εcu2 - εc2 / 3) / (εcu2 - εyd)

'MsgBox("vd = " & vd & " v1 = " & v1 & " v2 = " & v2)
If v2 < vd And vd < v1 Then

    ξ = ((1 - δ1) * (vd + w1d - w2d) + (1 + δ1) * wvd) / ((1 - δ1) * (1 - εc2
/ (3 * εcu2)) + 2 * wvd)

    Mrd = (ξ * ((1 - ξ) / 2 - εc2 / (3 * εcu2) * (0.5 - ξ + εc2 / (4 * εcu2)
* ξ)) + ((1 - δ1) * (w1d + w2d)) / 2 + wvd / (1 - δ1) * ((ξ - δ1) * (1 - ξ) - 1 / 3 *
(ξ * εyd / εcu2) ^ 2)) * (b * d ^ 2 * fcd)

ElseIf vd <= v2 Then

    Dim Δ, α, β, γ, ξ1, ξ2 As Single

    α = (1 - εc2 / (3 * εcu2) + wvd / (2 * (1 - δ1)) * (εcu2 + εyd) ^ 2 / (εc
u2 * εyd))
    β = -
(vd + w1d - w2d * εcu2 / εyd + wvd / (1 - δ1) * (1 + εcu2 / εyd * δ1))
    γ = -(w2d - wvd * δ1 / (2 * (1 - δ1))) * εcu2 / εyd * δ1

    Δ = β ^ 2 - 4 * α * γ

    ξ1 = (-β + Math.Sqrt(Δ)) / (2 * α)
    ξ2 = (-β - Math.Sqrt(Δ)) / (2 * α)
    ξ = Math.Max(ξ1, ξ2)
'MsgBox("ξ1 = " & ξ1 & "ξ2 = " & ξ2)
    Mrd = (ξ * ((1 - ξ) / 2 - εc2 / (3 * εcu2) * (0.5 - ξ + εc2 / (4 * εcu2)
* ξ)) + (1 - δ1) / 2 * (w1d + w2d * (ξ - δ1) / ξ * εcu2 / εyd) + wvd / (4 * (1 - δ1))
* (ξ * (1 + εyd / εcu2) - δ1) * (1 + εcu2 / εyd * ((ξ - δ1) / ξ)) * (1 - δ1 / 3 - 2
/ 3 * ξ * (1 + εyd / εcu2))) * (b * d ^ 2 * fcd)

ElseIf vd >= v1 Then

    Dim Δ, α, β, γ, ξ1, ξ2 As Single

    α = (1 - εc2 / (3 * εcu2) - wvd / (2 * (1 - δ1)) * (εcu2 - εyd) ^ 2 / (εc
u2 * εyd))
    β = (w2d + w1d * εcu2 / εyd - vd + wvd / (1 - δ1) * (εcu2 / εyd - δ1))
    γ = -(w1d + wvd / (2 * (1 - δ1))) * εcu2 / εyd

    Δ = β ^ 2 - 4 * α * γ

    ξ1 = (-β + Math.Sqrt(Δ)) / (2 * α)
    ξ2 = (-β - Math.Sqrt(Δ)) / (2 * α)
    ξ = Math.Max(ξ1, ξ2)

```

```

        'MsgBox("ξ1 = " & ξ1 & "ξ2 = " & ξ2)
        Mrd = (ξ * ((1 - ξ) / 2 - εc2 / (3 * εcu2) * (0.5 - ξ + εc2 / (4 * εcu2)
* ξ)) + (1 - δ1) / 2 * (ω1d * (1 - ξ) / ξ * εcu2 / εyd + ω2d) + ωvd / (4 * (1 - δ1))
* (1 - ξ * (1 - εyd / εcu2)) * (1 + εcu2 / εyd * ((1 - ξ) / ξ)) * (1 / 3 - δ1 + 2 / 3
* ξ * (1 - εyd / εcu2))) * b * d ^ 2 * fcd

    End If
    'MsgBox("ξ = " & ξ)
    Dim result As Single
    result = Mrd
    Return result

End Function

Public Function F_Mrd_c2(Asv As Single, Ash As Single, Nd As Single, Es As Single
, b As Single, h As Single, d1 As Single, fck As Single, fyk As Single) As Single()

    Dim Mrd_x, Mrd_y As Single
    'se ka8e gwnia to miso paei horizontal- to allo miso vertical
    Mrd_x = F_Mrd_c(Asv, Asv, Ash, Nd, Es, b, h, d1, fck, fyk)
    Mrd_y = F_Mrd_c(Ash, Ash, Asv, Nd, Es, b, h, d1, fck, fyk)

    Dim result(1) As Single
    result(0) = Mrd_x
    result(1) = Mrd_y
    Return result

End Function

'ypologismos elaxistoy ari8mou rabdwn ana pleura ypostylwmatos + ana pleura ari8m
os rabdwn se gwnia sundethra
Public Function F_nmin_c(b As Single, h As Single, d1 As Single, KP As Byte) As B
yte()

    'd1 = d2, KP -> kathgoria plastimothtas ( 1=mesh, 2=yppshlh)

    Dim nmin0, nmin(1) As Byte 'nmin0 elax kata ec2, nmin elax ari8mos rabdwn, ns
ari8mos rabdwn sygkrat sundethra APO TA DUO PERNW TO MEGALYTERO
    Dim dmax_gwniagwnia, d_gwniagwnia, n As Single

    Select Case KP
        Case 1 : nmin0 = 3 : dmax_gwniagwnia = 200 / 1000
        Case 2 : nmin0 = 3 : dmax_gwniagwnia = 150 / 1000
    End Select

    Dim dimension(1) As Single
    dimension(0) = b - 2 * d1
    dimension(1) = h - 2 * d1

    For i = 0 To 1 ' gia tis 2 diastaseis
        Dim c As Integer = 0
        Do
            n = nmin0 + c
            d_gwniagwnia = dimension(i) / (n - 1)
            c += 1
        Loop Until d_gwniagwnia <= dmax_gwniagwnia
        nmin(i) = n
    
```

Next

```
Dim result(1) As Byte
result(0) = nmin(0) 'b diastash
'result(0, 1) = ns(0) 'den xrhshmpoieitai
result(1) = nmin(1) 'h diastash
'result(1, 1) = ns(1) 'den xrhshmpoieitai
Return result
```

End Function

```
'ρmin/max kai φmin/max elegxos ANA PLEURA
Public Function F_fmin_max_c_old(b As Single, h As Single, Nv As Byte, Nh As Byte
, φmin As Single, φmax As Single, KP As Byte) As Single(,)
```

```
Dim ρ, ρmin, ρmax, Asi, Ac As Single
Dim φmin1, φmin2, φmax1, φmax2 As Single
Dim outcome1, outcome2 As Boolean
Ac = b * h
```

Select Case KP

```
Case 1
    ρmin = 0.01
    ρmax = 0.04
Case 2
    ρmin = 0.01
    ρmax = 0.04
```

End Select

'elegxos φmin gia ρmin

```
For φ = φmin To φmax Step 2
    φmin1 = φ
    φmin2 = φ
    Asi = Math.PI * φ ^ 2 / 4 * (2 * Nv + 2 * Nh - 4)
    ρ = Asi / Ac
```

```
If ρ < ρmin Then 'prospa8w kai ton sunduasmο tous
    Dim ρnew As Single
```

```
For φ2 = φ + 2 To φ + 4 Step 2
    If φ2 > φmax Then
        Exit For
    End If
```

```
Asi = (Math.PI * φ ^ 2 / 4 * (2 * Nv + 2 * Nh - 4 - 4)) + (Math.P
I * φ2 ^ 2 / 4 * 4) 'stis duo gwniakēs au3anw diametro
```

```
ρnew = Asi / Ac
```

```
If ρnew > ρmin Then
```

```
    φmin1 = φ
    φmin2 = φ2
```

```
    ρ = ρnew 'gia na perasei ton elegxo katw kai na bgei
```

```
Exit For
```

```
End If
```

Next

End If

```
If ρ > ρmin Then
```

```

        Exit For
    End If
Next

'telikos elegchos gia na dw ean eparkoun oi times pou e3agw
If ρ >= ρmin Then
    outcome1 = True
Else
    outcome1 = False
End If

'elegchos φmax gia ρmax
For φ = φmax To φmin Step -2
    φmax1 = φ
    φmax2 = φ
    Asi = Math.PI * φ ^ 2 / 4 * (2 * Nv + 2 * Nh - 4)
    ρ = Asi / Ac

    If ρ > ρmax Then 'prospa8w kai ton sunduasmο tous
        Dim ρnew As Single

        For φ2 = φ - 2 To φ - 4 Step -2
            If φ2 < φmin Then
                Exit For
            End If
            Asi = (Math.PI * φ ^ 2 / 4 * (2 * Nv + 2 * Nh - 4 - 4)) + (Math.P
I * φ2 ^ 2 / 4 * 4) 'stis duo gwniakes au3anw diametro
            ρnew = Asi / Ac
            If ρnew < ρmax Then
                φmax1 = φ
                φmax2 = φ2
                ρ = ρnew 'gia na perasei ton elegcho katw kai na bgei
                Exit For
            End If
        Next
    End If

    Next
End If

If ρ < ρmax Then
    Exit For
End If
Next

'telikos elegchos gia na dw ean eparkoun oi times pou e3agw
If ρ <= ρmax Then
    outcome2 = True
Else
    outcome2 = False
End If
MsgBox("")

Dim result(1, 2) As Single
result(0, 0) = φmin1
result(0, 1) = φmin2 'oi gwniakes rabdoi
result(0, 2) = -outcome1
result(1, 0) = φmax1
result(1, 1) = φmax2 'oi gwniakes rabdoi

```

```

result(1, 2) = -outcome2
Return result

```

```
End Function
```

```
Public Function F_fmin_max_c(b As Single, h As Single, Nv As Byte, Nh As Byte, KP
As Byte) As Single(,)
```

```

Dim ρ, ρmin, ρmax, Asi, Ac As Single
Dim φmin1, φmin2, φmax1, φmax2 As Single
Dim outcome1, outcome2 As Boolean
Ac = b * h

```

```
Select Case KP
```

```

Case 1
    ρmin = 0.01
    ρmax = 0.04
Case 2
    ρmin = 0.01
    ρmax = 0.04

```

```
End Select
```

```
'elegxos φmin gia ρmin
```

```
Dim φ As Single
```

```
For i = 0 To D_available.GetLength(0) - 1
```

```

    φ = D_available(i)
    φmin1 = φ
    φmin2 = φ
    Asi = Math.PI * φ ^ 2 / 4 * (2 * Nv + 2 * Nh - 4)
    ρ = Asi / Ac

```

```
If ρ < ρmin Then 'prospa8w kai ton sunduasmο tous
```

```
    Dim ρnew As Single
```

```
    Dim φ2 As Single
```

```
    For j = i + 1 To i + 2
```

```
        If j > D_available.GetLength(0) - 1 Then
```

```
            Exit For
```

```
        End If
```

```
        φ2 = D_available(j)
```

```
        Asi = (Math.PI * φ ^ 2 / 4 * (2 * Nv + 2 * Nh - 4 - 4)) + (Math.P
```

```
I * φ2 ^ 2 / 4 * 4) 'stis duo gwniakes au3anw diametro
```

```
        ρnew = Asi / Ac
```

```
        If ρnew > ρmin Then
```

```
            φmin1 = φ
```

```
            φmin2 = φ2
```

```
            ρ = ρnew 'gia na perasei ton elegxo katw kai na bgei
```

```
            Exit For
```

```
        End If
```

```
    Next
```

```
End If
```

```
If ρ > ρmin Then
```

```
    Exit For
```

```
End If
```

```
Next
```

```
'telikos elegchos gia na dw ean eparkoun oi times pou e3agw
If ρ >= ρmin Then
    outcome1 = True
Else
    outcome1 = False
End If

Dim result(1, 2) As Single
result(0, 0) = φmin1
result(0, 1) = φmin2 'oi gwniakes rabdoi
result(0, 2) = -outcome1
'-----den xrhshmpoiw
result(1, 0) = φmax1
result(1, 1) = φmax2 'oi gwniakes rabdoi
result(1, 2) = -outcome2
Return result

End Function

End Module
```