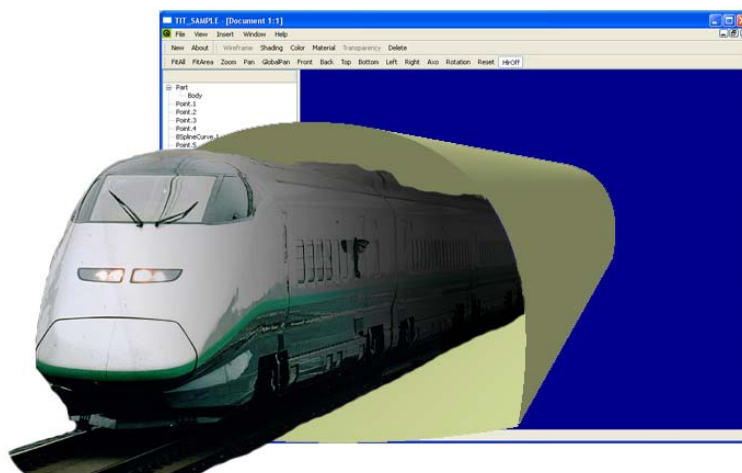




ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ & ΔΙΟΙΚΗΣΗΣ

Μεταπτυχιακή Διατριβή

Ανάπτυξη μεθοδολογίας και του αντίστοιχου λογισμικού για τη
διεπιδραστική σχεδίαση σηράγγων.



Διατριβή που υπεβλήθη για τη μερική ικανοποίηση των απαιτήσεων για την
απόκτηση Μεταπτυχιακού Διπλώματος Ειδίκευσης

υπό
Σωτηρίου Σ. Σαρακηνού

Χανιά 2007

Η διατριβή του Σωτήριου Σ. Σαρακηνού, εγκρίνεται

Ιωάννης Κ. Νικολός
Λέκτορας, Επιβλέπων

Βασίλειος Σ. Κουϊκόγλου
Καθηγητής

Νικόλαος Μπιλάλης
Αναπληρωτής Καθηγητής

Περιεχόμενα

ΠΡΟΛΟΓΟΣ	10
ΚΕΦΑΛΑΙΟ 1^ο: ΕΙΣΑΓΩΓΗ	11
1.1: ΓΕΝΙΚΑ	11
1.2: Η ΧΡΗΣΗ ΤΩΝ ΣΗΡΑΓΓΩΝ ΑΝΑ ΤΟΝ ΚΟΣΜΟ.	12
1.2: Η ΣΗΜΕΡΙΝΗ ΔΙΑΔΙΚΑΣΙΑ ΜΟΝΤΕΛΟΠΟΙΗΣΗΣ ΚΑΙ ΜΕΛΕΤΗΣ ΜΙΑΣ ΣΗΡΑΓΓΑΣ.	16
1.2.1: Βήμα 1 ^ο	16
1.2.2: Βήμα 2 ^ο	16
1.2.3: Βήμα 3 ^ο	18
1.2.4: Βήμα 4 ^ο	19
1.2.5: Βήμα 5 ^ο	19
1.2.6: Βήμα 6 ^ο	19
1.3: ΥΠΑΡΧΟΝΤΑ ΠΡΟΓΡΑΜΜΑΤΑ ΜΟΝΤΕΛΟΠΟΙΗΣΗΣ ΣΗΡΑΓΓΩΝ	20
1.3.1: FIDES-WinTUBE	20
1.3.2: Midas/GTS	22
ΚΕΦΑΛΑΙΟ 2^ο: ΔΥΝΑΤΟΤΗΤΕΣ ΚΑΙ ΔΟΜΗ ΤΟΥ ΛΟΓΙΣΜΙΚΟΥ ΠΟΥ ΑΝΑΠΤΥΧΘΗΚΕ	24
2.1: ΓΕΝΙΚΑ	24
2.2: ΤΟ ΠΕΡΙΒΑΛΛΟΝ ΤΡΙΔΙΑΣΤΑΤΗΣ ΑΠΕΙΚΟΝΙΣΗΣ.	24
2.3: ΓΕΩΜΕΤΡΙΚΕΣ ΟΝΤΟΤΗΤΕΣ	26
2.3.1: Σημεία και επίπεδα	26
2.3.2: Καμπύλες	29
2.3.3: Περιβάλλον διδιάστατης σχεδίασης	32
2.3.4: Επιφάνειες, όγκοι και λογικές πράξεις Boole.	35
2.3.5: Οδηγός δημιουργίας διατομών σιράγγων	39
2.3.6: Οδηγός δημιουργίας διασταυρώσεων σιράγγων	40
2.4: Η ΑΝΤΙΚΕΙΜΕΝΟΣΤΡΑΦΗΣ ΔΟΜΗ ΤΟΥ ΠΡΟΓΡΑΜΜΑΤΟΣ.	44
2.4.1: Κλάση bfc	45
2.4.2: Κλάση Point3D	46
2.4.3: Κλάση Plane3D	47
2.4.4: Κλάση Pln_T1	48
2.4.5: Κλάση Pln_T2	49
2.4.6: Κλάση GShape	49
2.4.7: Κλάση GEdge	49
2.4.8: Κλάση BSplineCurve	50
2.4.9: Κλάση pCircle	52
2.4.10: Κλάση GWire	52
2.4.11: Κλάση Sweep	53
2.4.12: Κλάση pBox	54
2.4.13: Κλάση pSphere	54
2.4.14: Κλάση BoolOp	55
ΚΕΦΑΛΑΙΟ 3^ο: ΠΑΡΑΔΕΙΓΜΑΤΑ	56
3.1: ΣΥΣΤΗΜΑ ΣΗΡΑΓΓΩΝ ΠΟΥ ΔΙΑΚΛΑΔΙΖΟΝΤΑΙ	56
3.2: ΣΗΡΑΓΓΑ ΜΕ ΕΝΔΙΑΜΕΣΗ ΠΛΑΤΥΝΣΗ.	58
3.2: ΣΥΣΤΗΜΑ ΠΑΡΑΛΛΗΛΩΝ ΣΗΡΑΓΓΩΝ ΜΕ ΒΟΗΘΗΤΙΚΗ ΣΗΡΑΓΓΑ	59
ΣΥΜΠΕΡΑΣΜΑΤΑ	69
ΒΙΒΛΙΟΓΡΑΦΙΑ	70
ΠΑΡΑΡΤΗΜΑ	71
I) ΠΑΡΟΥΣΙΑΣΗ ΤΗΣ ΒΙΒΛΙΟΘΗΚΗΣ OPEN CASCADE	71
II) ΟΙ ΒΙΒΛΙΟΘΗΚΕΣ ΤΗΣ OPEN CASCADE	72
III) ΟΙ ΚΛΑΣΕΙΣ ΤΗΣ OPEN CASCADE ΣΤΟ ΛΟΓΙΣΜΙΚΟ ΠΟΥ ΑΝΑΠΤΥΧΘΗΚΕ	72
1) Δημιουργία παράθυρου προβολής αντικειμένων.	73

2) Κλάσεις τοπολογίας	74
3) Γεωμετρικές κλάσεις	75

Ευρετήριο Εικόνων

Σχήμα 1.1: Σχηματική αναπαράσταση του Ευπαλίνειου ορύγματος στο οριζόντιο (α) και κατακόρυφο (β) επίπεδο [Wikipedia2].	13
Εικόνα 1.2: Εσωτερική άποψη της σήραγγας του Τάμεση. Γκραβούρα του 19 ^{ου} αιώνα [Thames].	13
Εικόνα 1.3: Η γραμμή North East MRT στη Σιγκαπούρη είναι πλήρως υπόγεια [NorthEast].	14
Εικόνα 1.4.: Σήραγγα που ενώνει δύο κτήρια στο αεροδρόμιο του Νπιτρόιτ [Hopson].	15
Εικόνα 1.5: Σχέδια σε σύστημα CAD της διατομής της σήραγγας σε συγκεκριμένες τομές της [Πάππου].	16
Εικόνα 1.6: (α) Εικόνες από τη διάνοξη σήραγγας και φάσεων κατασκευής της. (β) Διάφοροι τύποι φάσεων κατασκευής, ανάλογα με την τοπολογία της σήραγγας [Πάππου].	17
Εικόνα 1.7: Διακριτοποιημένη διατομή σήραγγας, λαμβάνοντας υπόψη τις ζώνες εδάφους και τα μέτρα υποστήριξης [Πάππου].	18
Σχήμα 1.8: Απεικόνιση και αξιολόγηση των αποτελεσμάτων ανάλυσης στη διατομή της σήραγγας [Πάππου].	19
Εικόνα 1.9: Τριδιάστατα πλέγματα για την ανάλυση σιράγγων μονού και διπλού κλάδου, με φάσεις κατασκευής. Προκύπτουν από διδιάστατα πλέγματα με ομοιόθετη αναπαραγωγή στην τρίτη διάσταση. Μια τέτοια περίπτωση αποτελεί ουσιαστικά το «όριο» στην πραγματοποίηση τριδιάστατης μελέτης, με τα σημερινά υπολογιστικά εργαλεία [Sofistik].	20
Εικόνα 1.10: Το περιβάλλον σχεδίασης του προγράμματος FIDES-WinTUBE [FIDES].	21
Εικόνα 1.11: Το γραφικό περιβάλλον του ειδικού οδηγού μοντελοποίησης σιράγγων του MIDAS/GTS [Shin].	23
Εικόνα 2.1: Το βασικό παράθυρο του λογισμικού και το σύστημα πολλαπλών εγγράφων.	24
Εικόνα 2.2: Παρουσίαση της δυνατότητας δημιουργίας διαφορετικών προβολών του ίδιου εγγράφου.	25
Εικόνα 2.3: Το βασικό παράθυρο του προγράμματος και το δέντρο των αντικειμένων.	26
Εικόνα 2.4: Σημείο στο χώρο, ορισμένο από τις τριδιάστατες συντεταγμένες του. Το σημείο αναπαρίσταται με το σύμβολο του σταυρού.	27
Εικόνα 2.5: Ειδική φόρμα εισαγωγής δεδομένων για τον ορισμό ενός σημείου στο χώρο με τις καρτεσιανές του συντεταγμένες.	27
Εικόνα 2.6: Επίπεδο στο χώρο. Ένα επίπεδο αναπαρίσταται με ένα τετράγωνο που έχει τον προσανατολισμό του επιπέδου.	28
Εικόνα 2.7: Ειδική φόρμα εισαγωγής δεδομένων για σχεδίαση δυσδιάστατου επιπέδου.	28
Εικόνα 2.8: Καμπύλη NURBS τρίτου βαθμού ορισμένη από σημεία ελέγχου. Φαίνεται επίσης η ειδική φόρμα εισαγωγής δεδομένων για τη σχεδίασή της.	29
Εικόνα 2.9: Κύκλος ορισμένος από τρία σημεία.	30
Εικόνα 2.10: Τόξο κύκλου ορισμένο από τρία σημεία. Το αρχικό και τελικό σημείο θέτουν τα όρια του τόξου. Παρουσιάζεται επίσης η ειδική φόρμα εισαγωγής δεδομένων για το τόξο.	30
Εικόνα 2.11: Σύρμα που προέκυψε από την ένωση μιας ευθύγραμμης και μιας καμπύλης ακμής. Και οι δύο ακμές είναι ορισμένες ως καμπύλες NURBS.	31
Εικόνα 2.12: Η ειδική φόρμα εισαγωγής δεδομένων για τον ορισμό ενός σύρματος.	31
Εικόνα 2.13: Το περιβάλλον διδιάστατης σχεδίασης. Στα δεξιά φαίνεται το επίπεδο πάνω στο οποίο υλοποιείται η σχεδίαση.	32
Εικόνα 2.14: Ειδική φόρμα εισαγωγής δεδομένων για τον ορισμό των ιδιοτήτων του πλέγματος στο περιβάλλον διδιάστατης σχεδίασης.	33
Εικόνα 2.15: Τόξο κύκλου που δημιουργήθηκε στο περιβάλλον διδιάστατης σχεδίασης. Πάνω αριστερά φαίνεται το ίχνος του κέρσora για την εισαγωγή σημείου στο επίπεδο.	34
Εικόνα 2.16: Το περιβάλλον διδιάστατης σχεδίασης, όταν έχει γίνει περιστροφή του σχεδίου στο παράθυρο προβολής. Η σχεδίαση συνεχίζεται στις δύο διαστάσεις.	34
Εικόνα 2.17: Σχήματα που προέκυψαν από διαδικασία οδήγησης. Στα αριστερά το σχήμα είναι επιφάνεια και προκύπτει από γενέτειρα καμπύλη. Στα δεξιά το σχήμα είναι στερεό που προκύπτει από γενέτειρα επιφάνεια.	35

Εικόνα 2.18: Αποτέλεσμα δύο διαδικασιών οδήγησης με χρήση της ίδιας καμπύλης για οδηγό. Πάνω δεξιά η ειδική φόρμα εισαγωγής δεδομένων για το αριστερό σχήμα.	36
Εικόνα 2.19: Ορθογώνιο παραλληλεπίπεδο ορισμένο πάνω σε ένα επίπεδο στο χώρο.	36
Εικόνα 2.20: Σφαίρα που δημιουργήθηκε ορίζοντας ένα σημείο ως το κέντρο της, καθώς και την ακτίνα της.	37
Εικόνα 2.21: Λογική πράξη αποκοπής (Cut).	38
Εικόνα 2.22: Ο οδηγός δημιουργίας διατομών σηράγγων.	39
Εικόνα 2.23: Ο οδηγός δημιουργίας διασταυρώσεων.	40
Εικόνα 2.24: Διασταύρωση τεταρτημόριου με σήραγγες ίδιας διατομής.	41
Εικόνα 2.25: Διασταύρωση μισού με σήραγγες ίδιας διατομής.	41
Εικόνα 2.26: Διασταύρωση τύπου Γωνίας.	42
Εικόνα 2.27: Διασταύρωση τύπου T με σήραγγες ίδιας διατομής.	42
Εικόνα 2.28: Πλήρης διασταύρωση με σήραγγες ίδιας διατομής, υπό γωνία 120 μοιρών.	43
Εικόνα 2.29: Πλήρης διασταύρωση με χρήση σηράγγων διαφορετικής διατομής.	43
Σχήμα 2.30: Το δέντρο των σχεδιαστικών κλάσεων.	45
Εικόνα 3.1: Σύνθεση σηράγγων που τέμνονται σε μορφή Y.	56
Εικόνα 3.2: Τριγωνικό επιφανειακό πλέγμα στη σύνθεση σηράγγων της εικόνας 3.1. Η παραγωγή του πλέγματος έγινε με χρήση διαφορετικού λογισμικού.	57
Εικόνα 3.3: Τετραπλευρικό επιφανειακό πλέγμα στη σύνθεση σηράγγων της εικόνας 3.1. Η παραγωγή του πλέγματος έγινε με χρήση διαφορετικού λογισμικού.	57
Εικόνα 3.4: Σήραγγα με ενδιάμεση πλάτυνση.	58
Εικόνα 3.5: Σήραγγα με ενδιάμεση πλάτυνση. Έχει γίνει χρήση διαφάνειας για καλύτερη προβολή του εσωτερικού της.	58
Εικόνα 3.6: Σύνθεση παράλληλων σηράγγων, που συνδέονται με βοηθητική σήραγγα. Διακρίνεται αριστερά το δέντρο των αντικειμένων.	59
Εικόνα 3.7: Ορισμός αρχικού επιπέδου σχεδίασης.	60
Εικόνα 3.8: Ορισμός σημείων στο περιβάλλον διδιάστατης σχεδίασης.	60
Εικόνα 3.9: Δημιουργία τόξων κύκλου, που θα αποτελέσουν τις διατομές των σηράγγων.	61
Εικόνα 3.10: Δημιουργία «συρμάτων» από τόξα κύκλου που αποτελούν τις γενέτειρες των σηράγγων.	61
Εικόνα 3.11: Οι γενέτειρες των σηράγγων, όπως φαίνονται στο τριδιάστατο περιβάλλον σχεδίασης.	62
Εικόνα 3.12: Δημιουργία του πρώτου μέρους των οδηγών των σηράγγων (ευθύγραμμο τμήμα).	62
Εικόνα 3.13: Δημιουργία του δεύτερου μέρους του οδηγού των σηράγγων (καμπύλη B-Spline).	63
Εικόνα 3.14: Ένωση των τμημάτων, που αποτελούν τον οδηγό των σηράγγων ώστε να δημιουργηθεί ένα σύρμα.	63
Εικόνα 3.15: Οι γενέτειρες των σηράγγων και ο οδηγός τους στο τρισδιάστατο περιβάλλον σχεδίασης.	64
Εικόνα 3.16: Δημιουργία της πρώτης σήραγγας.	64
Εικόνα 3.17: Δημιουργία της δεύτερης σήραγγας.	65
Εικόνα 3.18: Ορισμός του οδηγού της βοηθητικής σήραγγας.	65
Εικόνα 3.19: Ορισμός της γενέτειρας της βοηθητικής σήραγγας.	66
Εικόνα 3.20: Ορισμός της βοηθητικής σήραγγας.	66
Εικόνα 3.21: Αποκοπή βοηθητικού σχήματος από τη μία από τις δυο σήραγγες.	67
Εικόνα 3.22: Αποκοπή της μιας σήραγγας από τη βοηθητική.	67
Εικόνα 3.23: Η τελική σύνθεση. Κάθε διαφορετική επιφάνεια χρωματίστηκε με άλλο χρώμα για να ξεχωρίζουν μεταξύ τους.	68
Σχήμα A.1: Η ταξινόμηση των βιβλιοθηκών της Open Cascade σε ομάδες [OpenCascade].	72
Σχήμα A.2: Η σχέση των κλάσεων που δημιουργούν το παράθυρο προβολής αντικειμένων.	74

Σύντομο Βιογραφικό Σημείωμα

Ο Σωτήριος Σ. Σαρακηνός γεννήθηκε στο Αμαρούσιο Αττικής το 1981, μεγάλωσε και ολοκλήρωσε την εκπαίδευσή του στην Κέρκυρα στο 2^ο Γενικό Λύκειο Κέρκυρας. Το 1999 εισήχθη στο Τμήμα Μηχανικών Παραγωγής και Διοίκησης του Πολυτεχνείου Κρήτης στα Χανιά. Το 2005 αποφοίτησε με βαθμό 7,25 και στη συνέχεια εισήχθη στο Μεταπτυχιακό Πρόγραμμα Ειδίκευσης του ίδιου τμήματος, στον τομέα των Συστημάτων Παραγωγής. Με την εργασία αυτή ολοκληρώνει τις μεταπτυχιακές σπουδές του.

Ευχαριστίες

Με το πέρας της παρούσας μεταπτυχιακής εργασίας και των μεταπτυχιακών σπουδών μου, θα ήθελα να ευχαριστήσω τον Δρ. Ι. Κ. Νικολό για την υπομονή του, την αμέριστη συμβολή του κατά την εκπόνηση αυτής της εργασίας, καθώς και για την ευκαιρία που μου έδωσε να ασχοληθώ με κάτι τόσο ενδιαφέρον και εποικοδομητικό. Επίσης θα ήθελα να ευχαριστήσω τα μέλη της τριμελούς επιτροπής Καθηγητή κ. Βασίλειο Κουικόγλου και Αν. Καθηγητή κ. Νικόλαο Μπιλάλη για την τιμή που μου έκαναν να συμμετέχουν στην εξέταση της παρούσας εργασίας.

Ευχαριστώ επίσης την εταιρία SOFiSTiK Hellas A.E., και τους Δρ. Βύρωνα Πρωτοψάλτη και Δρ. Θεοδώρα Πάππου για την εμπιστοσύνη που μου δείξαν και για την ιδιαίτερα εποικοδομητική συνεργασία που είχαμε κατά την εκπόνηση της εργασίας αυτής. Η εργασία εκπονήθηκε στα πλαίσια ερευνητικού έργου που χρηματοδοτήθηκε από την εταιρία SOFiSTiK Hellas A.E.

Τέλος, θα ήθελα να ευχαριστήσω την οικογένειά μου που με στήριξαν και με στηρίζουν όλα αυτά τα χρόνια.

Η εργασία αυτή αφιερώνεται στην οικογένειά μου.

“A designer knows he has achieved perfection not
when there is nothing left to add, but when there is
nothing left to take away.”

Antoine de Saint-Exupery

Πρόλογος

Στην παρούσα εργασία παρουσιάζεται η ανάπτυξη μεθοδολογίας και του αντίστοιχου λογισμικού για την τριδιάστατη γεωμετρική μοντελοποίηση υπόγειων σηράγγων. Το λογισμικό αναπτύχθηκε σε γλώσσα προγραμματισμού C++ κάνοντας χρήση των βιβλιοθηκών για τρισδιάστατη σχεδίαση της Open Cascade, οι οποίες παρέχουν τη δυνατότητα ορισμού και διαχείρισης μεγάλου αριθμού από γεωμετρικές οντότητες. Η σχεδίαση των διαφόρων καμπυλών και επιφανειών βασίζεται στη λογική των καμπυλών και επιφανειών NURBS, που επιτρέπουν τη δημιουργία πολύπλοκων γεωμετρικών παραμετρικών σχημάτων. Οι καμπύλες και οι επιφάνειες ορίζονται δυναμικά, μέσω του γραφικού περιβάλλοντος του προγράμματος, σε περιβάλλον διδιάστατης ή τριδιάστατης σχεδίασης. Οι οντότητες, που σχεδιάζονται στο λογισμικό, αποθηκεύονται σε ένα δέντρο σχεδίασης, επιτρέποντας έτσι τη μετέπειτα διαχείρισή τους. Στο λογισμικό έχει προστεθεί επίσης η δυνατότητα εισαγωγής λογικών πράξεων μεταξύ των παραγόμενων επιφανειών και στερεών, έτσι ώστε να μπορούν να ορισθούν τεμνόμενα μέρη σηράγγων, καθώς και οι φάσεις της κατασκευής τους. Στη διάθεση του χρήστη βρίσκονται επίσης οδηγοί (wizards) για την αυτόματη δημιουργία διατομών σηράγγων και διάφορων τύπων διασταυρώσεων σηράγγων. Στην παρούσα εργασία παρουσιάζονται τα χαρακτηριστικά και οι λειτουργίες του αναπτυχθέντος λογισμικού και της αντίστοιχης μεθοδολογίας. Επίσης, παρουσιάζονται παραδείγματα σηράγγων, οι οποίες έχουν σχεδιαστεί με τη χρήση του αναπτυχθέντος λογισμικού, μαζί με ένα παράδειγμα που παρουσιάζει αναλυτικά τη διαδικασία σχεδίασης που ακολουθείται.

Κεφάλαιο 1^ο: Εισαγωγή

1.1: Γενικά

Οι μεγάλες εξελίξεις στον τομέα της υπολογιστικής μηχανικής έχουν αναδείξει νέες μεθόδους στον τρόπο που μοντελοποιούνται οι υπόγειες σήραγγες. Τα υπάρχοντα υπολογιστικά εργαλεία παρέχουν μεθοδολογίες για την ανάλυση σηράγγων με περίπλοκες συνθήσεις, με τη χρήση μεγάλου αριθμού εξωτερικών μεταβλητών, όπως οι μεταβολές στο έδαφος και στις υπόγειες γεωλογικές συνθήκες. Πανίσχυρα υπολογιστικά εργαλεία, που μέχρι πρότινος μπορούσαν να έχουν μόνο οι μεγάλες κατασκευαστικές εταιρείες, είναι πλέον διαθέσιμα για όλους στον τομέα αυτό.

Το πρώτο βήμα για τη σχεδίαση και ανάλυση σηράγγων είναι ο ορισμός της γεωμετρίας τους και του περιβάλλοντος εδάφους. Αυτό το βήμα της μοντελοποίησης σηράγγων είναι μια πλήρως τριδιάστατη διαδικασία. Τα περισσότερα υπάρχοντα λογισμικά στον τομέα χειρίζονται αυτό το μέρος της σχεδίασης σηράγγων ως διδιάστατο. Όμως, η τριδιάστατη φύση της δομής των σηράγγων, τα διάφορα στοιχεία υποστήριξης και οι γεωλογικές συνθήκες του υπεδάφους καθιστούν το πρόβλημα καθαρά τριδιάστατο. Τα περισσότερα πακέτα σχεδίασης με χρήση υπολογιστή (CAD) παρέχουν τα μέσα για τριδιάστατη απεικόνιση των σηράγγων και οποιασδήποτε άλλης περίπλοκης γεωμετρίας, καθώς και ένα μεγάλο αριθμό από εργαλεία. Όμως, η χρήση τέτοιων γενικών προγραμμάτων απαιτεί μεγάλη επένδυση σε χρόνο και εμπειρία, ενώ το κόστος τους είναι συνήθως ιδιαίτερα υψηλό. Επί πλέον, τα πακέτα αυτά δεν περιέχουν εργαλεία συγκεκριμένα για τη σχεδίαση σηράγγων, οδηγώντας σε μεγάλη πολυπλοκότητα στη διαδικασία της σχεδίασης και σε αυξημένο χρόνο για τη διαδικασία της μοντελοποίησης, η οποία είναι απλά η πρώτη φάση στη σχεδίαση και ανάλυση σηράγγων.

Για αυτή τη φάση της σχεδίασης σηράγγων απαιτείται ένα λογισμικό τριδιάστατης σχεδίασης (μοντελοποίησης) που θα είναι επικεντρωμένο στη μοντελοποίηση σηράγγων, δηλαδή θα παρέχει εργαλεία και οδηγούς σχεδίασης σηράγγων, χωρίς την πολυπλοκότητα των γενικών πακέτων τριδιάστατης σχεδίασης με χρήση υπολογιστή (CAD). Ένα λογισμικό αυτής της κατηγορίας θα πρέπει να είναι εύκολο στην εκμάθηση και χρήστη για τον ανειδίκευτο χρήστη παρόμοιων λογισμικών, ενώ θα πρέπει να παρέχει τη δυνατότητα συνεργασίας με άλλα λογισμικά ανάλυσης σηράγγων, μέσα από διεθνή πρωτόκολλα μεταφοράς δεδομένων [Nikolos].

Στην παρούσα εργασία αναπτύχθηκε μία μεθοδολογία και το αντίστοιχο λογισμικό για την τριδιάστατη σχεδίαση και μοντελοποίηση σηράγγων. Το λογισμικό κάνει χρήση των βιβλιοθηκών σχεδίασης της Open Cascade για την αναπαράσταση των γεωμετρικών οντοτήτων και για την εκτέλεση των διάφορων λειτουργιών της σχεδίασης [OpenCascade]. Η σχεδίαση των σηράγγων στο προτεινόμενο λογισμικό είναι πλήρως τριδιάστατη και βασίζεται σε καμπύλες και επιφάνειες NURBS [Piegl]. Η χρήση καμπύλων και επιφανειών NURBS επιτρέπει τη σχεδίαση πολύπλοκων γεωμετριών με τη χρήση ενός μικρού σχετικά αριθμού μεταβλητών, καθώς και τον ενιαίο ορισμό της γεωμετρίας, κάνοντας έτσι πιο εύκολη την απευθείας εισαγωγή της σε άλλα λογισμικά παραγωγής τριδιάστατου πλέγματος και ανάλυσης. Τα εργαλεία που παρέχονται σε αυτό το πρόγραμμα είναι εύκολα στη χρήση και παρέχουν στο χρήστη πολλές δυνατότητες για τη μοντελοποίηση σηράγγων. Η μοντελοποίηση σηράγγων ολοκληρώνεται σε μερικά βασικά βήματα:

1. Ορισμός σημείων στο χώρο και βασικών επιπέδων σχεδίασης.
2. Κατασκευή καμπυλών ή τμημάτων καμπυλών, που αποτελούν τις διατομές των σηράγγων.

3. Κατασκευή επιφανειών και όγκων.
4. Εκτέλεση λογικών πράξεων μεταξύ των παραγόμενων επιφανειών και όγκων.

Η γεωμετρία στη συνέχεια μπορεί να εξαχθεί σε αρχείο με τη χρήση των διεθνών πρωτόκολλων μεταφοράς δεδομένων STEP και IGES για να χρησιμοποιηθεί σε κάποιο λογισμικό παραγωγής πλέγματος ή ανάλυσης.

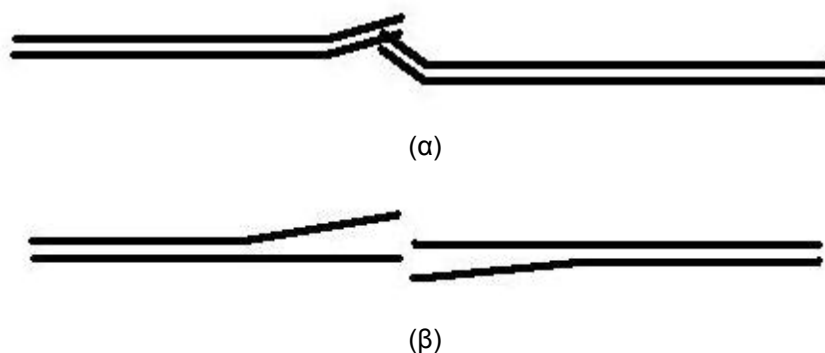
Η υπόλοιπη εργασία είναι οργανωμένη ως ακολούθως. Στο υπόλοιπο του πρώτου κεφαλαίου γίνεται μια ιστορική αναδρομή στις γνωστότερες σήραγγες όλων των εποχών, μια αναφορά στον τρόπο μοντελοποίησης και μελέτης σηράγγων, όπως γίνεται μέχρι σήμερα, καθώς και μια αναφορά στα λογισμικά που εξειδικεύονται στο συγκεκριμένο αντικείμενο. Στο δεύτερο κεφάλαιο παρουσιάζεται το λογισμικό που αναπτύχθηκε, με όλες τις δυνατότητές του, ενώ γίνεται εκτενής περιγραφή της δομής του. Στο τρίτο κεφάλαιο παρουσιάζονται παραδείγματα σηράγγων, που σχεδιάστηκαν με την προτεινόμενη μεθοδολογία, ενώ γίνεται αναλυτική περιγραφή της διαδικασίας σχεδίασης για ένα από αυτά. Τέλος, στο παράρτημα γίνεται μια περιγραφή της Open Cascade και των βιβλιοθηκών της, καθώς και μια περιγραφή των κλάσεων της, που χρησιμοποιήθηκαν στην ανάπτυξη του προτεινόμενου λογισμικού.

1.2: Η χρήση των σηράγγων ανά τον κόσμο.

Η χρονολογία που κατασκευάστηκαν οι πρώτες σήραγγες αμφισβητείται από πολλούς. Οι Αιγύπτιοι κατασκεύαζαν σήραγγες ως εισόδους των τάφων τους. Οι Βαβυλώνιοι κατασκεύαζαν το 2180 π.Χ. μια σήραγγα κάτω από τον ποταμό Ευφράτη, κάνοντας χρήση της μεθόδου εκσκαφής και επανεπίχωσης. Έκαναν εκτροπή των νερών του ποταμού, έσκαψαν ένα φρεάτιο στην κοίτη του και το έχτισαν με τούβλα πριν επαναφέρουν τα νερά. Το qanat ή kareez της Περσίας ήταν ένα σύστημα διαχείρισης των υδάτων, που χρησιμοποιούταν για την υδροδότηση καταυλισμών, ή για την άρδευση σε ζεστά και ξηρά κλίματα. Το αρχαιότερο και μεγαλύτερο γνωστό qanat βρίσκεται στην Ιρανική πόλη Γκοναμπάντ και παρέχει ακόμα πόσιμο νερό για περίπου 40.000 ανθρώπους, ύστερα από 2700 χρόνια [Infoplease].

Στον ελλαδικό χώρο κατασκευάζονταν σήραγγες για τη μεταφορά υδάτων. Πιο γνωστό είναι το Ευπαλίνειο όρυγμα στη Σάμο, που κατασκευάστηκε το 520 π.Χ. από τον Ευπαλίνο, γιο του Ναυστρόφου από τα Μέγαρα. Η σήραγγα έχει μήκος 1036 μέτρα και έχει άνοιγμα περίπου 1.80 επί 1.80 μέτρα. Το ιδιαίτερο χαρακτηριστικό της ήταν ότι ανοίχθηκε ταυτόχρονα και από τις δυο πλευρές του βουνού: αμφίστομον - όπως το χαρακτήρισε ο Ηρόδοτος (Θάλεια, 60) χάρις στον οποίον έγινε γνωστό. Οι δυο σήραγγες συναντήθηκαν περίπου στο μέσον, με αξιοθαύμαστη ακρίβεια (Σχήμα 1.1).

Ο Ευπαλίνος γνώριζε ότι το παραμικρό λάθος στον υπολογισμό της σήραγγας θα απέτρεπε τη συνάντηση των δυο ομάδων εκσκαφής. Γι' αυτό το λόγο χρησιμοποίησε συγκεκριμένες τεχνικές αποφυγής σφαλμάτων. Καθώς δυο παράλληλες γραμμές δεν τέμνονται, υπολόγισε ότι ένα λάθος μόνο δύο μέτρων θα τον έκανε να χάσει το σημείο συνάντησης. Έτσι, έχοντας υπολογίσει την αναμενόμενη θέση του σημείου συνάντησης, άλλαξε την κατεύθυνση εκσκαφής των δύο τμημάτων, όπως φαίνεται στο Σχήμα 1.1, ώστε να μπορεί να εγγυηθεί τη συνάντηση των δύο τμημάτων, ακόμα και αν αυτά ξεκίνησαν παράλληλα μεταξύ τους. Παρομοίως, καθώς υπήρχε πιθανότητα απόκλισης στο κάθετο επίπεδο, αύξησε την πιθανότητα συνάντησης των δύο τμημάτων, αυξάνοντας το ύψος και των δυο τμημάτων, όπως φαίνεται στο Σχήμα 1.1. Οι ρυθμίσεις αυτές, βέβαια, κατέληξαν άσκοπες, καθώς οι υπολογισμοί του Ευπαλίνου ήταν μόνο κατά τέσσερα εκατοστά λανθασμένοι [Kienast].



Σχήμα 1.1: Σχηματική αναπαράσταση του Ευπαλίνειου ορύγματος στο οριζόντιο (α) και κατακόρυφο (β) επίπεδο [Wikipedia2].

Άλλη σήραγγα άξια αναφοράς είναι η σήραγγα του καναλιού Σάπερτον στην Αγγλία, η οποία άνοιξε το 1789, είχε μήκος 3.5 χιλιόμετρα και επέτρεπε τη μεταφορά κάρβουνου και άλλων αγαθών με βάρκα. Η σήραγγα, που κατασκευάστηκε για την πρώτη ατμομηχανή, την ατμομηχανή Πενιντάρεν, κατασκευάστηκε από τον Ρίτσαρντ Τρέβιθικ και αποτελεί την πρώτη σήραγγα που κατασκευάστηκε για τη διέλευση αυτοκινούμενων ατμομηχανών σε ράγες. Η σήραγγα Μποξ στην Αγγλία, η οποία άνοιξε το 1841, ήταν η μακρύτερη σιδηροδρομική σήραγγα στην εποχή της, με μήκος 2.9 χιλιόμετρα. Η σήραγγα του Τάμεση (Εικόνα 1.2), κατασκευασμένη από τον Μαρκ Ισαμπάρντ Μπρουνέλ και το γιο του, άνοιξε το 1843 και ήταν η πρώτη σήραγγα κάτω από νερό. Αρχικά χρησιμοποιήθηκε ως υπόγεια διάβαση πεζών και πλέον αποτελεί τμήμα του μετρό του Λονδίνου. Οι σήραγγες Κομπλ Χιλ και Μάρεϋ Χιλ αποτελούν τις πρώτες σήραγγες που κατασκευάστηκαν κάτω από δρόμους, στη δεκαετία του 1850 [Wikipedia1].



Εικόνα 1.2: Εσωτερική άποψη της σήραγγας του Τάμεση. Γκραβούρα του 19^{ου} αιώνα [Thames].

Σήραγγες άξιες αναφοράς αποτελούν η σήραγγα Σεϊκάν στην Ιαπωνία, η οποία είναι η μακρύτερη σιδηροδρομική σήραγγα στον κόσμο με μήκος 53.9 χιλιόμετρα, από τα οποία τα 23.3 χιλιόμετρα βρίσκονται κάτω από τη θάλασσα, καθώς και η σήραγγα Τσάνελ στο στενό της Μάγχης μεταξύ Αγγλίας και Γαλλίας, η οποία είναι η δεύτερη μακρύτερη στον κόσμο, με συνολικό μήκος 50 χιλιόμετρα, 39 από τα οποία βρίσκονται κάτω από τον πυθμένα της θάλασσας. Η σήραγγα Lötschberg στην Ελβετία είναι η μακρύτερη ηπειρωτική σήραγγα, με συνολικό μήκος 34.5 χιλιόμετρα. Η σήραγγα Lærdal στην Νορβηγία είναι η μακρύτερη οδική σήραγγα στον κόσμο για διέλευση αυτοκινήτων, με μήκος 24.5 χιλιόμετρα [Wikipedia1].



Εικόνα 1.3: Η γραμμή North East MRT στη Σιγκαπούρη είναι πλήρως υπόγεια [NorthEast].



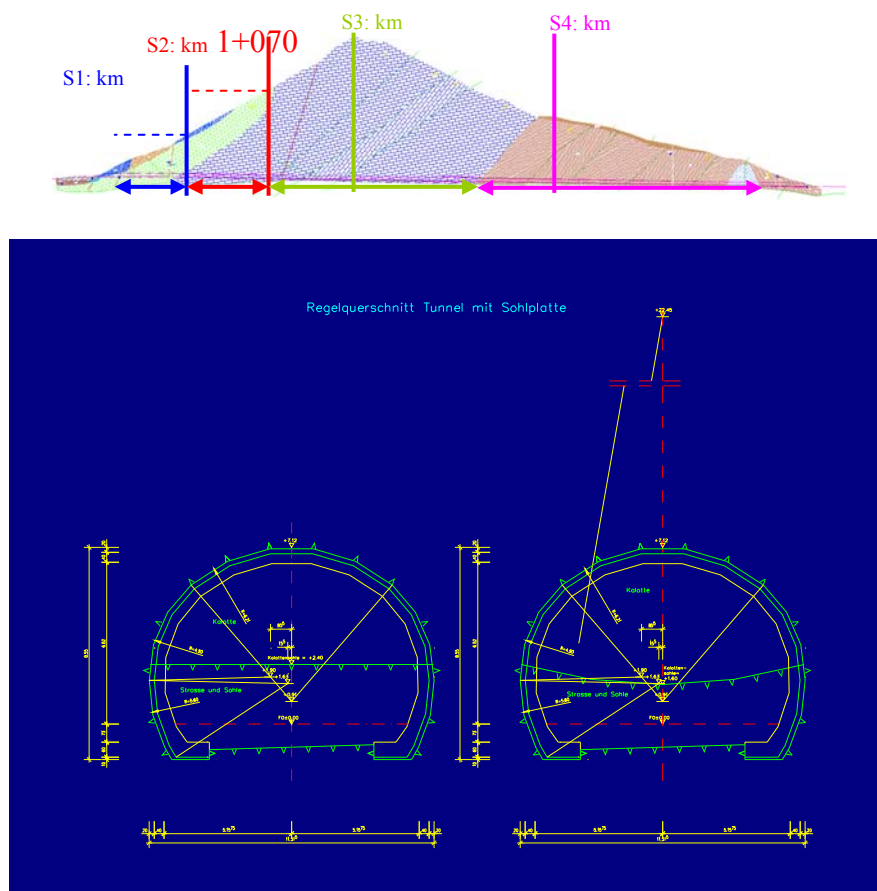
Εικόνα 1.4.: Σήραγγα που ενώνει δύο κτήρια στο αεροδρόμιο του Ντιτρόιτ [Hopson].

1.2: Η σημερινή διαδικασία μοντελοποίησης και μελέτης μιας σήραγγας.

Η διαδικασία που ακολουθείται ως τώρα για τη μοντελοποίηση και ανάλυση σήραγγων χωρίζεται σε έξι βήματα [Γιάππου]. Τα βήματα αυτά είναι:

1.2.1: Βήμα 1^ο

Αρχικά γίνεται η πρώτη σχεδίαση της σήραγγας με ακριβείς διαστάσεις σε ένα σύστημα CAD. Τα αρχικά σχέδια που παράγονται είναι διδιάστατα (τομές, όψεις, μηκοτομές, αξονοσυμμετρικά σχέδια, κ.λπ.).



Εικόνα 1.5: Σχέδια σε σύστημα CAD της διατομής της σήραγγας σε συγκεκριμένες τομές της [Γιάππου].

1.2.2: Βήμα 2^ο

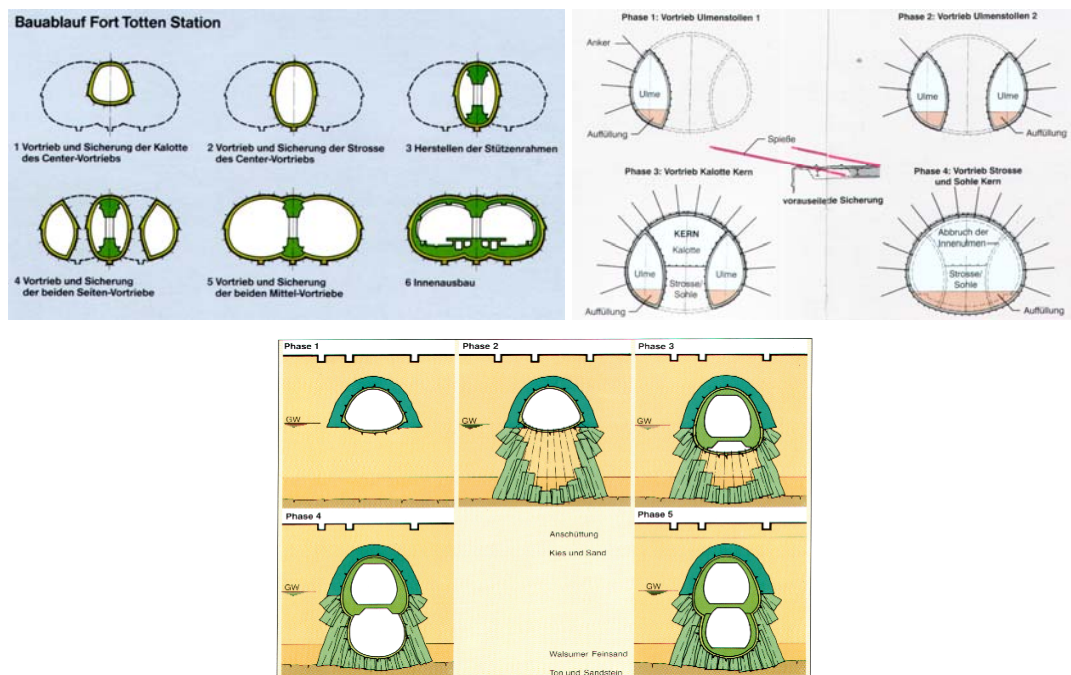
Αφού ορισθεί η γεωμετρία, εισάγεται σε κάποιο πρόγραμμα ανάλυσης σε δύο ή τρεις διαστάσεις. Τα γεωμετρικά στοιχεία από την προηγούμενη σχεδίαση εμπλουτίζονται με τον καθορισμό ζωνών εδάφους, επιφάνειας υπογείου ύδατος, επενδύσεων, εσωτερικών και εξωτερικών ορίων της γεωμετρίας, που αφορούν στις φάσεις διάνοιξης, τα οποία μπορεί να μεταβληθούν κατά την πορεία επίλυσης, επιπρόσθετων στοιχείων υποστήριξης ανά φάση, για τα οποία θα ληφθεί τελική

Κεφάλαιο 1^ο: Εισαγωγή

απόφαση μετά τον υπολογισμό, όπως αγκύρια και δοκοί προτορείας και φορτίων (γραμμικών, επιφανειακών κ.λπ.).



(α)



(β)

Εικόνα 1.6: (α) Εικόνες από τη διάνοιξη σήραγγας και φάσεων κατασκευής της. (β) Διάφοροι τύποι φάσεων κατασκευής, ανάλογα με την τοπολογία της σήραγγας [Πάππου].

Στην εικόνα 1.6 παρουσιάζονται ενδεικτικές εικόνες από διάνοιξη σήραγγας, με τις φάσεις κατασκευής της και τα τοποθετημένα μέτρα υποστήριξης (Εικόνα 1.6α), όπως και οι απαιτήσεις που τίθενται για τον καθορισμό των φάσεων κατασκευής, ανάλογα με τη μορφολογία του εδάφους, στη διαδρομή της σήραγγας (Εικόνα 1.6β). Με αφετηρία τη σχετικά απλή σήραγγα της Εικόνας 1.6, φαίνεται καθαρά η πολυπλοκότητα και ο καθαρά τριδιάστατος χαρακτήρας του μοντέλου, εάν αυτός ο απλός κλάδος είναι ένα τμήμα σήραγγας με αλληλοτεμνόμενους κλάδους.

1.2.3: Βήμα 3^ο

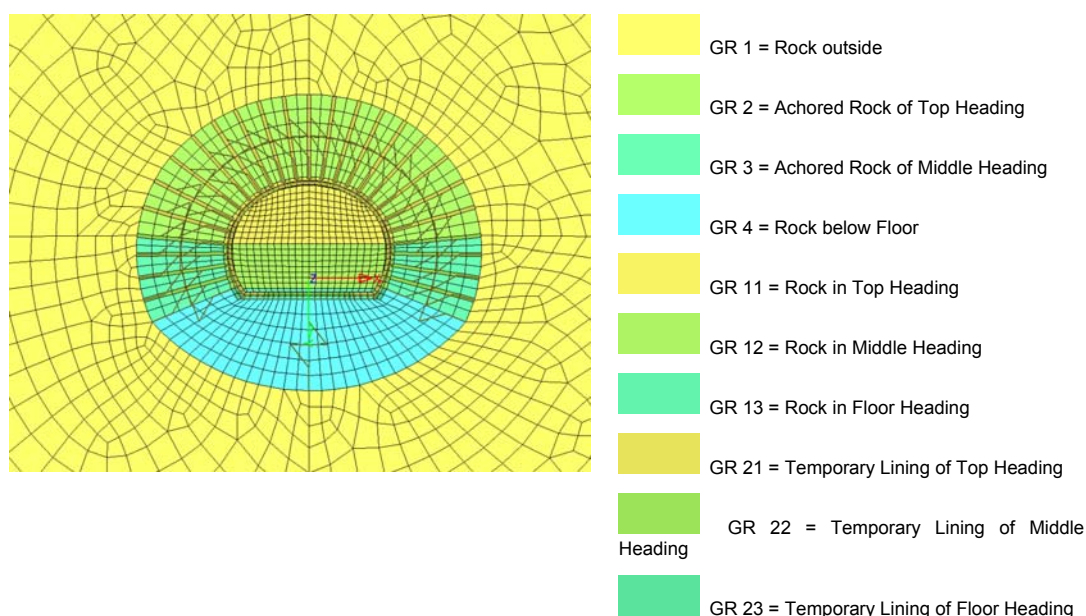
Το επόμενο βήμα της μοντελοποίησης και ανάλυσης σπηραγγων είναι η προσομοίωση – διακριτοποίηση με πλέγμα πεπερασμένων στοιχείων.

Για διδιάστατες αναλύσεις χρησιμοποιούνται:

- Επίπεδα ισοπαραμετρικά, ελαστοπλαστικά στοιχεία για την βραχομάζα,
- διδιάστατα ραβδωτά στοιχεία για τις επενδύσεις,
- διδιάστατα στοιχεία με ιδιότητες μη γραμμικού ελατηρίου.

Για την τριδιάστατη ανάλυση (στις λίγες απλές περιπτώσεις που επιχειρείται αυτή) χρησιμοποιούνται:

- Χωρικά ισοπαραμετρικά, ελαστοπλαστικά στοιχεία για την βραχομάζα,
- ισοπαραμετρικά στοιχεία κελύφους για τις επενδύσεις,
- τριδιάστατα στοιχεία με ιδιότητες μη γραμμικού ελατηρίου.



Εικόνα 1.7: Διακριτοποιημένη διατομή σήραγγας, λαμβάνοντας υπόψη τις ζώνες εδάφους και τα μέτρα υποστήριξης [Πάππου].

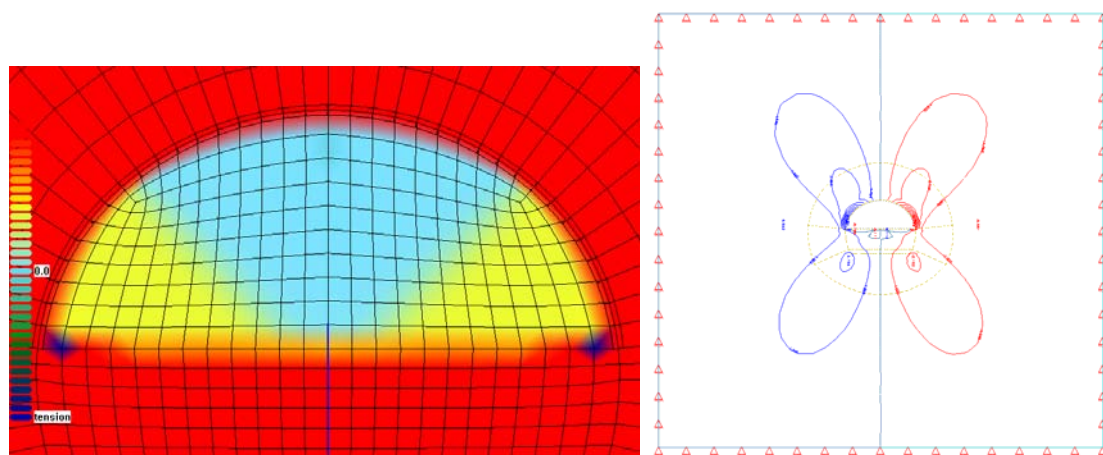
1.2.4: Βήμα 4^ο

Το επόμενο βήμα είναι η ανάλυση αντοχής, η οποία εκτελείται σε τρία στάδια:

- Τελικός καθορισμός και τρόπος ενεργοποίησης των φάσεων.
- Προσδιορισμός ιδιοτήτων των υλικών, ο οποίος αφορά στην προσομοίωση βραχομάζας με χρήση κατάλληλων κριτηρίων ελαστοπλαστικής αστοχίας, προσομοίωση ασυνεχειών αλλά και επενδύσεων (οπλισμένο σκυρόδεμα - μεταλλική επένδυση / πλαίσια υποστήριξης) και επιπρόσθετων στοιχείων υποστήριξης.
- Μη-γραμμική ελαστοπλαστική πολυσταδιακή ανάλυση για όλες τις φάσεις. Καταγραφή ιστορίας επιλύσεων για όλα τα αποτελέσματα (με αποθήκευση στη βάση δεδομένων).

1.2.5: Βήμα 5^ο

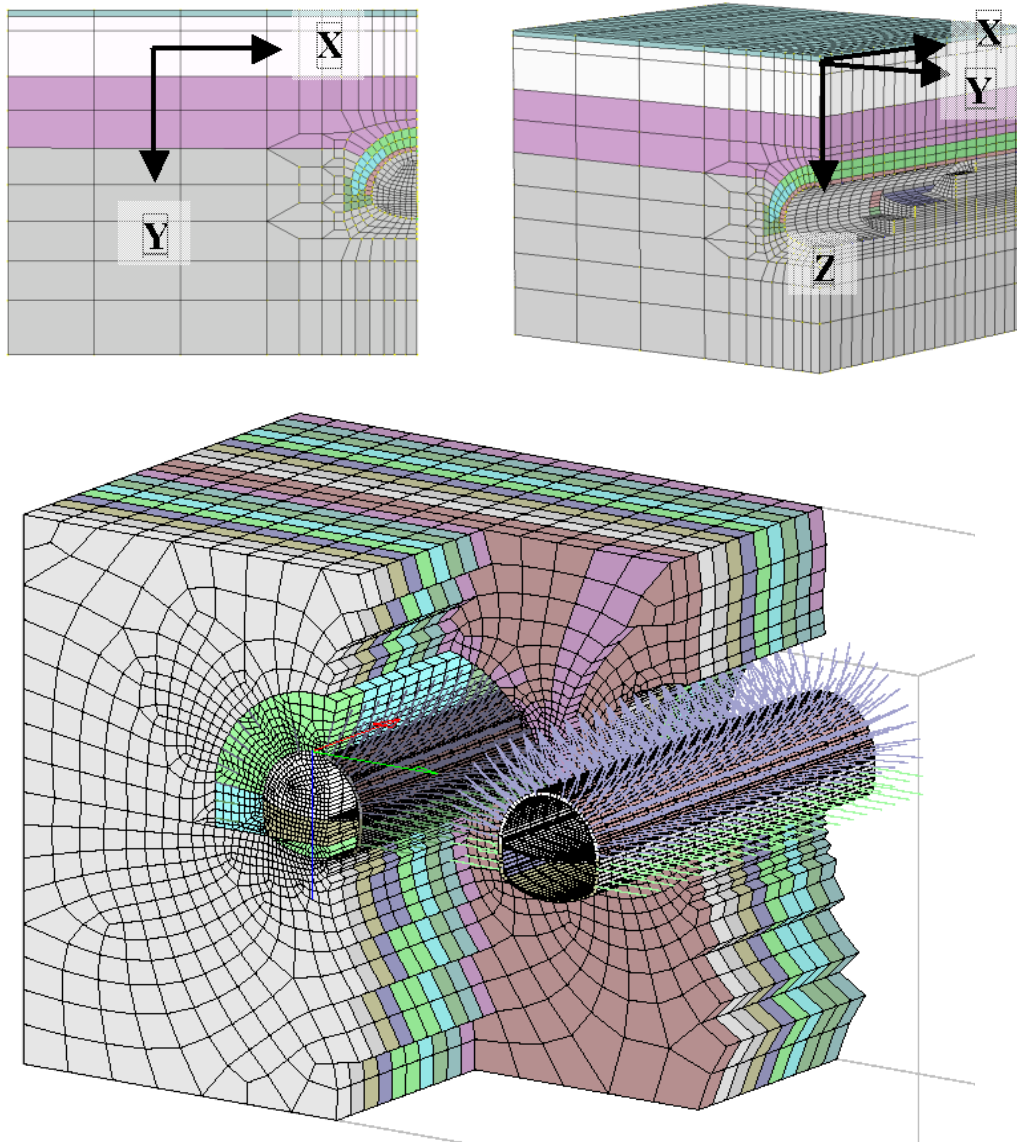
Στη συνέχεια πραγματοποιείται ο έλεγχος αποτελεσμάτων, δηλαδή μια γραφική απεικόνιση και αξιολόγηση, που αφορά σε απεικονίσεις κατανομών τάσεων, παραμορφώσεων και πλαστικοποιημένων περιοχών, ανά φάση και φόρτιση.



Σχήμα 1.8: Απεικόνιση και αξιολόγηση των αποτελεσμάτων ανάλυσης στη διατομή της σήραγγας [Πάππου].

1.2.6: Βήμα 6^ο

Τέλος γίνεται ο υπολογισμός περιβαλλουσών εντατικών μεγεθών και τελική διαστασιολόγηση προσωρινής και μόνιμης επένδυσης, αγκυρίων και των άλλων μέτρων υποστήριξης.



Εικόνα 1.9: Τριδιάστατα πλέγματα για την ανάλυση σηράγγων μονού και διπλού κλάδου, με φάσεις κατασκευής. Προκύπτουν από διδιάστατα πλέγματα με ομοιόθετη αναπαραγωγή στην τρίτη διάσταση. Μια τέτοια περίπτωση αποτελεί ουσιαστικά το «όριο» στην πραγματοποίηση τριδιάστατης μελέτης, με τα σημερινά υπολογιστικά εργαλεία [Sofistik].

1.3: Υπάρχοντα προγράμματα μοντελοποίησης σηράγγων

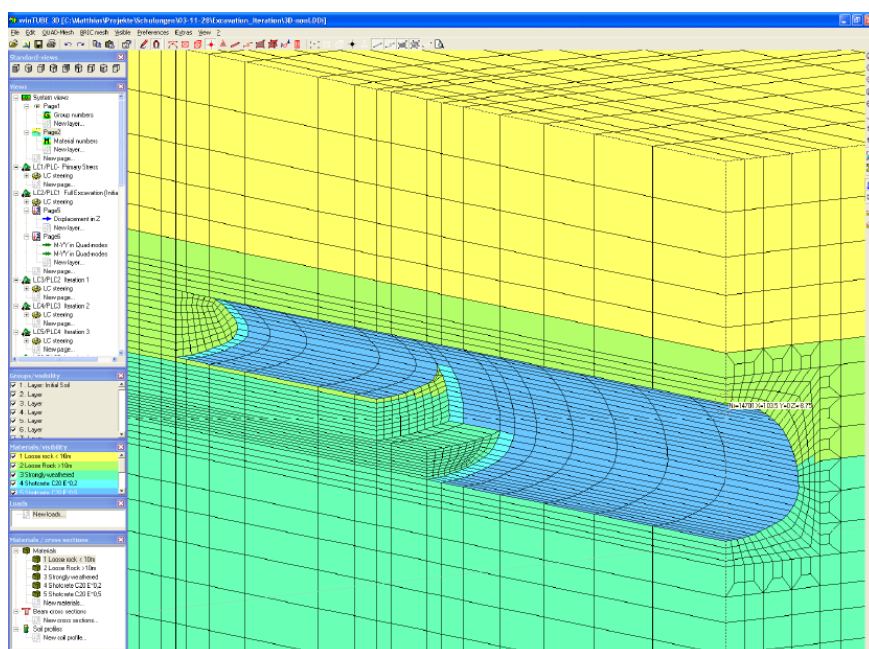
Τα υπάρχοντα λογισμικά, που είναι επικεντρωμένα στη μοντελοποίηση και ανάλυση σηράγγων, είναι περιορισμένα στον αριθμό λόγω της ιδιαιτερότητας του αντικειμένου και του περιορισμένου αγοραστικού κοινού. Τα πιο άξια αναφοράς περιγράφονται στη συνέχεια.

1.3.1: FIDES-WinTUBE

Το πρόγραμμα FIDES-WinTUBE [FIDES] της FIDES αναπτύχθηκε αρχικά ως ένα εξειδικευμένο εργαλείο για τη σχεδίαση σηράγγων. Περιέχει ένα μεγάλο αριθμό οδηγιών, που εκτελούν εύκολα διάφορες διαδικασίες, οι οποίες αφορούν στη

μοντελοποίηση σιράγγων. Οι δυνατότητες του προγράμματος περιγράφονται στη συνέχεια:

- Λειτουργίες CAD, πλήρης έλεγχος και διαχείριση της όψης των αντικειμένων προς μοντελοποίηση.
- Δυνατότητα εισαγωγής σχεδίων από αρχεία μορφής DXF του AutoCAD.
- Μεγάλος αριθμός αυτόματων οδηγών για σχεδίαση διατομών σιράγγων, διασταυρώσεων σιράγγων, σταδίων κατασκευής της σήραγγας, επιλογής φορτίων.
- Αυτόματη δημιουργία δομημένου και μη δομημένου πλέγματος, καθώς και σύνθεσης αυτών για τις παραγόμενες γεωμετρίες.
- Δημιουργία φάσεων εκσκαφής ή φάσεων κατασκευής.
- Δημιουργία μη γραμμικών διδιάστατων και τριδιάστατων οντοτήτων διεπαφής.
- Δημιουργία τριδιάστατων μοντέλων με εξώθηση απλών συστημάτων, δημιουργία μη δομημένου τριδιάστατου τετραεδρικού πλέγματος.
- Προσομοίωση αρμών έντασης.
- Εξαγωγή δεδομένων σε μορφή αρχείου κειμένου, του οποίου μπορεί να γίνει επεξεργασία εκτός λογισμικού.
- Δύο μέθοδοι για την προσομοίωση των σταδίων κατασκευής της σήραγγας:
 - Μέθοδος βήμα προς βήμα, όπου δημιουργείται πλέγμα κατά ομάδες για τα διάφορα στάδια κατασκευής.
 - Επαναληπτική μέθοδος, όπου χρησιμοποιείται δεδομένο πλέγμα, ενώ μεταβάλλονται οι τάσεις κατά τις επαναλήψεις.



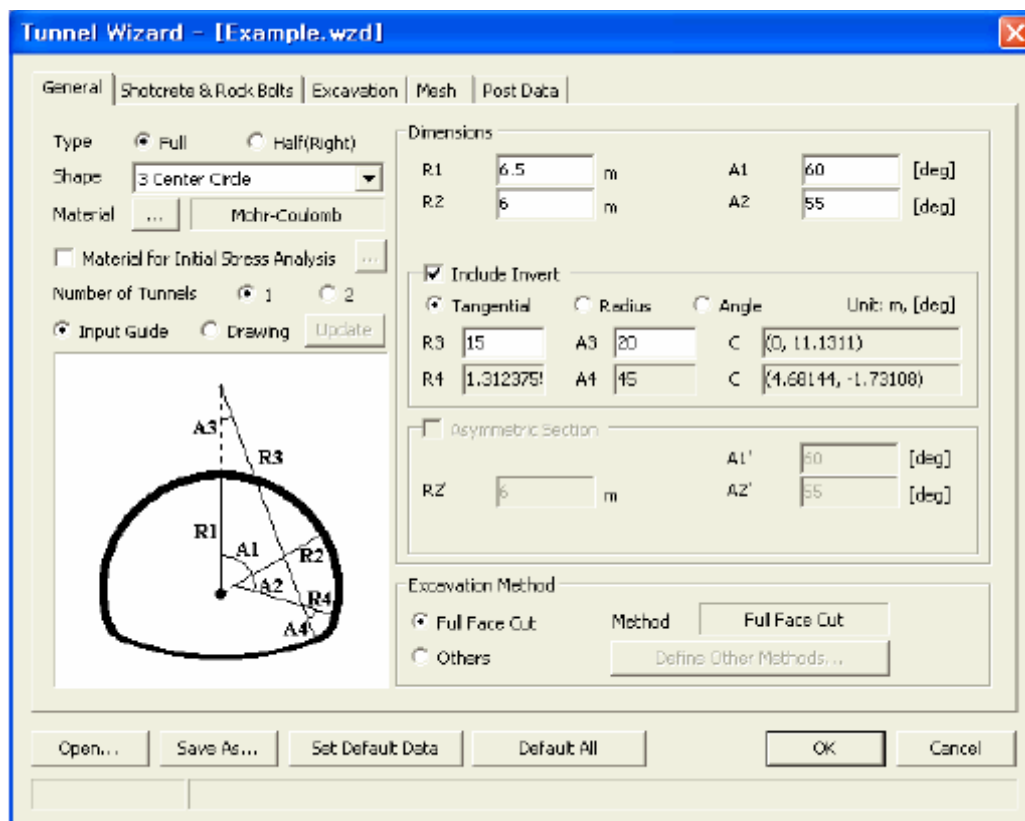
Εικόνα 1.10: Το περιβάλλον σχεδίασης του προγράμματος FIDES-WinTUBE [FIDES].

1.3.2: Midas/GTS

Το Midas/GTS [Midas/GTS] είναι ένα λογισμικό το οποίο παρέχει πολλές δυνατότητες για τη δημιουργία περίπλοκων γεωτεχνικών και δομικών μοντέλων. Το πακέτο παρέχει ένα μεγάλο αριθμό εργαλείων για τη μοντελοποίηση σιράγγων. Η σχεδίαση πραγματοποιείται με χρήση καμπύλων επιφανειών και όγκων. Οι καμπύλες προκύπτουν από κωνικές τομές ή γενικότερες splines, ενώ οι διατομές των σιράγγων μπορούν να δημιουργηθούν από ειδικούς οδηγούς και να τροποποιηθούν στη συνέχεια με φιλέτα, σπασίματα ή άλλες παρόμοιες διεργασίες. Οι επιφάνειες προκύπτουν από μπαλώματα επιφανειών τύπου Coons, NURBS, μπαλώματα πλέγματος, ή διανυσμάτων. Οι όγκοι είναι κυρίως βασικοί όγκοι, όπως κύλινδροι, κώνοι, παραλληλεπίπεδα, σφαίρες, οι οποίοι μπορούν να επεξεργαστούν περεταίρω με διεργασίες όπως διατμήσεις και διαιρέσεις. Πιο περίπλοκες γεωμετρίες μπορούν να παραχθούν με χρήση ειδικών μεθόδων όπως η εξώθηση, η περιστροφή και η οδήγηση. Όλες οι επιφάνειες και οι όγκοι μπορούν να χρησιμοποιηθούν σε λογικές πράξεις Boole για την παραγωγή ακόμα πιο περίπλοκων σχημάτων.

Το πρόγραμμα παρέχει επίσης έναν ειδικό οδηγό μοντελοποίησης σιράγγων [Shin] (Εικόνα 1.11), με τον οποίο μπορούν να παραχθούν γεωμετρίες σιράγγων σε πέντε βασικά βήματα:

1. Ορισμός της διατομής της σήραγγας και του τρόπου εκσκαφής.
Σε αυτό το βήμα ορίζεται η διατομή της σήραγγας από έναν αριθμό προκαθορισμένων μορφών διατομών, όπως η κυκλική, η κυκλική με τρία κέντρα και η κυκλική με πέντε κέντρα, ενώ έχει τη δυνατότητα ορισμού και μη συμμετρικής διατομής σήραγγας. Στο ίδιο βήμα ορίζεται το υλικό του εδάφους, από τέσσερα βασικά μοντέλα προσομοίωσης εδάφους.
2. Ορισμός τοιχώματος και αγκυρίων (rock bolts).
Στο δεύτερο βήμα ορίζεται ο τύπος της υποστήριξης του τοιχώματος της σήραγγας, ορίζοντας τον τύπο της και όλες τις ιδιότητές της. Επίσης δηλώνεται το υλικό των αγκυρίων και οι ιδιότητές τους.
3. Ορισμός σταδίων κατασκευής/ εκσκαφής.
Στο τρίτο βήμα ορίζεται η διαδικασία της εκσκαφής και κατασκευής της σήραγγας.
4. Ορισμός δεδομένων πλέγματος.
Στο τέταρτο βήμα καθορίζεται η τοποθεσία της σήραγγας και τα όρια της επιφάνειας του υπεδάφους.
5. Προετοιμασία αναφορών με δεδομένα εξόδου.
Αφού ορισθούν όλα τα απαραίτητα δεδομένα για τη μοντελοποίηση της σήραγγας, ο οδηγός παρέχει αναφορά με όλα τα δεδομένα εξόδου, τα οποία μπορούν να υποστούν μετεπεξεργασία.



Εικόνα 1.11: Το γραφικό περιβάλλον του ειδικού οδηγού μοντελοποίησης σηράγγων του MIDAS/GTS [Shin].

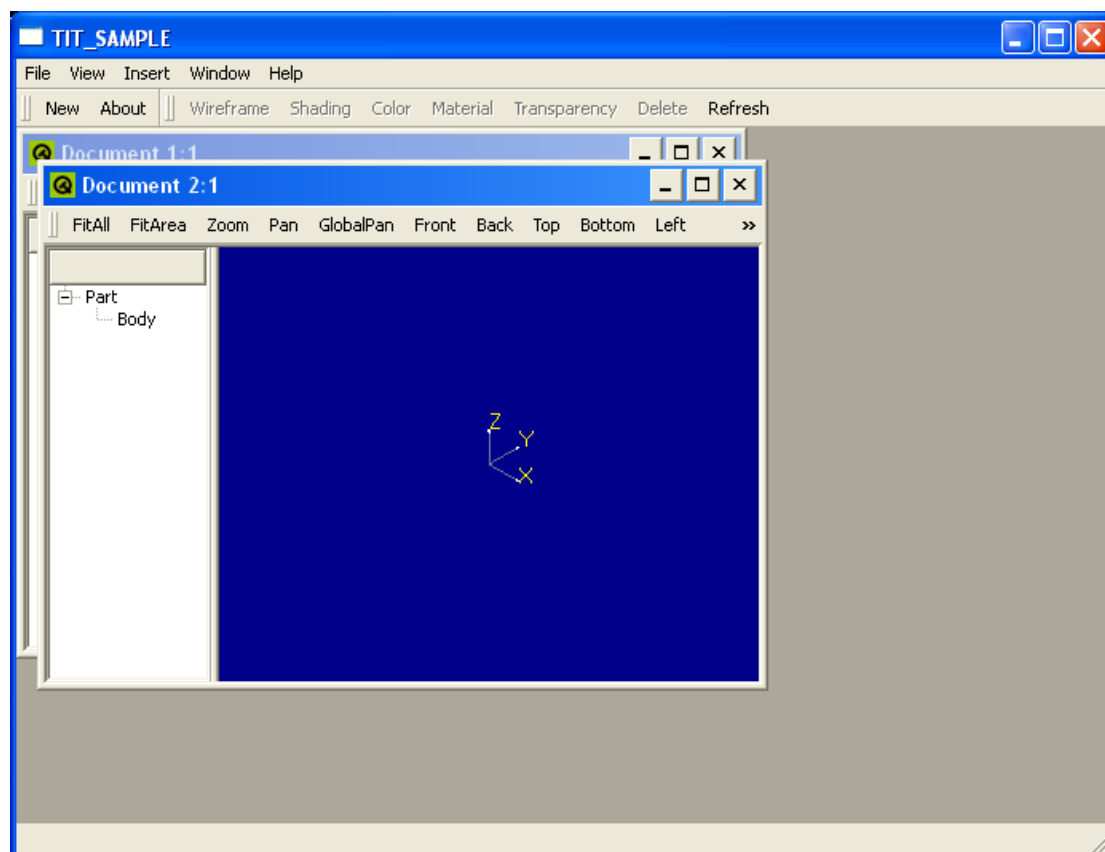
Κεφάλαιο 2^ο: Δυνατότητες και δομή του λογισμικού που αναπτύχθηκε

2.1: Γενικά

Το λογισμικό έχει αναπτυχθεί σε γλώσσα προγραμματισμού C++ κάνοντας χρήση του γραφικού περιβάλλοντος της γλώσσας Qt και των βιβλιοθηκών CAD της Open CasCade. Η C++ είναι μια αμιγώς αντικειμενοστραφής γλώσσα προγραμματισμού, συνεπώς το λογισμικό βασίστηκε σε αυτή τη φιλοσοφία. Στη συνέχεια θα γίνει η περιγραφή των κλάσεων του λογισμικού, ενώ ιδιαίτερη έμφαση θα δοθεί στο δέντρο των κλάσεων, που περιγράφουν τα γεωμετρικά στοιχεία του λογισμικού.

2.2: Το περιβάλλον τριδιάστατης απεικόνισης.

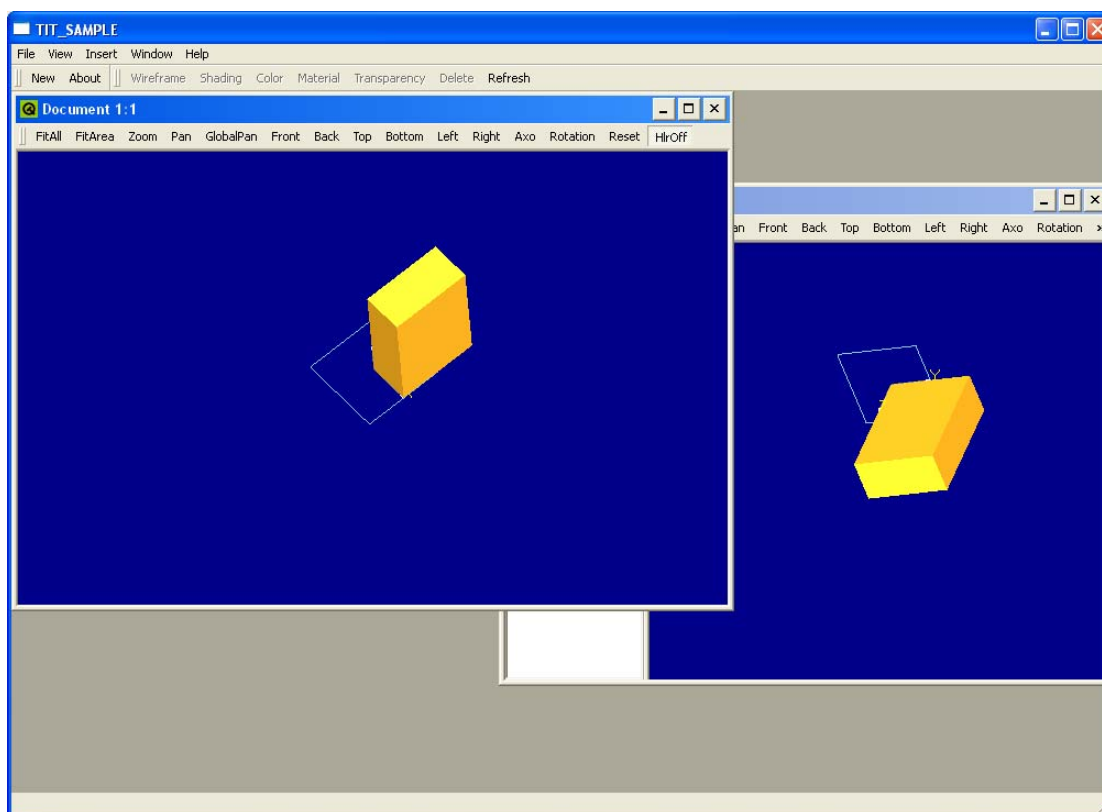
Το λογισμικό έχει τη μορφή Multi – Document Interface (MDI), δηλαδή είναι στην ουσία ένα βασικό λογισμικό που μπορεί να υποστηρίξει ταυτόχρονα πολλά σχέδια – έγγραφα (Documents), παρέχοντας τις ίδιες λειτουργίες για όλα, αλλά απολύτως ανεξάρτητα μεταξύ τους (Εικόνα 2.1).



Εικόνα 2.1: Το βασικό παράθυρο του λογισμικού και το σύστημα πολλαπλών εγγράφων.

Κάθε έγγραφο αποτελεί αντικείμενο της κλάσης Document του λογισμικού. Κάθε αντικείμενο της κλάσης Document περιέχει αναφορές σε όλα τα σχεδιαστικά αντικείμενα, που εισάγονται στο συγκεκριμένο έγγραφο, την αναφορά στο δέντρο των αντικειμένων για το συγκεκριμένο δέντρο, ενώ μπορεί να περιέχει ένα αριθμό προβολών (Views), δηλαδή παράθυρα που παρουσιάζουν το σχέδιο που έχει

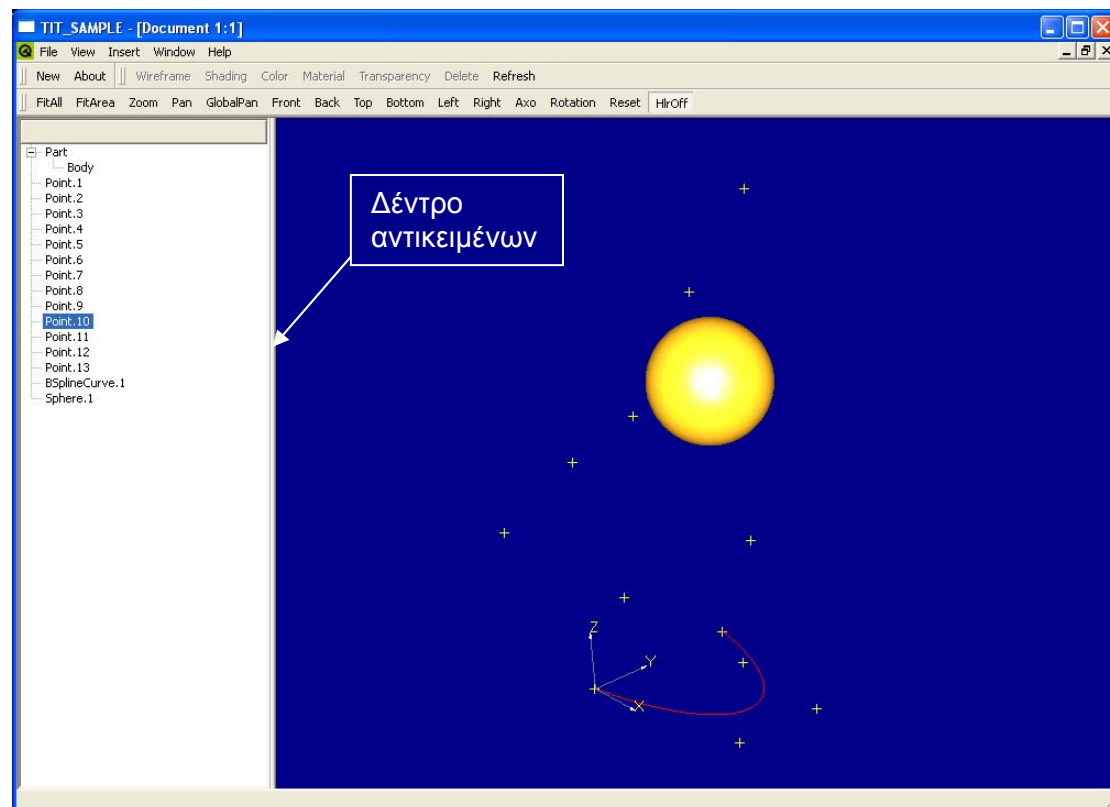
δημιουργηθεί στο έγγραφο (Εικόνα 2.2). Κάθε προβολή μπορεί να εμφανίζει το σχέδιο σε διαφορετική όψη ή μεγέθυνση, αλλά κάθε αλλαγή που εκτελείται στο σχέδιο εφαρμόζεται σε όλες τις προβολές του συγκεκριμένου εγγράφου. Κάθε έγγραφο πρέπει να έχει τουλάχιστον μια προβολή για την εμφάνιση του σχεδίου.



Εικόνα 2.2: Παρουσίαση της δυνατότητας δημιουργίας διαφορετικών προβολών του ίδιου εγγράφου.

Η σχεδίαση σηράγγων υποστηρίζεται από ένα πλήρως τριδιάστατο γραφικό περιβάλλον για την εμφάνιση και διαχείριση των γεωμετρικών οντοτήτων. Λειτουργίες όπως μετακίνηση, περιστροφή και μεγέθυνση, καθώς και δυνατότητα εμφάνισης προκαθορισμένων όψεων του μοντέλου (Πρόοψη, Κάτοψη, Αριστερή Όψη, Δεξιά Όψη, κ.λπ.) συμβάλλουν στην ευχρηστία του γραφικού περιβάλλοντος. Επίσης, παρέχεται δυνατότητα προβολής των αντικειμένων σε μοντέλο σύρματος, ή μοντέλο με σκίαση, με ή χωρίς διαφάνεια. Τέλος, υπάρχει δυνατότητα χρωματισμού των γεωμετρικών οντοτήτων για να μπορεί να γίνεται πιο εύκολα ο διαχωρισμός τους. Οι λειτουργίες αυτές μπορούν να γίνουν ξεχωριστά σε κάθε προβολή ενός εγγράφου. Όμως οι λειτουργίες που αλλάζουν τις οπτικές ιδιότητες των οντοτήτων, όπως η εμφάνιση σε μοντέλο σύρματος ή ο χρωματισμός, εφαρμόζονται στις οντότητες και συνεπώς γίνονται σε όλες τις προβολές ενός εγγράφου.

Η δημιουργία σχεδιαστικών αντικειμένων, όπως σημεία, καμπύλες ή επιφάνειες στο σχέδιο, πραγματοποιείται από το βασικό παράθυρο, ενώ το νέο αντικείμενο εμφανίζεται σε όλες τις όψεις του εγγράφου. Κάθε νέο αντικείμενο αποθηκεύεται σε ευρετικούς πίνακες (Hash tables), που περιέχονται στο αντικείμενο της συγκεκριμένης κλάσης Document, ενώ το όνομα του αντικειμένου εμφανίζεται στο δέντρο των αντικειμένων, που εμφανίζεται στα αριστερά της τελευταίας όψης του εγγράφου, για να μπορεί να χρησιμοποιηθεί περαιτέρω από το χρήστη (Εικόνα 2.3).



Εικόνα 2.3: Το βασικό παράθυρο του προγράμματος και το δέντρο των αντικειμένων.

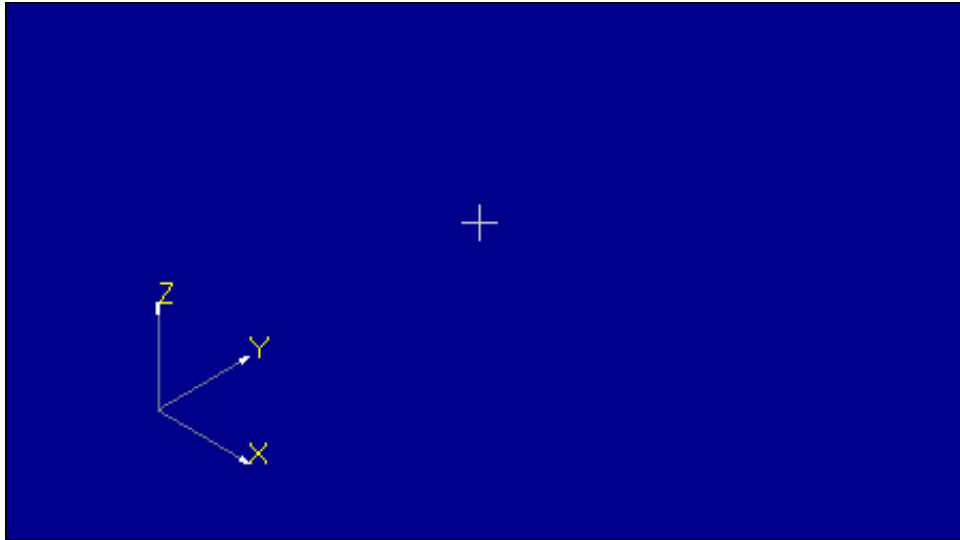
Η αποθήκευση των αντικειμένων σε ευρετικούς πίνακες γίνεται για την πιο εύκολη ανάκτηση των συγκεκριμένων αντικειμένων, σε περίπτωση που ο χρήστης χρειαστεί να τα επεξεργαστεί περαιτέρω ή να τα συμπεριλάβει στη δημιουργία κάποιου άλλου αντικειμένου. Ο τρόπος δημιουργίας όλων των σχεδιαστικών αντικειμένων περιγράφεται σε επόμενη παράγραφο.

2.3: Γεωμετρικές οντότητες

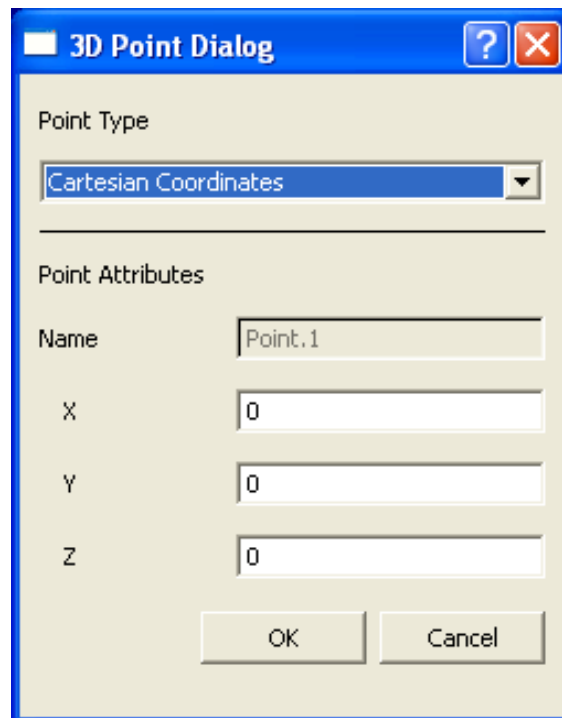
Στη συνέχεια θα γίνει περιγραφή των δυνατοτήτων σχεδίασης του λογισμικού. Μέσα από την περιγραφή της σχεδίασης των διάφορων γεωμετρικών οντοτήτων, που δύνανται να δημιουργηθούν με το λογισμικό, θα γίνει εμφανής και η διαδικασία σχεδίασης σηράγγων που ακολουθείται.

2.3.1: Σημεία και επίπεδα

Το πρώτο βήμα της σχεδίασης σηράγγων στην παρούσα μεθοδολογία είναι ο ορισμός σημείων στο χώρο (Εικόνα 2.4), τα οποία θα αποτελέσουν τη βάση για την κατασκευή άλλων γεωμετρικών οντοτήτων. Ο βασικός ορισμός σημείων στον τριδιάστατο χώρο πραγματοποιείται με τον ορισμό των τριδιάστατων καρτεσιανών συντεταγμένων τους (με τη χρήση ειδικής φόρμας εισαγωγής δεδομένων (Εικόνα 2.5), ή μέσω ενός αρχείου κειμένου για την εισαγωγή πολλών σημείων ταυτόχρονα).

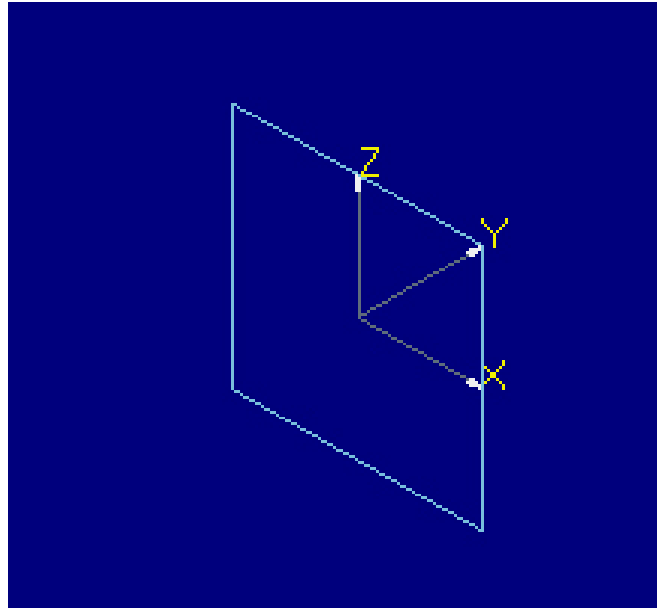


Εικόνα 2.4: Σημείο στο χώρο, ορισμένο από τις τριδιάστατες συντεταγμένες του. Το σημείο αναπαρίστανται με το σύμβολο του σταυρού.



Εικόνα 2.5: Ειδική φόρμα εισαγωγής δεδομένων για τον ορισμό ενός σημείου στο χώρο με τις καρτεσιανές του συντεταγμένες.

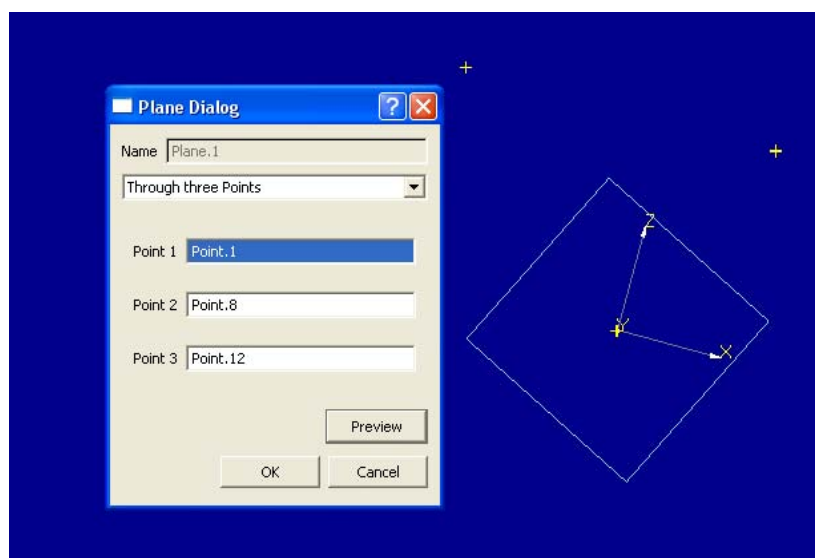
Τα επίπεδα στο χώρο (Εικόνα 2.6) είναι σημαντικές οντότητες, καθώς χρησιμοποιούνται ως βάσεις για τον ορισμό άλλων γεωμετρικών οντοτήτων. Στο λογισμικό χρησιμοποιούνται για τον ορισμό της διεύθυνσης άλλων γεωμετρικών οντοτήτων, ή ως επίπεδα σχεδίασης για διατομές σφαιρών, ή κατευθυντήριων (οδηγών) γραμμών.



Εικόνα 2.6: Επίπεδο στο χώρο. Ένα επίπεδο αναπαρίστανται με ένα τετράγωνο που έχει τον προσανατολισμό του επιπέδου.

Ένα επίπεδο μπορεί να ορισθεί είτε με τη βοήθεια τριών σημείων στο χώρο (τα οποία ανήκουν σε αυτό), είτε ως το επίπεδο που προκύπτει από τον ορισμό των παραμέτρων A , B , C , D της εξίσωσης: $Ax + By + Cz = D$

Σε κάθε περίπτωση, ο ορισμός του επιπέδου πραγματοποιείται από ειδική φόρμα εισαγωγής δεδομένων (Εικόνα 2.7). Η φόρμα παρέχει και τους δυο τρόπους ορισμού του επιπέδου. Στην περίπτωση του επιπέδου που ορίζεται από σημεία, ο χρήστης πρέπει να επιλέξει τα σημεία από το δέντρο των αντικειμένων και το όνομα κάθε σημείου θα εμφανιστεί στο ανάλογο πεδίο. Στην περίπτωση του επιπέδου ορισμένου από εξίσωση, ο χρήστης πρέπει να συμπληρώσει τις τιμές των παραμέτρων στα κατάλληλα πεδία της φόρμας.

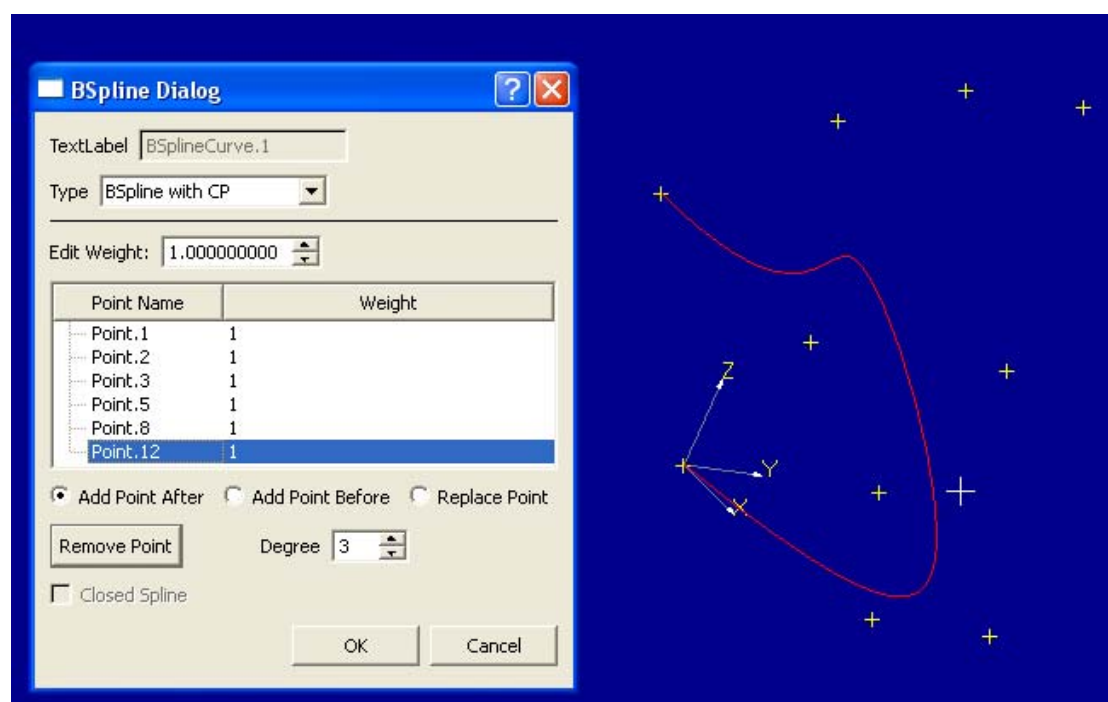


Εικόνα 2.7: Ειδική φόρμα εισαγωγής δεδομένων για σχεδίαση δυσδιάστατου επιπέδου.

2.3.2: Καμπύλες

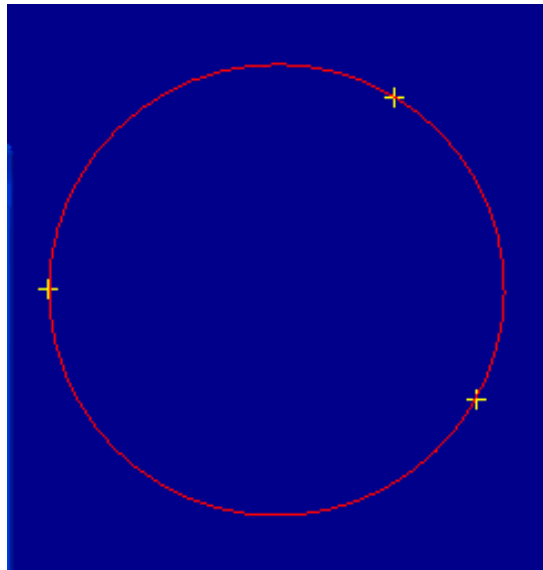
Μετά τον ορισμό των βασικών στοιχείων υποδομής, δηλαδή των σημείων στο χώρο και των επιπέδων, ο χρήστης είναι σε θέση να ορίσει καμπύλες, οι οποίες θα βασίζονται σε αυτά τα στοιχεία. Οι καμπύλες θα μπορέσουν να λειτουργήσουν ως διατομές, ή ως κατευθυντήριες γραμμές (οδηγοί) των υπό μοντελοποίηση σηράγγων. Οι καμπύλες που ορίζονται στο λογισμικό είναι κατά κύριο λόγο NURBS καμπύλες, λόγω της μεγάλης ποικιλομορφίας τους και παραμετροποίησής τους.

Μια καμπύλη NURBS (Εικόνα 2.8) μπορεί να ορισθεί από ένα αριθμό σημείων, τα οποία θα λειτουργούν ως σημεία ελέγχου της καμπύλης, ή ως σημεία παρεμβολής της. Στην περίπτωση των σημείων ελέγχου, ο χρήστης θα πρέπει να ορίσει επίσης το βαθμό της καμπύλης και τα αντίστοιχα βάρη των σημείων ελέγχου. Στην περίπτωση των σημείων παρεμβολής ο χρήστης θα πρέπει να ορίσει τον ελάχιστο επιτρεπόμενο βαθμό της καμπύλης. Και στις δύο περιπτώσεις όλες οι επιλογές πραγματοποιούνται μέσα από ειδική φόρμα εισαγωγής δεδομένων (Εικόνα 2.8). Η επιλογή των σημείων γίνεται από το δέντρο των αντικειμένων, ενώ κάθε αλλαγή στην καμπύλη εμφανίζεται δυναμικά στο παράθυρο προβολής.

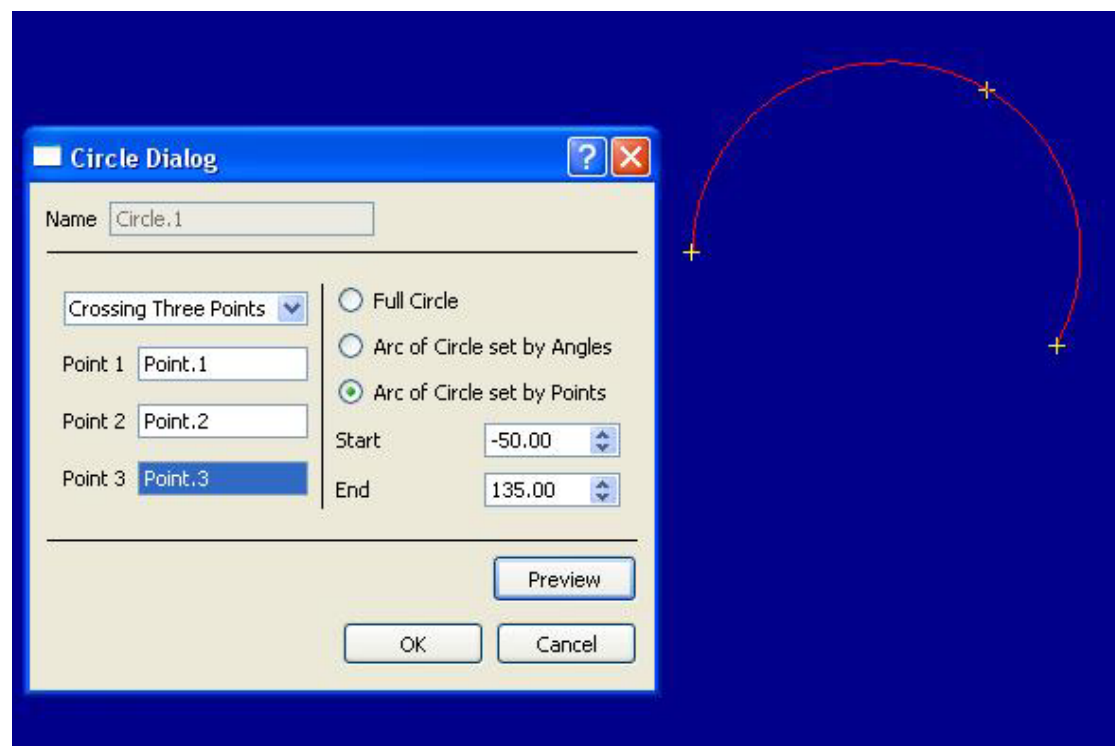


Εικόνα 2.8: Καμπύλη NURBS τρίτου βαθμού ορισμένη από σημεία ελέγχου. Φαίνεται επίσης η ειδική φόρμα εισαγωγής δεδομένων για τη σχεδίασή της.

Ο χρήστης έχει επίσης τη δυνατότητα, πέρα από τον ορισμό γενικών καμπύλων NURBS να ορίσει κύκλους ή τόξα κύκλων. Για τον ορισμό ενός κύκλου (Εικόνα 2.9), ο χρήστης πρέπει να επιλέξει τρία σημεία, από τα οποία θα περνά ο κύκλος. Για τον ορισμό ενός τόξου κύκλου (Εικόνα 2.10), ο χρήστης έχει τη δυνατότητα είτε να επιλέξει τρία σημεία από τα οποία θα περνά το τόξο, με το πρώτο και το τρίτο σημείο να ορίζουν τα όρια του τόξου, είτε να επιλέξει τρία σημεία, από τα οποία θα περνά το τόξο, ενώ το μήκος του να ορίζεται από δύο γωνίες που μετρούνται από το πρώτο σημείο και προς δύο αντίθετες κατευθύνσεις κατά μήκος του κύκλου. Τόσο οι κύκλοι, όσο και τα τόξα κύκλων ορίζονται από την ίδια φόρμα εισαγωγής δεδομένων (Εικόνα 2.10). Η επιλογή των σημείων γίνεται από το δέντρο των αντικειμένων.



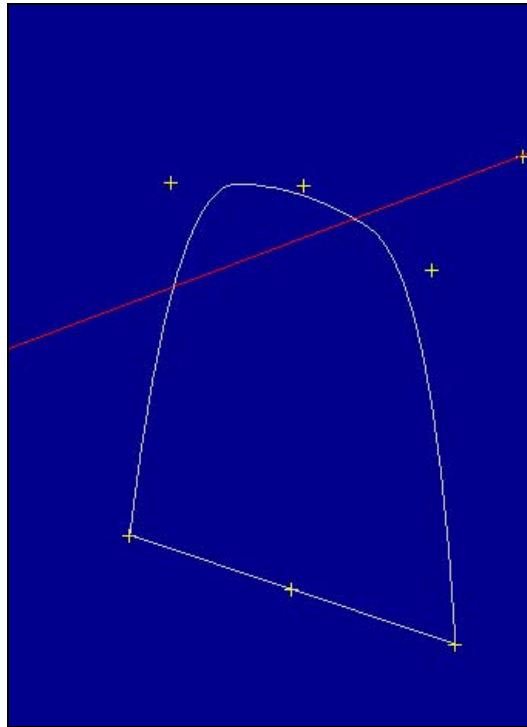
Εικόνα 2.9: Κύκλος ορισμένος από τρία σημεία.



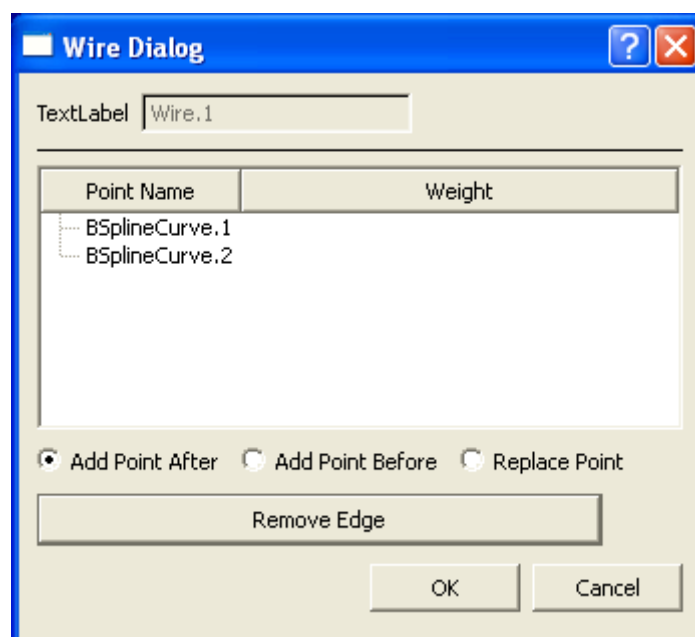
Εικόνα 2.10: Τόξο κύκλου ορισμένο από τρία σημεία. Το αρχικό και τελικό σημείο θέτουν τα όρια του τόξου. Παρουσιάζεται επίσης η ειδική φόρμα εισαγωγής δεδομένων για το τόξο.

Όλες οι καμπύλες που σχεδιάζονται με το λογισμικό, άσχετα με τον τρόπο σχεδίασης, χαρακτηρίζονται ως ακμές (Edges). Ακμή είναι μια καμπύλη η οποία έχει συνέχεια πρώτης παραγώγου. Ειδική περίπτωση αποτελούν οι πρωτοβάθμιες καμπύλες NURBS, οι οποίες μπορεί να παρουσιάζουν ασυνέχεια της πρώτης παραγώγου στα σημεία ελέγχου, όμως χαρακτηρίζονται και αυτές ως ακμές.

Ενώνοντας ένα αριθμό από διαδοχικές ακμές, οι οποίες πρέπει να έχουν σημείο επαφής τα ακραία σημεία τους, δημιουργείται ένα σύρμα (Wire). Σύρμα (Εικόνα 2.11) είναι μια περίπλοκη καμπύλη, η οποία δεν είναι απαραίτητο να παρουσιάζει συνέχεια πρώτης παραγώγου στα σημεία επαφής των ακμών από τις οποίες αποτελείται. Ο ορισμός ενός σύρματος γίνεται μέσα από ειδική φόρμα εισαγωγής δεδομένων, και η επιλογή των ακμών που αποτελούν το σύρμα γίνεται από το δέντρο των δεδομένων (Εικόνα 2.12).



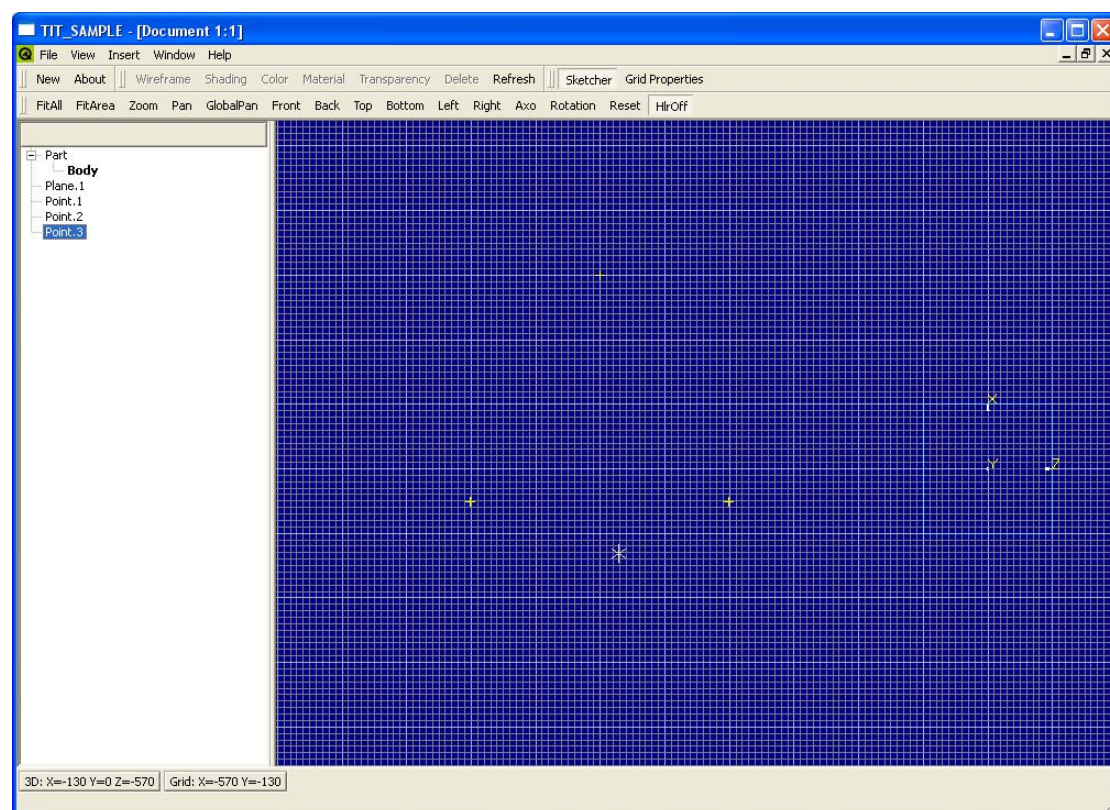
Εικόνα 2.11: Σύρμα που προέκυψε από την ένωση μιας ευθύγραμμης και μιας καμπύλης ακμής. Και οι δύο ακμές είναι ορισμένες ως καμπύλες NURBS.



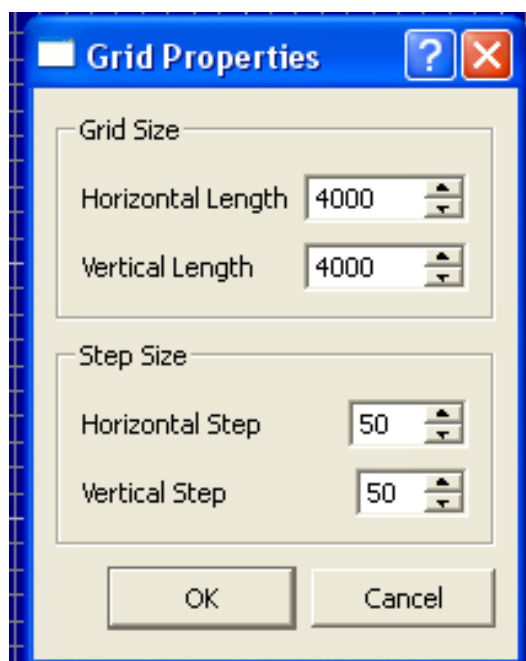
Εικόνα 2.12: Η ειδική φόρμα εισαγωγής δεδομένων για τον ορισμό ενός σύρματος.

2.3.3: Περιβάλλον διδιάστατης σχεδίασης

Το περιβάλλον διδιάστατης σχεδίασης είναι μια κατάσταση λειτουργίας του λογισμικού που ενεργοποιείται από τον χρήστη μετά την επιλογή ενός επιπέδου από το δέντρο των αντικειμένων. Τότε η προβολή περιστρέφεται για να ταυτιστεί με το σύστημα συντεταγμένων του επιλεγμένου επιπέδου και ένα διδιάστατο πλέγμα (Εικόνα 2.13) εμφανίζεται για να ορίσει πλέον την επιφάνεια εργασίας. Το διδιάστατο πλέγμα ανταποκρίνεται στο τοπικό σύστημα συντεταγμένων του επιλεγμένου επιπέδου. Ο χρήστης έχει τη δυνατότητα προσαρμογής του μεγέθους του πλέγματος, καθώς και της πύκνωσής του. Η αλλαγή του πλέγματος πραγματοποιείται από ειδική φόρμα εισαγωγής δεδομένων (Εικόνα 2.14).



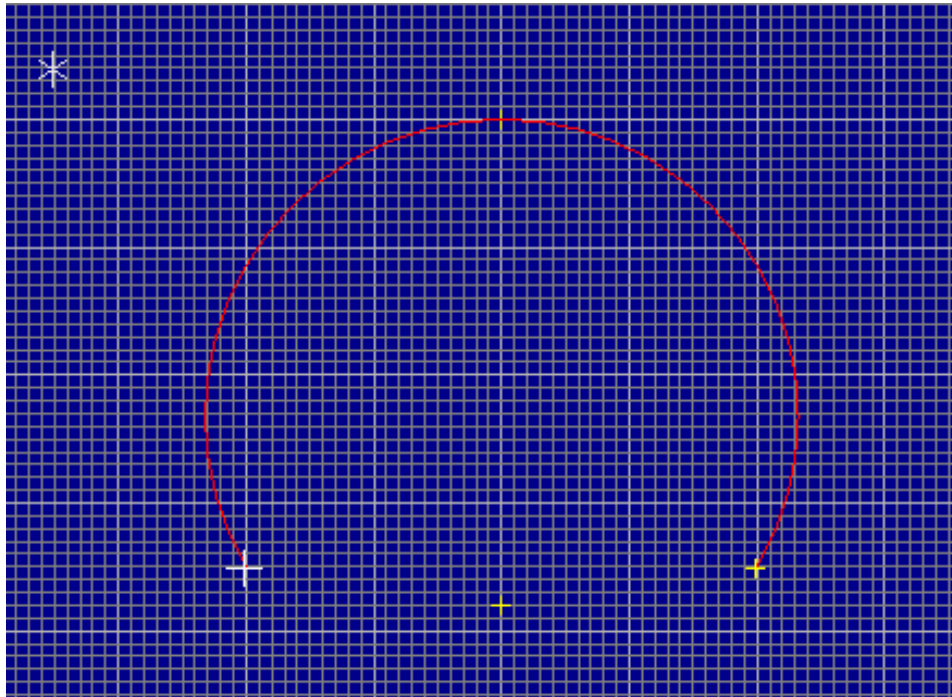
Εικόνα 2.13: Το περιβάλλον διδιάστατης σχεδίασης. Στα δεξιά φαίνεται το επίπεδο πάνω στο οποίο υλοποιείται η σχεδίαση.



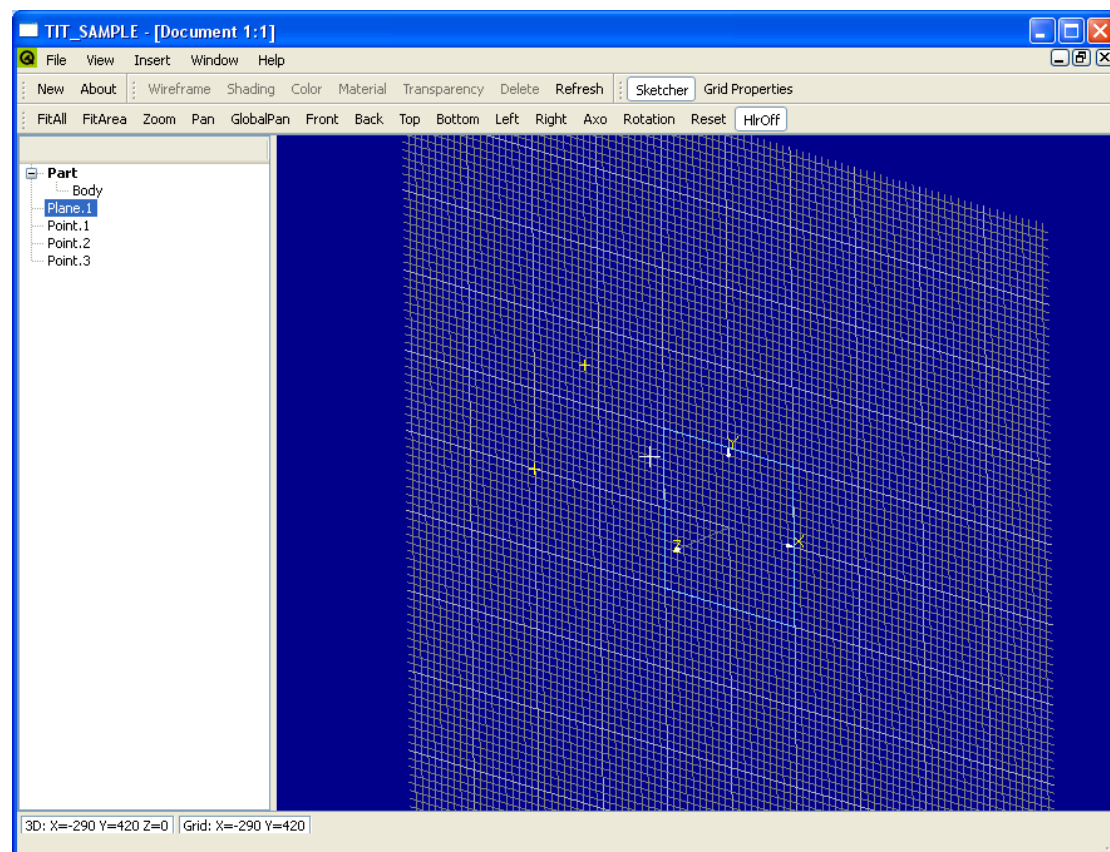
Εικόνα 2.14: Ειδική φόρμα εισαγωγής δεδομένων για τον ορισμό των ιδιοτήτων του πλέγματος στο περιβάλλον διδιάστατης σχεδίασης.

Όταν το λογισμικό βρίσκεται στη λειτουργία διδιάστατης σχεδίασης όλες οι λειτουργίες του λογισμικού γίνονται όπως και στην τριδιάστατη λειτουργία. Όμως, στην κατάσταση διδιάστατης σχεδίασης, η εισαγωγή σημείων γίνεται με απλή επιλογή της θέσης του με τον κέρσορα πάνω στο πλέγμα (Εικόνα 2.15), με τη λειτουργία «αρπαγής στο πλέγμα» (snap to grid). Με αυτή τη λειτουργία, ο κέρσοράς έλκεται στο κοντινότερο σημείο τομής του πλέγματος και μόνο σε εκείνο το σημείο μπορεί ο χρήστης να εισάγει σημείο. Όλα τα σημεία που εισάγονται με αυτή τη λειτουργία θεωρούνται ακόμη τριδιάστατα σημεία και έχουν τις ίδιες ιδιότητες με τα σημεία που εισάγονται με ορισμό των συντεταγμένων τους.

Στη λειτουργία διδιάστατης σχεδίασης όλες οι λειτουργίες του παράθυρου προβολής (εστίαση, μετακίνηση, περιστροφή κ.λπ.) πραγματοποιούνται όπως και στην τριδιάστατη σχεδίαση. Στην περίπτωση της περιστροφής το πλέγμα περιστρέφεται και αυτό, ακολουθώντας το επίπεδο σχεδίασης (Εικόνα 2.16). Κατά την εισαγωγή σημείων, ενώ το πλέγμα έχει περιστραφεί, τα σημεία και πάλι εισάγονται πάνω στο πλέγμα, άσχετα με την περιστροφή.



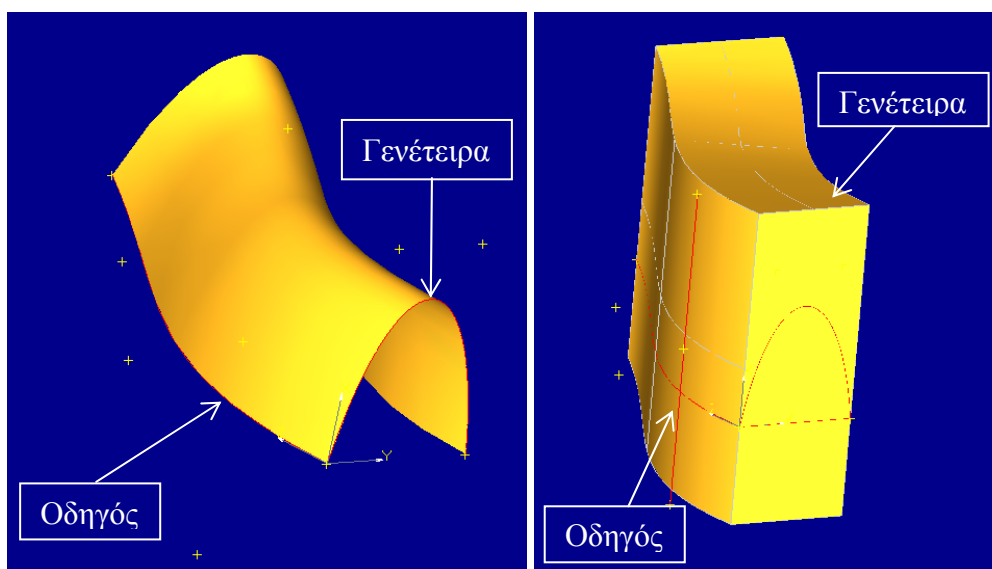
Εικόνα 2.15: Τόξο κύκλου που δημιουργήθηκε στο περιβάλλον διδιάστατης σχεδίασης. Πάνω αριστερά φαίνεται το ίχνος του κέρσorsa για την εισαγωγή σημείου στο επίπεδο.



Εικόνα 2.16: Το περιβάλλον διδιάστατης σχεδίασης, όταν έχει γίνει περιστροφή του σχεδίου στο παράθυρο προβολής. Η σχεδίαση συνεχίζεται στις δύο διαστάσεις.

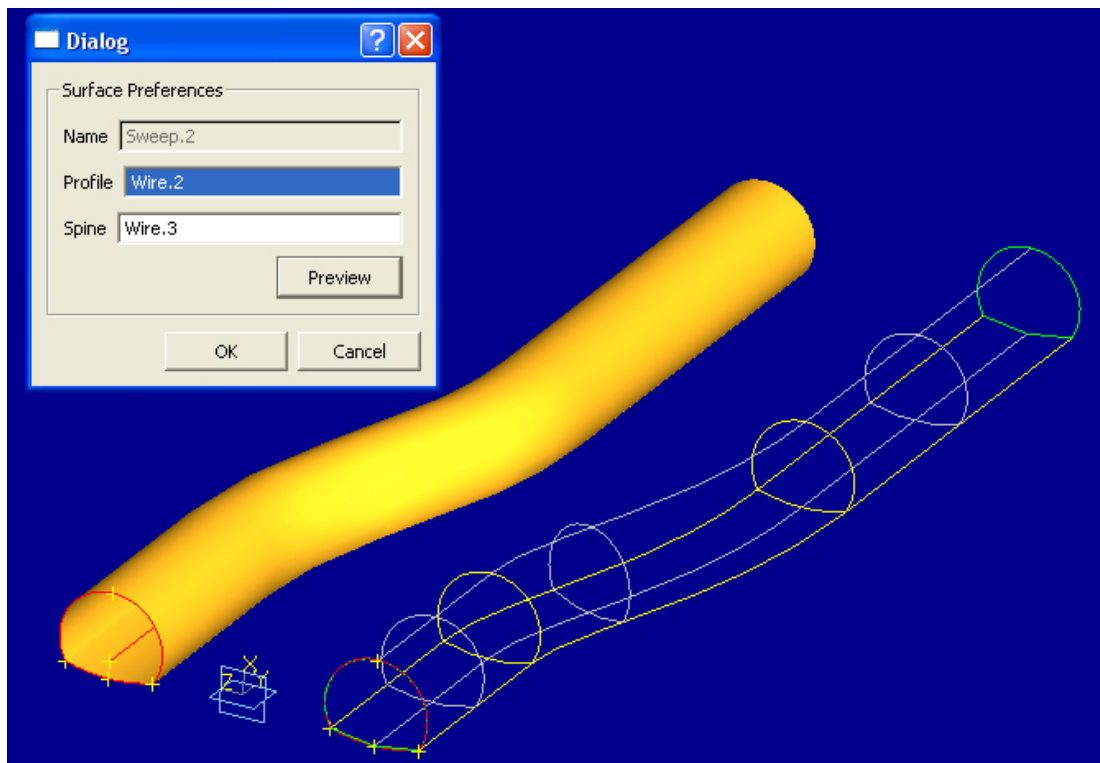
2.3.4: Επιφάνειες, όγκοι και λογικές πράξεις Boole.

Οι επιφάνειες και οι όγκοι που αναπαριστούν σήραγγες ορίζονται στο λογισμικό μέσω των μεθόδων οδήγησης (Sweep) και εξώθησης (Extrude). Οδήγηση είναι το σχήμα που προκύπτει από την «οδήγηση» μιας γενέτειρας γεωμετρίας πάνω σε μια διαδρομή που ονομάζουμε οδηγό. Η γενέτειρα γεωμετρία μπορεί να είναι είτε καμπύλη είτε επιφάνεια. Η γεωμετρία που προκύπτει από τη διαδικασία της οδήγησης εξαρτάται από τον τύπο της γενέτειρας γεωμετρίας. Στην περίπτωση που η γενέτειρα γεωμετρία είναι καμπύλη, τότε το σχήμα που προκύπτει είναι επιφάνεια (Εικόνα 2.17). Στην περίπτωση που η γενέτειρα γεωμετρία είναι επιφάνεια, τότε το σχήμα που προκύπτει είναι στερεό (Εικόνα 2.17).



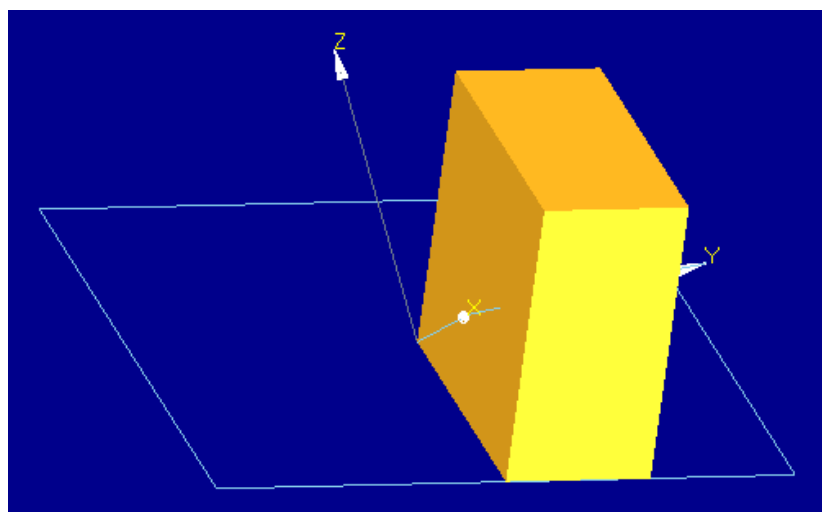
Εικόνα 2.17: Σχήματα που προέκυψαν από διαδικασία οδήγησης. Στα αριστερά το σχήμα είναι επιφάνεια και προκύπτει από γενέτειρα καμπύλη. Στα δεξιά το σχήμα είναι στερεό που προκύπτει από γενέτειρα επιφάνεια.

Η καμπύλη οδηγός μπορεί να έχει οποιαδήποτε μορφή, ενώ μπορεί να είναι από απλός κύκλος μέχρι περίπλοκο σύρμα, που προκύπτει από ένα μεγάλο αριθμό ακμών. Επίσης, η ίδια καμπύλη οδηγός μπορεί να χρησιμοποιηθεί για την παραγωγή διαφορετικών σχημάτων, χρησιμοποιώντας διαφορετικές γενέτειρες γεωμετρίες (Εικόνα 2.18). Τα σχήματα που θα προκύψουν από την ίδια καμπύλη οδηγό θα είναι παράλληλα μεταξύ τους. Η μέθοδος εξώθησης πραγματοποιείται με τον ίδιο τρόπο, χρησιμοποιώντας ως καμπύλη οδηγό ένα οποιοδήποτε ευθύγραμμο τμήμα. Η εισαγωγή των δεδομένων γίνεται από ειδική φόρμα εισαγωγής δεδομένων (Εικόνα 2.18) και η επιλογή των οντοτήτων που παίρνουν μέρος στη διαδικασία γίνεται με επιλογή τους από το δέντρο των αντικειμένων.



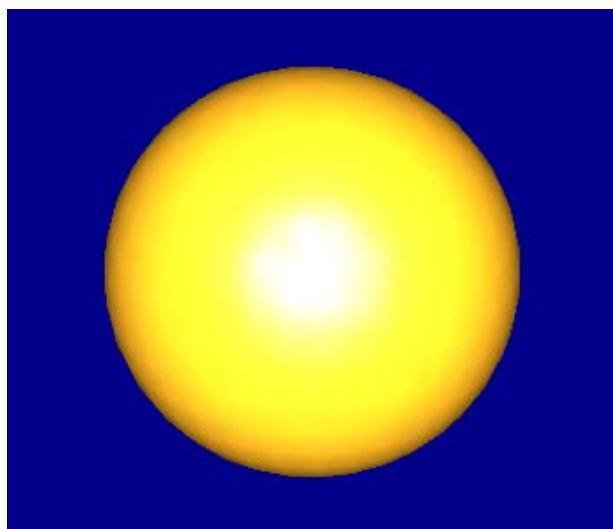
Εικόνα 2.18: Αποτέλεσμα δύο διαδικασιών οδήγησης με χρήση της ίδιας καμπύλης για οδηγό. Πάνω δεξιά η ειδική φόρμα εισαγωγής δεδομένων για το αριστερό σχήμα.

Βασικά στερεά, όπως ορθογώνια παραλληλεπίπεδα και σφαίρες μπορούν επίσης να δημιουργηθούν με το λογισμικό. Τα στερεά αυτά μπορούν να χρησιμοποιηθούν ως βοηθητικές οντότητες για άλλες άλλα στάδια της σχεδίασης μιας σήραγγας. Ένα ορθογώνιο παραλληλεπίπεδο (Εικόνα 2.19) μπορεί να σχεδιαστεί ορίζοντας τις διαστάσεις του και το επίπεδο το οποίο θα ορίζει την κατεύθυνση δημιουργίας του στερεού. Η εισαγωγή των δεδομένων πραγματοποιείται από ειδική φόρμα εισαγωγής δεδομένων και η επιλογή του βασικού επιπέδου γίνεται από το δέντρο των αντικειμένων. Το βασικό επίπεδο πρέπει να έχει ήδη ορισθεί από τον χρήστη.



Εικόνα 2.19: Ορθογώνιο παραλληλεπίπεδο ορισμένο πάνω σε ένα επίπεδο στο χώρο.

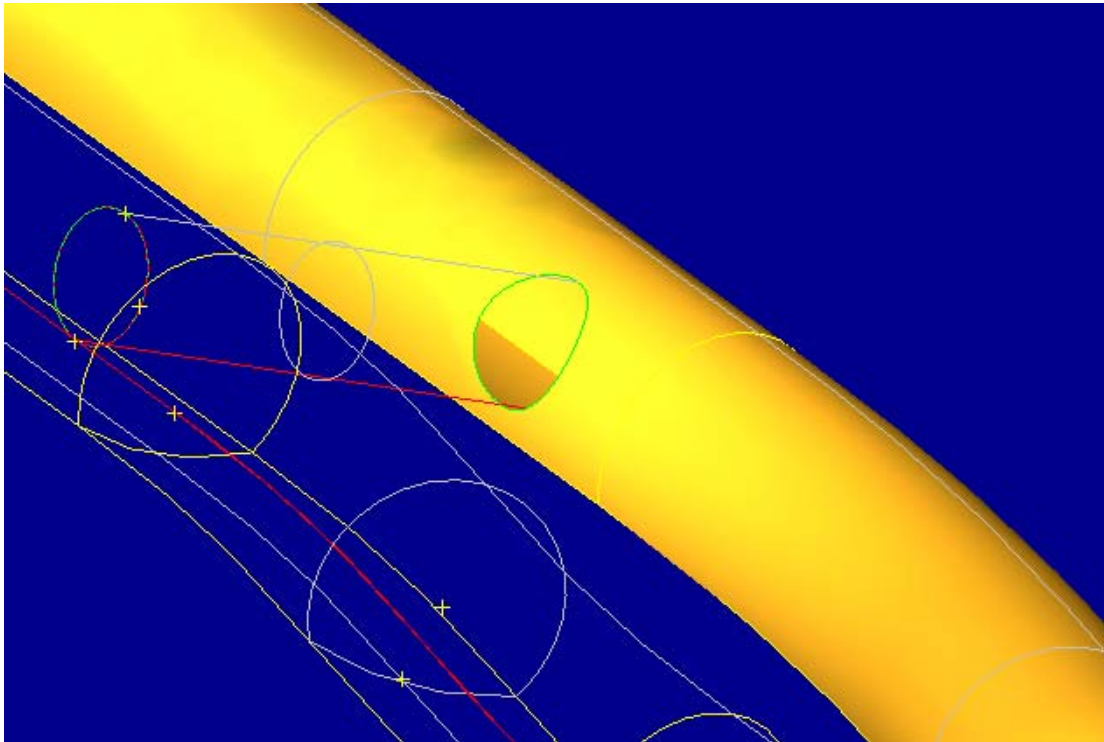
Η σφαίρα (Εικόνα 2.20) μπορεί να σχεδιαστεί ορίζοντας μόνο το κέντρο και την ακτίνα της. Το κέντρο της σφαίρας αναπαρίσταται από ένα σημείο στο χώρο, που επιλέγεται από το δέντρο των αντικειμένων.



Εικόνα 2.20: Σφαίρα που δημιουργήθηκε ορίζοντας ένα σημείο ως το κέντρο της, καθώς και την ακτίνα της.

Όταν κάποιες επιφάνειες ή στερεά τέμνονται, μπορεί να γίνει περεταίρω επεξεργασία σε αυτά, χρησιμοποιώντας λογικές πράξεις Boole. Το λογισμικό υποστηρίζει τέσσερεις λογικές πράξεις. Αυτές είναι οι ακόλουθες: Αποκοπή (Cut), Ένωση (Fuse), Τομή (Common) και Διατομή (Section). Η πράξη Αποκοπή (Cut(Shape1, Shape2)) δημιουργεί ένα καινούριο σχήμα αφαιρώντας το σχήμα 1 από το σχήμα 2 (Εικόνα 2.21). Ως σχήμα θεωρείται οποιαδήποτε οντότητα έχει δημιουργηθεί εκτός από σημείο ή επίπεδο. Η πράξη Ένωση ενώνει δύο σχήματα για να δημιουργήσει ένα καινούριο. Η πράξη Τομή δημιουργεί ένα καινούριο σχήμα, που προκύπτει από την κοινή περιοχή των σχημάτων που παίρνουν μέρος στην πράξη. Η πράξη Διατομή βρίσκει την κοινή διατομή των δυο τεμνόμενων σχημάτων.

Οι λογικές πράξεις Boole είναι δυνατόν να πραγματοποιηθούν και μεταξύ σχημάτων διαφορετικού τύπου, δηλαδή μεταξύ επιφανειών και στερεών, αλλά με ορισμένους περιορισμούς. Όταν ένα στερεό και μια επιφάνεια παίρνουν μέρος σε μια πράξη αποκοπής τότε το στερεό θα πρέπει να είναι ο αφαιρέτης και το σχήμα που θα προκύψει θα είναι επιφάνεια. Στην περίπτωση της Ένωσης και της Τομής το σχήμα που θα προκύψει θα είναι εκείνου του τύπου με τον μικρότερο βαθμό (στην περίπτωση επιφάνειας και στερεού θα είναι επιφάνεια). Σε κάθε άλλη περίπτωση τα αποτελέσματα δεν μπορεί να είναι ελεγχόμενα.



Εικόνα 2.21: Λογική πράξη αποκοπής (Cut).

2.3.5: Οδηγός δημιουργίας διατομών σηράγγων

Οι διατομές σηράγγων σχεδιάζονται από ένα αριθμό διαδοχικών τόξων κύκλου. Για αυτό το λόγο έχει προστεθεί στο πρόγραμμα ο οδηγός (wizard) δημιουργίας διατομών σηράγγων (Εικόνα 2.22). Σε αυτή τη φόρμα ο χρήστης μπορεί να ορίσει ένα αριθμό από διαδοχικά τόξα κύκλου, που θα αποτελούν μια διατομή σήραγγας. Για τον ορισμό κάθε τόξου ο χρήστης πρέπει να δώσει την ακτίνα του και τη γωνία που μετριέται από τον κατακόρυφο άξονα και την ακτίνα του τόξου στο ακραίο σημείο του.

	Radius\ Length	Angle	Partitioning	
1	7.045	66	-0.7	
2	10	18	-0.7	
3	3	64	-0.7	
4				<input checked="" type="checkbox"/> Finalize

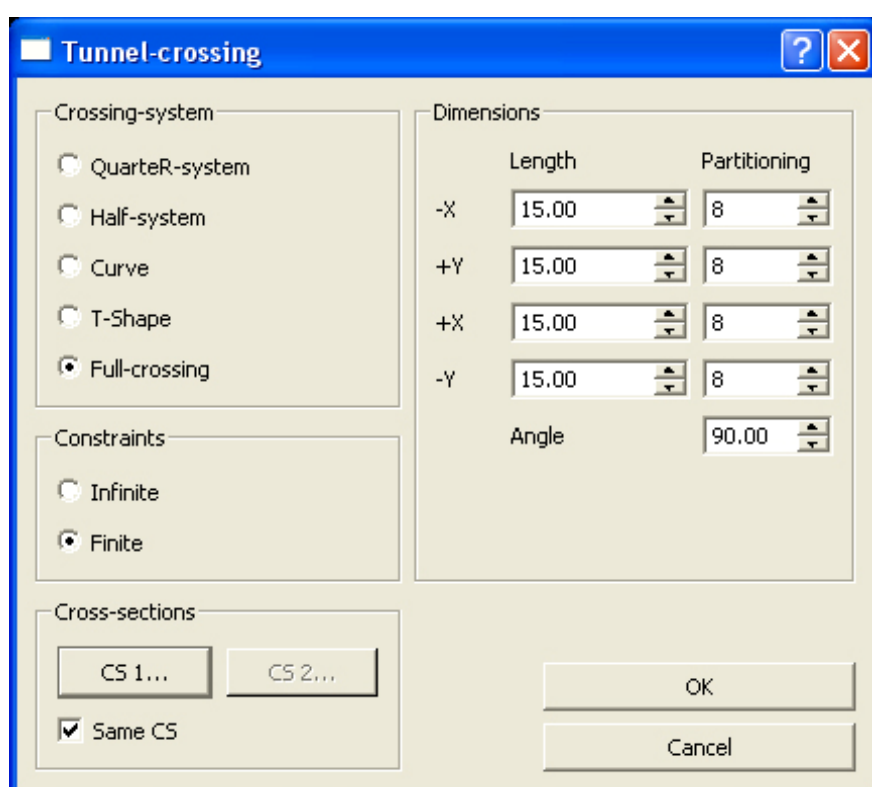
Εικόνα 2.22: Ο οδηγός δημιουργίας διατομών σηράγγων.

Αφού ορισθούν, τα τόξα σχεδιάζονται ώστε να έχουν συνέχεια πρώτης παραγώγου στα ακραία σημεία τους με τα γειτονικά τους τόξα. Το πρώτο και τελευταίο τόξο είναι κάθετα στον κατακόρυφο άξονα. Το τελευταίο τόξο κύκλου υπολογίζεται αυτόματα από το λογισμικό, καθώς ορίζεται έτσι ώστε να καταλήγει στον κάθετο άξονα και η

εφαπτομένη του σε αυτό το σημείο να δημιουργεί γωνία 90 μοιρών με τον άξονα. Έτσι ορίζεται μια διατομή σήραγγας κατά το ήμισυ. Το υπόλοιπο μισό ορίζεται ως το κατοπτρικό του μισού της διατομής ως προς τον κατακόρυφο άξονα. Η διαδικασία αυτή είναι διδιάστατη, αλλά ο χρήστης δύναται να ορίσει το επίπεδο στο οποίο θα λάβει χώρα.

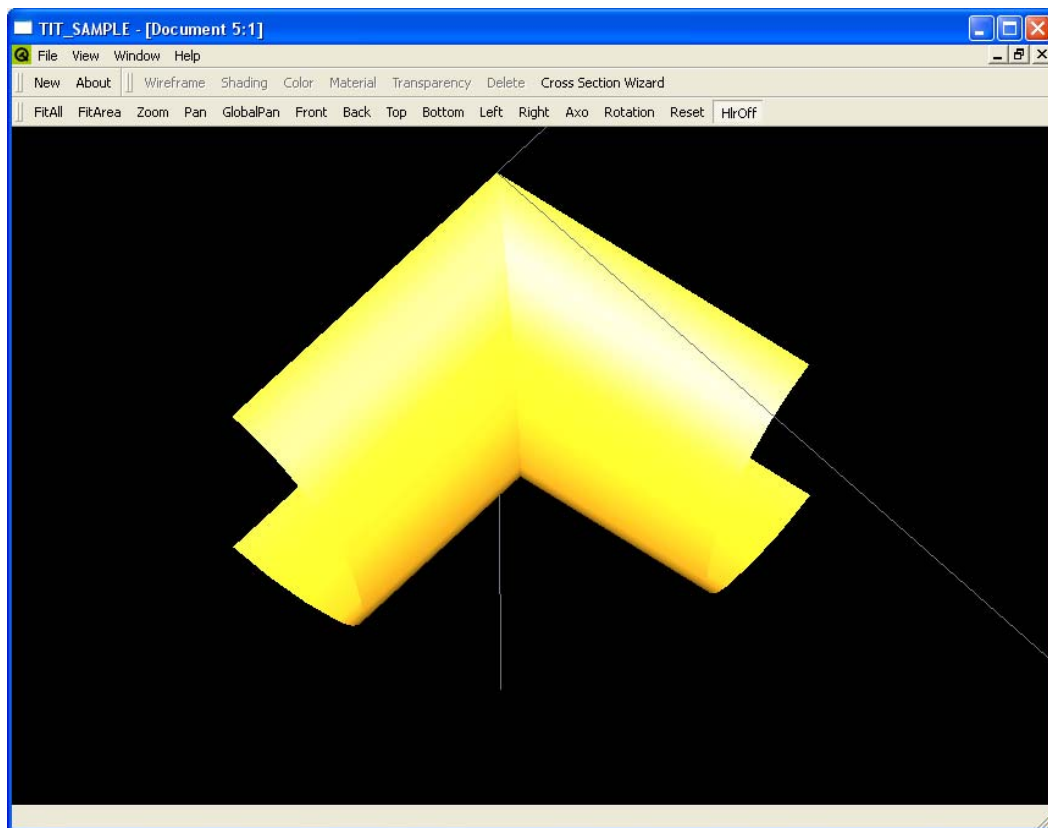
2.3.6: Οδηγός δημιουργίας διασταυρώσεων σηράγγων

Ο οδηγός δημιουργίας διασταυρώσεων σηράγγων (Εικόνα 2.23) είναι μια προσθήκη στο λογισμικό για την ευκολότερη δημιουργία τυπικών διασταυρώσεων σηράγγων ίδιας ή διαφορετικής διατομής. Ο χρήστης πρέπει να δώσει ένα αριθμό δεδομένων αναφορικά με τις διασταυρούμενες σήραγγες, όπως ο τύπος της διατομής κάθε σήραγγας, ο τύπος της διασταύρωσης, το μήκος κάθε μέρους της σήραγγας, καθώς και η γωνία με την οποία θα διασταυρώνονται οι δυο σήραγγες μεταξύ τους.

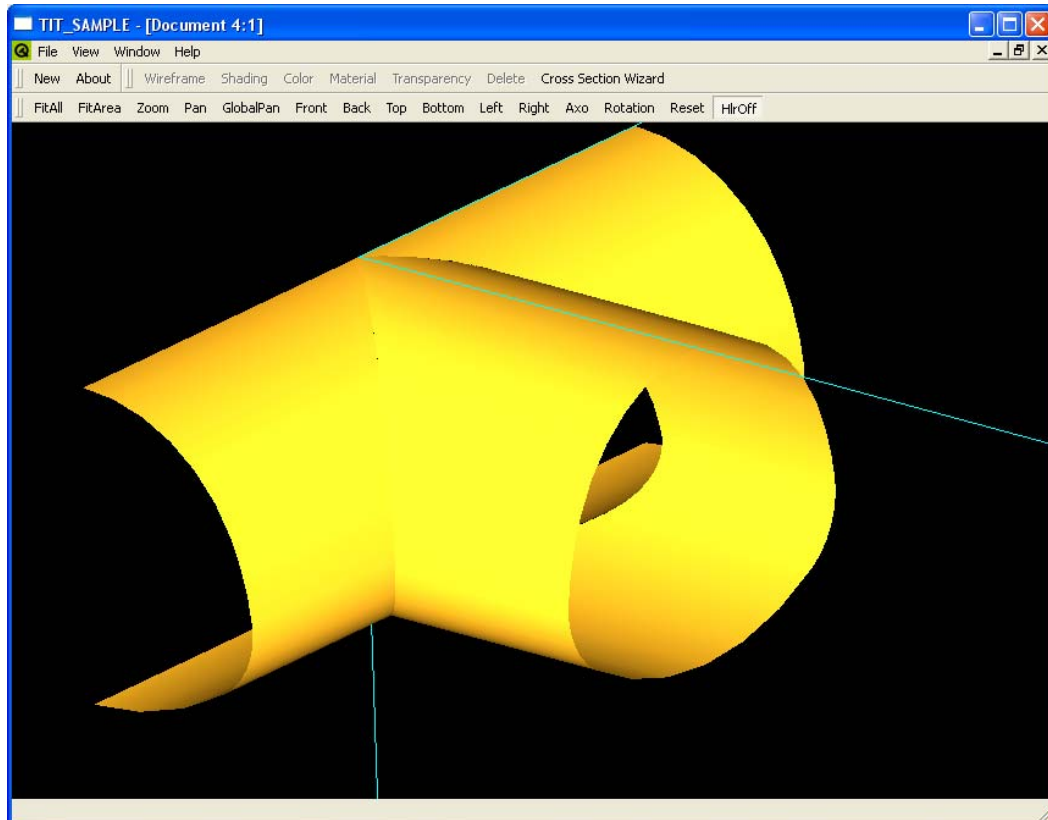


Εικόνα 2.23: Ο οδηγός δημιουργίας διασταυρώσεων.

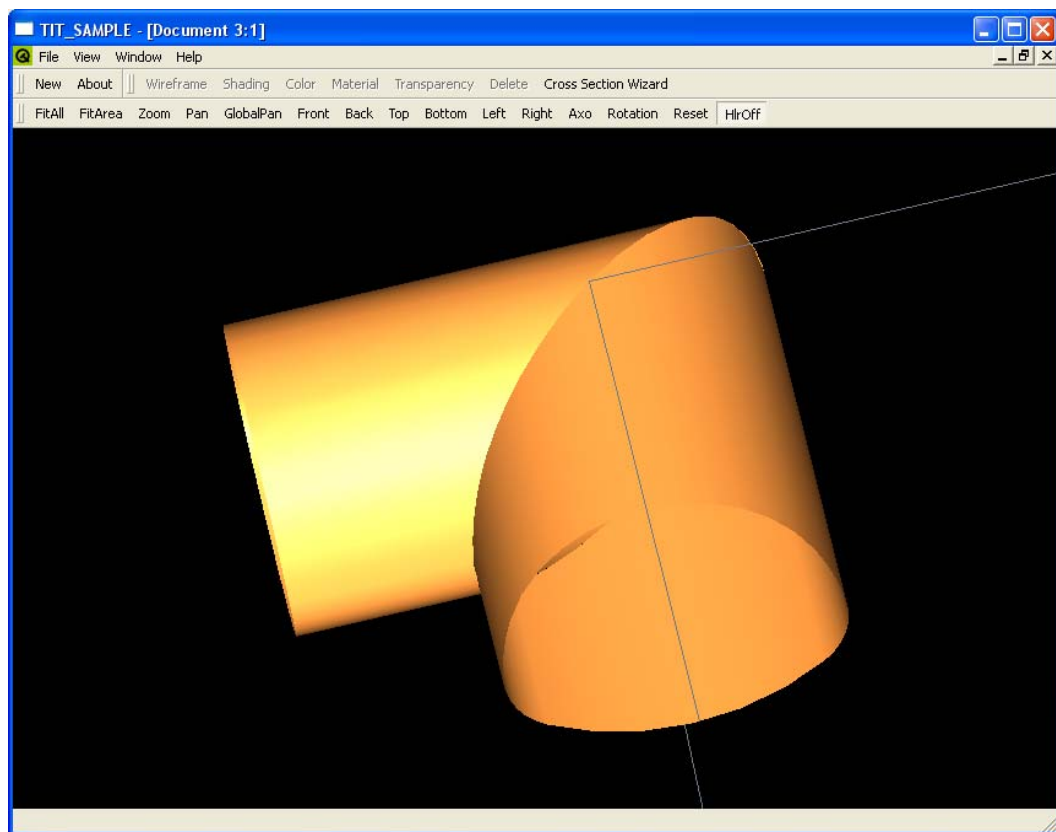
Οι τύποι των διατομών των σηράγγων ορίζονται από τον οδηγό δημιουργίας διατομής σήραγγας, ο οποίος μπορεί να ενεργοποιείται μέσα από τον οδηγό διασταυρώσεων σηράγγων. Οι τύποι διασταυρώσεων που μπορεί να δημιουργηθούν με τον οδηγό είναι η διασταύρωση τεταρτημόριου (Εικόνα 2.24), η διασταύρωση μισού (Εικόνα 2.25), η διασταύρωση τύπου γωνίας (Εικόνα 2.26), η διασταύρωση τύπου T (Εικόνα 2.27) και η πλήρης διασταύρωση (Εικόνα 2.28). Όλοι οι τύποι διασταυρώσεων υποστηρίζουν σήραγγες με διαφορετικές διατομές, εκτός από τη διασταύρωση τύπου γωνίας, αφού η μορφή της υπαγορεύει τη χρήση σηράγγων με ίδιες διατομές.



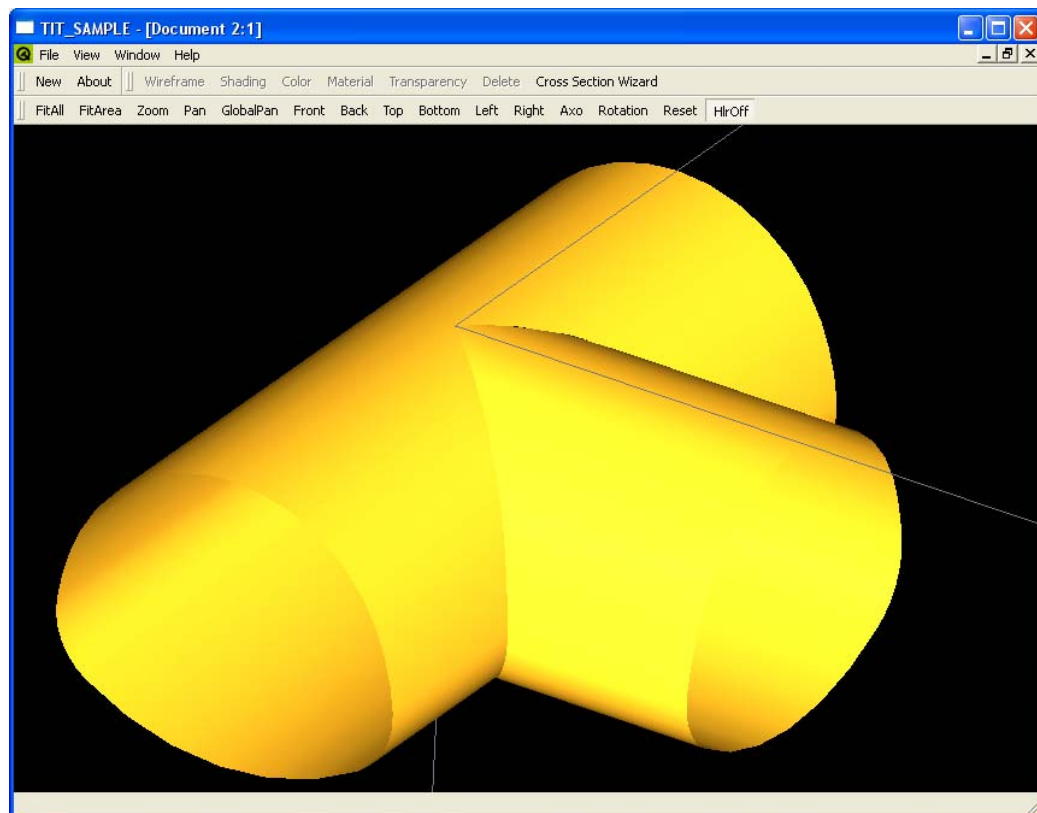
Εικόνα 2.24: Διασταύρωση τεταρτημόριου με σήραγγες ίδιας διατομής.



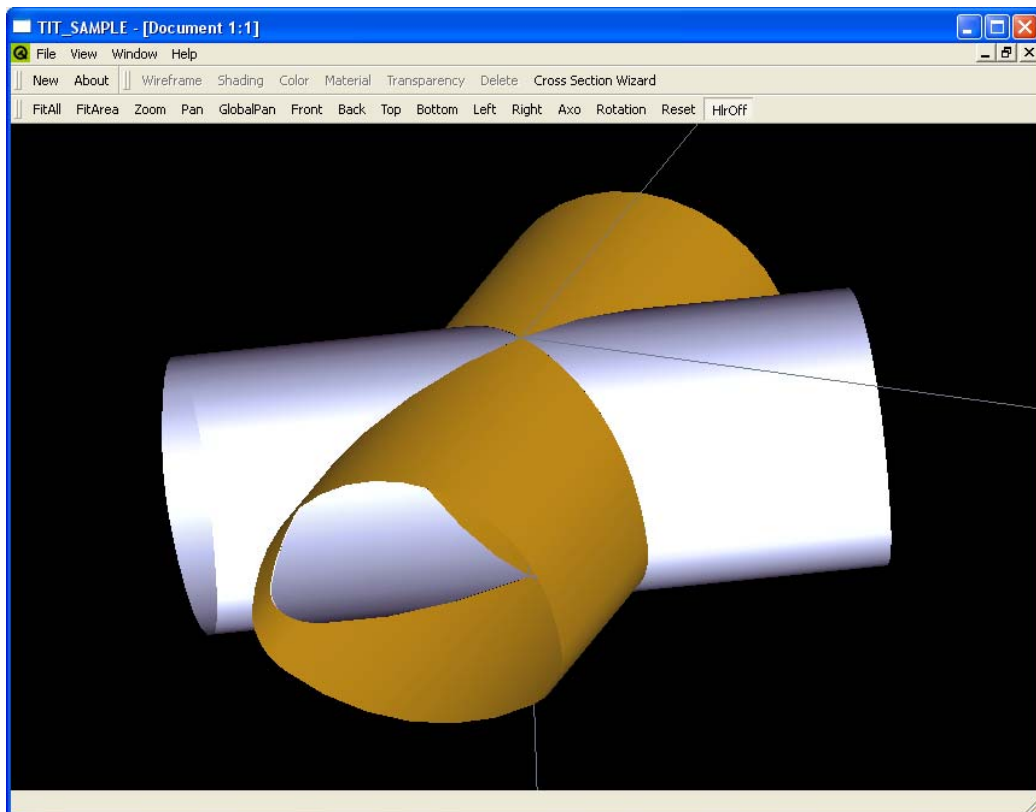
Εικόνα 2.25: Διασταύρωση μισού με σήραγγες ίδιας διατομής.



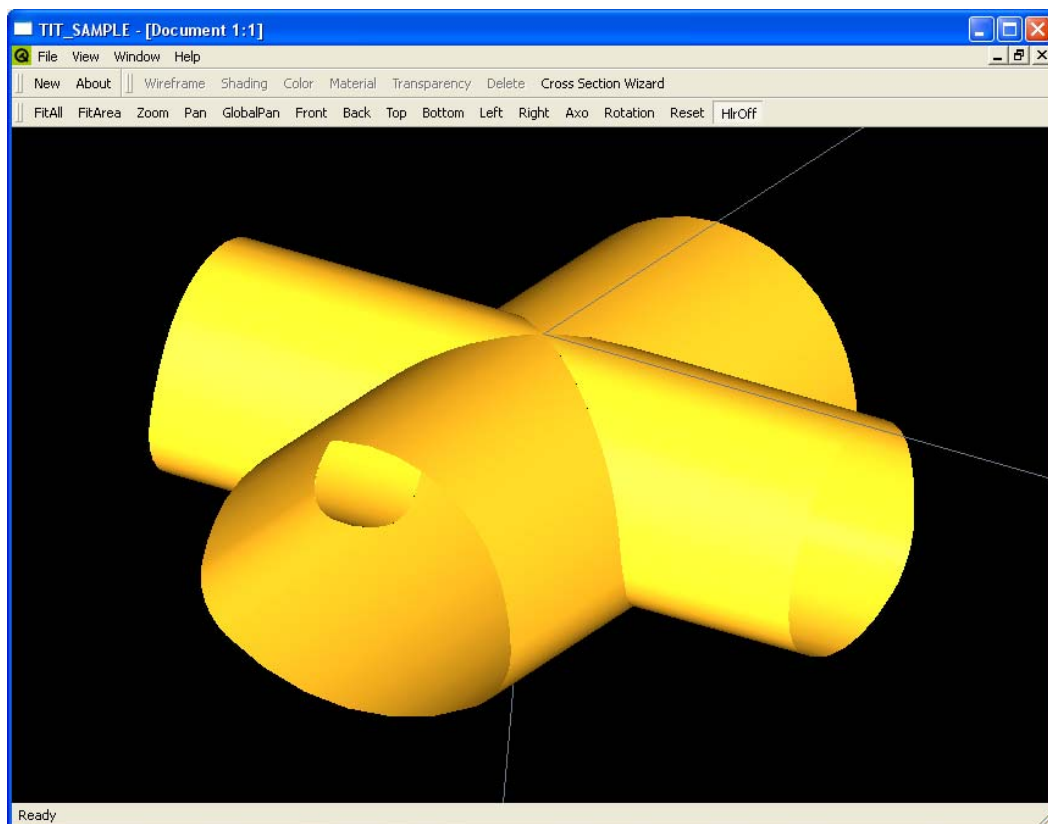
Εικόνα 2.26: Διασταύρωση τύπου Γωνίας.



Εικόνα 2.27: Διασταύρωση τύπου T με σήραγγες ίδιας διατομής.



Εικόνα 2.28: Πλήρης διασταύρωση με σήραγγες ίδιας διατομής, υπό γωνία 120 μοιρών.



Εικόνα 2.29: Πλήρης διασταύρωση με χρήση σηράγγων διαφορετικής διατομής.

2.4: Η αντικειμενοστραφής δομή του προγράμματος.

Προκειμένου να επιτευχθεί η δημιουργία ενός πολύ ευέλικτου λογισμικού, τόσο στην ανάπτυξή του, όσο και στη χρήση του, επιλέχθηκε από την αρχή της ανάπτυξής του ο αντικειμενοστραφής τρόπος προγραμματισμού και η γλώσσα C++, ως μια γλώσσα με πολλές δυνατότητες σε αυτό τον τομέα, η οποία παρέχει στον προγραμματιστή πολλές ελευθερίες και ευκολίες. Η χρήση της αντικειμενοστραφούς λογικής επέτρεψε πολλές φορές την αναθεώρηση της βασικής δομής του προγράμματος, χωρίς να χρειαστεί η επανασχεδίαση μεγάλου μέρους του κώδικα. Επίσης, η χρήση της Open Cascade ως βάσης για όλες τις σχεδιαστικές λειτουργίες του προγράμματος αποδείχτηκε καταλυτικός παράγοντας στην ανάπτυξη, καθώς και στην επιλογή της γλώσσας προγραμματισμού, αφού οι βιβλιοθήκες της είναι γραμμένες σε C++.

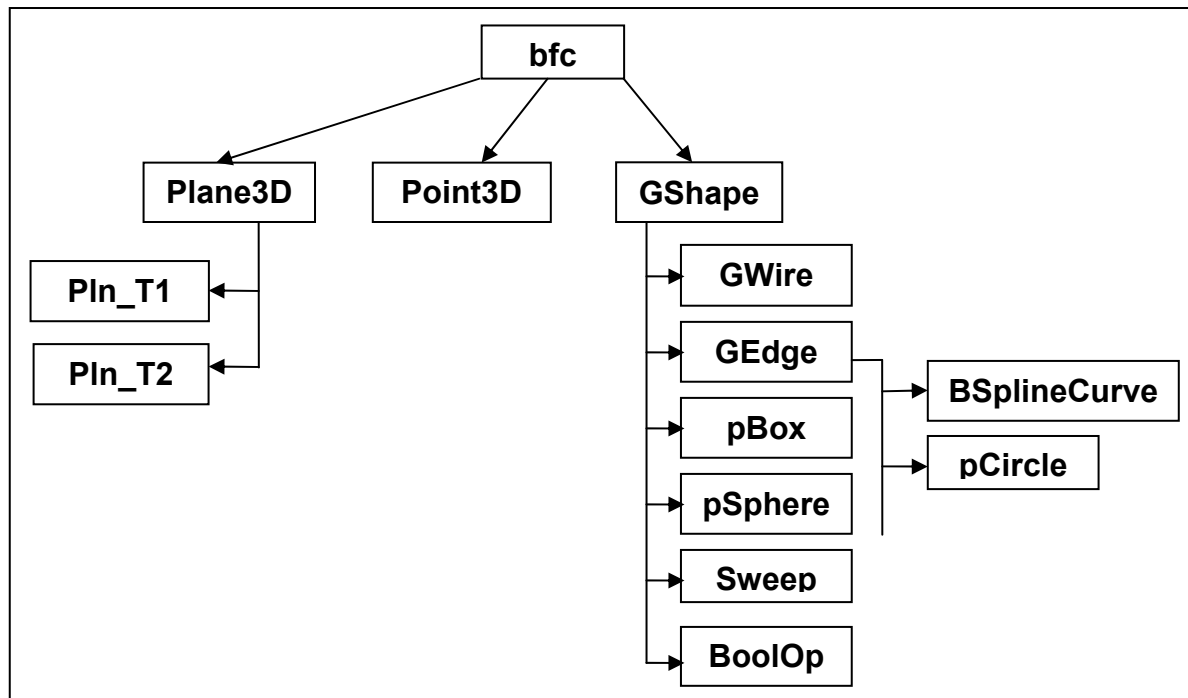
Για το σωστότερο χειρισμό των δυνατοτήτων που παρέχει η γλώσσα προγραμματισμού και η λογική του λογισμικού, σχεδιάστηκε από την αρχή η βασική δομή του, η οποία παρουσιάζεται στο σχήμα 2.30, όπως εξελίχθηκε στην τελική της μορφή. Ο σχεδιασμός αφορούσε βασικά τις σχεδιαστικές κλάσεις του προγράμματος και πώς αυτές θα συνυπήρχαν με το υπόλοιπο πρόγραμμα. Από την αρχή υιοθετήθηκε η ιδέα της χρήσης μιας βασικής κλάσης (bfc), την οποία θα κληρονομούν όλες οι υπόλοιπες σχεδιαστικές κλάσεις και θα λειτουργεί ως μέσο διεπαφής μεταξύ των συναρτήσεων υπολογισμού των γεωμετρικών οντοτήτων και του γραφικού περιβάλλοντος. Αρχικά όλες οι σχεδιαστικές κλάσεις κληρονομούνταν κατευθείαν από αυτή τη βασική κλάση. Αργότερα κατέστη απαραίτητο η χρήση συγκεκριμένων «ενδιάμεσων» κλάσεων (π.χ. GShape, GEdge), έτσι ώστε να υπάρχει μια ομαδοποίηση των γεωμετρικών οντοτήτων, που κληρονομούσαν τις ενδιάμεσες κλάσεις. Η λογική αυτή έδωσε τη δυνατότητα χρήσης γενικών μεθόδων σχεδίασης, που αφορούσαν ένα μεγάλο αριθμό διαφορετικών γεωμετρικών οντοτήτων, αλλά με παρόμοιες ιδιότητες. Για παράδειγμα, ενώ ο ορισμός ενός κύκλου και μιας καμπύλης NURBS είναι διαφορετικός εν γένει, η καταγωγή τους από την ίδια ομάδα κλάσεων έδωσε τη δυνατότητα να μπορούν να χρησιμοποιηθούν εναλλακτικά και οι δύο για τον σκοπό της δημιουργίας μιας επιφάνειας.

Στη συνέχεια περιγράφονται οι σχεδιαστικές κλάσεις και η κληρονομικότητά τους, καθώς και ο τρόπος σχεδίασης της κάθε οντότητας, όσον αφορά στον προγραμματισμό τους.

Στο σχήμα 2.30 φαίνονται όλες οι σχεδιαστικές κλάσεις του προγράμματος, δηλαδή οι κλάσεις που περιγράφουν κάποιο γεωμετρικό στοιχείο, όπως σημείο, καμπύλη, κ.λπ. Στο σχήμα φαίνεται επίσης η ιεραρχία αυτών των κλάσεων, με αυτές που βρίσκονται πιο χαμηλά στο δέντρο να είναι αυτές που κληρονομούν τις πιο πάνω.

Τα βέλη στο σχήμα δείχνουν την ιεραρχία. Η αρχή ενός βέλους δείχνει την κλάση που κληρονομείται, ενώ το τέλος του δείχνει την κλάση που κληρονομεί την προηγούμενη. Για παράδειγμα, η κλάση BsplinesCurve κληρονομεί την κλάση GEdge, η οποία με τη σειρά της κληρονομεί την κλάση GShape. Τέλος, η GShape κληρονομεί την κλάση bfc. Η τελευταία, δεν κληρονομεί κάποια άλλη κλάση, αποτελώντας έτσι την πιο βασική κλάση ανάμεσα στις σχεδιαστικές κλάσεις του προγράμματος.

Στη συνέχεια θα αναλύσουμε περαιτέρω τις κλάσεις του σχήματος και θα περιγράψουμε τις μεταξύ τους σχέσεις.



Σχήμα 2.30: Το δέντρο των σχεδιαστικών κλάσεων.

2.4.1: Κλάση bfc

Η κλάση bfc είναι η πιο βασική σχεδιαστική κλάση του προγράμματος και στην ουσία κληρονομείται από όλες τις υπόλοιπες κλάσεις. Η bfc είναι μια αυθαίρετη κλάση (abstract), δηλαδή αντικείμενα αυτής της κλάσης δεν μπορούν ποτέ να δημιουργηθούν απευθείας, όμως μπορούν να δημιουργηθούν δείκτες σε αντικείμενα τύπου bfc ή που είναι αναφορές σε κλάσεις που κληρονομούν την bfc.

Κληρονομείται απευθείας από τις κλάσεις Plane3D, Point3D, GShape.

Ιδιότητες

➤ enum bfc::otype

Αυτή η απαρίθμηση (enumeration) ορίζει τον τύπο του αντικειμένου.

Τίτλος	Τιμή	Περιγραφή
bfc::Pnt	0	Τριδιάστατο καρτεσιανό σημείο, περιγράφεται από την κλάση Point3D.
bfc::BSCurve	1	Καμπύλη BSpline, περιγράφεται από την κλάση BSplineCurve.
bfc::Wir	2	Γραμμή τύπου σύρματος, περιγράφεται από την κλάση Wire.
bfc::Sweep	3	Διεργασία

bfc::Boolean	4	Λογική πράξη μεταξύ αντικειμένων, περιγράφεται από την κλάση BoolOp.
bfc::GPlane	5	Επίπεδο στο χώρο, περιγράφεται από την κλάση Plane3D.
bfc::Sphere	6	Σφαίρα, περιγράφεται από την κλάση pSphere.
bfc::Box	7	Ορθογώνιο παραλληλεπίπεδο, περιγράφεται από την κλάση pBox.
bfc::Circle	8	Κύκλος ή τόξο κύκλου, περιγράφεται από την κλάση pCircle.

- IObject : Handle(AIS_InteractiveObject)

Αυτή η ιδιότητα περιέχει το αλληλεπιδρών αντικείμενο, που φαίνεται στο παράθυρο.

- type : otype

Αυτή η ιδιότητα περιγράφει τον τύπο του αντικειμένου και παίρνει τιμή από το enumeration otype.

- name : QString

Αυτή η ιδιότητα περιέχει το όνομα του αντικειμένου, όπως αυτό εμφανίζεται στο δέντρο των αντικειμένων του εγγράφου.

- visible : bool

Αυτή η ιδιότητα παίρνει την τιμή true αν το αντικείμενο φαίνεται στο Viewer και false αν όχι.

Μέθοδοι

- virtual void Draw()=0

Υπολογίζει το αντικείμενο με βάση τον τύπο του. Η συνάρτηση αυτή παίρνει την τιμή μηδέν, καθιστώντας έτσι την κλάση αυθαίρετη. Ο ορισμός της συνάρτησης γίνεται ξεχωριστά σε κάθε κλάση που κληρονομεί την bfc.

2.4.2: Κλάση Point3D

Η κλάση Point3D είναι εκείνη που περιγράφει ένα τριδιάστατο καρτεσιανό σημείο.

Κληρονομεί την κλάση bfc.

Ιδιότητες

- X, Y, Z : double

Αυτές οι τρεις ιδιότητες κρατάνε την τιμή των X, Y, Z συντεταγμένων του σημείου αντίστοιχα.

- ThePoint : gp_Pnt

Αυτή η μεταβλητή τύπου `gp_Pnt` είναι ένα αντικείμενο της κλάσης `gp_Pnt` της `Open CasCade`, η οποία περιγράφει ένα τριδιάστατο καρτεσιανό σημείο.

Μέθοδοι

➤ `Point3D()`

Κατασκευαστής (constructor) της κλάσης `Point3D`. Αρχικοποιεί τις ιδιότητες της κλάσης. Το σημείο που δημιουργείται από προεπιλογή είναι το (0, 0, 0). Επίσης δίνει τιμές στις ιδιότητες που κληρονομούνται από την `bfc`. Ο τύπος του αντικειμένου (type) είναι `Pnt`.

➤ `Point3D(double c_x, double c_y, double c_z, QString PntName)`

Επιπλέον κατασκευαστής (constructor) της κλάσης `Point3D`. Αρχικοποιεί τις ιδιότητες της κλάσης με τιμές που προέρχονται από τα ορίσματα του. Επίσης δίνει τιμές στις ιδιότητες που κληρονομούνται από την `bfc`. Ο τύπος του αντικειμένου (type) είναι `Pnt`.

➤ `Void Draw()`

Κατασκευάζει το σημείο και δημιουργεί το αλληλεπιδρόν του αντικείμενο, που θα εμφανιστεί στο `Viewer`. Η μέθοδος κληρονομείται από την κλάση `bfc` και ορίζεται ξανά εδώ για τα αντικείμενα τύπου `Point3D`. Για να γίνει ο ορισμός της τοπολογίας του σημείου, η κλάση `gp_Pnt` της `Open CasCade` ορίζει την κλάση `Geom_CartesianPoint`, η οποία ένα σημείο στο χώρο. Η κλάση `Geom_CartesianPoint` κληρονομεί τη γενική κλάση `Geom_Point`, από την οποία ορίζεται η κλάση `AlS_Point`, που απεικονίζει το σημείο στο πρόγραμμα με το σύμβολο «+».

➤ `void Modify(double c_x, double c_y, double c_z, QString PntName)`

Αλλάζει τις τιμές των συντεταγμένων και το όνομα του σημείου με αυτά που δίνονται από τα ορίσματα.

2.4.3: Κλάση `Plane3D`

Η κλάση `Plane3D` ορίζει ένα γενικό επίπεδο στο χώρο. Η κλάση κληρονομείται από δυο άλλες κλάσεις, που ορίζουν το επίπεδο με διαφορετικό τρόπο. Η πληροφορία αποθηκεύεται στη συνέχεια στην κλάση `Plane3D` και ανακτάται σε περίπτωση που χρειαστεί.

Κληρονομεί την κλάση `bfc`.

Κληρονομείται από τις κλάσεις `Pln_T1` και `Pln_T2`.

Ιδιότητες

➤ `plntype: int`

Ορίζει τον τύπο του επιπέδου, που περιγράφεται από το αντικείμενο της κλάσης. Παίρνει την τιμή 0 αν το επίπεδο ορίζεται από τρία σημεία στο χώρο και την τιμή 1 αν περιγράφεται από τη συνάρτηση $Ax + By + Cz = D$.

➤ `Pln: gp_Pln`

Η ιδιότητα αυτή είναι ένα αντικείμενο της κλάσης `gr_Pln` της `Open Cascade`. Η κλάση `gr_Pnt` περιγράφει ένα επίπεδο τοποθετημένο στο χώρο με σύστημα συντεταγμένων τέτοιο ώστε το επίπεδο να ορίζεται από το σημείο εφαρμογής και τους X και Y άξονες του συστήματος συντεταγμένων, το οποίο αποκαλείται «τοπικό σύστημα συντεταγμένων» του επιπέδου.

- `elements: QList<void*>`

Λίστα στην οποία αποθηκεύονται τα δεδομένα που ορίζουν το επίπεδο. Η λίστα αυτή μπορεί να περιέχει στοιχεία οποιασδήποτε μεταβλητής ή αντικείμενα κλάσεων, επιτρέποντας έτσι την αποθήκευση δεδομένων είτε αυτά είναι οι συντελεστές μιας συνάρτησης, είτε αντικείμενα της κλάσης `Point3D`. Όταν το επίπεδο επανασχεδιάζεται η λίστα αυτή συνδυάζεται με την ιδιότητα `plntype` και ανακτώνται τα δεδομένα που όρισαν αρχικά το επίπεδο.

Μέθοδοι

- `Plane3D()`

Ο κατασκευαστής της κλάσης. Ορίζει τις ιδιότητες της κλάσης, που κληρονομούνται από την κλάση `bfc`. Ο τύπος αντικειμένου είναι `Plane`.

- `virtual void Draw()`

Υπολογίζει τη γεωμετρική οντότητα του επιπέδου, ανάλογα με τον τρόπο σχεδίασης. Η μέθοδος αυτή κληρονομείται από τις κλάσεις που βρίσκονται κάτω από την κλάση `Plane3D` στην ιεραρχία και ξαναορίζεται σε αυτές ξεχωριστά.

2.4.4: Κλάση `Pln_T1`

Η κλάση `Pln_T1` ορίζει ένα επίπεδο στο χώρο, το οποίο ορίζεται από τρία σημεία στο χώρο.

Κληρονομεί την κλάση `Plane3D`.

Ιδιότητες

- `*P1, *P2, *P3: Point3D`

Αντικείμενα της κλάσης `Point3D`, που αντιπροσωπεύουν τα σημεία που ορίζουν το επίπεδο.

Μέθοδοι

- `Pln_T1()`

Ο κατασκευαστής του αντικειμένου της κλάσης `Pln_T1`.

- `void Draw()`

Υπολογίζει το επίπεδο με τη χρήση των στοιχείων που έχει δώσει ο χρήστης.

- `void setPnt(int count, Point3D *pnt)`

Ορίζει ένα από τα σημεία που περιγράφουν το επίπεδο. Το σημείο που ορίζεται περιγράφεται από τη μεταβλητή `count`.

2.4.5: Κλάση PIn_T2

Η κλάση PIn_T2 ορίζει ένα επίπεδο στο χώρο, που περιγράφεται από την εξίσωση $Ax + By + Cz = D$.

Κληρονομεί την κλάση Plane3D.

Ιδιότητες

- A, B, C, D: double

Οι συντελεστές της εξίσωσης, που περιγράφουν το επίπεδο.

Μέθοδοι

- PIn_T2()

Ο κατασκευαστής του αντικειμένου της κλάσης PIn_T2. Γίνεται αρχικοποίηση των συντελεστών της συνάρτησης και η αρχική τους τιμή ορίζεται 0.

- void Draw()

Υπολογίζει το επίπεδο με βάση τα δεδομένα που έδωσε ο χρήστης.

2.4.6: Κλάση GShape

Η κλάση GShape ορίζει μια κατηγορία αντικειμένων ως σχήματα. Τα σχήματα μπορεί να είναι οποιοδήποτε είδος τοπολογίας εκτός του σημείου. Η τοπολογία ορίζει τη σχέση μεταξύ απλών γεωμετρικών οντοτήτων, τα οποία μπορούν έτσι να συνδεθούν μεταξύ τους για να δημιουργήσουν περίπλοκα σχήματα. Η κλάση GShape είναι η πιο γενική κλάση τοπολογίας στο λογισμικό, ώστε να περιλαμβάνει όλα τα πιθανά σχήματα. Στο πρόγραμμα δεν δημιουργείται ποτέ αντικείμενο αυτής της κλάσης. Αντίθετα, οι ιδιότητές του χρησιμοποιούνται για να χαρακτηρίσουν τα αντικείμενα των κλάσεων που την κληρονομούν.

Κληρονομεί την κλάση bfc.

Κληρονομείται από τις κλάσεις GWire, GEdge, BoolOp, pBox, pSphere, pSphere.

Ιδιότητες

- Shape : TopoDS_Shape

Η ιδιότητα αυτή είναι ένα αντικείμενο της κλάσης TopoDS_Shape της Open CasCade. Η κλάση TopoDS_Shape είναι η πιο γενική κλάση τοπολογίας της Open CasCade και ορίζει ένα γενικό σχήμα.

2.4.7: Κλάση GEdge

Η κλάση GEdge ορίζει μια κατηγορία αντικειμένων ως ακμές. Μια ακμή είναι ένα στοιχείο τοπολογίας που περιγράφει μια συνεχή γραμμή, χωρίς κρίσιμα σημεία. Ορίζεται από μια καμπύλη, η οποία περιορίζεται από σημεία ή διανύσματα στο μήκος της. Στο πρόγραμμα ποτέ δεν δημιουργείται αντικείμενο της κλάσης GEdge, αντίθετα κληρονομείται από τις κλάσεις που πρέπει να χαρακτηριστούν ως ακμές.

Κληρονομεί την κλάση GShape.

Κληρονομείται από την κλάση BSplineCurve.

Ιδιότητες

- Edge : TopoDS_Edge

Η ιδιότητα αυτή είναι ένα αντικείμενο της κλάσης TopoDS_Edge της Open Cascade. Η κλάση TopoDS_Edge περιγράφει την τοπολογία μιας ακμής.

2.4.8: Κλάση BSplineCurve

Η κλάση BSplineCurve ορίζει μια καμπύλη BSpline στο χώρο.

Κληρονομεί την κλάση GEdge.

Ιδιότητες

- OpenCurve : bool

Παίρνει την τιμή true όταν η καμπύλη είναι μη περιοδική, ενώ false όταν είναι περιοδική.

- Points : QList<Point3D*>

Λίστα αντικειμένων της κλάσης Point3D. Ορίζει τα σημεία ελέγχου της καμπύλης.

- W : QList<double>

Λίστα από μεταβλητές double. Ορίζει τα βάρη των σημείων ελέγχου της καμπύλης. Αυτή η λίστα έχει το ίδιο μέγεθος με τη λίστα Points και το στοιχείο W[i] είναι το βάρος του σημείου Points[i].

- Degree : int

Ορίζει το βαθμό καμπυλότητας της καμπύλης. Έχει ελάχιστη τιμή 1 και μέγιστη τιμή ίση με N-1 όπου N ο αριθμός των σημείων ελέγχου, όπως ορίζεται από τη θεωρία των καμπυλών BSpline.

- Curve : Handle(Geom_BSplineCurve)

Ειδικός δείκτης σε αντικείμενο της κλάσης Geom_BSplineCurve της Open Cascade. Ορίζει την περιγραφή της καμπύλης και τον υπολογισμό της για περαιτέρω χρήση από τις κλάσεις της Open Cascade. Μέσω αυτής ορίζεται το αντικείμενο της κλάσης TopoDS_Edge για αυτήν την καμπύλη.

Μέθοδοι

- void AddPoint(Point3D* pnt,int pos)

Προσθέτει το σημείο pnt στη λίστα Points και στη θέση pos.

- BSplineCurve(bool open)

Ο κατασκευαστής (constructor) του αντικειμένου της κλάσης BSplineCurve. Αν η μεταβλητή open έχει τιμή true, τότε η καμπύλη είναι μη περιοδική, διαφορετικά είναι περιοδική.

- void Draw()

Υπολογίζει την καμπύλη B-Spline. Για την κατασκευή καμπύλης B-Spline γίνεται χρήση της κλάσης `Geom_BSplineCurve`, η οποία δημιουργεί μια καμπύλη B-Spline με χρήση σημείων ελέγχου, που ορίζονται από την κλάση `gp_Pnt`. Η σύνταξη του constructor της κλάσης είναι η ακόλουθη:

```
Geom_BSplineCurve  
(const TColgp_Array1OfPnt& Poles,  
const TColStd_Array1OfReal& Weights,  
const TColStd_Array1OfReal& Knots,  
const TColStd_Array1OfInteger& Multiplicities,  
const Standard_Integer Degree,  
const Standard_Boolean Periodic = Standard_False);
```

- `TColgp_Array1OfPnt& Poles`

Ένας πίνακας που περιέχει τα σημεία ελέγχου. Η κλάση `TColgp_Array1OfPnt` δημιουργεί ένα μονοδιάστατο πίνακα από κλάσεις `gp_Pnt`.

- `TColStd_Array1OfReal& Weights`

Ένας πίνακας από πραγματικούς αριθμούς, που ορίζουν τα βάρη των σημείων ελέγχου της καμπύλης. Ο πίνακας αυτός έχει το ίδιο μέγεθος με τον πίνακα `Poles` και το στοιχείο `Weights[i]` αντιστοιχεί στο βάρος του σημείου `Poles[i]`. Αν κατά τον ορισμό της κλάσης `Geom_BSplineCurve` δεν οριστεί αυτός ο πίνακας, τότε η καμπύλη θεωρείται μη ρητή και όλα τα βάρη λαμβάνονται ίσα με 1.

- `TColStd_Array1OfReal& Knots`

- `TColStd_Array1OfInteger& Multiplicities`

Οι πίνακες `Knots` και `Multiplicities` ορίζουν το διάνυσμα κόμβων της καμπύλης. Ο πίνακας `Knots` περιέχει τους κόμβους, ενώ ο πίνακας `Multiplicities` την πολλαπλότητα του κάθε κόμβου μέσα στο διάνυσμα. Για παράδειγμα το ζεύγος `knots` και `Multiplicities`

(0,1,2,3,4)

(3,1,1,2,3)

θα έδινε το διάνυσμα κόμβων

(0,0,0,1,2,3,3,4,4,4)

Οι δυο πίνακες πρέπει να έχουν το ίδιο μέγεθος, πράγμα που φαίνεται από τον ορισμό τους.

- `Standard_Integer Degree`

Ο βαθμός της καμπύλης ως ακέραιος αριθμός. Ο βαθμός της καμπύλης δεν μπορεί να είναι μικρότερος από ένα και μεγαλύτερος από τον αριθμό των σημείων ελέγχου μείον ένα.

- `Standard_Boolean Periodic`

Η μεταβλητή αυτή ορίζει αν η καμπύλη είναι περιοδική. Αν είναι true, η καμπύλη είναι περιοδική και false αν όχι. Από προεπιλογή έχει τιμή false.

2.4.9: Κλάση pCircle

Η κλάση pCircle περιγράφει ένα κύκλο ή τόξο κύκλου.

Κληρονομεί την κλάση GEdge.

Ιδιότητες

- Circtype: int

Ακέραιος αριθμός, που υποδηλώνει αν το αντικείμενο είναι κύκλος ή τόξο κύκλου. Αν έχει τιμή 0 το αντικείμενο είναι κύκλος. Αν έχει τιμή 1 το αντικείμενο είναι τόξο κύκλου, ορισμένο από τις γωνίες αρχής και τέλους. Αν έχει την τιμή 2 το αντικείμενο είναι τόξο κύκλου, που περνά από τρία σημεία και περιορίζεται από το πρώτο και τρίτο σημείο.

- *pnt1, *pnt2, *pnt3: Point3D

Αντικείμενα τύπου Point3D, που ορίζουν τα σημεία από τα οποία περνά ο κύκλος.

- start, end: double

Οι γωνίες αρχής και τέλους, στην περίπτωση που το αντικείμενο είναι τόξο κύκλου και το μήκος του ορίζεται από γωνίες. Οι δυο γωνίες μετρούνται ως προς το πρώτο σημείο που ορίζει το τόξο και προς τις δυο αντίθετες κατευθύνσεις σχεδίασής του.

Μέθοδοι

- pCircle()

Ο κατασκευαστής ενός αντικειμένου της κλάσης pCircle. Εδώ ορίζονται οι ιδιότητες της κλάσης, που κληρονομούνται από την bfc. Ο τύπος του αντικειμένου είναι Circle. Επίσης αρχικοποιείται η ιδιότητα Circtype.

- virtual void Draw()

Υπολογίζει τον κύκλο ή το τόξο κύκλου, ανάλογα με την τιμή που έχει η μεταβλητή Circtype. Η συνάρτηση αυτή κληρονομείται από την κλάση bfc.

2.4.10: Κλάση GWire

Η κλάση Wire περιγράφει αντικείμενα τύπου «σύρματος».

Κληρονομεί την κλάση GShape.

Ιδιότητες

- TWire : TopoDS_Wire

Η ιδιότητα αυτή είναι ένα αντικείμενο της κλάσης TopoDS_Wire της Open CasCade. Η κλάση TopoDS_Wire περιγράφει την τοπολογία ενός σύρματος.

- edges : QList<GEdge*>

Λίστα αντικειμένων της κλάσης GEdge. Αυτή η ιδιότητα περιέχει τις ακμές από τις οποίες αποτελείται το «σύρμα».

Μέθοδοι

- Wire()

Ο κατασκευαστής (constructor) της κλάσης Wire. Δημιουργεί ένα κενό αντικείμενο Wire.

- bool AddCurve(GEdge *curv)

Προσθέτει το αντικείμενο της κλάσης GEdge στο τέλος της λίστας edges.

- void Draw()

Υπολογίζει το «σύρμα», προσθέτοντας σε ένα κενό σύρμα τις ακμές που υπάρχουν στη λίστα edges.

2.4.11: Κλάση Sweep

Η κλάση Sweep περιγράφει μια διαδικασία «οδήγησης» για τη δημιουργία επιφάνειας ή όγκου.

Κληρονομεί την κλάση GShape.

Ιδιότητες

- spine : GWire*

Ο οδηγός του σχήματος.

- profile : GShape *

Η γενέτειρα του σχήματος.

- prname : QString

Το όνομα του αντικειμένου, που επιλέχθηκε ως γενέτειρα του σχήματος.

- spname : QString

Το όνομα του αντικειμένου, που επιλέχθηκε ως οδηγός του σχήματος.

Μέθοδοι

- Sweep()

Ο κατασκευαστής (constructor) της κλάσης Sweep. Δημιουργεί ένα κενό αντικείμενο Sweep.

- void Draw()

Υπολογίζει το σχήμα Sweep με τον οδηγό και τη γενέτειρα που έχουν δοθεί.

- void SetValues(GWire* sp, GShape *pr)

Ορίζει τον οδηγό sp και τη γενέτειρα pr του σχήματος.

- void ChangeProfile(GShape *pr)

Η μέθοδος αυτή έχει προστεθεί για ευκολία. Αλλάζει τη γενέτειρα του σχήματος με το αντικείμενο GShape pr.

- void ChangeSpine(GWire* sp)

Η μέθοδος αυτή έχει προστεθεί για ευκολία. Αλλάζει τον οδηγό του σχήματος με το αντικείμενο GWire sp.

2.4.12: Κλάση pBox

Η κλάση pBox περιγράφει τη δημιουργία ενός ορθογώνιου παραλληλεπίπεδου.

Κληρονομεί την κλάση GShape.

Ιδιότητες

- sPln : Plane3D *

Η ιδιότητα αυτή ορίζει το επίπεδο αναφοράς του παραλληλεπίπεδου. Το τοπικό τρισσορθογώνιο σύστημα του παραλληλεπίπεδου είναι ίδιο με εκείνο του επιπέδου αναφοράς. Από αυτό ορίζεται ο προσανατολισμός και η θέση του αντικειμένου.

- DX, DY, DZ : double

Οι τρεις αυτές ιδιότητες ορίζουν το μέγεθος του ορθογώνιου παραλληλεπίπεδου. Η εφαρμογή τους γίνεται πάνω στο τοπικό τρισσορθογώνιο σύστημα του παραλληλεπίπεδου.

Μέθοδοι

- pBox()

Ο κατασκευαστής (constructor) της κλάσης pBox. Δημιουργεί ένα κενό αντικείμενο της κλάσης.

- void Draw()

Υπολογίζει το ορθογώνιο παραλληλεπίπεδο δεδομένου των ιδιοτήτων του που πρέπει να έχουν ήδη ορισθεί.

2.4.13: Κλάση pSphere

Η κλάση pSphere περιγράφει μια σφαίρα στο χώρο.

Κληρονομεί την κλάση GShape.

Ιδιότητες

- Center : Point3D *

Ορίζει το κέντρο της σφαίρας.

- R : double

Ορίζει την ακτίνα της σφαίρας.

Μέθοδοι

- pSphere()

Ο κατασκευαστής (constructor) της κλάσης pSphere. Δημιουργεί ένα κενό αντικείμενο της κλάσης pSphere.

- void Draw()

Υπολογίζει τη σφαίρα με δεδομένα το κέντρο της και την ακτίνα της, που πρέπει να έχουν ορισθεί από πριν. Για τον ορισμό της σφαίρας γίνεται χρήση της κλάσης BRepPrimAPI_MakeSphere της Open CasCade.

2.4.14: Κλάση BoolOp

Η κλάση BoolOp κατασκευάζει σχήματα, που προκύπτουν από λογικές πράξεις μεταξύ δύο σχημάτων. Τα σχήματα που λαμβάνουν μέρος στην πράξη είναι αντικείμενα της κλάσης GShape.

Κληρονομεί την κλάση GShape.

Ιδιότητες

- Shape1, Shape2 : GShape *

Αντικείμενα τύπου GShape τα οποία παίρνουν μέρος στην πράξη.

- s1name, s2name : QString

Τα ονόματα των σχημάτων που λαμβάνουν μέρος στην πράξη, όπως αυτά φαίνονται στο δέντρο των αντικειμένων.

- btype : int

Ακέραιος, ο οποίος υποδηλώνει το είδος της πράξης που αντιπροσωπεύει το συγκεκριμένο αντικείμενο BoolOp.

Μέθοδοι

- BoolOp()

Ο κατασκευαστής του αντικειμένου της κλάσης BoolOp. Γίνεται αρχικοποίηση των ιδιοτήτων, που κληρονομούνται από την κλάση bfc. Ο τύπος του αντικειμένου είναι Boolean.

- void Draw()

Υπολογίζει το αποτέλεσμα της πράξης, δεδομένου του τύπου της πράξης. Στην περίπτωση που η πράξη είναι η αποκοπή (Cut) τότε αν το σχήμα που κάνει την αφαίρεση είναι επιφάνεια, πριν την πράξη μετατρέπεται σε στερεό.

- void ChangeShape1(GShape *sh1)

Ορίζει το σχήμα που θα είναι πρώτο στην πράξη. Στην περίπτωση της αποκοπής είναι το σχήμα από το οποίο αφαιρείται υλικό.

- void ChangeShape2(GShape *sh2)

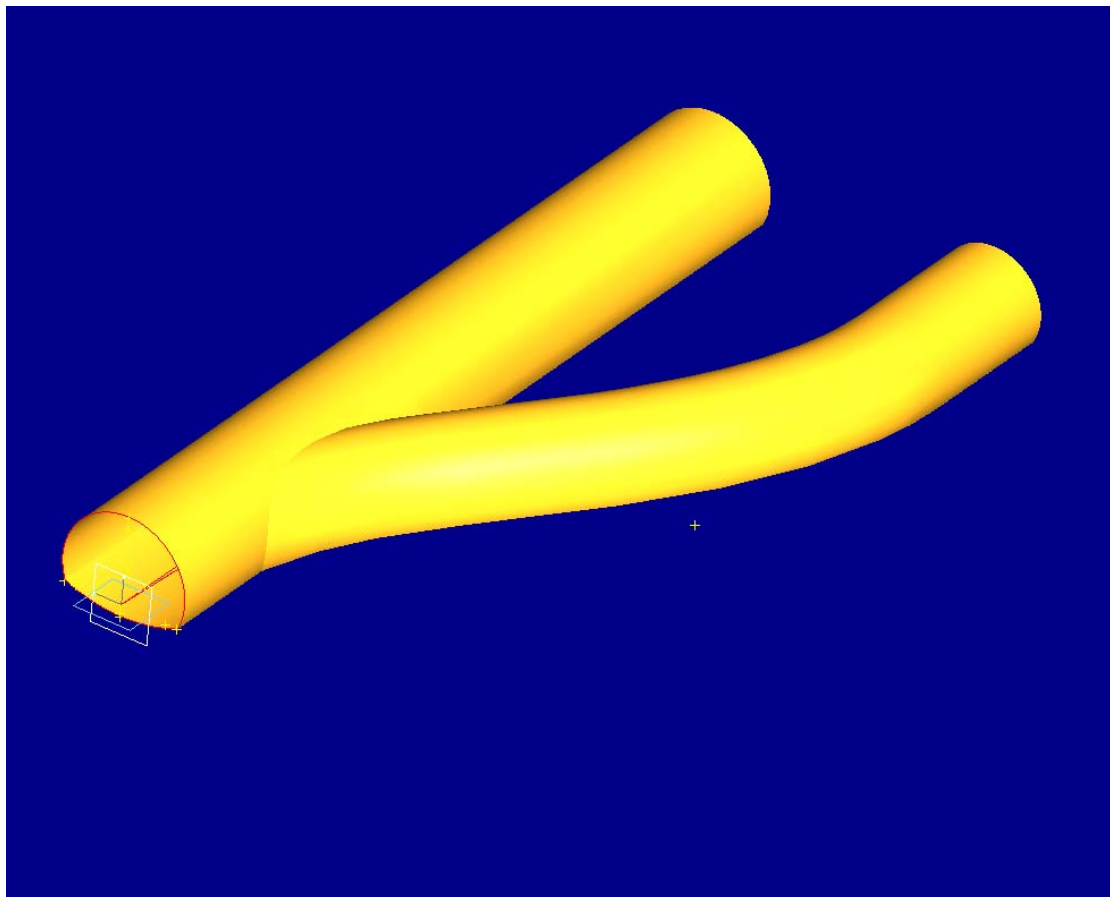
Ορίζει το σχήμα που θα είναι δεύτερο στην πράξη. Στην περίπτωση της αποκοπής είναι το σχήμα που αφαιρεί υλικό.

Κεφάλαιο 3^ο: Παραδείγματα

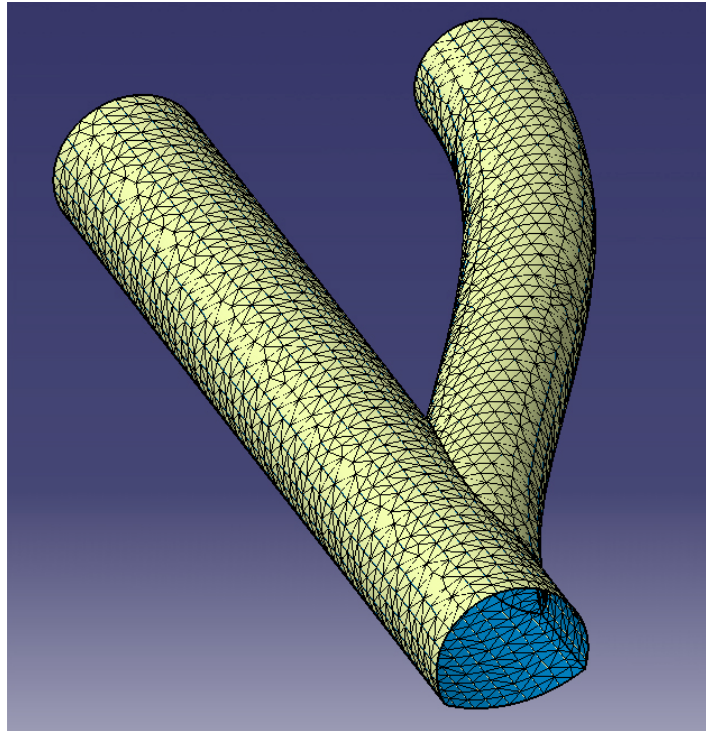
Σε αυτό το κεφάλαιο παρουσιάζονται παραδείγματα γεωμετριών, που σχεδιάστηκαν με τη χρήση του αναπτυχθέντος λογισμικού. Επίσης θα γίνει λεπτομερής περιγραφή της δημιουργίας ενός από τα παραδείγματα, που περιέχονται στο κεφάλαιο αυτό.

3.1: Σύστημα σηράγγων που διακλαδίζονται

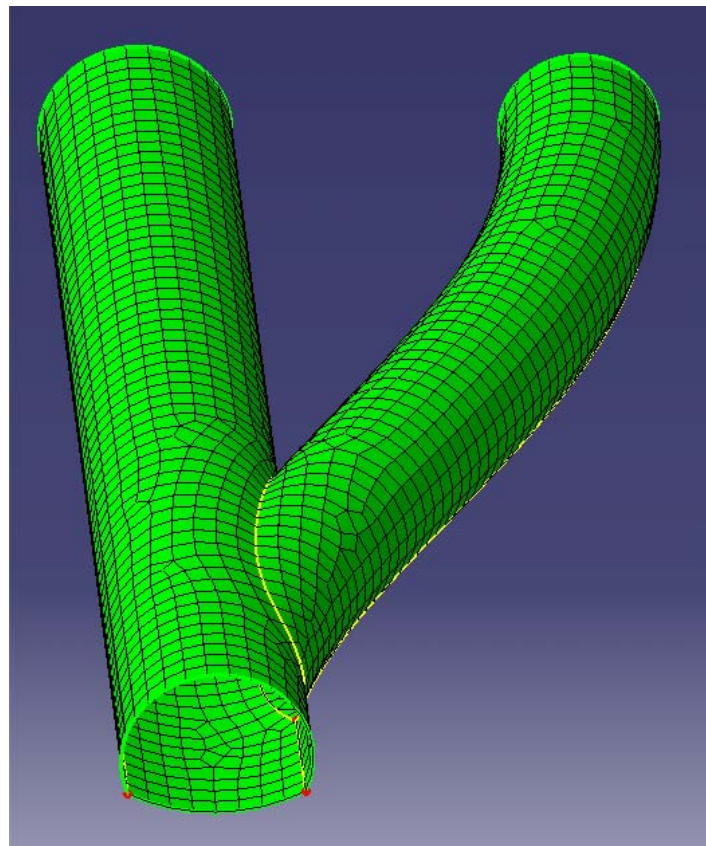
Στις Εικόνες 3.1, 3.2 και 3.3 παρουσιάζεται μια σύνθεση δύο σηράγγων οι οποίες τέμνονται ώστε να δημιουργούν ένα σχήμα τύπου Υ. Οι δυο σήραγγες έχουν διαφορετικές διατομές. Μετά τη σχεδίαση έγινε εξαγωγή του σχήματος σε αρχείο τύπου IGES, ώστε να εισαχθεί σε πρόγραμμα επιφανειακού πλέγματος. Το αποτέλεσμα αυτής της διεργασίας φαίνεται στις εικόνες 3.2 και 3.3 με δυο διαφορετικούς τύπους επιφανειακού πλέγματος.



Εικόνα 3.1: Σύνθεση σηράγγων που τέμνονται σε μορφή Υ.



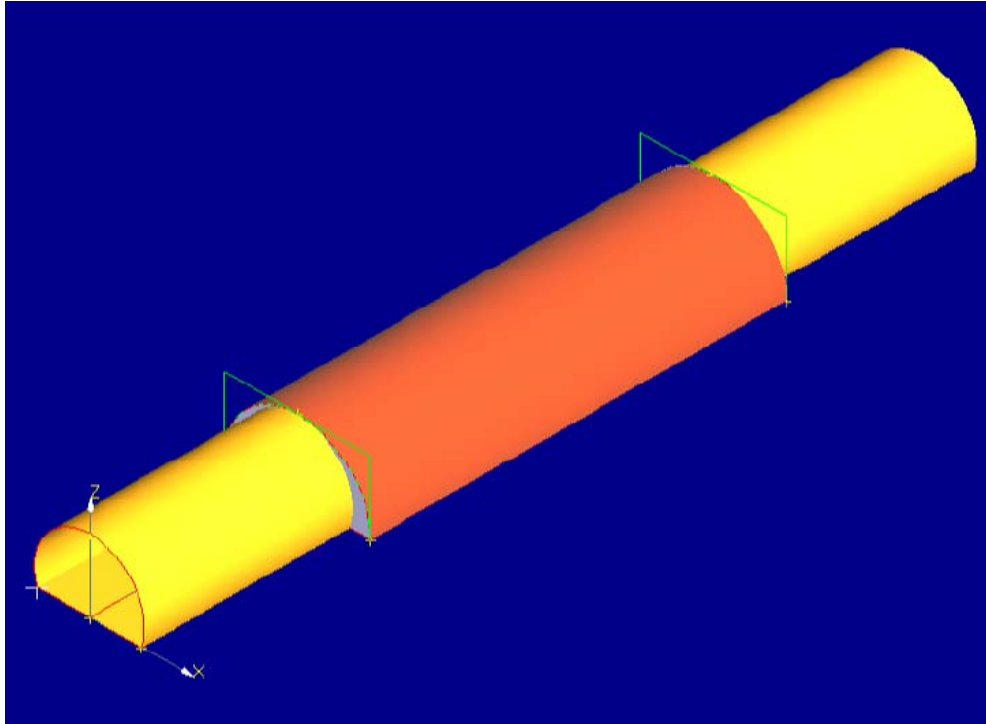
Εικόνα 3.2: Τριγωνικό επιφανειακό πλέγμα στη σύνθεση σηράγγων της εικόνας 3.1. Η παραγωγή του πλέγματος έγινε με χρήση διαφορετικού λογισμικού.



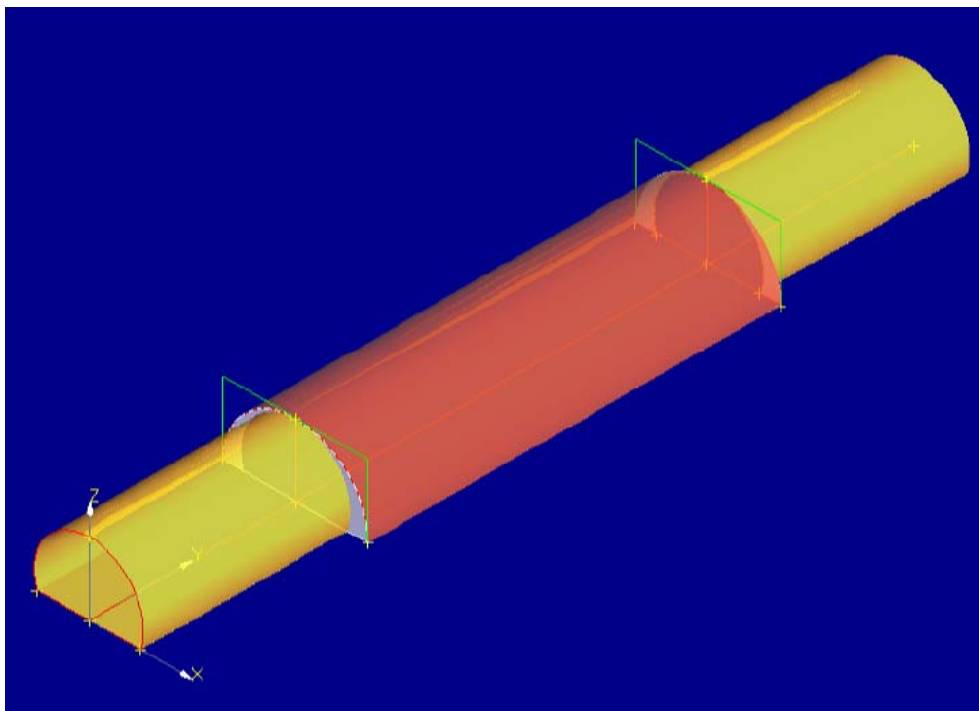
Εικόνα 3.3: Τετραπλευρικό επιφανειακό πλέγμα στη σύνθεση σηράγγων της εικόνας 3.1. Η παραγωγή του πλέγματος έγινε με χρήση διαφορετικού λογισμικού.

3.2: Σήραγγα με ενδιάμεση πλάτυνση.

Στις εικόνες 3.4 και 3.5 παρουσιάζεται μια σήραγγα, στην οποία δημιουργήθηκε πλάτυνση στο μέσο της. Δύο διαφορετικές διατομές χρησιμοποιήθηκαν για τον ορισμό των τριών διαδοχικών τμημάτων της σήραγγας.



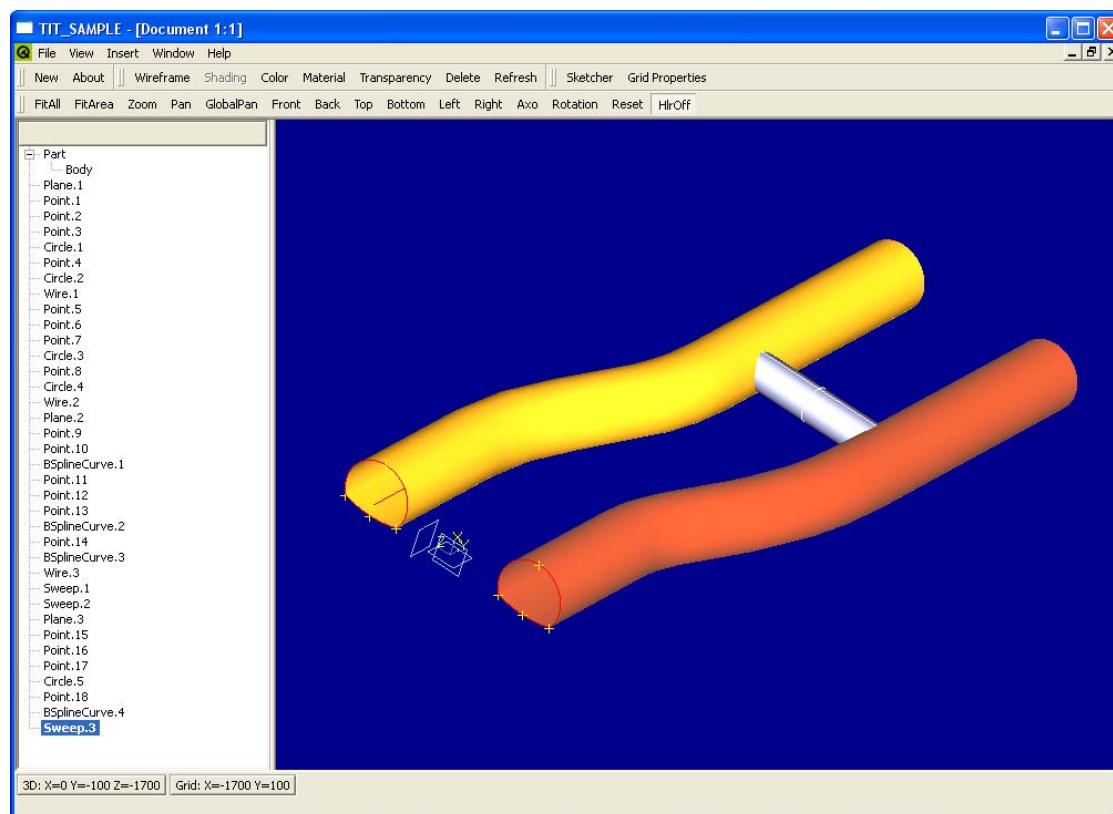
Εικόνα 3.4: Σήραγγα με ενδιάμεση πλάτυνση.



Εικόνα 3.5: Σήραγγα με ενδιάμεση πλάτυνση. Έχει γίνει χρήση διαφάνειας για καλύτερη προβολή του εσωτερικού της.

3.2: Σύστημα παράλληλων σηράγγων με βοηθητική σήραγγα

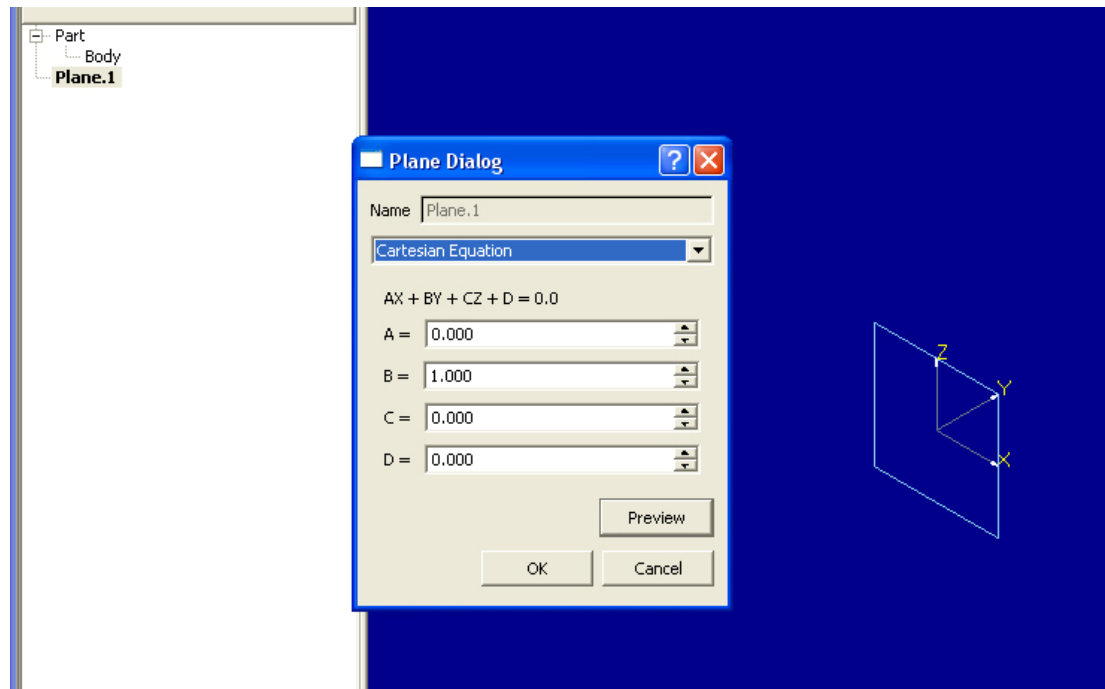
Στην Εικόνα 3.6 παρουσιάζεται ένα σύστημα παράλληλων σηράγγων, οι οποίες συνδέονται με μία βοηθητική σήραγγα. Οι δυο βασικές σήραγγες έχουν όμοιες διατομές ενώ επιτυγχάνεται παραλληλία χρησιμοποιώντας την ίδια καμπύλη ως οδηγό για τη δημιουργία τους. Η βοηθητική σήραγγα έχει μικρότερη διατομή από τις βασικές. Στη συνέχεια θα γίνει περιγραφή της διαδικασίας σχεδίασης αυτής της σύνθεσης.



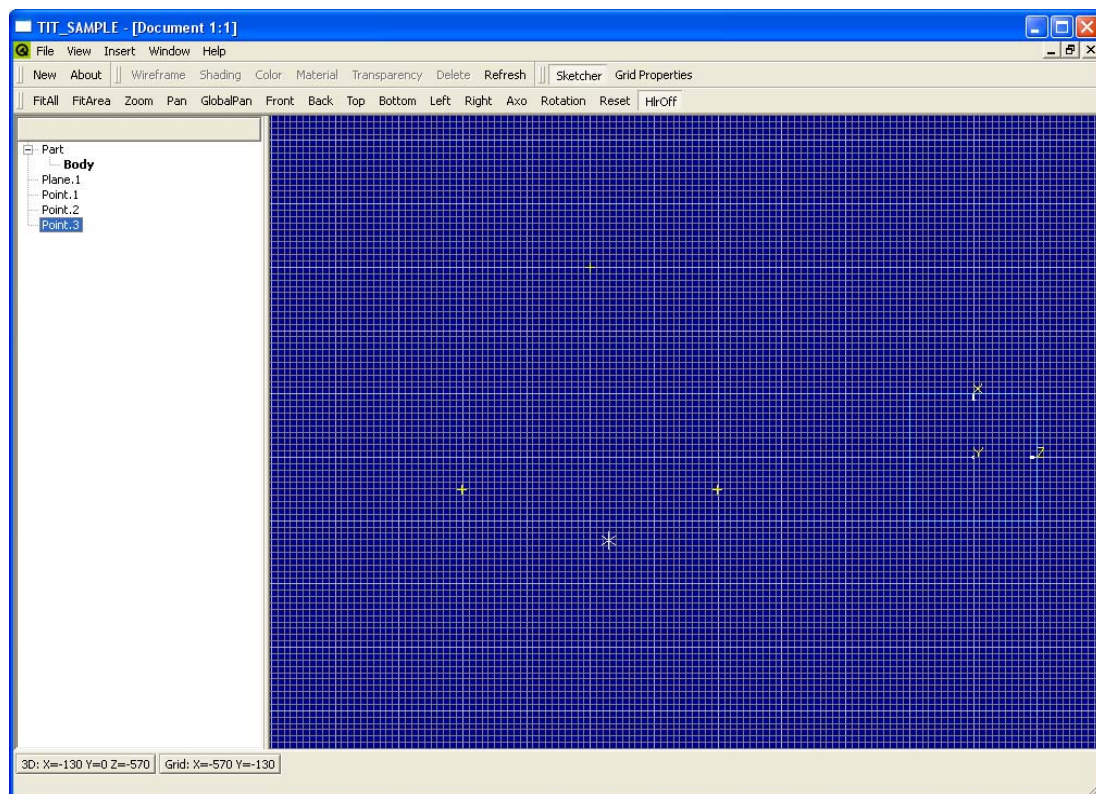
Εικόνα 3.6: Σύνθεση παράλληλων σηράγγων, που συνδέονται με βοηθητική σήραγγα. Διακρίνεται αριστερά το δέντρο των αντικειμένων.

Το πρώτο βήμα στη σχεδίαση είναι ο ορισμός ενός επιπέδου, πάνω στο οποίο θα πραγματοποιηθεί η σχεδίαση των διατομών των σηράγγων (Εικόνα 3.7).

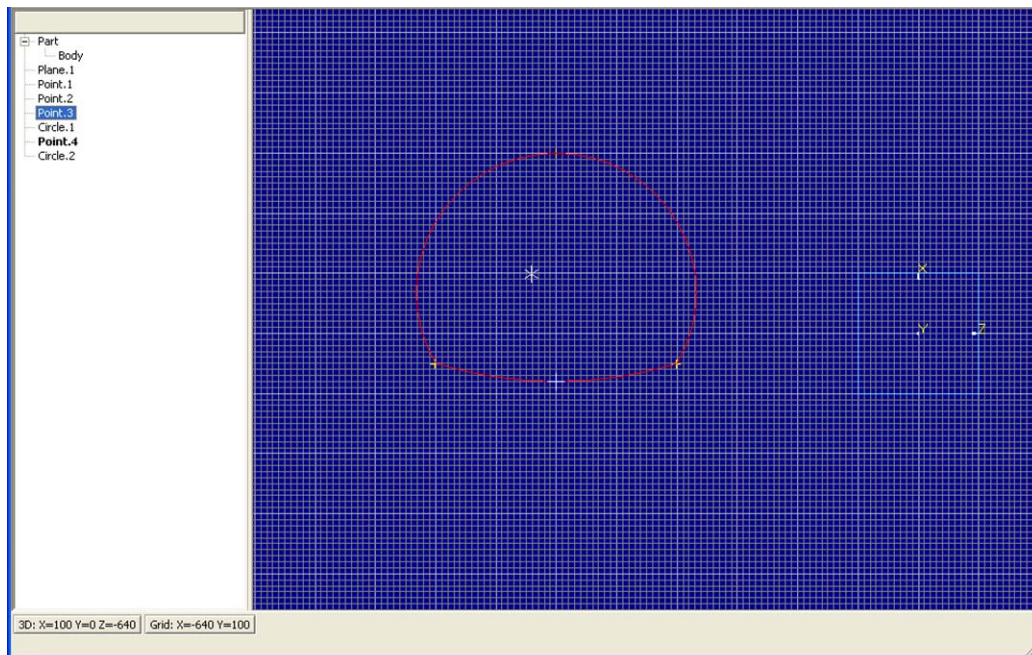
Αφού ορισθεί το αρχικό επίπεδο σχεδίασης μπαίνουμε στη λειτουργία διδιάστατης σχεδίασης για να ορίσουμε τις διατομές των σηράγγων. Αρχικά θα ορίσουμε ένα αριθμό σημείων. Στη συνέχεια θα δημιουργήσουμε ένα αριθμό από τόξα κύκλου βασισμένα στα σημεία αυτά. Τα τόξα κύκλου θα βοηθήσουν στη δημιουργία των διατομών των σηράγγων.



Εικόνα 3.7: Ορισμός αρχικού επιπέδου σχεδίασης.

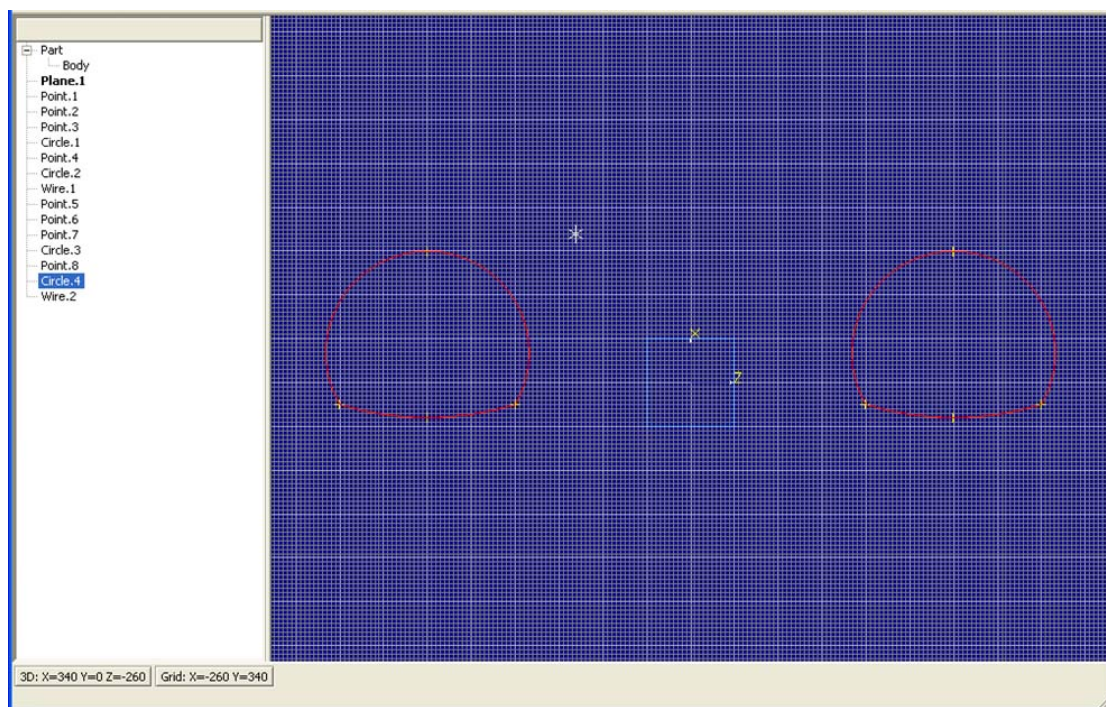


Εικόνα 3.8: Ορισμός σημείων στο περιβάλλον διάστατης σχεδίασης.

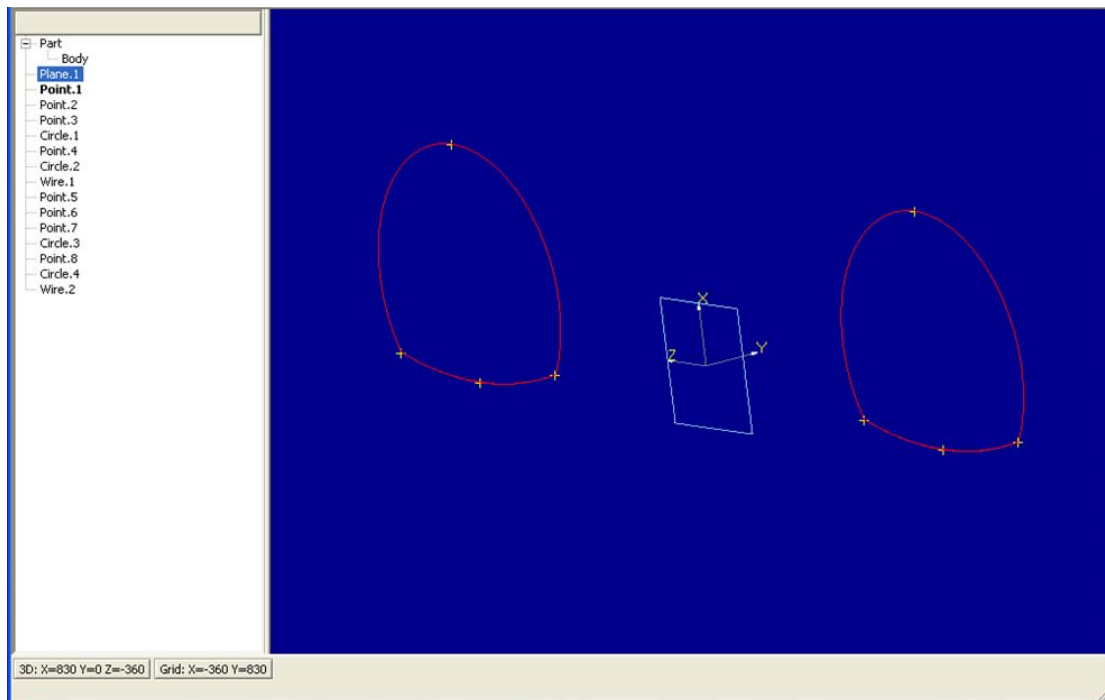


Εικόνα 3.9: Δημιουργία τόξων κύκλου, που θα αποτελέσουν τις διατομές των σηράγγων.

Για κάθε σήραγγα χρειάζεται να δημιουργήσουμε διαφορετική γενέτειρα. Αφού δημιουργηθούν τα τόξα κύκλου ενώνονται για να δημιουργηθούν «σύρματα», που θα αποτελούν τις γενέτειρες των σηράγγων (Εικόνες 3.9 και 3.10).

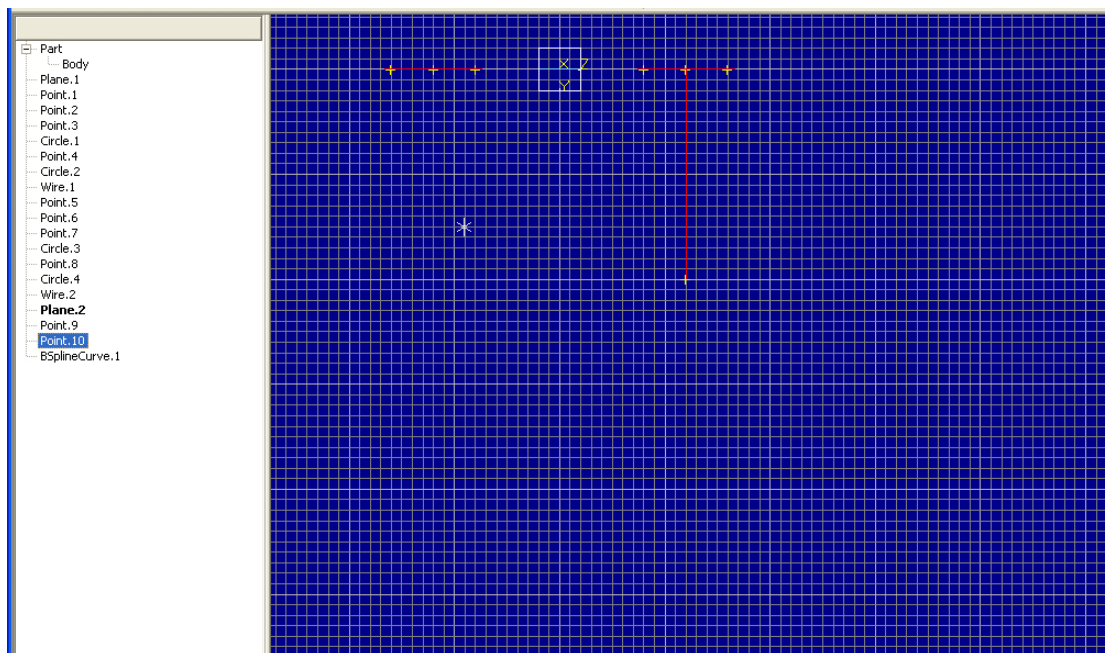


Εικόνα 3.10: Δημιουργία «συρμάτων» από τόξα κύκλου που αποτελούν τις γενέτειρες των σηράγγων.

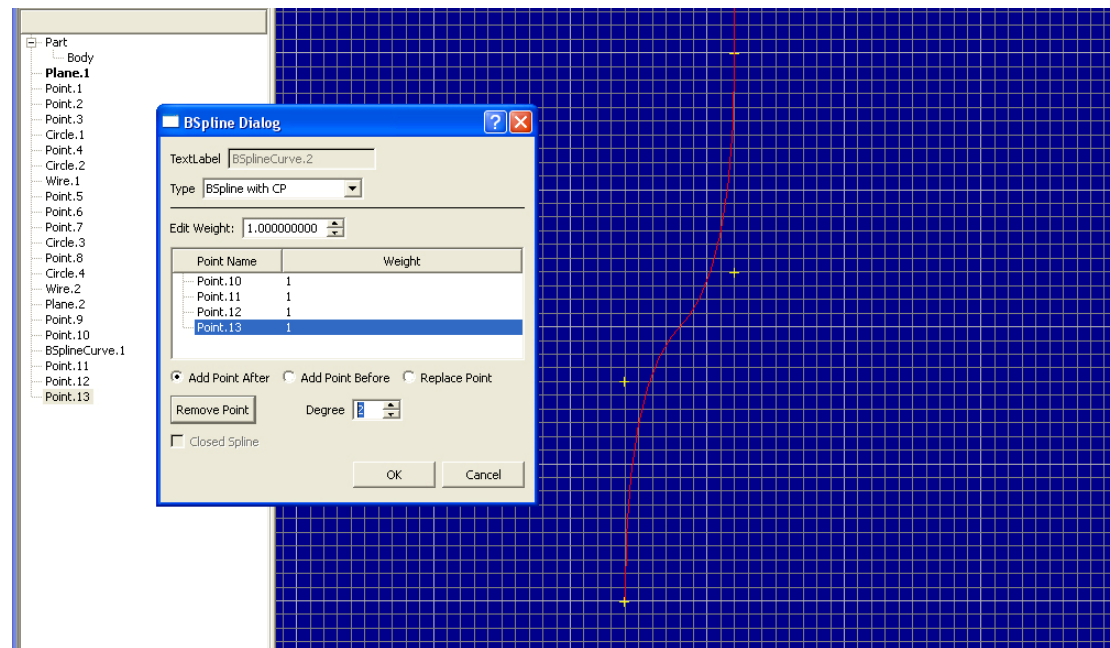


Εικόνα 3.11: Οι γενέτειρες των σηράγγων, όπως φαίνονται στο τριδιάστατο περιβάλλον σχεδίασης.

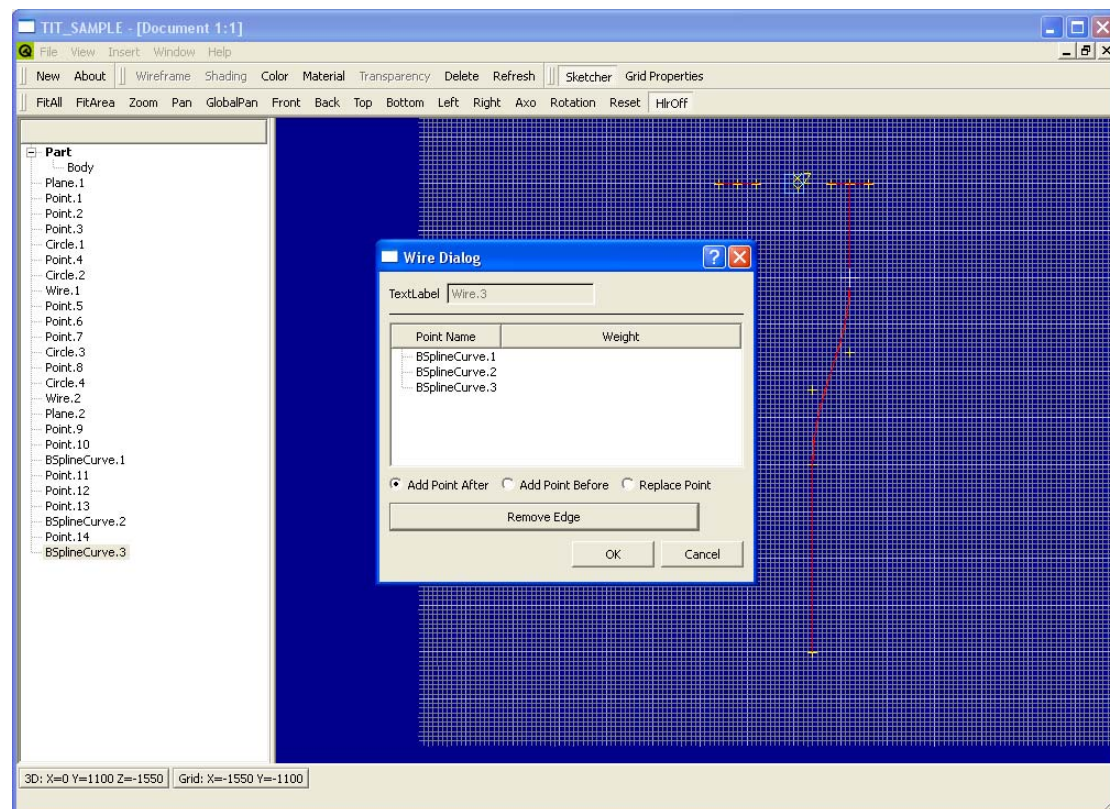
Το επόμενο βήμα είναι ο ορισμός του οδηγού των σηράγγων. Για να γίνει αυτό ορίζουμε ένα επίπεδο κάθετο στο αρχικό και μπαίνουμε ξανά στο διδιάστατο περιβάλλον σχεδίασης. Η γενέτειρα αποτελείται από τρία μέρη, τα οποία στη συνέχεια ενώνονται για να αποτελέσουν ένα «σύρμα».



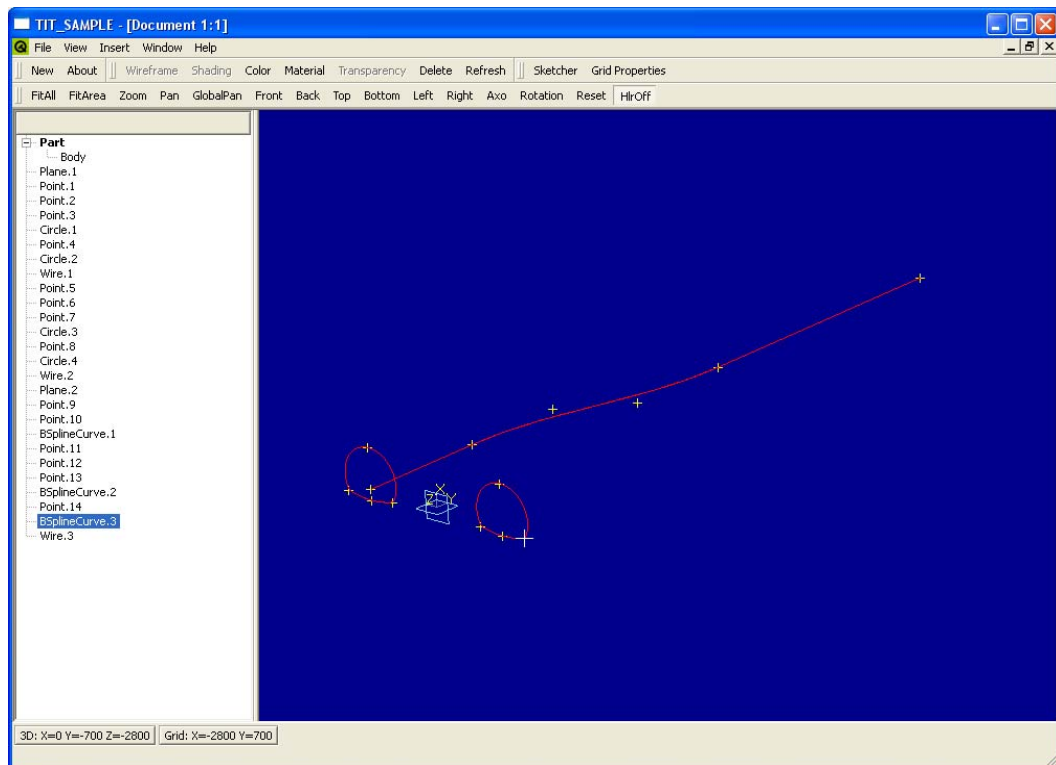
Εικόνα 3.12: Δημιουργία του πρώτου μέρους των οδηγών των σηράγγων (ευθύγραμμο τμήμα).



Εικόνα 3.13: Δημιουργία του δεύτερου μέρους του οδηγού των σηράγγων (καμπύλη B-Spline).

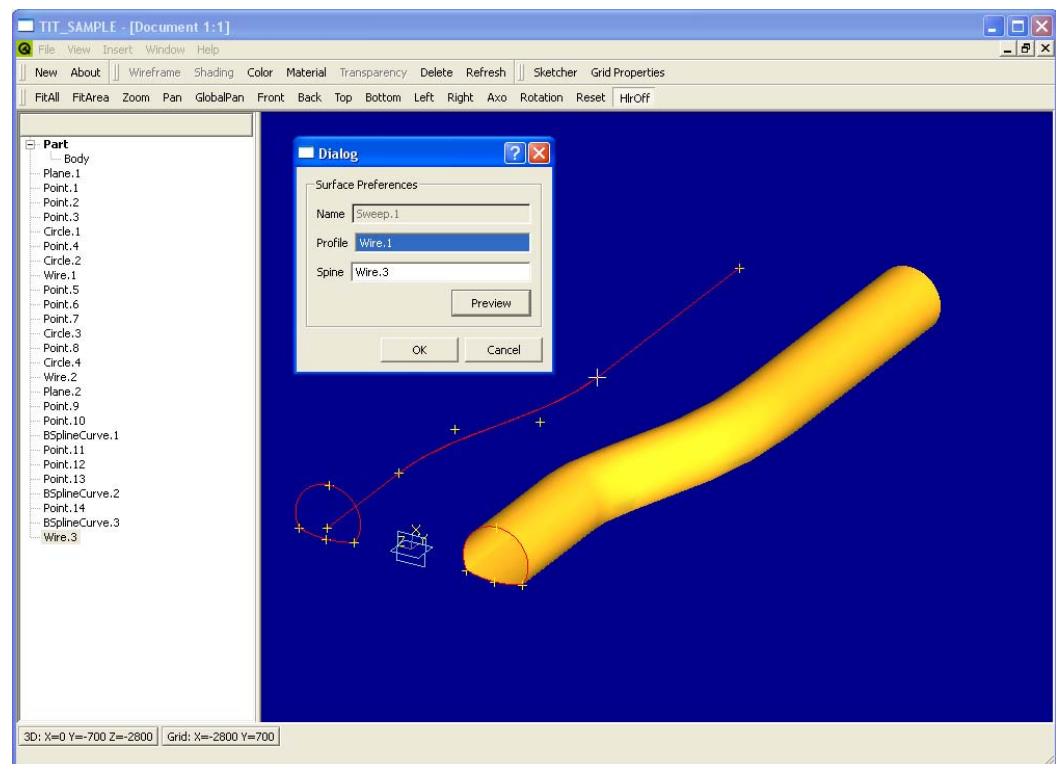


Εικόνα 3.14: Ένωση των τμημάτων, που αποτελούν τον οδηγό των σηράγγων ώστε να δημιουργηθεί ένα σύρμα.

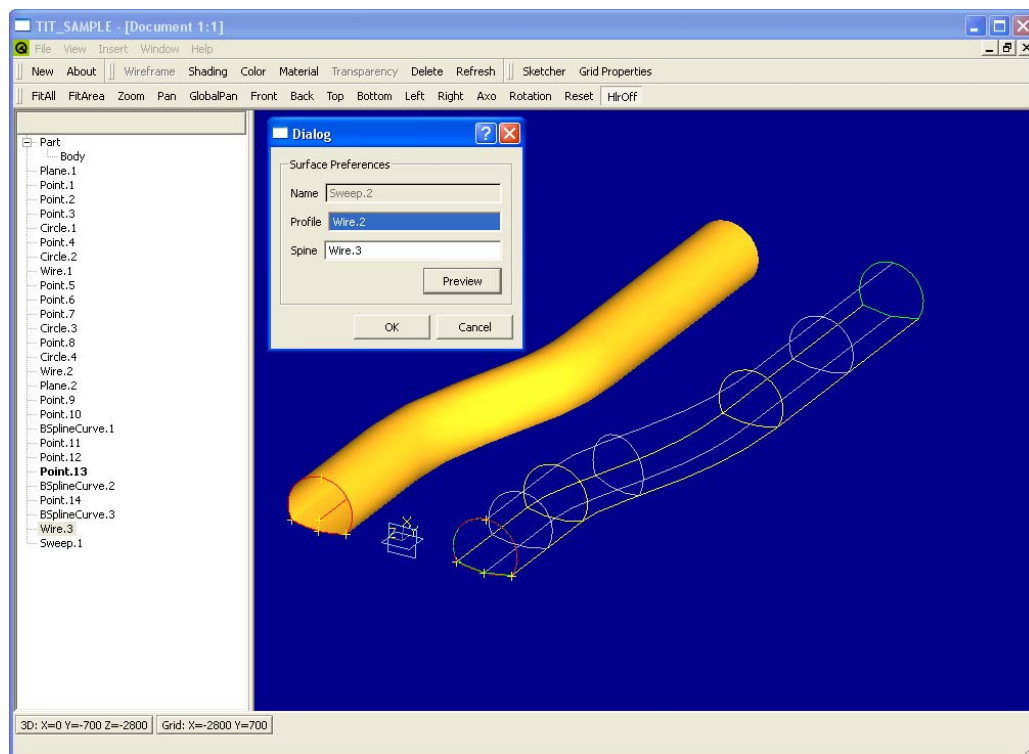


Εικόνα 3.15: Οι γενέτειρες των σηράγγων και ο οδηγός τους στο τρισδιάστατο περιβάλλον σχεδίασης.

Με χρήση των γενετειρών και του οδηγού, ορίζουμε τις σήραγγες, που είναι επιφάνειες οδήγησης (Εικόνες 3.16, 3.17).

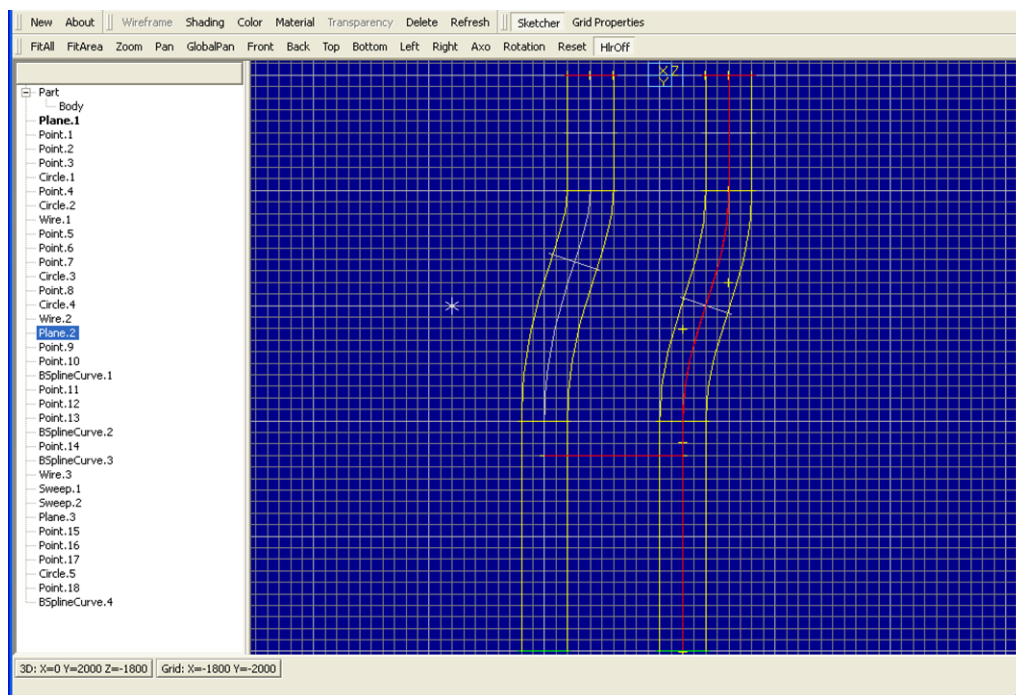


Εικόνα 3.16: Δημιουργία της πρώτης σήραγγας.

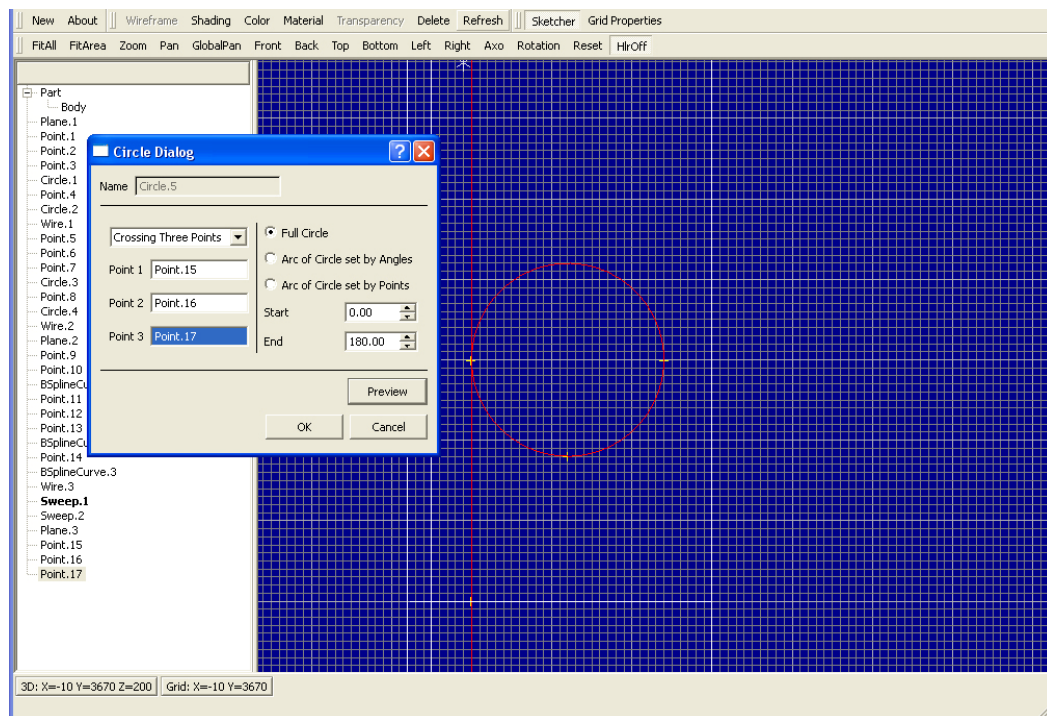


Εικόνα 3.17: Δημιουργία της δεύτερης σήραγγας.

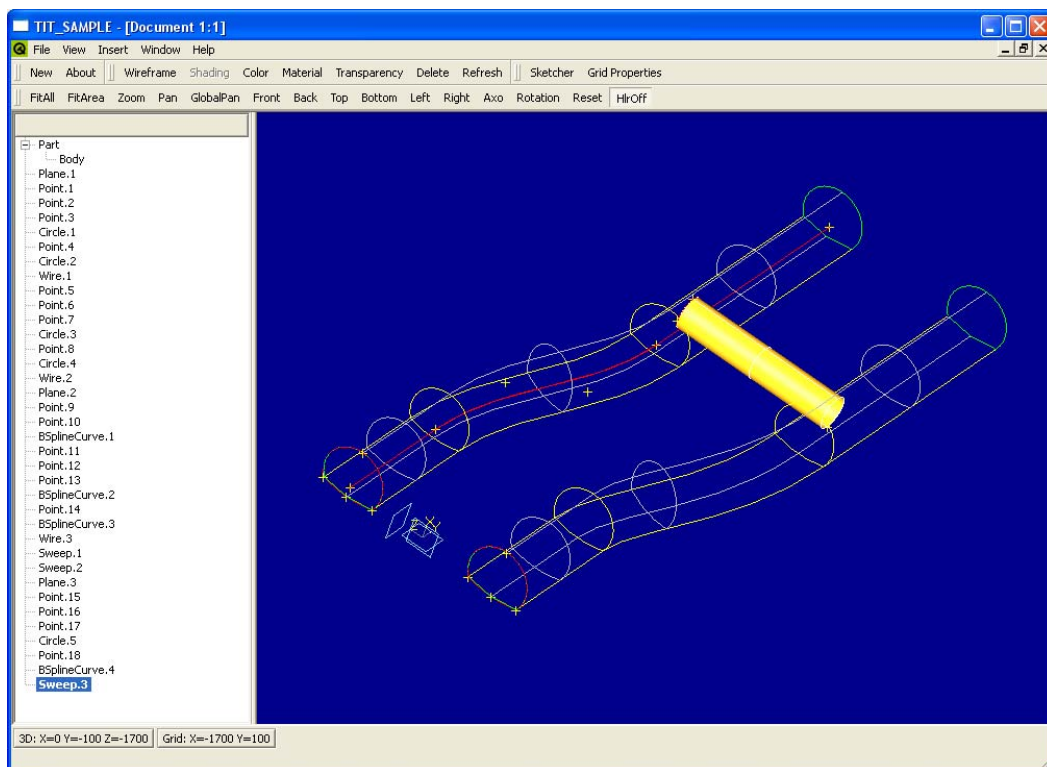
Για να δημιουργήσουμε τη βοηθητική σήραγγα ακολουθούμε παρόμοια διαδικασία. Αρχικά μπαίνουμε σε περιβάλλον διδιάστατης σχεδίασης (Εικόνα 3.18) για να ορίσουμε τον οδηγό και στη συνέχεια αλλάζουμε επίπεδο σχεδίασης για να ορίσουμε τη γενέτειρα. Επιλέγουμε η γενέτειρα να αποτελείται από ένα κύκλο μόνο (Εικόνα 3.19).



Εικόνα 3.18: Ορισμός του οδηγού της βοηθητικής σήραγγας.

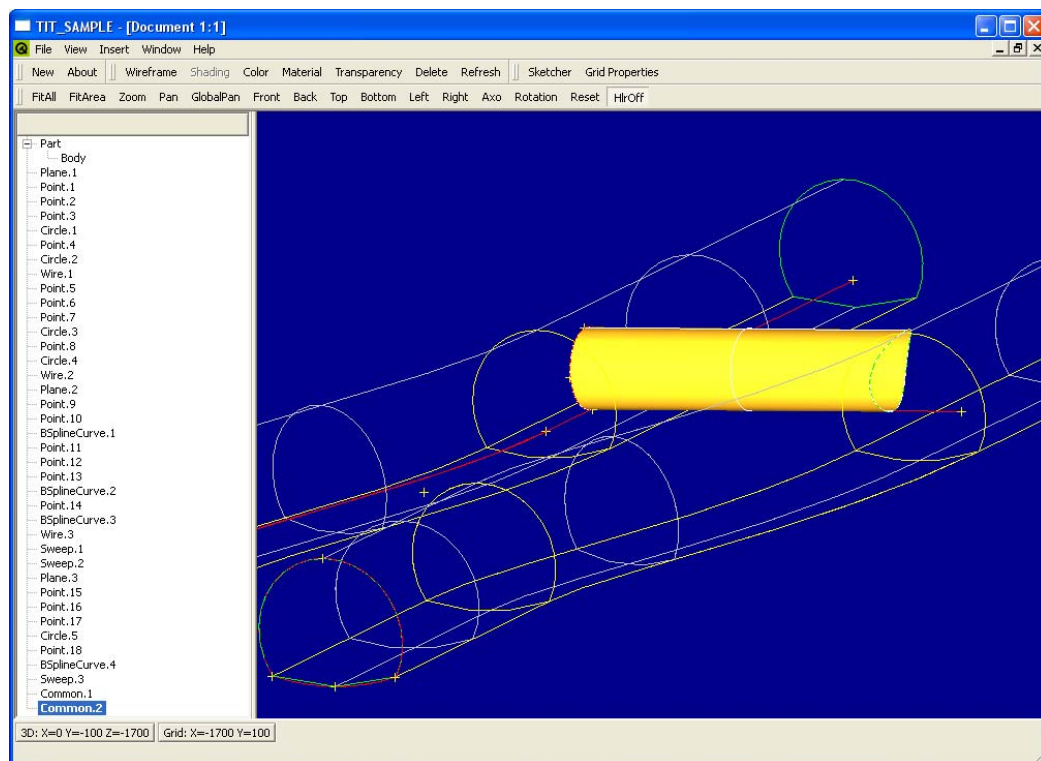


Εικόνα 3.19: Ορισμός της γενέτειρας της βοηθητικής σήραγγας.

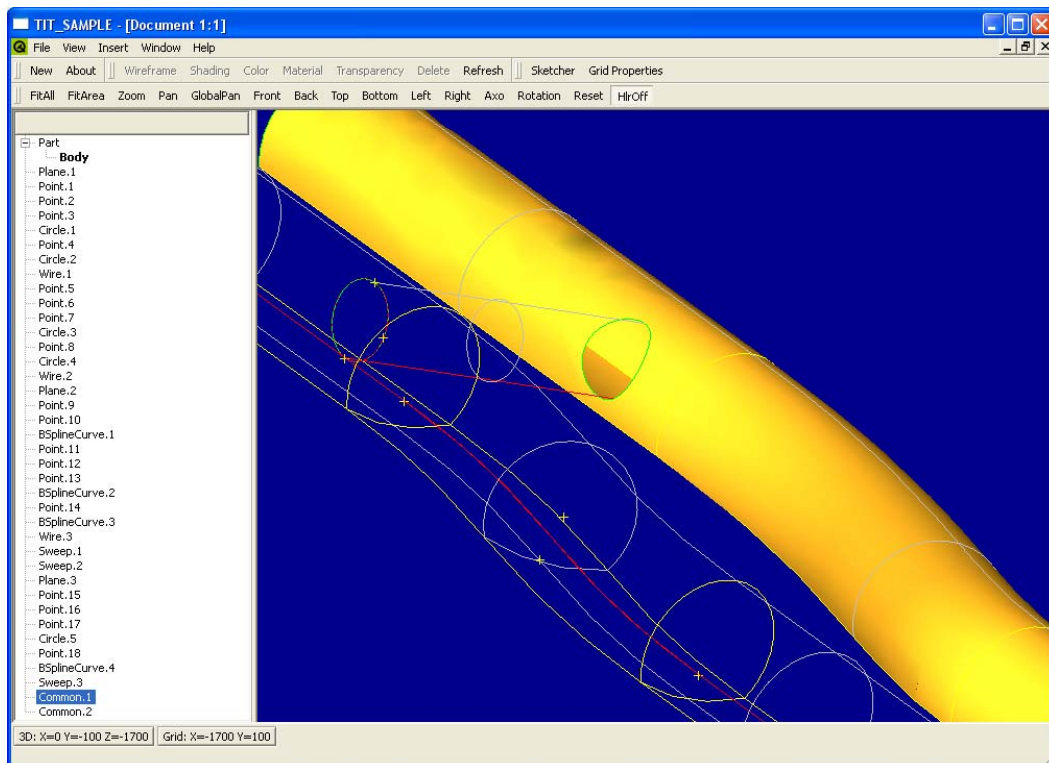


Εικόνα 3.20: Ορισμός της βοηθητικής σήραγγας.

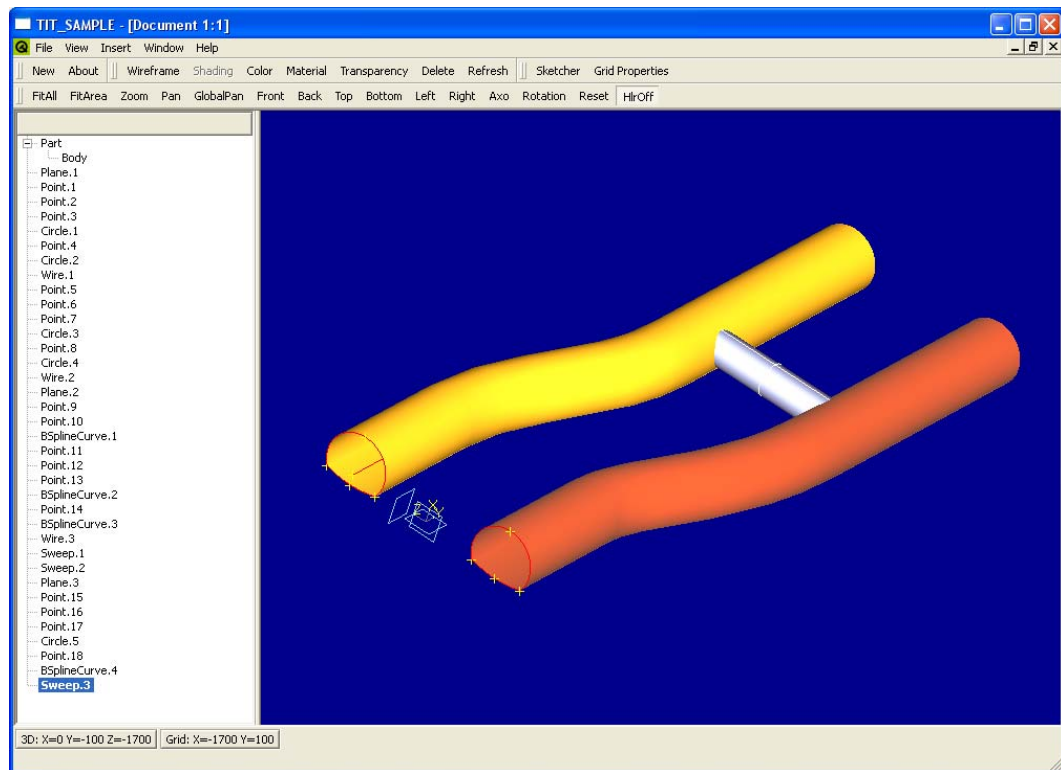
Το τελευταίο βήμα στη σχεδίαση θα είναι η αποκοπή των σχημάτων που τέμνονται μεταξύ τους, για να δημιουργήσουμε το τελικό σχήμα (Εικόνες 3.21-3.22).



Εικόνα 3.21: Αποκοπή βοηθητικού σχήματος από τη μία από τις δυο σήραγγες.



Εικόνα 3.22: Αποκοπή της μιας σήραγγας από τη βοηθητική.



Εικόνα 3.23: Η τελική σύνθεση. Κάθε διαφορετική επιφάνεια χρωματίστηκε με άλλο χρώμα για να ξεχωρίζουν μεταξύ τους.

Συμπεράσματα

Σε αυτή την εργασία αναπτύχθηκε μεθοδολογία και το αντίστοιχο λογισμικό για τη μοντελοποίηση σηράγγων, με γνώμονα τη φιλικότητα προς το χρήστη, παρέχοντας ταυτόχρονα ένα πλήρως τρισδιάστατο περιβάλλον σχεδίασης. Το πρόγραμμα κάνει χρήση των βιβλιοθηκών τρισδιάστατης σχεδίασης της Open Cascade για τον ορισμό των γεωμετρικών οντοτήτων και την απεικόνισή τους. Η μοντελοποίηση των σηράγγων βασίζεται σε καμπύλες και επιφάνειες NURBS, παρέχοντας έτσι τη δυνατότητα δημιουργίας οποιασδήποτε γεωμετρίας σήραγγας, χρησιμοποιώντας ενιαία παραμετροποίηση. Η χρήση της παραμετροποίησης NURBS και των πρωτοκόλλων IGES και STEP καθιστούν το προτεινόμενο λογισμικό συμβατό με τα περισσότερα λογισμικά δημιουργίας πλέγματος και ανάλυσης. Η σχεδίαση ενισχύεται από το γραφικό περιβάλλον διεπαφής με την τριδιάστατη απεικόνιση και διαχείριση των γεωμετρικών οντοτήτων. Η χρήση των βιβλιοθηκών της Open Cascade επιτρέπει την ανάπτυξη λογισμικού για γεωμετρική μοντελοποίηση με πολλαπλές εφαρμογές, χωρίς να χρειάζεται μεγάλη επάρκεια σε χρόνο και ανθρώπινο δυναμικό.

Η λογική της σχεδίασης βασίστηκε στη λογική σχεδίασης που ακολουθούν υπάρχοντα λογισμικά που κυκλοφορούν στο εμπόριο και ειδικεύονται στον ίδιο τομέα, καθιστώντας το αντίστοιχο με αυτά. Οι αδυναμίες του προγράμματος βασίζονται κυρίως στις ελλείψεις που ακόμα έχει, αλλά που πρόκειται να καλυφθούν στο εγγύτερο μέλλον. Οι ελλείψεις του είναι κυρίως η αδυναμία του στο να αποθηκεύει τις γεωμετρίες που σχεδιάζονται σε ειδική μορφή αρχείου, το οποίο θα αποθηκεύει όλη τη διαδικασία σχεδίασης, η αδυναμία του στο να δημιουργεί όγκους που θα αναπαριστούν τη μάζα του εδάφους, η αδυναμία του στο να ορίζει τις φάσεις της εκσκαφής, καθώς και η έλλειψη κάποιων δομικών στοιχείων των σηράγγων, όπως είναι τα αγγύρια. Όλες αυτές οι ελλείψεις αποτελούν και άμεσους μελλοντικούς στόχους, καθώς και η απευθείας συνεργασία του με κάποιο πρόγραμμα παραγωγής πλέγματος, παρά το γεγονός ότι αυτό είναι ήδη εφικτό, μέσω των αρχείων μεταφοράς δεδομένων IGES και STEP, από ανεξάρτητα λογισμικά δημιουργίας πλέγματος.

Βιβλιογραφία

[Nikolos] Sarakinos, S., Nikolos I.K., Pappou, T., *A software tool for 3D geometric modelling of tunnels*, CD proceedings of the EUROTUN2007 conference.

[OpenCascade] <http://www.opencascade.org/>.

[Piegl] Piegl, L., Tiller, W., *The NURBS Book*, Springer (1997).

[Shin] D.S., Shin, H.S., Lee, C., Cho, J.H., Shin, *An enhanced pre & post processing techniques for three dimensional tunnel analysis: Tunnel Modelling Wizard*, Key Engineering Materials, Vols.353-358 (2007), pp. 2798-2801.

[Midas/GTS] <http://eng.midasuser.com/gts/>.

[FIDES] <http://www.sofistik.com/en/solutions/structural-fea/tunnel-design/>.

[Sofistik] Sofistik G.m.b.H. 1993. SOFiSTiK Computer Software, Manuals.

[Thames] <http://en.wikipedia.org/wiki/Image:Thamestunnel.jpg>.

[Kienast] H.J., Kienast, *Die Wasserleitung des Eupalinos auf Samos (Samos XIX.)*, Rudolph Habelt, Bonn, 1995, ISBN 3-7749-2713-8.

[Hopson] <http://www.stevhopson.com/LightTunnel3.htm>.

[NorthEast] http://en.wikipedia.org/wiki/Image:North_east_Line_tunnels.jpg.

[Πάππου] Επικοινωνία με Δρ. Θεοδώρα Πάππου.

[Wikipedia1] <http://en.wikipedia.org/wiki/Tunnels>.

[Infoplease] <http://www.infoplease.com/ce6/sci/A0861654.html>.

[Wikipedia2] http://en.wikipedia.org/wiki/Tunnel_of_Eupalinos.

ΠΑΡΑΡΤΗΜΑ

Ι) Παρουσίαση της βιβλιοθήκης Open Cascade

Η βιβλιοθήκη Open Cascade είναι μια συλλογή από αντικειμενοστραφείς βιβλιοθήκες, γραμμένες σε γλώσσα C++, για τη γρήγορη ανάπτυξη λογισμικών επικεντρωμένων στη σχεδίαση. Οι βιβλιοθήκες αυτές βοηθούν στην ταχύτερη ανάπτυξη εφαρμογών σχεδιομελέτης σε υπολογιστή με διδιάστατη ή τριδιάστατη απεικόνιση, εφαρμογών κατασκευής ή ανάλυσης, προσομοίωσης, ή εργαλείων απεικόνισης [OpenCascade].

Η Open CasCade μπορεί να παρέχει τις παρακάτω υπηρεσίες:

- Εργαλεία διδιάστατης ή τριδιάστατης γεωμετρικής μοντελοποίησης για τη σχεδίαση περίπλοκων αντικειμένων:
 - Βασικούς όγκους, όπως ορθογώνιο παραλληλεπίπεδο, κύλινδρος, κώνος και δακτύλιος.
 - Λογικές πράξεις μεταξύ αντικειμένων (Boolean Operations)
 - Τροποποίηση αντικειμένων με φιλέτα και σπασίματα.
 - Διαδικασίες μοντελοποίησης με χρήση διαδικασιών offset, shelling, hollowing και sweep.
 - Υπολογισμός ιδιοτήτων, όπως εμβαδό, όγκος, κέντρο βάρους και καμπυλότητα.
 - Υπολογισμός γεωμετρίας με χρήση προβολών, παρεμβολών και προσεγγίσεων.
- Υπηρεσίες απεικόνισης, οι οποίες επιτρέπουν τη διαχείριση και προβολή αντικειμένων σε διάφορες όψεις:
 - Τριδιάστατη περιστροφή.
 - Εστίαση.
 - Σκίαση.
- Αλγόριθμους για τη μεταφορά δεδομένων:
 - Συσχέτιση μεταξύ μη γεωμετρικών δεδομένων και της γεωμετρίας.
 - Παραμετροποίηση των μοντέλων.
 - Μεταφορά γεωμετρικών δεδομένων με χρήση αρχείων μορφής IGES και STEP.

II) Οι βιβλιοθήκες της Open CasCade

Τα αντικείμενα της Open CasCade έχουν σχεδιαστεί για να είναι χωρισμένα σε υπομονάδες και πλήρως επεκτάσιμα. Γι' αυτό το λόγο υπάρχουν διαφορετικές κλάσεις για τον ορισμό των δομών δεδομένων (γεωμετρικής σχεδίασης, απεικόνισης και επιλογής), την εφαρμογή περίπλοκων αλγορίθμων και την παροχή APIs.

Οι κλάσεις της Open Cascade, που συσχετίζονται μεταξύ τους, είναι ομαδοποιημένες σε πακέτα, που διακρίνονται από το όνομά τους. Το όνομα κάθε κλάσης έχει ως πρόθεμα το όνομα του πακέτου στο οποίο ανήκει. Για παράδειγμα οι κλάσεις που περιγράφουν τριδιάστατα γεωμετρικά αντικείμενα ανήκουν στο πακέτο Geom. Σε αυτό το πακέτο η κλάση, που περιγράφει μια επιφάνεια Bezier, καλείται BezierSurface, και το πλήρες της όνομα είναι Geom_BezierSurface.

Τέλος, οι βιβλιοθήκες είναι ταξινομημένες σε έξι ομάδες με τη μικρότερη (Foundation Classes) να περιέχει δύο βιβλιοθήκες και τη μεγαλύτερη (Modeling Algorithms) να περιέχει οκτώ βιβλιοθήκες. Οι ομάδες αυτές συνοψίζονται στον πίνακα του σχήματος A.1 [OpenCascade].

Foundation Classes	Modeling Data	Modeling Algorithms	Visualization	Data Exchange	Application Framework
<ul style="list-style-type: none"> Kernel Classes Math Utilities 	<ul style="list-style-type: none"> 2D Geometry 3D Geometry Geometry Utilities Topology 	<ul style="list-style-type: none"> Construction of Primitives Boolean Operations Fillets and Chamfers Offsets, Drafts Sewing and Sweeps Features Hidden Line Removal Geometric Tools Topological Tools 	<ul style="list-style-type: none"> Services Common to 2D and 3D 2D Visualization 3D Visualization 	<ul style="list-style-type: none"> IGES STEP AP203 AP214 	<ul style="list-style-type: none"> Data Framework Data Storage Application Desktop

Σχήμα A.1: Η ταξινόμηση των βιβλιοθηκών της Open Cascade σε ομάδες [OpenCascade].

III) Οι κλάσεις της Open Cascade στο λογισμικό που αναπτύχθηκε

Στο κεφάλαιο 2 αναλύθηκε η δομή των σχεδιαστικών κλάσεων του προγράμματος, όσον αφορά την κληρονομικότητά τους και τη λειτουργία τους. Σε αυτές τις κλάσεις χρησιμοποιείται ένας μεγάλος αριθμός κλάσεων της Open Cascade, που εκτελούν διάφορες λειτουργίες. Στη συνέχεια αναλύονται οι κλάσεις της Open Cascade που χρησιμοποιούνται στο λογισμικό που αναπτύχθηκε.

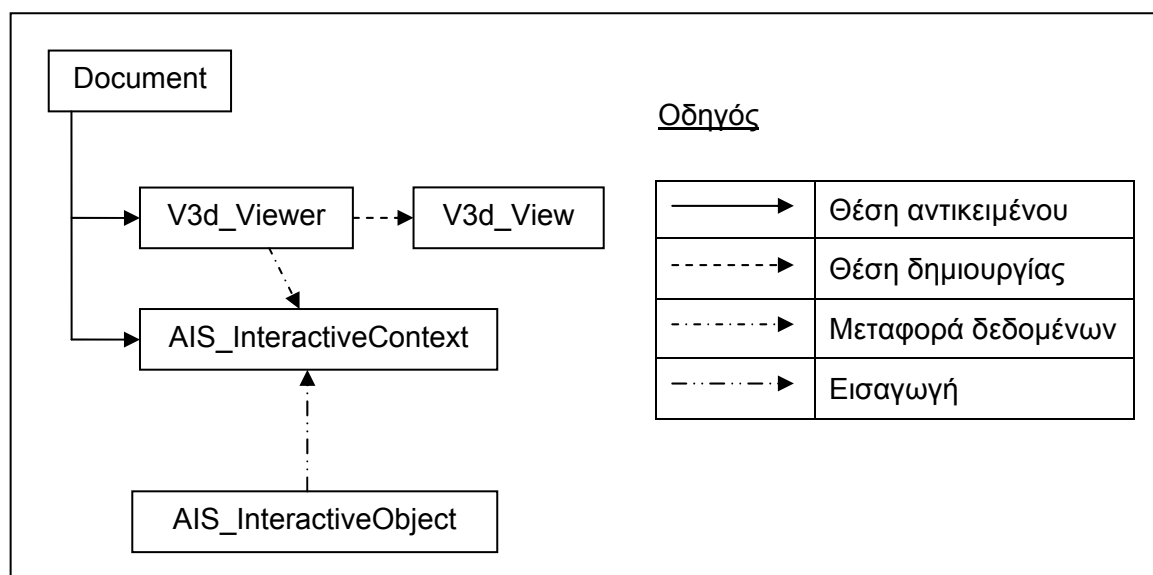
1) Δημιουργία παράθυρου προβολής αντικειμένων.

Το λογισμικό βασίζεται στη λογική MDI (Multi Document Interface), δηλαδή είναι στην ουσία μια πλατφόρμα που μπορεί να περιέχει μεγάλο αριθμό σχεδίων, τα οποία μπορούν να διαχειρίζονται ξεχωριστά. Για να επιτευχθεί αυτό, έχει δημιουργηθεί στο λογισμικό μια κλάση με το όνομα Document, κάθε αντικείμενο της οποίας είναι ένα διαφορετικό σχέδιο, αλλά και ένα διαφορετικό παράθυρο προβολής στο λογισμικό. Γι' αυτό το σκοπό κάθε αντικείμενο της κλάσης Document έχει ως ιδιότητες ένα αντικείμενο της κλάσης V3d_Viewer, έναν πίνακα με αντικείμενα της κλάσης V3d_View και ένα αντικείμενο της κλάσης AIS_InteractiveContext της Open Cascade.

Η κλάση V3d_Viewer είναι μια κλάση που λειτουργεί ως πλατφόρμα για τη δημιουργία αντικειμένων της κλάσης V3d_View, ορίζοντας τις αρχικές τιμές για όλα τα αντικείμενα της κλάσης V3d_View, που δημιουργούνται μέσα από αυτή, ενώ ορίζει τη γενική κατάσταση της προβολής, όπως τον φωτισμό. Η κλάση αυτή είναι υπεύθυνη και για το πλέγμα που εμφανίζεται στην κατάσταση διδιάστατης σχεδίασης, καθώς ορίζει το μέγεθός του, τη θέση του και την κατάσταση λειτουργίας του.

Η κλάση V3d_View ορίζει μια όψη των αντικειμένων, που εισάγονται στο λογισμικό και επιτρέπει τον ορισμό και την αλλαγή παραμέτρων που αφορούν αυτή την όψη. Οι αρχικές παράμετροι του αντικειμένου της κλάσης V3d_View ορίζονται από το αντικείμενο της κλάσης V3d_Viewer, που βρίσκεται μέσα στην κλάση Document. Μέσα από την κλάση V3d_View μπορεί να ορισθεί η γωνία προβολής του σχεδίου, η θέση του ματιού του θεατή σε σχέση με το σχέδιο κ.α. Άρα γίνεται σαφές ότι μέσα από το αντικείμενο αυτής της κλάσης πραγματοποιούνται οι λειτουργίες της περιστροφής, της μετατόπισης και της εστίασης του σχεδίου, ανάμεσα σε άλλες λειτουργίες.

Η κλάση AIS_InteractiveContext επιτρέπει την επεξεργασία της οπτικής συμπεριφοράς και της επιλογής των γεωμετρικών οντοτήτων, που εισάγονται στο παράθυρο προβολής. Πάνω απ' όλα ένα αντικείμενο της κλάσης AIS_InteractiveContext περιέχει μια λίστα από αντικείμενα της κλάσης AIS_InteractiveObject, η οποία είναι η κλάση που ορίζει την οπτική συμπεριφορά μιας γεωμετρίας. Κάθε γεωμετρική κλάση του λογισμικού διαθέτει ως ιδιότητα ένα αντικείμενο της κλάσης AIS_InteractiveObject, με το όνομα IObject, που αντιπροσωπεύει τη γεωμετρική οντότητα στο παράθυρο προβολής. Για να δημιουργηθεί ένα αντικείμενο AIS_InteractiveObject πρέπει να μετατραπεί από ένα αντικείμενο τοπολογίας. Το αντικείμενο της κλάσης AIS_InteractiveContext στο κάθε Document ορίζεται θέτοντάς του το V3d_Viewer, στο οποίο θα εισάγει τα σχεδιαστικά αντικείμενα. Πέραν αυτού, η κλάση AIS_InteractiveContext μπορεί να λειτουργήσει ως «μεσάζοντας» για να μπορούν να ορίζονται οι ιδιότητες των Interactive Objects που εισάγονται σε αυτή, όπως το χρώμα, το υλικό και η κατάσταση προβολής του αντικειμένου. Στο σχήμα A.2 παρουσιάζεται η σχέση των κλάσεων που περιγράφησαν σε αυτή την ενότητα, καθώς και η σειρά δημιουργίας τους στο λογισμικό.



Σχήμα Α.2: Η σχέση των κλάσεων που δημιουργούν το παράθυρο προβολής αντικειμένων.

2) Κλάσεις τοπολογίας

Η τοπολογία στην Open Cascade ορίζεται μέσω του πακέτου TopoDS. Οι κλάσεις που περιέχονται σε αυτό το πακέτο είναι οι TopoDS_Shape, TopoDS_Compound, TopoDS_Edge, TopoDS_Shell, TopoDS_CompSolid, TopoDS_Face, TopoDS_Solid και TopoDS_Wire.

Η τοπολογία ορίζει τη σχέση μεταξύ απλών γεωμετρικών οντοτήτων, που μπορούν έτσι να συσχετιστούν και να συνδεθούν μεταξύ τους, για να δημιουργήσουν πιο περίπλοκα σχήματα. Το πακέτο TopoDS περιέχει κλάσεις για μοντελοποίηση και δημιουργία δομών δεδομένων που είναι πλήρως τοπολογικές.

Η κλάση TopoDS_Shape είναι η πιο γενική από όλες τις κλάσεις και κληρονομείται από όλες τις άλλες. Περιγράφει μια γενική τοπολογική οντότητα και μπορεί να ορισθεί ως οποιοσδήποτε τύπος γεωμετρίας.

Η κλάση TopoDS_Compound περιγράφει μια σύνθετη τοπολογία, που μπορεί να αποτελείται από ποικιλία τοπολογικά κατώτερων οντοτήτων.

Η κλάση TopoDS_Edge περιγράφει την τοπολογία μιας ακμής. Μπορεί να ορίζεται από μια καμπύλη οποιοσδήποτε μορφής, αρκεί να έχει συνέχεια πρώτης παραγώγου.

Η κλάση TopoDS_Shell περιγράφει την τοπολογία ενός κελυφους. Το κέλυφος μπορεί να είναι μια οποιαδήποτε επιφάνεια, η οποία έχει συνέχεια πρώτης παραγώγου στις δυο παραμετρικές διευθύνσεις της.

Η κλάση TopoDS_CompSolid περιγράφει μια σύνθετη τοπολογία στερεών. Ένα CompSolid μπορεί να αποτελείται από πολλά διαφορετικά Solids ενωμένα μεταξύ τους.

Η κλάση TopoDS_Face περιγράφει την τοπολογία μιας επίπεδης επιφάνειας. Η επιφάνεια συνήθως προκύπτει από ένα επίπεδο στο χώρο.

Η κλάση TopoDS_Solid περιγράφει την τοπολογία ενός στερεού. Το στερεό μπορεί να είναι οποιασδήποτε μορφής.

Η κλάση `TopoDS_Wire` περιγράφει την τοπολογία μιας σύνθετης καμπύλης, που αποτελείται από ένα μεγάλο αριθμό αντικειμένων `TopoDS_Edge` και μπορεί να μην έχει συνέχεια πρώτης παραγώγου στα σημεία ένωσης των ακμών.

3) Γεωμετρικές κλάσεις

Για τη δημιουργία όλων των γεωμετρικών οντοτήτων χρησιμοποιήθηκαν συγκεκριμένες κλάσεις της `Open Cascade`, που ορίζουν την κάθε ξεχωριστή γεωμετρία. Για να μπορέσουν να εμφανιστούν στο παράθυρο προβολής, έπρεπε στη συνέχεια να μετατραπούν σε στοιχεία τοπολογίας και αυτά με τη σειρά τους σε `interactive objects`. Τελικά, όλες οι σχεδιαστικές κλάσεις του προγράμματος περιέχουν και τις τρεις αυτές απόψεις της ίδιας οντότητας. Οι γεωμετρικές κλάσεις της `Open Cascade`, που χρησιμοποιήθηκαν στο αναπτυχθέν λογισμικό, περιγράφονται στη συνέχεια.

- **gp_Pnt**: Περιγράφει ένα καρτεσιανό σημείο στο χώρο. Για να δημιουργηθεί πρέπει να ορισθούν οι συντεταγμένες του σημείου. Καθώς το σημείο δεν έχει τοπολογία, η κλάση μπορεί να μετατραπεί απευθείας σε `AIS_Point`, που είναι `interactive object` κλάση ειδική για σημεία και απεικονίζει το σημείο στο παράθυρο με το σύμβολο $+$.
- **Geom_BSplineCurve**: Περιγράφει μια καμπύλη NURBS. Για να δημιουργηθεί πρέπει να ορισθούν τα σημεία ελέγχου, το δυνάμωμα των κόμβων, ο βαθμός της καμπύλης και σε περίπτωση που είναι ρητή, τα αντίστοιχα βάρη των σημείων. Το αντικείμενο της κλάσης μετατρέπεται στη συνέχεια σε `TopoDS_Edge`.
- **BRepOffsetAPI_MakePipe**: Περιγράφει τη δημιουργία μιας γεωμετρίας που προκύπτει από οδήγηση. Για να δημιουργηθεί πρέπει να ορισθούν η γενέτειρα γεωμετρία και η καμπύλη οδηγός. Το αντικείμενο μετατρέπεται στη συνέχεια σε `TopoDS_Shape`.
- **BRepAlgo_Cut, BRepAlgo_Fuse, BRepAlgo_Section, BRepAlgo_Common**: Όλες οι κλάσεις περιγράφουν τις λογικές πράξεις `Boole` που είναι εφικτές από το λογισμικό. Για να δημιουργηθούν χρειάζεται να ορισθούν οι γεωμετρίες, οι οποίες παίρνουν μέρος στην κάθε πράξη. Τα αντικείμενα μετά μετατρέπονται σε `TopoDS_Shape`.
- **gp_Pln**: Περιγράφει ένα επίπεδο στο χώρο. Το αντικείμενο προέρχεται από την κλάση `gce_MakePln`, που παρέχει περισσότερες επιλογές για διαφορετικούς τρόπους ορισμού ενός επιπέδου στο χώρο. Καθώς ένα επίπεδο δεν έχει τοπολογία, το αντικείμενο `gp_Pln` μετατρέπεται απευθείας σε `AIS_Plane`, που είναι το αντίστοιχο `interactive object` για το επίπεδο και εμφανίζει το επίπεδο στο παράθυρο προβολής με τη μορφή ενός τετραγώνου.
- **BRepPrimAPI_MakeSphere**: Περιγράφει μια σφαίρα στο χώρο. Η κλάση παρέχει πολλούς τρόπους ορισμού της σφαίρας. Στο πρόγραμμα υιοθετήθηκε η μεθοδολογία ορισμού της από ένα σημείο στο χώρο, που αποτελεί το κέντρο της, καθώς και την ακτίνα της.
- **BRepPrimAPI_MakeBox**: Περιγράφει ένα ορθογώνιο παραλληλεπίπεδο στο χώρο. Η κλάση παρέχει πολλούς τρόπους ορισμού του παραλληλεπίπεδου. Στο λογισμικό υιοθετήθηκε η μεθοδολογία ορισμού του από ένα επίπεδο στο χώρο, που ορίζει την κατεύθυνσή του και τις διαστάσεις του.
- **gce_Circ**: Παρέχει ένα μεγάλο αριθμό από συναρτήσεις για τον ορισμό ενός κύκλου στο χώρο. Στο λογισμικό υιοθετήθηκε ο τρόπος ορισμού του από τρία

σημεία στο χώρο. Το αντικείμενο μετατρέπεται αυτόματα σε αντικείμενο `gr_Circ`, που περιγράφει ένα κύκλο στο χώρο και στη συνέχεια σε `TopoDS_Edge`.

- **GC_MakeArcOfCircle:** Παρέχει ένα μεγάλο αριθμό από συναρτήσεις για τον ορισμό ενός τόξου κύκλου. Στο λογισμικό υιοθετήθηκαν δύο τρόποι ορισμού. Ο πρώτος ορίζει ένα τόξο που περνά από τρία σημεία στο χώρο. Για αυτό τον ορισμό πρέπει να δηλωθούν τα σημεία από τα οποία περνά το τόξο και είναι αντικείμενα τύπου `gr_Pnt`. Ο δεύτερος ορίζει ένα τόξο κύκλου, που περνά από τρία σημεία και περιορίζεται από δυο γωνίες που μετρούνται προς τις αντίθετες κατευθύνσεις του τόξου στο πρώτο σημείο. Για αυτόν τον ορισμό πρέπει να ορισθούν τα σημεία, που είναι αντικείμενα `gr_Pnt` και οι ακτίνες.