

Ετήσια Τεχνική Έκθεση Έτος 2014



ΘΑΛΗΣ – Πολυτεχνείο Κρήτης

Πλατφόρμα προηγμένων μαθηματικών μεθόδων και λογισμικού για την επίλυση προβλημάτων πολλαπλών πεδίων (multi physics, multidomain) σε σύγχρονες υπολογιστικές αρχιτεκτονικές: Εφαρμογή σε προβλήματα Περιβαλλοντικής Μηχανικής και Ιατρικής (MATENVMED- MIS 379416)

Δράση 3.1

Υλοποίηση σε Clusters, Grids και Cloud



Πίνακας Περιεχομένων

1. ΣΚΟΠΟΣ	3
1.1. ΥΛΟΠΟΙΗΣΗ ΣΕ CLUSTERS, GRIDS ΚΑΙ CLOUD	3
1.2. ΣΥΓΚΕΡΑΣΜΟΣ ΑΡΙΘΜΗΤΙΚΩΝ ΜΕΘΟΔΩΝ & ΛΟΓΙΣΜΙΚΟΥ	4
2. ΜΕΘΟΔΟΛΟΓΙΑ.....	5
2.1. ΥΛΟΠΟΙΗΣΗ ΜΕΘΟΔΩΝ IR ΣΤΗ FEniCS	5
2.1.1. Υλοποίηση στη FEniCS	6
2.1.2. RabbitMQ	7
2.1.3. Σειριακή Υλοποίηση	7
2.1.4. Παράλληλη Υλοποίηση	7
2.2. ΣΥΓΚΕΡΑΣΜΟΣ ΑΡΙΘΜΗΤΙΚΩΝ ΜΕΘΟΔΩΝ & ΛΟΓΙΣΜΙΚΟΥ	8
2.2.1. Σειριακή Υλοποίηση της GEO	8
2.2.2. Παράλληλη Υλοποίηση της GEO	8
3. ΑΠΟΤΕΛΕΣΜΑΤΑ	9
3.1. ΑΞΙΟΛΟΓΗΣΗ ΑΛΓΟΡΙΘΜΩΝ	9
3.1.1. Πειράματα για αξιολόγηση της υλοποίησης αλγορίθμων	9
3.1.2. Πειράματα	10
4. ΠΑΡΑΔΟΤΕΑ.....	20
5. ΣΥΝΕΡΓΑΣΙΕΣ.....	20
6. ΜΕΛΛΟΝΤΙΚΕΣ ΔΡΑΣΕΙΣ.....	20
7. ΒΙΒΛΙΟΓΡΑΦΙΑ.....	21
ΠΑΡΑΡΤΗΜΑ Ι.....	21
ΠΑΡΑΡΤΗΜΑ ΙΙ	25



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



1. ΣΚΟΠΟΣ

Η Δράση 3 αναφέρεται γενικά στη χρήση σύγχρονων υπολογιστικών αρχιτεκτονικών αφενός μεν στην υλοποίηση των αριθμητικών μεθόδων και του αντίστοιχου επιστημονικού λογισμικού, που θα αναπτυχθούν στα πλαίσια της Δράσης 2 (ιδιαίτερα 2.2 και 2.3), αφετέρου δε στην ανάπτυξη της πλατφόρμας λογισμικού της Δράσης 4. Η υλοποίησή του πραγματοποιείται από δύο διαφορετικές δράσεις (3.1 και 3.2), ανάλογα με την κατηγορία αρχιτεκτονικών που χρησιμοποιούνται. Η ΚΕΟ3 έχει αναλάβει την δράση 3.1: Υλοποίηση σε Clusters, Grids και Cloud και η ΚΕΟ2 την δράση 3.2: Υλοποίηση σε FPGAs.

Η Δράση 4 έχει σαν στόχο την δημιουργία ενός Περιβάλλοντος Επίλυσης Προβλημάτων. Η Δράση 4.1 εστιάζει στην ανάπτυξη πλατφόρμας λογισμικού, για προβλήματα πολλαπλών πεδίων καθώς και επικύρωση της για την επίλυση προβλημάτων της Περιβαλλοντικής Μηχανικής και της Ιατρικής. Στην πλατφόρμα θα ενσωματωθούν όλοι οι παράλληλοι αλγόριθμοι που υλοποιήθηκαν στην πλατφόρμα FEniCS (στη δράση 3) και θα παρέχει ταυτόχρονη πρόσβαση, μέσω Internet, σε πολλούς δυνητικούς χρήστες.

1.1. Υλοποίηση σε Clusters, Grids και Cloud

Αρχικά η υλοποίηση έλαβε χώρα σε επίπεδο συστοιχιών υπολογιστών (Clusters). Η υλοποίηση των μεθόδων αφορά τόσο σε ομοιογενή, συμμετρικά σχήματα όπου γίνεται χρήση ενός συγκεκριμένου προγραμματιστικού μοντέλου ανταλλαγής μηνυμάτων (MPI), όσο και σε ετερογενή, μη συμμετρικά σχήματα, όπου ο παραλληλισμός των μεθόδων επιτελείται σε πολλαπλά επίπεδα. Σύμφωνα με αυτά τα συνεργατικά σχήματα εφαρμόστηκε συνδυασμός μοντέλων ανταλλαγής μηνυμάτων (MPI) με μοντέλα προγραμματισμού κοινής μνήμης (OpenMP, Pthreads) για την περισσότερο ευέλικτη αξιοποίηση των επεξεργαστών πολλαπλών πυρήνων που περιέχονται σε κάθε κόμβο του Cluster.

Κατά το επόμενο στάδιο της συγκεκριμένης δράσης η υλοποίηση έγινε σε περιβάλλον Cloud μέσω χρήσης υπηρεσιών διαδικτύου (web services). Στη περίπτωση αυτή το παρεχόμενο λογισμικό είναι σε θέση να αξιοποιεί με καλύτερο τρόπο τους διαθέσιμους υπολογιστικούς πόρους, ιδιαίτερα κατά την περίπτωση όπου υποβάλλονται μέσω του Cloud αιτήσεις για παράλληλη εκτέλεση από περισσότερους του ενός χρήστες. Για το λόγο αυτό στη δράση 4.1 υλοποιήθηκε το μοντέλο Λογισμικό σαν Υπηρεσία (SaaS – Software as a Service), που δίνει τη δυνατότητα ενός φιλικού προς το χρήστη διαδικτυακού γραφικού περιβάλλοντος που μπορεί να χρησιμοποιηθεί όχι μόνο από σταθμούς εργασίας αλλά και από φορητές συσκευές. Επιπλέον, για να ληφθεί υπόψη η ποικιλομορφία των υποπροβλημάτων για ένα δοθέν πρόβλημα πολλαπλών πεδίων / πολλαπλής φυσικής, η προτεινόμενη εφαρμογή έχει τη δυνατότητα να υπολογίσει και να παρέχει τα βέλτιστα VMs (σε πλήθος και δυνατότητες) σύμφωνα με τους διαθέσιμους πόρους, το μέγεθος του συνολικού προβλήματος και το μέγεθος των υποπροβλημάτων του.

1.2. Συγκερασμός Αριθμητικών Μεθόδων & Λογισμικού

Κεντρική επιδίωξη της παρούσας δράσης ήταν η σχεδίαση μιας αρχιτεκτονικής η οποία έθεσε τις βάσεις για μια ενοποιημένη προσέγγιση αντιμετώπισης των σύνθετων προβλημάτων που μας απασχολούν καλύπτοντας τόσο την αναγκαιότητα σχεδίασης/ανάπτυξης λογισμικού με δυνατότητα ενσωμάτωσης σύγχρονων υπολογιστικών συστημάτων όσο και ανάπτυξης ενός λειτουργικού περιβάλλοντος επίλυσης σύνθετων προβλημάτων, το οποίο ενσωματώνει μεθόδους και λογισμικό της Δράσης 3, επιτρέπει την εκμετάλλευση των σύγχρονων υπολογιστικών συστημάτων και διευκολύνει την αποδοτική χρήση των λογισμικών μονάδων. Η αρχιτεκτονική αυτή προσαρμόστηκε σε ήδη υπάρχουσα πλατφόρμα ανοικτού λογισμικού (FEniCS) και το περιβάλλον αυτό αποτελεί και την πλατφόρμα αξιολόγησης των μεθόδων επίλυσης σύνθετων ΜΔΕ και δυνητικά την επικύρωσή τους σε σημαντικά προβλήματα της Περιβαλλοντικής Μηχανικής και της Ιατρικής.

Το cloud computing είναι ένα νέο πρότυπο παράδοσης υπηρεσιών υπολογισμού στο Διαδίκτυο. Οι cloud τεχνολογίες μπορούν να παρέχουν αξιόπιστες λύσεις σε σύνθετα και μεγάλα προβλήματα ανάλυσης δεδομένων. Στον τομέα της επεξεργασίας δεδομένων από σύνθετα φυσικά συστήματα του πραγματικού κόσμου, όπως η μηχανή στροβίλων αερίου, η ατμοσφαιρική ρύπανση ή η υποβρύχια ακουστική, απαιτούνται υπολογιστικά απαιτητικές μέθοδοι επίλυσης διαφορικών εξισώσεων. Η χρησιμοποίηση cloud computing τεχνολογιών για τη γρήγορη και αξιόπιστη μαθηματική επεξεργασία των συνεχώς προερχόμενων δεδομένων από τέτοια φυσικά συστήματα μπορεί να οδηγήσει σε ολοκληρωμένα συστήματα που εξάγουν πολύτιμη γνώση.

Το cloud computing αποτελεί ένα νέο μοντέλο στον τομέα των Τεχνολογιών Πληροφορικής και Επικοινωνιών. Ο βασικός του στόχος είναι να παρέχει πρόσβαση σε όλους τους υπολογιστικούς πόρους (όπως για παράδειγμα σε εφαρμογές, δίκτυα, αποθηκευτικούς χώρους, διακομιστές, υπηρεσίες κτλ.) απευθείας από το διαδίκτυο σε συνδυασμό με την ελάχιστη προσπάθεια διαχείρισης από μεριάς τελικού χρήστη. Το cloud computing στοχεύει στον διαμοιρασμό των πόρων από τους παρόχους προς τους τελικούς χρήστες με τη μορφή υπηρεσιών. Τα κύρια χαρακτηριστικά του cloud computing που το διαφοροποιούν από άλλες τεχνικές είναι η κατά απαίτηση διάθεση υπολογιστικών πόρων, η απομακρυσμένη πρόσβαση σε αυτούς μέσω διαδικτύου και η ευελιξία των παρεχόμενων υπηρεσιών. Η ευελιξία επιτρέπει την αναβάθμιση ή υποβάθμιση των υπολογιστικών πόρων σύμφωνα με τις απαιτήσεις του τελικού χρήστη. Επιπλέον, η συνεχής αύξηση των αποθηκευτικών χώρων των δεδομένων έχει δημιουργήσει μία τεράστια ποσότητα πολύπλοκων και διάχυτων ψηφιακών δεδομένων. Η απόσπαση χρήσιμης γνώσης από μεγάλα ψηφιακά datasets απαιτεί έξυπνες και ευκόλως επεκτάσιμες υπηρεσίες ανάλυσης, εργαλεία προγραμματισμού και εφαρμογές. Έτσι το cloud computing εισάγει ένα σύνολο τεχνολογιών για την διανομή υπολογιστικών πόρων και υπηρεσιών στους τελικούς χρήστες ανάλογα με τις απαιτήσεις τους για την επιτυχή διαδικασία ανάλυσης δεδομένων. Επίσης, η

δυνατότητα της ελαστικότητας και της επεκτασιμότητας έχει κάνει το cloud computing να είναι μια αναδυόμενη τεχνολογία αναφορικά με τις αναλύσεις μεγάλου όγκου δεδομένων οι οποίες απαιτούν παραλληλισμό, πολύπλοκες ροές ανάλυσης και υψηλό υπολογιστικό φόρτο εργασίας.

Στη συγκεκριμένη δράση η υλοποίηση έγινε σε περιβάλλον Cloud μέσω χρήσης υπηρεσιών διαδικτύου (web services). Στη περίπτωση αυτή το παρεχόμενο λογισμικό είναι σε θέση να αξιοποιεί με καλύτερο τρόπο τους διαθέσιμους υπολογιστικούς πόρους, ιδιαίτερα κατά την περίπτωση όπου υποβάλλονται μέσω του Cloud αιτήσεις για παράλληλη εκτέλεση από περισσότερους του ενός χρήστες. Για το λόγο αυτό στη δράση 4.1 υλοποιήθηκε το μοντέλο Λογισμικό σαν Υπηρεσία (SaaS – Software as a Service), που δίνει τη δυνατότητα ενός φιλικού προς το χρήστη διαδικτυακού γραφικού περιβάλλοντος που μπορεί να χρησιμοποιηθεί όχι μόνο από σταθμούς εργασίας αλλά και από φορητές συσκευές.

2. Μεθοδολογία

2.1. Υλοποίηση Μεθόδων IR στη FEniCS

Η επίλυση μεγάλων και σύνθετων Μερικών Διαφορικών Εξισώσεων (ΜΔΕ) είναι ένα πρόβλημα που αντιμετωπίζεται κυρίως με τεχνικές διακριτοποίησης πεδίου [1], [2]. Αυτή η προσέγγιση περιλαμβάνει την αποσύνθεση στο επίπεδο της γραμμικής άλγεβρας μετά την διακριτοποίηση του πεδίου και της εξίσωσης με την επιθυμητή μέθοδο, δηλαδή: Πεπερασμένων Διαφορών (ΠΔ) ή πεπερασμένων στοιχείων (ΠΣ). Το κύριο χαρακτηριστικό αυτών των μεθόδων είναι η μη ευελιξία για την επιλογή διαφορετικών μεθόδων για κάθε υποπεδίο του αρχικού προβλήματος. Η μεθοδολογία χαλάρωσης στη διεπαφή είναι μια ενδιαφέρουσα εναλλακτική λύση [3], [4]. Εδώ, το πεδίο των ΜΔΕ αποσυντίθεται σε υποπεδία, για λόγους που προέρχονται από την φυσική ή για λόγους παραλληλισμού, και την ίδια στιγμή αρχικές μαντεψιές καθορίζονται στις διεπαφές μεταξύ των υποπεδίων. Τα υποπροβλήματα επιλύονται και νέες τιμές στις διεπαφές υπολογίζονται με συγκεκριμένες μεθόδους χαλάρωσης στη διεπαφή (επιβάλλοντας τις σωστές συνθήκες για το πρόβλημα) επαναληπτικά έως ότου επιτευχθεί η σύγκλιση.

Περιβάλλοντα επίλυσης προβλημάτων πολλαπλών πεδίων / πολλαπλών φυσικών που υλοποιούν τη μεθοδολογία χαλάρωσης στη διεπαφή θα πρέπει να είναι ικανά να φιλοξενούν και να ενσωματώνουν μια ποικιλία υφιστάμενων επιλυτών ΜΔΕ και μεθόδων χαλάρωσης στη διεπαφή. Αυτοί οι επιλυτές πρέπει να παρέχουν μια ελάχιστη λειτουργικότητα συμπεριλαμβανομένων των εξής: ορισμό πεδίου και ΜΔΕ, δημιουργία πλέγματος, σχήμα διακριτοποίησης, εκτίμηση της λύσης και της παραγώγου σε οποιοδήποτε σημείο του πεδίου συμπεριλαμβανομένων των ορίων / διεπαφών. Οι υφιστάμενες υλοποιήσεις έχουν διάφορα μειονεκτήματα, μεταξύ των οποίων το ότι εξαρτώνται σε μεγάλο βαθμό από πλατφόρμες πρακτόρων και την βιβλιοθήκη PELLPACK, αποκαλύπτοντας έτσι την ανάγκη μιας νέας υλοποίησης απαλλαγμένης από αυτούς τους περιορισμούς.

Στην παρούσα μελέτη υλοποιούμε μια από τις μεθόδους χαλάρωσης στη διεπαφή, που παρουσιάστηκαν 2^η Ετήσια Έκθεση Προόδου, τη geometric (GEO)



contraction based στη FEniCS. Η FEniCS [5] είναι μια συλλογή ελεύθερου λογισμικού για αυτοματοποιημένη, αποδοτική επίλυση διαφορικών εξισώσεων. Οι λόγοι για την επιλογή της FEniCS είναι, μεταξύ άλλων, ότι είναι αξιόπιστη, αποδοτική, ελεύθερη και υποσχόμενη να υποστηρίζεται και στο μέλλον. Μια παράλληλη υλοποίηση της GEO στη FEniCS με τη βοήθεια του RabbiMQ (προσανατολισμένο σε μηνύματα middleware) [6] παρουσιάζεται, επίσης. Στη συνέχεια παρουσιάζονται σύντομα η FEniCS, το RabbiMQ και ζητήματα τόσο της σειριακής, όσο και της παράλληλης υλοποίησης.

2.1.1. Υλοποίηση στη FEniCS

Σκοπεύοντας στην ενσωμάτωση της μεθόδου GEO, μαζί με άλλες μεθόδους χαλάρωσης στη διεπαφή, σε ένα ολοκληρωμένο περιβάλλον επίλυσης προβλημάτων που να αφορά προβλήματα πολλαπλών πεδίων / πολλαπλών φυσικών την υλοποιήσαμε στη FEniCS. Η FEniCS παρέχει κλάσεις και μεθόδους με τις οποίες είναι δυνατός ο καθορισμός των ιδιοτήτων των υποπεδίων του προβλήματος (γεωμετρία πεδίου, τελεστής ΜΔΕ και οριακές συνθήκες ή/και συνθήκες διεπαφών). Οι μέθοδοί της μπορούν επίσης να δημιουργήσουν ή/και να βελτιώσουν πλέγματα (τριγωνικά στοιχεία) για κάθε υποπεδίο, να επιλύσουν τα τοπικά προβλήματα ΜΔΕ και να απεικονίσουν τα υπολογισμένα αποτελέσματα για ολόκληρο το πεδίο και για τις διεπαφές. Ο προεπιλεγμένος επιλυτής στα προγράμματα της FEniCS είναι η αραιή LU αποσύνθεση, καθώς είναι ισχυρή για κάποιες χιλιάδες αγνώστους στο σύστημα εξισώσεων. Παρόλα αυτά, η αραιή LU αποσύνθεση γίνεται αργή και πολύ απαιτητική σε μνήμη για μεγάλα προβλήματα. Για αυτό το λόγο η FEniCS παρέχει εναλλακτικά και επαναληπτικές μεθόδους, όπως οι preconditioned Krylov επιλυτές, οι οποίοι είναι γρηγορότεροι και απαιτούν σημαντικά λιγότερη μνήμη. Μια πλήρης λίστα των διαθέσιμων Krylov επιλυτών και preconditioners μπορεί να βρεθεί στην τεκμηρίωση της FEniCS [5]. Στην πραγματικότητα, οι υλοποιήσεις των επιλυτών που τίθενται σε δράση εξαρτώνται από την επιλογή του πακέτου γραμμικής άλγεβρας. Η FEniCS υποστηρίζει διάφορα πακέτα γραμμικής άλγεβρας, που αποκαλούνται backends στην ορολογία της FEniCS. Το PETSc είναι το προεπιλεγμένο backend και εναλλακτικά backends που υποστηρίζονται είναι τα uBLAS, Epetra (Trilinos) και MTL4.

Για την καλύτερη κατανόηση του τρόπου με τον οποίο ένα πρόβλημα επιλύεται στη FEniCS πειραματιστήκαμε με το ακόλουθο πρόβλημα-μοντέλο, που είναι το Poisson πρόβλημα:

$$\begin{aligned} -\nabla^2 u(x) &= f(x), \quad x \text{ in } \Omega, \\ u(x) &= u_0(x), \quad x \text{ on } \partial\Omega. \end{aligned}$$

Εδώ, η $u(x)$ είναι η άγνωστη συνάρτηση, η $f(x)$ είναι μια ορισμένη συνάρτηση, το ∇^2 είναι ο τελεστής Laplace, το Ω είναι το χωρικό πεδίο και το $\partial\Omega$ είναι το όριο του Ω . Μια ΜΔΕ σαν αυτή, μαζί με ένα πλήρες σύνολο από οριακές συνθήκες αποτελούν ένα πρόβλημα οριακών τιμών, που πρέπει να οριστεί πλήρως πριν την επίλυσή του στη FEniCS.

2.1.2. RabbitMQ

Για την παράλληλη υλοποίηση στην πλατφόρμα FEniCS χρησιμοποιήθηκε το RabbitMQ [6]. Το RabbitMQ είναι ένας ελαφρύς, αξιόπιστος, επεκτάσιμος και φορητός μεσολαβητής μηνυμάτων. Δίνει στις εφαρμογές μια κοινή πλατφόρμα για αποστολή και λήψη μηνυμάτων. Το RabbitMQ τρέχει σε όλα τα κύρια λειτουργικά συστήματα και είναι εύκολο στη χρήση. Υποστηρίζει έναν τεράστιο αριθμό από πλατφόρμες ανάπτυξης, μεταξύ των οποίων και την Python, που είναι η πλατφόρμα ανάπτυξης της FEniCS. Το RabbitMQ βασίζεται στο Advanced Message Queuing Protocol (AMQP). Το AMQP είναι ένα πρωτόκολλο μηνυμάτων που ασχολείται με εκδότες και καταναλωτές. Οι εκδότες παράγουν τα μηνύματα, οι καταναλωτές τα λαμβάνουν και τα επεξεργάζονται.

Στην παρούσα μελέτη δημιουργήθηκε ένα σύστημα κλήσης απομακρυσμένης διαδικασίας (Remote Procedure Call - RPC), βασισμένο στο RabbitMQ καθότι υπάρχει μεγάλη ανάγκη για επικοινωνία κατά τη διάρκεια της παράλληλης επίλυσης των προβλημάτων ΜΔΕ. Στη συνέχεια θα περιγράψουμε τη διαδικασία δημιουργίας ενός RPC συστήματος (που επιστρέφει Fibonacci αριθμούς): ένας πελάτης και ένας server χρησιμοποιώντας το RabbitMQ.

2.1.3. Σειριακή Υλοποίηση

Η geometric (GEO) contraction based μέθοδος υλοποιείται σαν ένα FEniCS πρόγραμμα γραμμένο στην προγραμματιστική γλώσσα Python. Η βιβλιοθήκη DOLFIN χρησιμοποιείται για την εισαγωγή κλάσεων χρήσιμων για τη δημιουργία των υποπεδίων του προβλήματος και τη δημιουργία πλεγμάτων (τριγωνικά στοιχεία) σε αυτά τα υποπεδία. Στη συνέχεια δηλώνουμε και εφαρμόζουμε τις οριακές συνθήκες, καθώς επίσης και τις αρχικές μαντεψιές στις διεπαφές των υποπεδίων. Το πρόβλημα ΜΔΕ πρέπει να εκφραστεί σαν ένα παραμετρικό (variational) πρόβλημα και στη συνέχεια να οριστεί στο πρόγραμμα. Μετά τον υπολογισμό της λύσης, πραγματοποιείται και ο υπολογισμός της παραγώγου.

Η δημιουργία κατάλληλων συναρτήσεων για την λήψη των τιμών της λύσης και της παραγώγου στα σημεία των διεπαφών (όρια των υποπροβλημάτων), ο υπολογισμός των νέων χαλαρωμένων τιμών και το πέρασμά τους πίσω στα υποπροβλήματα σαν ανανεωμένες τιμές για τις διεπαφές ήταν οι κυριότερες προκλήσεις της υλοποίησης της GEO.

2.1.4. Παράλληλη Υλοποίηση

Δεδομένου ότι η μέθοδος GEO είναι εγγενώς παραλληλίσιμη, στην παράλληλη υλοποίησή της κάθε κόμβος επιλύει ένα υποπεδίο. Οι υπολογισμένες τιμές της λύσης και της παραγώγου στα σημεία των διεπαφών από κάθε κόμβο (που επιλύει ένα υποπεδίο που έχει παραπάνω από ένα γειτονικά υποπεδία) στέλνονται σαν ένα RabbitMQ μήνυμα, έτσι ώστε η νέα επανάληψη να ξεκινήσει και στον άλλο κόμβο επίσης. Αυτό το σχήμα, που δεν εμπεριέχει ξεχωριστούς κόμβους για να διαχειρίζονται τις διεπαφές εξυπηρετεί στη μείωση του αριθμού των μηνυμάτων που

ανταλλάσσονται, σε μια προσπάθεια να ελαχιστοποιηθεί ο συνολικός χρόνος επικοινωνίας. Άρα, το μεσαίο πεδίο το επιλύει ένας client και τα άλλα δύο ένας server το καθένα. Αυτοί οι servers χειρίζονται και τις διεπαφές.

2.2. Συγκερασμός Αριθμητικών Μεθόδων & Λογισμικού

2.2.1. Σειριακή Υλοποίηση της GEO

Η geometric (GEO) contraction based μέθοδος υλοποιείται σαν ένα FEniCS πρόγραμμα γραμμένο στην προγραμματιστική γλώσσα Python. Η βιβλιοθήκη DOLFIN [7] χρησιμοποιείται για την εισαγωγή κλάσεων χρήσιμων για τη δημιουργία των υπο-πεδίων του προβλήματος και τη δημιουργία πλεγμάτων (τριγωνικά στοιχεία) σε αυτά τα υπο-πεδία. Στη συνέχεια δηλώνουμε και εφαρμόζουμε τις οριακές συνθήκες, καθώς επίσης και τις αρχικές μαντεψιές στις διεπαφές των υπο-πεδίων. Το πρόβλημα ΜΔΕ πρέπει να εκφραστεί σαν ένα παραμετρικό (variational) πρόβλημα και στη συνέχεια να οριστεί στο πρόγραμμα. Μετά τον υπολογισμό της λύσης, πραγματοποιείται και ο υπολογισμός της παραγώγου.

Η δημιουργία κατάλληλων συναρτήσεων για την λήψη των τιμών της λύσης και της παραγώγου στα σημεία των διεπαφών (όρια των υπο-προβλημάτων), ο υπολογισμός των νέων χαλαρωμένων τιμών και το πέρασμά τους πίσω στα υπο-προβλήματα σαν ανανεωμένες τιμές για τις διεπαφές ήταν οι κυριότερες προκλήσεις της υλοποίησης της GEO. Οι ανανεωμένες τιμές σε ένα σημείο της διεπαφής x υπολογίζονται ως εξής:

$$u^{(k+1)}(x) = u^{(k)}(x) - \rho \left(\frac{\partial u_L^{(k)}(x)}{\partial n} - \frac{\partial u_R^{(k)}(x)}{\partial n} \right), k = 1, 2, \dots$$

όπου k είναι η επανάληψη, u είναι η υπολογισμένη λύση στο σημείο της διεπαφής x , $\frac{\partial u_L^{(k)}(x)}{\partial n}$, $-\frac{\partial u_R^{(k)}(x)}{\partial n}$ είναι οι τιμές των προς τα έξω κανονικών παραγώγων στα δύο γειτονικά υποπεδία και ρ είναι η παράμετρος χαλάρωσης που χρησιμοποιείται για να επιταχύνει τη σύγκλιση. Μια νέα επανάληψη ξεκινάει με το που περαστούν οι χαλαρωμένες τιμές των διεπαφών πίσω στα υπο-πεδία.

Πιο αναλυτικά, τμήματα του FEniCS προγράμματος που υλοποιεί τα παραπάνω παρουσιάζεται στο Παράρτημα Ι.

2.2.2. Παράλληλη Υλοποίηση της GEO

Δεδομένου ότι η μέθοδος GEO είναι εγγενώς παραλληλίσιμη, στην παράλληλη υλοποίησή της κάθε κόμβος επιλύει ένα υπο-πεδίο. Οι υπολογισμένες τιμές της λύσης και της παραγώγου στα σημεία των διεπαφών από κάθε κόμβο (που επιλύει ένα υπο-πεδίο που έχει παραπάνω από ένα γειτονικά υπο-πεδία) στέλνονται σαν ένα RabbiMQ μήνυμα, έτσι ώστε η νέα επανάληψη να ξεκινήσει και στον άλλο κόμβο επίσης. Αυτό το σχήμα, που δεν εμπεριέχει ξεχωριστούς κόμβους για να διαχειρίζονται τις διεπαφές εξυπηρετεί στη μείωση του αριθμού των μηνυμάτων που ανταλλάσσονται, σε μια προσπάθεια να ελαχιστοποιηθεί ο συνολικός χρόνος επικοινωνίας. Άρα, το μεσαίο πεδίο το επιλύει ένας client και τα άλλα δύο ένας server το καθένα. Αυτοί οι servers

χειρίζονται και τις διεπαφές. Ενδεικτικά τμήματα του κώδικα δίνονται στο Παράρτημα Π.

3. ΑΠΟΤΕΛΕΣΜΑΤΑ

Σε αυτό το στάδιο όλοι οι αλγόριθμοι έχουν υλοποιηθεί παράλληλα στην πλατφόρμα FEniCS. Η αξιολόγηση των αλγορίθμων που υλοποιήθηκαν, για τις Δράσεις 3.1 και 4.1, παρουσιάζεται ενιαία στα επόμενα κεφάλαια.

3.1. Αξιολόγηση Αλγορίθμων

3.1.1. Πειράματα για αξιολόγηση της υλοποίησης αλγορίθμων

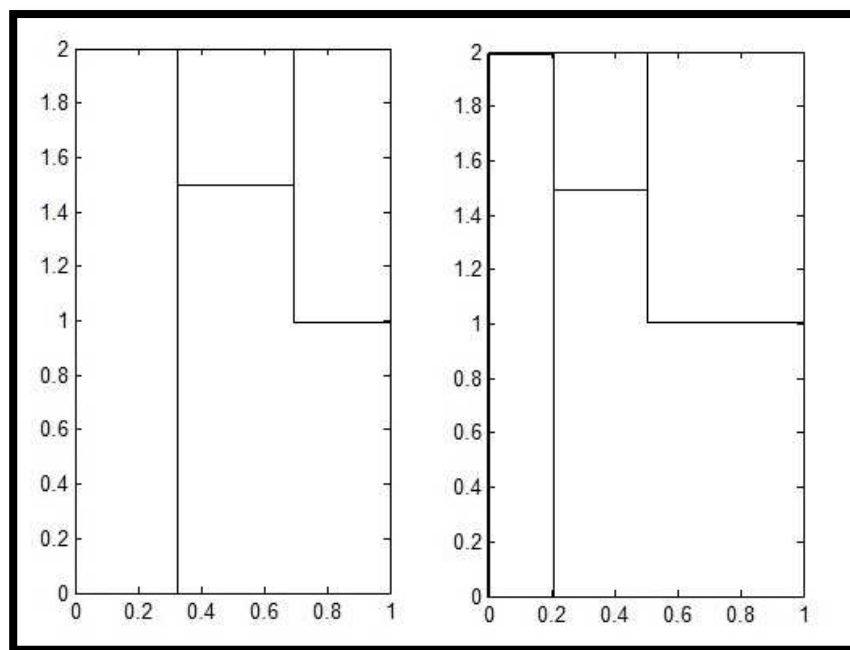
Για την εξέταση της ορθότητας και της απόδοσης της υλοποίησης, χρησιμοποιείται το ακόλουθο ελλειπτικό πρόβλημα:

$$\begin{aligned} Lu(x, y) &\equiv \nabla u(x, y) + \gamma^2 u(x, y) = f(x, y), \quad (x, y) \in \Omega \\ u(x, y) &= u^b(x, y), \quad (x, y) \in \Omega \end{aligned}$$

με $f(x, y)$ και $u^b(x, y)$ επιλεγμένα έτσι ώστε η πραγματική λύση να είναι:

$$u(x, y) = e^{y(x=4)} x(x-1)(x-0.7)y(y-0.5) \quad (3)$$

Πιο συγκεκριμένα μελετώνται δύο διαφορετικά προβλήματα ΜΔΕ που αποτελούνται από την προηγούμενη διαφορική εξίσωση και οριακές συνθήκες και τα δύο διαφορετικά πεδία που αποτυπώνονται στην εικόνα 6.



Εικόνα 1: Τα πεδία του ομοιόμορφου (αριστερά) και του ανομοιόμορφου (δεξιά) προβλήματος

3.1.2. Πειράματα

Για το ομοιόμορφο πρόβλημα τα σημεία των διεπαφών είναι στα $x_1 = \frac{1}{3}$ και $x_2 = \frac{2}{3}$ και για το ανομοιόμορφο πρόβλημα στα $x_1 = \frac{1}{5}$ και $x_2 = \frac{1}{2}$ και το $\gamma^2 = 2$. Τα πειράματα εκτελούνται για διάφορες τιμές της παραμέτρου διακριτοποίησης h , η οποία έχει θεωρηθεί ίση τόσο στην κατεύθυνση x όσο και στην κατεύθυνση y . Τα προκύπτοντα μεγέθη του πλέγματος παρουσιάζονται στον πίνακα 2.

Ομοιόμορφο πρόβλημα				
Περίπτωση	H	Αριστερό υποπεδίο	Μεσαίο υποπεδίο	Δεξί υποπεδίο
Π1	0.1	4x21	4x6	4x11
Π2	0.05	8x41	8x11	8x21
Π3	0.025	14x81	14x21	14x41
Π4	0.0125	28x161	28x41	28x81
Π5	0.00625	55x321	55x81	55x161
Π6	0.003125	108x641	108x161	108x321
Π7	0.0015625	214x1281	214x321	214x641



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



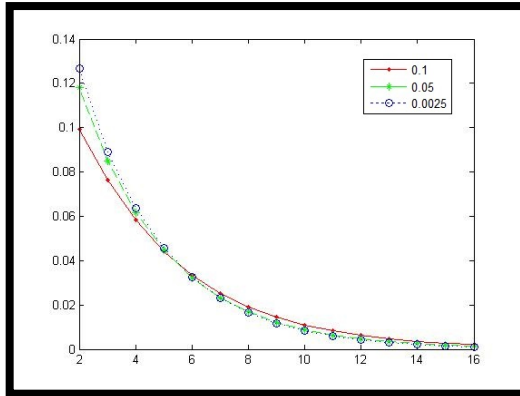
ΕΣΠΑ
2007-2013
πρόγραμμα για την ανάπτυξη
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ

Ανομοιόμορφο πρόβλημα				
Περίπτωση	H	Αριστερό υποπεδίο	Μεσαίο υποπεδίο	Δεξί υποπεδίο
Π1	0.1	3x21	4x6	6x11
Π2	0.05	5x41	7x11	11x21
Π3	0.025	9x81	13x21	21x41
Π4	0.0125	17x161	25x41	41x81
Π5	0.00625	33x321	49x81	81x161
Π6	0.003125	65x641	97x161	161x321
Π7	0.0015625	129x1281	193x321	321x641

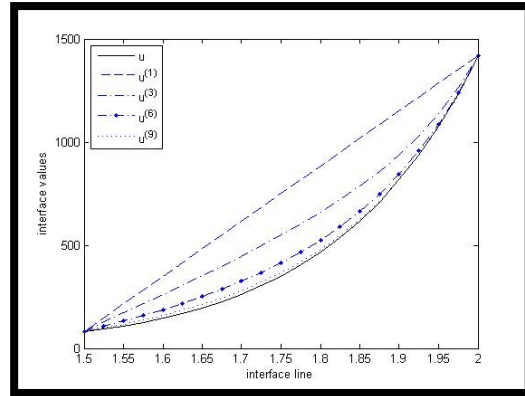
Πίνακας 1: Περιπτώσεις ελέγχου που εξετάστηκαν μαζί με το βήμα διακριτοποίησης και το μέγεθος πλέγματος για το αριστερό, μεσαίο και δεξί υποπεδίο του ομοιόμορφου και του ανομοιόμορφου προβλήματος

Ο αριθμός των σημείων των διεπαφών σε κάθε περίπτωση ισούται με τον αριθμό των σημείων της y διάστασης του μεσαίου υποπεδίου. Αυξάνεται, δηλαδή, από τα 6 έως τα 321 σημεία τόσο για το ομοιόμορφο, όσο και για το ανομοιόμορφο πρόβλημα. Στην ομοιόμορφη περίπτωση το αριστερό υπο-πεδίο είναι το μεγαλύτερο πρόβλημα από τα τρία με σημαντική διαφορά από τα υπόλοιπα. Στην ανομοιόμορφη περίπτωση το δεξί υπο-πεδίο είναι αυτό με το βαρύτερο έργο. Οι διεπαφές έχουν όλες ίσο αριθμό σημείων και άρα απαιτούν ίσο φόρτο εργασίας για την επεξεργασία τους.

Το ιστορικό της σύγκλισης αποτυπώνεται στις επόμενες εικόνες. Πιο συγκεκριμένα, στο αριστερό γράφημα παρουσιάζεται η νόρμα μεγίστου της σχετικής διαφοράς διαδοχικών επαναλήψεων για τη διεπαφή $x_1 = \frac{1}{3}$ του ομοιόμορφου προβλήματος για $h = 0.1, 0.05, 0.025$. Όπως άλλωστε υπόσχεται η μεθοδολογία χαλάρωσης στη διεπαφή, παρατηρούμε ότι ο ρυθμός σύγκλισης είναι ανεξάρτητος του τοπικού βήματος διακριτοποίησης h . Για το ομοιόμορφο πρόβλημα και για τη διεπαφή $x_1 = \frac{1}{3}$ στο δεξί γράφημα παρουσιάζονται η ακριβής και οι υπολογισμένες λύσεις για τις επαναλήψεις 1, 3, 6, 10 για $h = 0.05$.



Εικόνα 2: Η νόρμα μεγίστου της σχετικής διαφοράς διαδοχικών επαναλήψεων για τη διεπαφή $x_1=1/3$ του ομοιόμορφου προβλήματος για $h=0.1, 0.05, 0.0025$.



Εικόνα 3: Η ακριβής και οι υπολογισμένες λύσεις για το ομοιόμορφο πρόβλημα και για τη διεπαφή $x_1=1/3$ για τις επαναλήψεις 1, 3, 6, 10 για $h=0.05$.

Στη συνέχεια παρουσιάζονται αναλυτικά τα αποτελέσματα που προκύπτουν για όλες τις διεπαφές και για τα δύο προβλήματα για κάθε μία από τις περιπτώσεις Π1, Π2, Π3. Οι περιπτώσεις ελέγχου επιλέχτηκαν για να διερευνηθεί η συμπεριφορά της παράλληλης υλοποίησης και όχι για λόγους ακρίβειας, οπότε γι' αυτό το λόγο δεν παρουσιάζουμε εδώ τις γραφικές παραστάσεις των περιπτώσεων Π4, Π5, Π6 και Π7.



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο

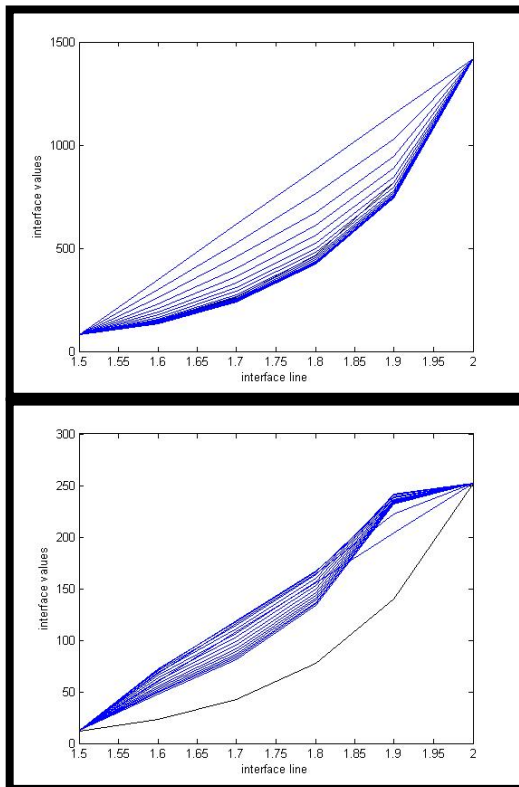


ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

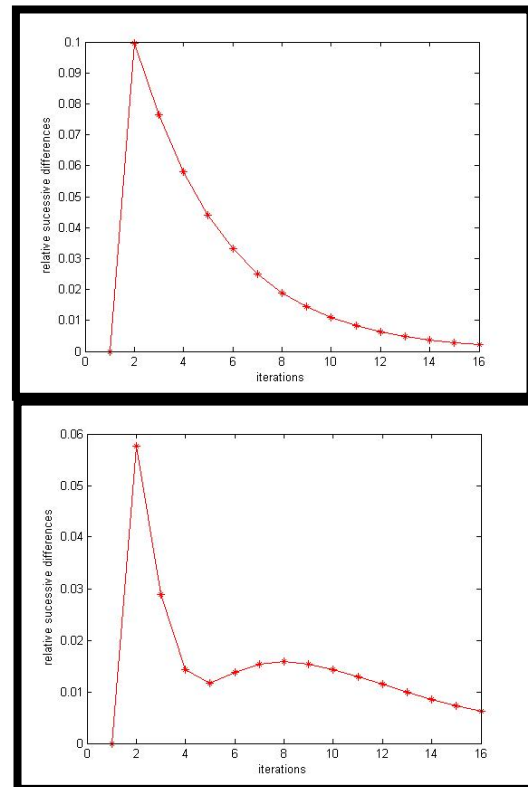
Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



Οι γραφικές παραστάσεις για την περίπτωση Π1 και το ομοιόμορφο πρόβλημα είναι οι ακόλουθες:



Εικόνα 4: Η ακριβής και οι υπολογισμένες λύσεις – ομοιόμορφο πρόβλημα, περίπτωση Π1, διεπαφή 1 (πάνω) και διεπαφή 2 (κάτω).



Εικόνα 5: Σχετικές διαφορές διαδοχικών επαναλήψεων - ομοιόμορφο πρόβλημα, περίπτωση Π1, διεπαφή 1 (πάνω) και διεπαφή 2 (κάτω)



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο

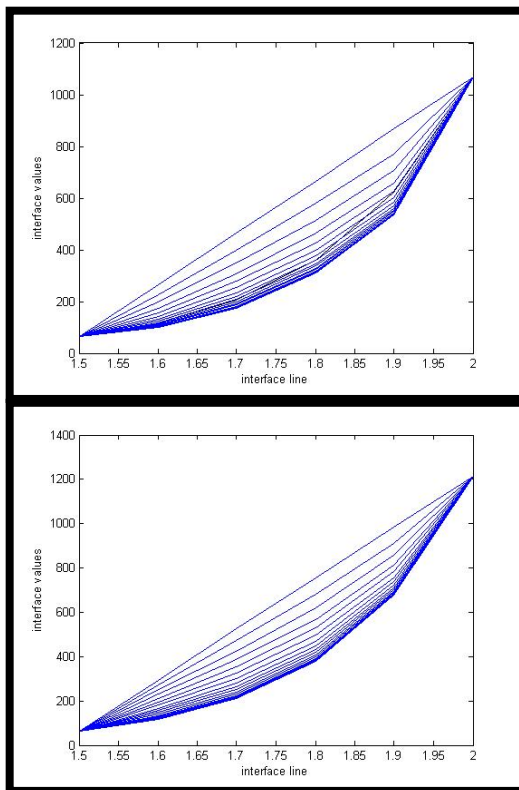


ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

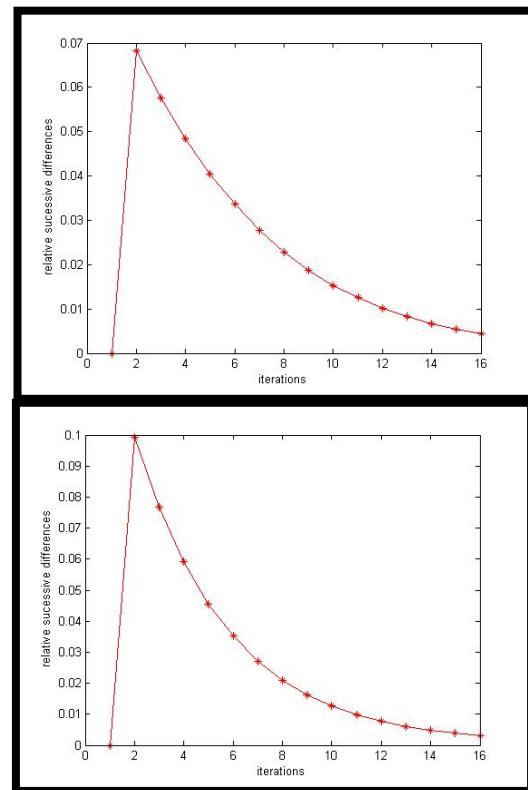
Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



Αντίστοιχα για το ανομοιόμορφο πρόβλημα:



Εικόνα 6: Η ακριβής και οι υπολογισμένες λύσεις – ανομοιόμορφο πρόβλημα, περίπτωση Π1, διεπαφή 1 (πάνω) και διεπαφή 2 (κάτω).



Εικόνα 7: Σχετικές διαφορές διαδοχικών επαναλήψεων – ανομοιόμορφο πρόβλημα, περίπτωση Π1, διεπαφή 1 (πάνω) και διεπαφή 2 (κάτω).



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο

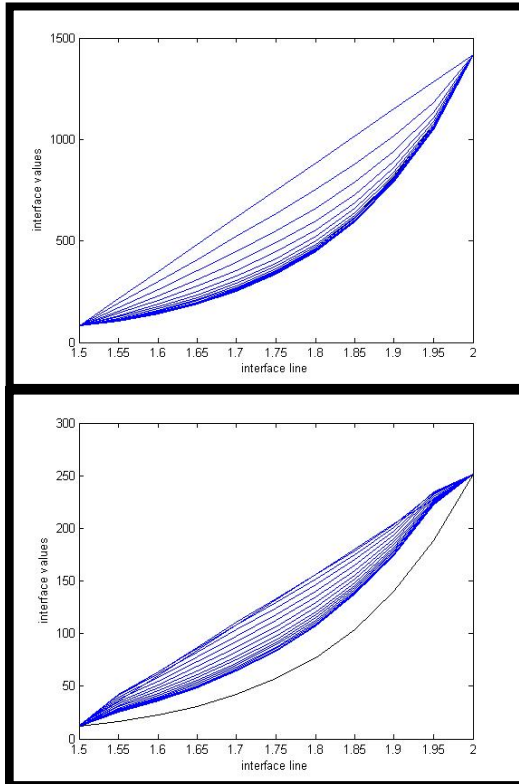


ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

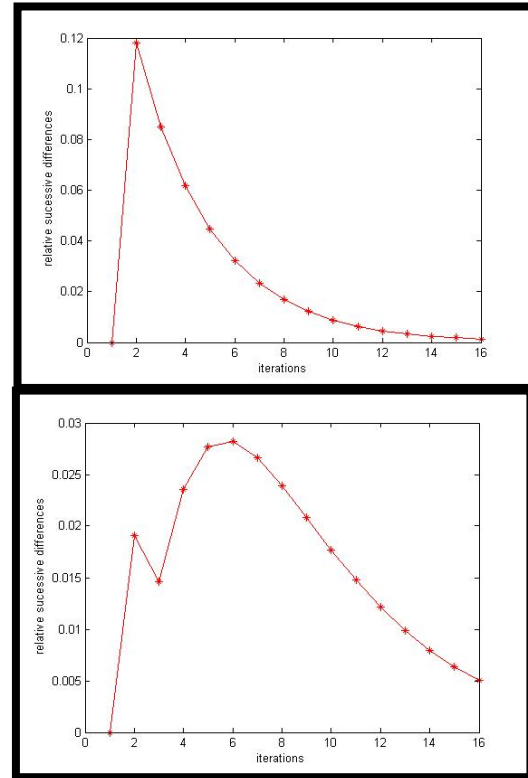
Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



Ακολουθούν οι γραφικές παραστάσεις για την περίπτωση Π2. Αρχικά για το ομοιόμορφο πρόβλημα:



Εικόνα 8: Η ακριβής και οι υπολογισμένες λύσεις – ομοιόμορφο πρόβλημα, περίπτωση Π2, διεπαφή 1 (πάνω) και διεπαφή 2 (κάτω).



Εικόνα 9: Σχετικές διαφορές διαδοχικών επαναλήψεων - ομοιόμορφο πρόβλημα, περίπτωση Π2, διεπαφή 1 (πάνω) και διεπαφή 2 (κάτω).



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο

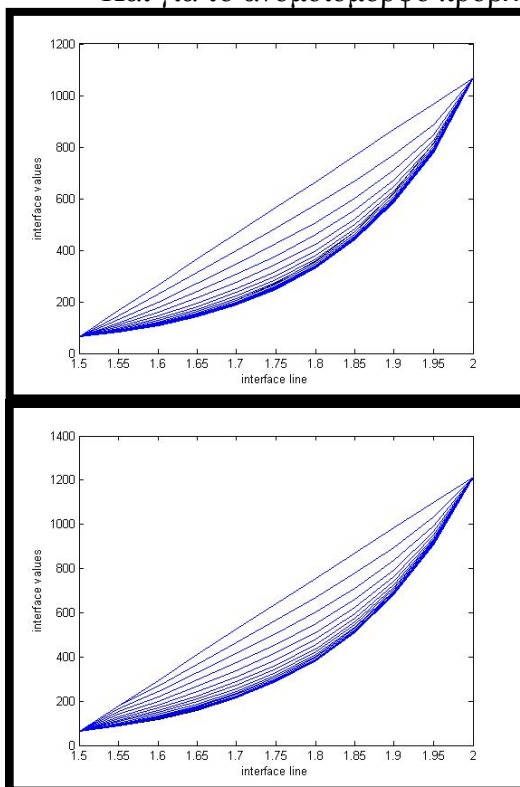


ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

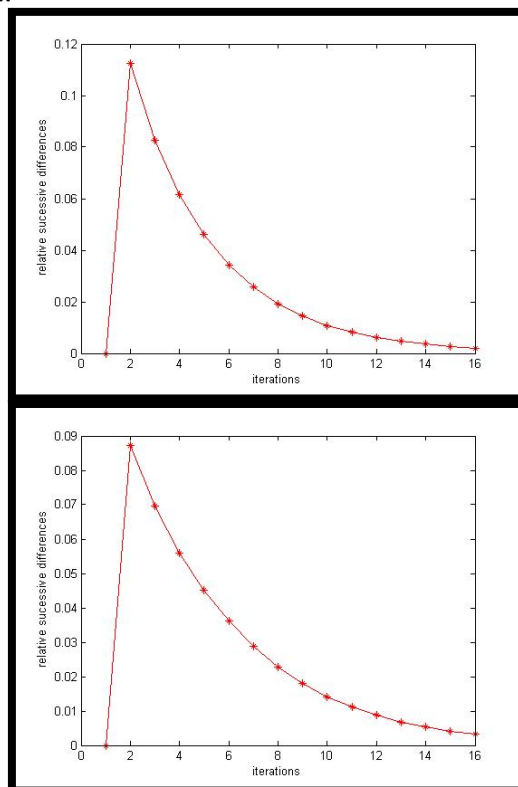
Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



Και για το ανομοιόμορφο πρόβλημα:



Εικόνα 10: Η ακριβής και οι υπολογισμένες λύσεις – ανομοιόμορφο πρόβλημα, περίπτωση Π2, διεπαφή 1 (πάνω) και διεπαφή 2 (κάτω).



Εικόνα 11: Σχετικές διαφορές διαδοχικών επαναλήψεων – ανομοιόμορφο πρόβλημα, περίπτωση Π2, διεπαφή 1 (πάνω) και διεπαφή 2 (κάτω).



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο

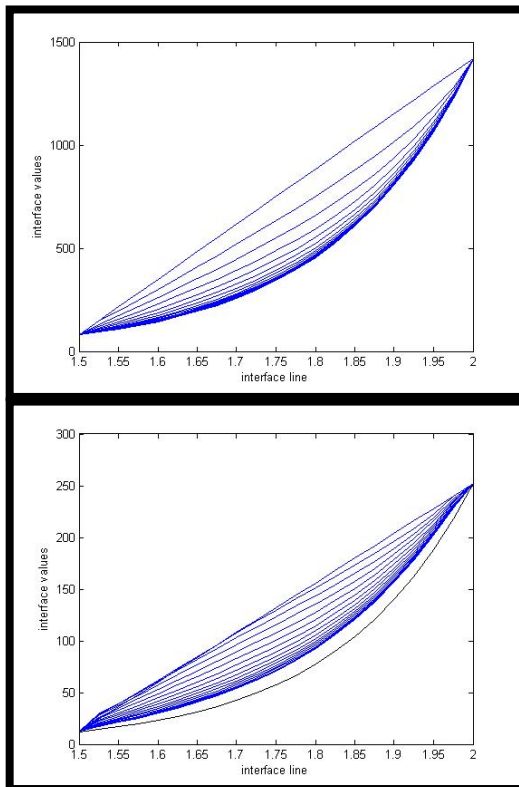


ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

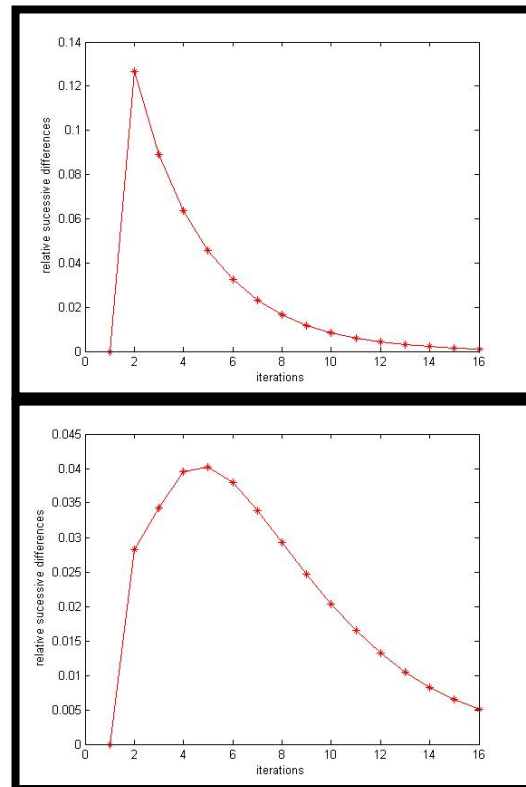
Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



Ακολουθούν οι γραφικές παραστάσεις για την περίπτωση Π3. Αρχικά για το ομοιόμορφο πρόβλημα:



Εικόνα 12: Η ακριβής και οι υπολογισμένες λύσεις – ομοιόμορφο πρόβλημα, περίπτωση Π3, διεπαφή 1 (πάνω) και διεπαφή 2 (κάτω).



Εικόνα 13: Σχετικές διαφορές διαδοχικών επαναλήψεων - ομοιόμορφο πρόβλημα, περίπτωση Π3, διεπαφή 1 (πάνω) και διεπαφή 2 (κάτω).



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο

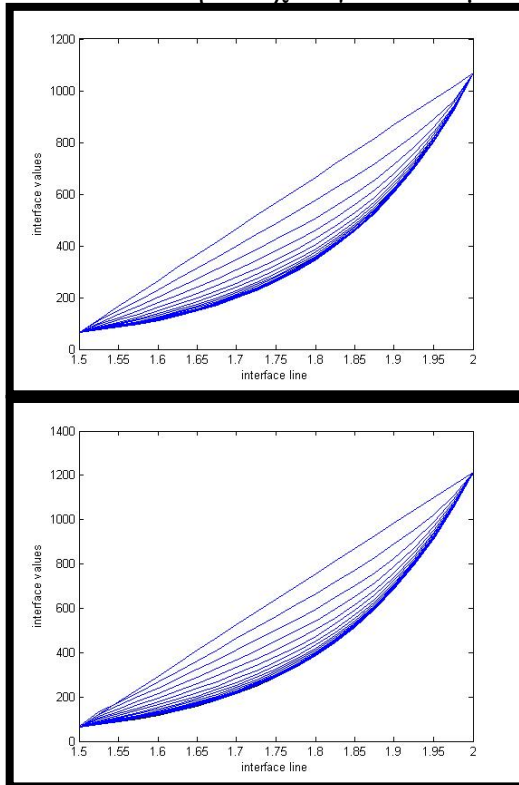


ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

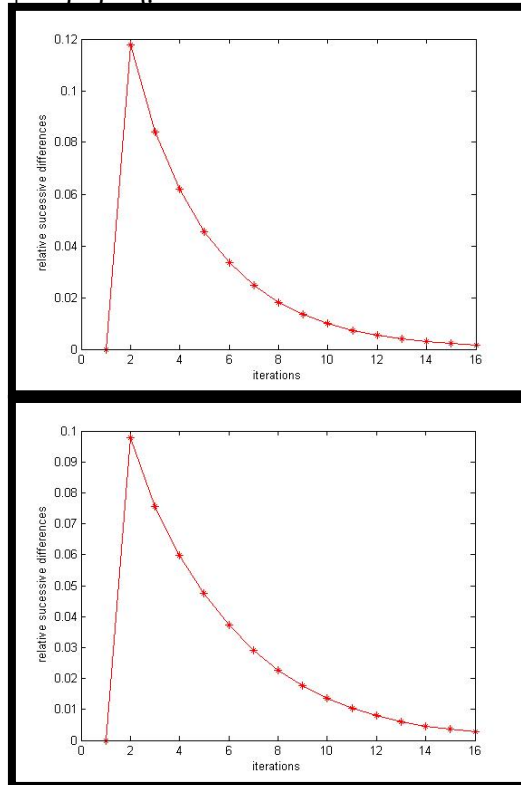
Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



Και στη συνέχεια για το ανομοιόμορφο πρόβλημα:



Εικόνα 14: Η ακριβής και οι υπολογισμένες λύσεις – ανομοιόμορφο πρόβλημα, περίπτωση Π3, διεπαφή 1 (πάνω) και διεπαφή 2 (κάτω).



Εικόνα 15: Σχετικές διαφορές διαδοχικών επαναλήψεων – ανομοιόμορφο πρόβλημα, περίπτωση Π3, διεπαφή 1 (πάνω) και διεπαφή 2 (κάτω).



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



Στον πίνακα 3 παρουσιάζονται οι συνολικοί χρόνοι εκτέλεσης της σειριακής και της παράλληλης υλοποίησης.

	Π1	Π2	Π3	Π4	Π5	Π6	Π7
Ομοιόμορφο πρόβλημα							
Σειριακή Υλοποίηση	1.900	2.664	5.216	12.883	40.883	178.217	996.348
Παράλληλη Υλοποίηση	3.495	4.068	6.140	13.595	39.808	154.633	723.731
Ανομοιόμορφο πρόβλημα							
Σειριακή Υλοποίηση	2.294	2.854	5.111	13.251	37.266	179.561	1057.644
Παράλληλη Υλοποίηση	3.720	4.264	5.732	10.633	34.172	126.607	918.838

Πίνακας 2: Χρόνοι συνολικής εκτέλεσης σειριακής και παράλληλης υλοποίησης

Για το ομοιόμορφο πρόβλημα η παράλληλη υλοποίηση αρχίζει να γίνεται γρηγορότερη από την περίπτωση Π5 και μετά, ενώ για το ανομοιόμορφο από την περίπτωση Π4 και μετά. Αυτό είναι λογικό μιας και στο ομοιόμορφο πρόβλημα ένα από τα 3 υπο-προβλήματα είναι σημαντικά μεγαλύτερο από τα υπόλοιπα και άρα είναι το κυρίαρχο στο χρόνο εκτέλεσης και της σειριακής και της παράλληλης υλοποίησης.

Η σειριακή εκτέλεση πραγματοποιείται σε έναν κόμβο με 4 Intel(R) Xeon(R) CPU E5-2620, 2.00GHz επεξεργαστές και 2GB RAM και η παράλληλη σε 3 κόμβους με την ίδια διαμόρφωση, στην Cloud Υποδομή του Εργαστηρίου Αναγνώρισης Προτύπων, του τμήματος Μηχανικών Η/Υ και Πληροφορικής, του Πανεπιστημίου Πατρών.

Ο χρόνος των σειριακών εκτελέσεων περιλαμβάνει το συνολικό χρόνο των υπολογισμών για τα τρία πεδία και τις δύο διεπαφές για 16 επαναλήψεις. Στις παράλληλες εκτελέσεις, ο συνολικός χρόνος αντιστοιχεί στο χρόνο που απαιτείται για την επίλυση του μεγαλύτερου από τα τρία υπο-πεδία μαζί με τους υπολογισμούς για τις διεπαφές και την απαραίτητη επικοινωνία, για 16 επαναλήψεις. Οι περιπτώσεις ελέγχου επιλέχθηκαν για να διερευνηθεί η συμπεριφορά της παράλληλης υλοποίησης και όχι για λόγους ακρίβειας. Όσο τα πλέγματα γίνονται λεπτότερα, ο φόρτος εργασίας αυξάνεται σημαντικά, ενώ ο χρόνος επικοινωνίας αυξάνεται με μια τάξη μεγέθους μικρότερη. Αυτός είναι ο κύριος λόγος που παρατηρείται κέρδος για λεπτά πλέγματα.

Τα αποτελέσματα της παραπάνω υλοποίησης παρουσιάστηκαν στο διεθνές συνέδριο NUMAN 2014 (Chania, Greece, 9/2014), στην εργασία: A. Korfiati, P. Tsobanopoulou and S. Likothanassis, “Serial and Parallel Implementation of the Interface Relaxation Method GEO”.

4. ΠΑΡΑΔΟΤΕΑ

1. Η παρούσα Ετήσια Τεχνική Έκθεση Προόδου φυσικού αντικείμενου.
2. Δύο (2) επιστημονικά άρθρα σε πρακτικά διεθνών συνεδρίων σχετικά με το αντικείμενο του έργου. Παρακάτω δίνονται οι αναφορές των άρθρων (το πλήρες κείμενο επισυνάπτεται):
 - **Korfiati, P. Tsompanopoulou, S. Likothanassis, “Serial and Parallel Implementation of the Interface Relaxation Method GEO”, Numerical Analysis Conference (NumAn 2014), September 2-5, Chania, Greece, 2014.**
 - **P. Alefragis, A. Spyrou and S. Likothanassis, “Application of a hybrid parallel Monte Carlo PDE Solver on rectangular multi-domains” Numerical Analysis Conference (NumAn 2014), September 2-5, Chania, Greece, 2014.**

5. ΣΥΝΕΡΓΑΣΙΕΣ

Κατά τη διάρκεια του έργου η ΚΕΟ 3, ανέπτυξε τις παρακάτω συνεργασίες:

- Με την ΚΕΟ 2
 - σε θέματα υλοποίησης της μεθόδου Interface Relaxation στην πλατφόρμα FEniCS (με την Αν. Καθηγήτρια Γ. Τσομπανοπούλου) και
 - της μεθόδου Monte Carlo σε pThreads (με τον Καθηγητή Μ. Βάβαλη και τον υποψήφιο διδάκτορα Μ. Μαρούδα).

6. ΜΕΛΛΟΝΤΙΚΕΣ ΔΡΑΣΕΙΣ

Στην τελευταία φάση του έργου θα γίνει αξιοποίηση της υποδομής Cloud, σαν περιβάλλον λειτουργίας της πλατφόρμας FEniCS, όπως έχει εμπλουτιστεί με την παράλληλη υλοποίηση των μεθόδων IR και Monte Carlo. Έχει γίνει ο σχεδιασμός της Πλατφόρμας IRaaS (Interface Relaxation as a Service), που θα αποτελεί ένα ολοκληρωμένο Περιβάλλον επίλυσης Προβλημάτων.

Η προτεινόμενη Cloud εφαρμογή είναι ένα περιβάλλον επίλυσης προβλημάτων πολλαπλών πεδίων / πολλαπλών φυσικών που υλοποιεί τη μεθοδολογία χαλάρωσης στη διεπαφή. Ο χρήστης δίνει ένα πρόβλημα σαν είσοδο, επιλέγει την IR μέθοδο που ταιριάζει καλύτερα με το συγκεκριμένο πρόβλημα και έχει πρόσβαση στη λύση σε ένα κοντά στο βέλτιστο χρόνο εκτέλεσης, από οποιοδήποτε μέρος και οποιαδήποτε συσκευή. Συγχρόνως, η εφαρμογή δεσμεύει τους ελάχιστους δυνατούς πόρους αυτόματα, στο παρασκήνιο, χωρίς παρέμβαση του χρήστη.

Για την επίτευξη των παραπάνω στόχων, εγκαταστάθηκε στις υποδομές του εργαστηρίου Αναγνώρισης Προτύπων μια συγκεκριμένη πλατφόρμα λογισμικού για την παροχή υπηρεσιών Cloud, το CloudStack. Το CloudStack αποτελεί ένα εργαλείο που ελέγχει τη δεξαμενή των υπολογιστικών πόρων καθώς επίσης διαχειρίζεται το δίκτυο και τον αποθηκευτικό χώρο. Η αρχιτεκτονική του συστήματος είναι βασισμένη στο μοντέλο IaaS και SaaS της τεχνολογίας Cloud και ταυτόχρονα στη διαχείριση της κατανομής των υπολογιστικών πόρων του Cloud έτσι ώστε να δημιουργηθούν ομάδες εικονικών μηχανών για παράλληλη και κατανεμημένη επεξεργασία. Υποστηρίζει ένα περιβάλλον για την επίλυση διαφόρων ειδών προβλημάτων και ανάλυσης δεδομένων παρέχοντας μία απλή και φιλική προς τον χρήστη διεπαφή, αυτόματη διανομή του προβλήματος στους κατάλληλους υπολογιστικούς πόρους, τον απαραίτητο αποθηκευτικό χώρο για τα δεδομένα εισόδου και των αποτελεσμάτων ενός προβλήματος καθώς και την απαιτούμενη επικοινωνία μεταξύ των συστατικών στοιχείων του συστήματος.

7. ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1] Tony F. Chan and Tarek P. Mathew. Domain decomposition algorithms. In Acta Numerica 1994, pages 61-43. Cambridge University Press, 1994.
- [2] T. F. Chan and D. C. Reasco. A survey of preconditioners for domain decomposition. Technical Report /DCS/RR-414, Yale University, 1985.
- [3] J.R. Rice, E. Vavalis, and D. Yang. Analysis of a non-overlapping domain decomposition method for elliptic PDEs. J. Comput. Appl. Math., 87:11-19, 1997.
- [4] Tsompanopoulou, P., Vavalis, E.: Analysis of an interface relaxation method for composite elliptic differential equations. Journal of Computational and Applied Mathematics 226 2, 370–387 (2009).
- [5] Logg, A., Mardal, K. A., Wells, G. N. et al.: Automated Solution of Differential Equations by the Finite Element Method. Springer, (2012).
- [6] Rabbitmq, 2014. Online. Available: www.rabbitmq.com/documentation.html(accessed July 11, 2014).
- [7] Logg, A., Wells G. N.: DOLFIN: Automated Finite Element Computing. ACM Transactions on Mathematical Software 37, 2 (2010).

ΠΑΡΑΡΤΗΜΑ Ι

Ενδεικτικό τμήμα κώδικα, για την σειριακή υλοποίηση της GEO IR.

Αρχικά γίνεται η δημιουργία των υπο-πεδίων. Ενδεικτικά παρουσιάζουμε τη δημιουργία του αριστερού υπο-πεδίου για το ομοιόμορφο πρόβλημα και για την περίπτωση, όπου $h = 0.5$.



```
meshdown = RectangleMesh(0, 0, 0.3333, 2, 7, 40, 'left')
Vdown = FunctionSpace(meshdown, 'Lagrange', 1)
```

Στη συνέχεια ορίζουμε τις οριακές συνθήκες και τις συνθήκες των διεπαφών. Ενδεικτικά, η πρώτη (αριστερή) διεπαφή:

```
int1_expr = Expression('4135.5*x[1]-6077.6')
class Interface1(SubDomain):
    def inside(self, x, on_boundary):
        return on_boundary and abs(x[0] - 0.3333) <
        tol and abs(x[1] - 2) > tol and abs(x[1] - 1.5) > tol
    Gamma_4_left = DirichletBC(Vdown, int1_expr,
    Interface1())
```

Ακολουθεί ο ορισμός του παραμετρικού προβλήματος. Πάλι για το αριστερό υπο-πεδίο του ομοιόμορφου προβλήματος:

```
u = TrialFunction(Vdown)
v = TestFunction(Vdown)
f = interpolate(f_expr, Vdown)
a = (inner(nabla_grad(u), nabla_grad(v))+g2*inner(u,
v))*dx
L = f*v*dx
```

Και στη συνέχεια ο υπολογισμός της λύσης:

```
u = Function(Vdown)
solve(a == L, u, bcsdown)
```



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



Και της παραγώγου:

```
V_g = VectorFunctionSpace(meshdown, 'Lagrange', 1)
v = TestFunction(V_g)
w = TrialFunction(V_g)
a = inner(w, v)*dx
L = inner(grad(u), v)*dx
grad_u = Function(V_g)
solve(a == L, grad_u)
```

Παρουσιάζεται ο τρόπος με τον οποίο λαμβάνουμε τις τιμές της λύσης και της παραγώγου στα σημεία των διεπαφών:

```
vertex_to_dof_map =
Vdown.dofmap().vertex_to_dof_map(meshdown)
dof_to_vertex_map =
Vdown.dofmap().dof_to_vertex_map(meshdown)
m = u.vector()
dofs_at_vertices = m[dof_to_vertex_map]
coor = meshdown.coordinates()

grad_u_x, grad_u_y = grad_u.split(deepcopy=True)

for vdown in vertices(meshdown):
    ldown = coor[vdown.index()]
    ldown[0], ldown[1],
dofs_at_vertices[vdown.index()],
dofs_at_vertices_g_x_down[vdown.index()],
dofs_at_vertices_g_y_down[vdown.index()]
```

Και ακολουθεί ο τρόπος που τις επαναθέτουμε, αφού τις επεξεργαστούμε:

```

        downbound=(dofs_at_vertices[vdown.index()]+dofs_at_v
vertices_up[vup.index()])/2+omega*(-
dofs_at_vertices_g_x_down[vdown.index()]+dofs_at_vertices_g
_x_up[vup.index()])

        upbound=(dofs_at_vertices[vdown.index()]+dofs_at_verti
ces_up[vup.index()])/2+omega*(-
dofs_at_vertices_g_x_down[vdown.index()]+dofs_at_vertices_g
_x_up[vup.index()])

        u.vector().set_local(dofs_at_vertices.array()[vertex_to_do
f_map])
        downbound=Function(Vdown)
        downbound.assign(u)

```

Γράφουμε τα αποτελέσματα σε αρχεία ως εξής:

```

import numpy as np

mattxt3=file('dom1.txt', 'w')
mattxt3.write('%%iteration x y ul ulx uly \n')
for vdown in vertices(meshdown):
    ldown = coor[vdown.index()]
    mattxt3=file('dom1.txt', 'a')
    np.savetxt(mattxt3, np.c_[iter, ldown[0], ldown[1],
dofs_at_vertices[vdown.index()],
dofs_at_vertices_g_x_down[vdown.index()],
dofs_at_vertices_g_y_down[vdown.index()]], delimiter=' ',
fmt='%d %f %f %f %f %f', newline=os.linesep)

```

Και σχεδιάζουμε γραφικές παραστάσεις:

```

import matplotlib.pyplot as plt
plt.plot(uupsto175)
plt.ylabel('u [0.3333,1.75]')
plt.xlabel('iterations')
plt.show()

```



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



ΠΑΡΑΡΤΗΜΑ II

Ενδεικτικό τμήμα κώδικα, για την παράλληλη υλοποίηση της GEO IR.

Ο client αρχικοποιεί την επικοινωνία:

```
def __init__(self):
    self.credentials = pika.PlainCredentials('test', 'test')
    self.connection =
pika.BlockingConnection(pika.ConnectionParameters('150.140.1
39.89',5672,'/',self.credentials))
    self.channel = self.connection.channel()
    result=self.channel.queue_declare(exclusive=True)
    self.callback_queue = result.method.queue
    self.channel.basic_consume(self.on_response,
no_ack=True,
                                queue=self.callback_queue)
    self.connection2=
pika.BlockingConnection(pika.ConnectionParameters('150.140.1
39.89',5672,'/',self.credentials))
    self.channel2 = self.connection2.channel()

    result2=self.channel2.queue_declare(exclusive=True)
    self.callback_queue2 = result2.method.queue
    self.channel2.basic_consume(self.on_response,
no_ack=True,
                                queue=self.callback_queue2)
```

Περιμένει τις απαντήσεις από τους servers:


```
def on_response(self, ch, method, props, body):
    #print "on_response result ch=%s guess=%s" %
    (ch,self.channel)
    if ch == self.channel:
        self.response = body
    else:
        self.response2 = body
```

Αφού πρώτα τους στείλει τις τιμές στις διεπαφές που υπολόγισε επιλύοντας το πρόβλημα στην εκάστοτε επανάληψη:

```
self.response = None
self.corr_id = str(uuid.uuid4())
self.channel.basic_publish(exchange='',
                           routing_key='hello',

properties=pika.BasicProperties(
    reply_to =
self.callback_queue,
    correlation_id =
self.corr_id,
    ),
    body=inter2)
```

Αντίστοιχα ο κάθε server επιλύει το πρόβλημά του, λαμβάνει από τον client τη λύση και την παράγωγο που υπολόγισε στη διεπαφή και υπολογίζει τις καινούργιες τιμές στη διεπαφή:

```
response=(interfaceup+interfaced)/2+omega*(-
gradientd+gradientup)
```

Το αποτέλεσμα αυτό το χρησιμοποιεί για την επόμενη επανάληψή του. Το στέλνει, όμως και στον client για να προχωρήσει και αυτός με την επόμενη επανάληψη του.



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΕΚΠΑΙΔΕΥΣΗ ΚΑΙ ΔΙΑ ΒΙΟΥ ΜΑΘΗΣΗ
επένδυση στην κοινωνία της γνώσης

ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



ΕΣΠΑ
2007-2013
πρόγραμμα για την ανάπτυξη
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ

```
ch.basic_publish(exchange="",  
                  routing_key=props.reply_to,  
                  properties=pika.BasicProperties(correlation_id = \  
                                                    props.correlation_id),  
                  body=json.dumps(response.tolist()))  
ch.basic_ack(delivery_tag = method.delivery_tag)
```



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

