

Προσομοίωση Τετραδιάστατης
Συμπαγούς Θεωρίας Βαθμίδας $U(1)$
στο Grid με τον Αλγόριθμο
Πλακετών

Σκίπης Δημήτρης

3 Οκτωβρίου 2006

Λέξεις Κλειδιά - Key Words

Αλγόριθμος των πλακετών, αλγόριθμος των συνδέσμων, προσομοιώσεις Monte Carlo, αλγόριθμος Metropolis, προσομοίωση κβαντικής θεωρίας βαθμίδας, κβαντική θεωρία βαθμίδας σε πλέγμα, Grid

Plaquette algorithm, link algorithm, Monte Carlo simulations, Metropolis algorithm, gauge theory simulation, lattice quantum field theory, Grid

Περίληψη - Summary

Η παρούσα διπλωματική εργασία αφορά την πιστοποίηση του αλγόριθμου των πλακετών για την προσομοίωση κβαντικών θεωριών βαθμίδας $U(1)$ σε πλέγμα, τον οποίο δημιούργησε ο συγγραφέας. Παρουσιάζονται ενδεικτικά στοιχεία στατιστικής μηχανικής καθώς κβαντικών θεωριών πεδίου. Επίσης παρουσιάζονται τα αποτελέσματα που πιστοποιούν την ορθότητα του αλγόριθμου και συγκριτικά του αλγόριθμου με παλαιότερους. Τέλος δίνεται μία εκτενής παρουσίαση του Grid, καθώς ήταν το υπολογιστικό περιβάλλον στο οποίο εκτελέστηκαν όλοι οι υπολογισμοί της παρούσας εργασίας.

The current diploma thesis is concerned with the validation of the plaquette algorithm for the simulation of $U(1)$ gauge field theories on a lattice, which the author created. In the beginning statistical mechanics and quantum gauge theories are presented. Then the results are given that validate the correctness of the algorithm, as well as, comparatives and benchmarks with the older algorithm of links. In the end, an extensive presentation is given of the Grid, which was the computational environment in which all the calculations for this diploma thesis were undertaken.

Αφιερώνεται στους δασκάλους μου

Περιεχόμενα

Ευχαριστίες	vi
Πρόλογος	viii
1 Στοιχεία στατιστικής μηχανικής	1
1.1 Η συνάρτηση επιμερισμού	2
1.2 Προσομοιώσεις Monte Carlo	4
1.2.1 Η εκτιμήτρια	5
1.2.2 Δειγματοληψία σημαντικότητας	6
1.2.3 Διαδικασίες Markov	6
1.2.4 Εργοδικότητα	7
1.2.5 Λεπτομερής ισορροπία	7
1.2.6 Λόγοι αποδοχής	7
1.3 Η μέθοδος Metropolis και το μοντέλο Ising	9
2 Κβαντική Θεωρία πεδίου σε πλέγμα και προσομοιώσεις Monte Carlo	13
2.1 Περί πλεγμάτων	13
2.2 Συναρτησιακό ολοκλήρωμα και στατιστική μηχανική	14
2.3 Αβελιανά πεδία βαθμίδας και Κβαντική Ηλεκτροδυναμική	18
2.4 Κβαντική θεωρία πεδίου στο πλέγμα	19
2.5 Ο κλασικός αλγόριθμος πάνω στους συνδέσμους	22
3 Το υπολογιστικό περιβάλλον του Grid	24
3.1 Τι είναι το Grid;	24
3.2 Το LCG	25
3.3 Η αρχιτεκτονική του LCG	26
3.3.1 Γενικά	26
3.3.2 Η διεπαφή του χρήστη	26
3.3.3 Υπολογιστικά Στοιχεία και Στοιχεία Αποθήκευσης	27
3.3.4 Το Σύστημα Πληροφόρησης	27
3.3.5 Διαχείριση δεδομένων	28
3.3.6 Διαχείριση εργασιών	29
3.3.7 Η ροή των εργασιών	29
3.4 Τα πιστοποιητικά proxy certificates	31
3.5 Διαχείριση Εργασιών	34
3.5.1 Η γλώσσα ορισμού εργασιών JDL	34

3.5.2	Υποβολή εργασίας	35
3.5.3	Λειτουργίες εργασίας	36
4	Ο αλγόριθμος των πλακετών	39
4.1	Θεωρητική υποστήριξη	42
4.2	Η αρχή της λεπτομερούς ισορροπίας	43
4.3	Υλοποίηση και βελτιστοποιήσεις	44
4.3.1	Θέματα υλοποίησης	44
4.3.2	Βελτιστοποιήσεις	46
4.4	Σύγκριση χρόνων αυτοσυσχέτισης	48
4.5	Σύγκριση επιδόσεων ως προς τον χρόνο	50
5	Αποτελέσματα και ανάλυση αποτελεσμάτων	51
5.1	Παρατηρήσιμες ποσότητες	51
5.2	Συγκριτικά αποτελεσμάτων	54
6	Συμπεράσματα	56
Α'	Η μέθοδος multihistogram	57
Β'	Η μέθοδος jackknife	59
Γ'	Πηγαίος κώδικας του προγράμματος mapper	61
Δ'	Πηγαίος κώδικας του προγράμματος sweeper	64
Ε'	Πηγαίος κώδικας του προγράμματος cumulants	68

Ευχαριστίες

Η παρούσα εργασία αποτελεί την Διπλωματική Εργασία μου στα πλαίσια των σπουδών μου στην Σχολή των Ηλεκτρολόγων Μηχανικών και Μηχανικών Ηλεκτρονικών Υπολογιστών. Η εκπόνησή της πραγματοποιήθηκε από τον Ιούνιο του 2005 έως τον Φεβρουάριο του 2006, υπό την επίβλεψη των καθηγητών του Εθνικού Μετσοβίου Πολυτεχνείου Δρ. Κ. Φαράκου, Δρ. Γ. Κουτσούμπα και Δρ. Ν. Κοζύρη.

Με την ευκαιρία της παρουσίας της εργασίας μου, θεωρώ χρέος μου να εκφράσω τις ευχαριστίες μου σε όσους συνέβαλαν αποφασιστικά στην προσπάθεια. Έτσι, θα ήθελα, καταρχήν, να ευχαριστήσω το φίλο και συνεργάτη μου Ορέστη Βάντζο για την μεγάλη συνεισφορά του στην διαμόρφωση του αλγόριθμου, που ήταν το αντικείμενο της παρούσας διπλωματικής.

Επίσης, θα ήθελα να ευχαριστήσω ιδιαίτερος τον Δρ. Κωνσταντίνο Φαράκο, που μου δίδαξε το αντικείμενο των Κβαντικών Θεωριών σε Πλέγμα και με κατεύθυνε βήμα προς βήμα κατά την εκπόνηση της διπλωματικής μου εργασίας. Η καθοδήγηση και οι συνεχείς συμβουλές του, αποτέλεσμα της μεγάλης του πείρας στο αντικείμενο, καθώς και το άριστο κλίμα συνεργασίας μας, ήταν αποφασιστικά στοιχεία για την επιτυχή ολοκλήρωση της εργασίας.

Ευχαριστίες, επιπλέον, πρέπει να δώσω στον Δρ. Γεώργιο Κουτσούμπα για τις εύστοχες παρατηρήσεις του και τη καθοδήγησή που και εκείνος μου προσέφερε, στον Δρ. Νεκτάριο Κοζύρη και την ομάδα υποστήριξης του κόμβου Isabella του Grid για την υλικοτεχνική υποστήριξη που είχα (Χωρίς το Grid αυτή η διπλωματική εργασία δεν θα ήταν εφικτή.), στον Δρ. Κωνσταντίνο Αναγνωστόπουλο για τις πολυτιμότες παρατηρήσεις και συμβουλές του σε ό,τι αφορούσε τους αλγόριθμους Monte Carlo, και στον διδακτορικό φοιτητή Σπύρο Βρέντζο για την συνεργασία μας.

Τέλος, θα ήθελα να ευχαριστήσω την οικογένειά μου καθώς τους καθηγητές μου στο Πολυτεχνείο Δρ. Ευάγγελο Γαζή, Δρ. Νικόλαο Τράκα, Δρ. Σωφρόνιο Παπαδόπουλο και Δρ. Γεώργιο Ζουπάνο για την υποστήριξη που μου έδειξαν καθόλη την δύσκολη φοίτησή μου στην Σχολή των Ηλεκτρολόγων Μηχανικών.

Και φυσικά δεν θα μπορούσα ποτέ να παραβλέψω τους Νίκο Δημητρόπουλο, Δώρα και Βαγγελιώ Κοντογιάννη, Τάσο Δημά, Σοφρά Νικολέττα και Isabel Llorente-Garcia, η φίλια των οποίων είναι το πολυτιμότερο πράγμα που θα έχω ποτέ.

Παρατηρήσεις επί του κειμένου

Δυστυχώς, πολλές φορές η ελληνική γλώσσα δεν έχει κατάλληλους ελληνικούς όρους για να αντιστοιχίσει σε επιστημονικούς όρους που χρησιμοποιούνται στην διεθνή βιβλιογραφία. Για αποφυγή παρεξηγήσεων, λοιπόν, την πρώτη φορά σε κάθε κεφάλαιο που επιχειρείται η μετάφραση κάποιου διεθνούς όρου στην ελληνική, θα παρατίθεται σε παρενθέσεις και ο αντίστοιχος διεθνής όρος.

Πρόλογος

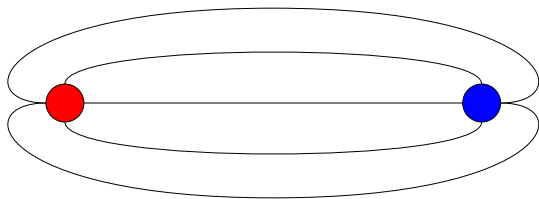
Η παρούσα εργασία αφορά την μελέτη της τετραδιάστατης συμπαγούς θεωρίας βαθμίδας $U(1)$ (compact $U(1)$ gauge theory) χωρίς φερμιόνια με τον αλγόριθμο πλακετών. Ο αλγόριθμος πλακετών δημιουργήθηκε το καλοκαίρι του 2005 από τον συγγραφέα και τον Ορέστη Βάντζο, όταν ο πρώτος μελετούσε το πρόβλημα των συμπαγών θεωριών βαθμίδας σε πλέγμα χωρίς φερμιόνια και ο δεύτερος είχε ορισμένες πρωτοποριακές ιδέες. Πρωταρχικός σκοπός της εργασίας και του παρόντος κειμένου είναι η πιστοποίηση του αλγορίθμου των πλακετών και η σύγκριση των ιδιοτήτων του σε σχέση με τους υπάρχοντες αλγόριθμους. Ο αλγόριθμος παρουσιάζεται αναλυτικά στο σχετικό κεφάλαιο, με μια μικρή αναφορά σε θεωρητική υποστήριξη του καθώς και τα υπολογιστικά αποτελέσματα που έδωσε. Επίσης, καθώς η όλη εκπόνηση της εργασίας έγινε με την βοήθεια της νεοεμφανιζόμενης τεχνολογίας του Grid, ένα μεγάλο τμήμα του παρόντος κειμένου αφιερώνεται και στην παρουσίαση αυτής της τεχνολογίας.

Πριν παρουσιαστούν όμως ενδελεχώς τα αντικείμενα της εργασίας, κρίνεται σκόπιμη μια μικρή ιστορική αναδρομή των χβαντικών θεωριών σε πλέγμα (lattice gauge theories). Η ιστορική αυτή αναδρομή δείχνει καθαρά την ανάγκη, η οποία οδήγησε στην δημιουργία των σχετικών θεωριών και των επιτυχιών που έχουν αυτές έως σήμερα. Είναι βασικό ο αναγνώστης να καταλάβει ότι οι θεωρίες σε πλέγμα δεν είναι ένα απλό μαθηματικό τέχνασμα ώστε να γίνουν δυνατοί κάποιοι υπολογισμοί, αλλά υπάρχουν σοβαροί κυρίως θεωρητικοί λόγοι που κάνουν τις θεωρίες σε πλέγμα ισοδύναμες με τις αναλυτικές χβαντικές θεωρίες πεδίου, και σε πολλές περιπτώσεις πολύ πιο επιτυχείς από αυτές.

Κατά τα τέλη της δεκαετίας του '60, λοιπόν, η χβαντική ηλεκτροδυναμική ήταν ήδη αρκετά επιτυχής, αλλά η βασική θεωρία είχε ήδη "γίνει". Δύσκολοι υπολογισμοί παρέμεναν ακόμα να γίνουν, και παραμένουν ακόμα, παρόλα αυτά όμως δεν υπήρχε κάποια βασική εννοιολογική καινοτομία.

Αυτά ήταν τα χρόνια στα οποία ο "οκταπλός τρόπος" (eightfold way) για την περιγραφή πολλαπλοτήτων σωματιδίων κέρδιζε ευρεία αναγνώριση. Η ιδέα των κουάρκ υπήρχε τριγύρω, αλλά παράλληλα υπήρχε και μεγάλος σκεπτικισμός για το αν μπορούσε κάποιος να της αναθέσει κάποια φυσική υπόσταση· ίσως ήταν τελικά απλά κάποιο μαθηματικό τρίκ. Μερικοί φυσικοί δούλευαν πάνω στις ασθενείς αλληλεπιδράσεις και η βασική ηλεκτροασθενής ενοποίηση είχε αρχίσει να αναδύεται. Στα πειράματα στο SLAC παρατηρούνταν σημαντικές ανελαστικές σκεδάσεις ηλεκτρονίου-πρωτονίου σε μεγάλες γωνίες, και αυτό μεταφράστηκε σαν ένδειξη για δομή κατωτέρου επιπέδου. Έτσι ο όρος παρτόνιο (parton) άρχισε να μπαίνει στο προσκήνιο. Και παρόλο που συμπτωματικά υπήρχαν συλλογισμοί που συνέδεαν τα κουάρκ με τα παρτόνια, οι φυσικοί έτειναν να είναι επιφυλακτικοί στο να κάνουν τέτοιες αναγωγές.

Ένα σημαντικό χαρακτηριστικό εκείνης της περιόδου ήταν ότι η επέκταση της χβαντικής ηλεκτροδυναμικής σε μια θεωρία πεδίου μεσονίων-νουκλεονίων αποτύγχανε παταγωδώς.



Σχήμα 1: Ένας “σωλήνας” γλουονικής ροής που συνδέει ένα κουάρκ με ένα αντικουάρκ. Η τάση του δεσμού είναι 14 τόνοι.

Το ανάλογο της ηλεκτρομαγνητικής σταθεράς σύζευξης (electromagnetic coupling constant) υπολογιζόταν περίπου στο 15, σε αντίθεση με την αποδεκτή τιμή των $1/137$. Αυτό σήμαινε ότι οι διορθώσεις υψηλότερης τάξης στις διαταρακτικές διαδικασίες ήταν σημαντικά μεγαλύτερες από τους υπολογισμούς πρώτης τάξης. Δεν υπήρχε κάποια γνωστή στοιχειώδης παράμετρος στην οποία μπορούσε κάποιος να αναπτύξει.

Καθώς μπαίναμε στην δεκαετία του '70, τα παρτόνια όλο και πιο συχνά ταυτίζονταν με τα κουάρκ. Αυτή η μεταστροφή εμφανίστηκε λόγω δύο δραματικών θεωρητικών επιτυχιών. Πρώτη ήταν η απόδειξη της επανακανονικοποιησιμότητας (renormalizability) των μη Αβελιανών θεωριών βαθμίδας. Το γεγονός της επανακανονικοποιησιμότητας έδινε την σιγουριά ότι αυτές οι κομψές μαθηματικές δομές μπορεί να έχουν κάτι να κάνουν με την πραγματικότητα. Δεύτερον ήταν η ανακάλυψη της ασυμπτωτικής ελευθερίας, του γεγονότος δηλαδή ότι οι αλληλεπιδράσεις των μη Αβελιανών θεωριών γίνοντουσαν ασθενέστερες σε μικρές αποστάσεις. Πραγματικά, αυτό γρήγορα συνδέθηκε με τις σημειακές δομές για τις οποίες υπήρξαν ενδείξεις στα πειράματα του SLAC. Μέσα από αυτές τις ιδέες αναπτύχθηκε η κβαντική χρωμοδυναμική, η θεωρία της δυναμικής των κουάρκ.

Η εγκυρότητα αυτής της εικόνας εξαρτόταν από την έννοια του “περιορισμού” (confinement). Ενώ υπήρχαν σημαντικές ενδείξεις για την υφιστάμενη δομή των κουάρκ, δεν είχαν παρατηρηθεί ποτέ ελεύθερα κουάρκ. Αυτό ήταν αρκετά προβληματικό δεδομένης της σχεδόν ελεύθερης φύσης των αλληλεπιδράσεών τους μέσα στα νουκλεόνια. Αυτό αναδεικνύει την ερώτηση του τί είναι στοιχειώδες. Είναι τα φυσικά σωματίδια που βλέπουμε στο εργαστήριο τα στοιχειώδη αντικείμενα ή είναι τα κουάρκ και τα γλουόνια;

Η πάλη με αυτό το παράδοξο οδήγησε στην πλέον καθιερωμένη εικόνα του σωλήνα ροής (flux tube) του περιορισμού. Τα γλουονικά πεδία είναι ανάλογα των φωτονίων με την διαφορά ότι φέρουν φορτίο το ένα σε σχέση με το άλλο. Ο προαναφερθής σωλήνας είναι ένα πραγματικό αντικείμενο, το οποίο μεγαλώνει σε μήκος όσο το κουάρκ και το αντικουάρκ απομακρύνονται το ένα από το άλλο. Η παραγόμενη δύναμη είναι σταθερή σε μεγάλες αποστάσεις και σε φυσικές μονάδες είναι περίπου 14 τόνοι.

Ο λόγος για τον οποίο ένα κουάρκ δεν μπορεί να απομονωθεί είναι ίδιος για τον οποίο ένα κομμάτι κλωστής δεν μπορεί να έχει μόνο μία άκρη. Το φαινόμενο του περιορισμού δεν μπορεί να παραχθεί στην θεωρία διαταραχών· όταν η σύζευξη “σβήνεται” το φάσμα γίνεται ελεύθερα κουάρκ και γλουόνια, τα οποία είναι δραματικά διαφορετικά από τα πιόνια και τα πρωτόνια της θεωρίας με τις αλληλεπιδράσεις.

Τα μέσα της δεκαετίας του '70 υπήρξαν μια ενθουσιώδης περίοδος για την σωματιδιακή φυσική, με μία σειρά δραματικών γεγονότων να φέρνουν επανάσταση στον χώρο. Πρώτον ήταν η ανακάλυψη του σωματιδίου J/ψ . Η ερμηνεία αυτού του αντικειμένου και των συντρόφων του σαν δέσιμες καταστάσεις βαριών κουάρκ παρείχε το αντίστοιχο του από-

μου του υδρογόνου στην κβαντική χρωμοδυναμική. Η ιδέα των κουάρκ έγινε πλέον ακαταμάχητη και έτσι η κβαντική θεωρία πεδίου “ζαναγεννήθηκε”. Η μη Αβελιανή θεωρία βαθμίδας $SU(3)$ των ισχυρών αλληλεπιδράσεων συνδυασμένη με την ηλεκτρασθενή θεωρία έγινε το ανθεκτικό στο χρόνο “Καθιερωμένο Πρότυπο” (Standard Model).

Αυτή η περίοδος όμως έγινε θεατής και σε ορισμένες αξιοσημείωτες διαπιστώσεις στο θεωρητικό επίπεδο. Μη γραμμικές συμπεριφορές σε διάφορες κλασικές θεωρίες πεδίου έδειξαν ότι έχουν σοβαρές επιπτώσεις για τα κβαντικά αντίστοιχά τους. Η ανακάλυψη τέτοιων φαινομένων δημιούργησε σοβαρές υπόνοιες: η θεωρία πεδίου μπορεί να έχει πολύ περισσότερη δομή από ότι φαίνεται από μια απλή ανάλυση με διαγράμματα Feynman. Αλλά αυτό με την σειρά του έχει σοβαρά αποτελέσματα για τους πρακτικούς υπολογισμούς. Η κβαντική θεωρία είναι διαβόητη για τους απειρισμούς της, που απαιτούν κανονικοποίηση. Η “γυμνή” μάζα και το φορτίο που υπεισέρχονται στην θεωρία είναι εξ αρχής άπειρες ποσότητες. Δεν είναι οι φυσικά παρατηρήσιμες οντότητες, οι οποίες πρέπει να καθοριστούν με όρους φυσικών διαδικασιών. Για να γίνει κάποιος υπολογισμός, απαιτείται κάποιος “ρυθμιστής” (regulator) για να δαμάσει τους απειρισμούς, και όταν φυσικές ποσότητες συνδέονται η μία με την άλλη, κάθε εξάρτηση από “ρυθμιστές” θα πρέπει να απαλείφεται.

Η ανάγκη για τον χειρισμό των απειρισμών ήταν, φυσικά, γνωστή από τις αρχές της κβαντικής ηλεκτροδυναμικής. Αλλά όλοι οι κοινοί “ρυθμιστές” βασιζόνταν στα διαγράμματα Feynman: οι θεωρητικοί υπολόγιζαν τα διαγράμματα μέχρι κάποιο να αποκλίνει και μετά εκείνο το διάγραμμα θα αποκοβόταν. Πολλά σενάρια κατασκευάστηκαν για αυτό τον σκοπό. Με την αυξανόμενη όμως συνειδητοποίηση ότι μη διαταρακτικά φαινόμενα είναι αρκετά σημαντικά, έγινε προφανές ότι χρειαζόταν η ύπαρξη ενός “μη διαταρακτικού ρυθμιστή” (non-perturbative regulator), ο οποίος δεν θα εξαρτόταν από τα διαγράμματα.

Το απαραίτητο εργαλείο εμφανίστηκε στην θεωρία πλέγματος του Wilson. Ο Wilson το παρουσίασε σαν ένα παράδειγμα ενός μοντέλου με περιορισμό. Η ανάπτυξη με βάση την ισχυρή σταθερά ζεύξης είχε πεπερασμένη ακτίνα σύγκλισης, επιτρέποντας μια σαφή επίδειξη του περιορισμού αν και σε ένα μη φυσικό όριο. Το παραγώμενο φάσμα είχε ακριβώς τις επιθυμητές ιδιότητες.

Ο πρωταρχικός ρόλος του πλέγματος είναι να παράσχει μια μη διαταρακτική αποκοπή (non-perturbative cutoff). Καθώς ο χώρος δεν μπορεί να είναι πραγματικά ένας κρύσταλλος, το πλέγμα είναι ένα καθαρά μαθηματικό τέχνασμα. Παρέχει ένα ελάχιστο μήκος κύματος μέσω του βήματος του πλέγματος a . Τα αθροίσματα σε διαδρομές γίνονται καλά ορισμένα απλά ολοκληρώματα. Αποφεύγοντας τις δυσκολίες σύγκλισης της διαταρακτικής θεωρίας, το πλέγμα παρέχει ένα σαφές μονοπάτι προς τον ορισμό μιας κβαντικής θεωρίας πεδίου.

Αυτή η προσέγγιση όμως είχε και ένα καταπληκτικό παρελκόμενο. Κάνοντας το σύστημα προσεκτικά διακριτό, γίνεται επαρκώς καλά ορισμένο ώστε να μπορεί να μεταφραστεί στην γλώσσα των ηλεκτρονικών υπολογιστών. Αυτό έγινε αρκετά άμεσα και κατά την ίδια περίοδο κατά την οποία οι υπολογιστές γινόντουσαν αρκετά ισχυροί.

Φτάνοντας λοιπόν στο σήμερα, έχουμε ένα αρκετά “δουλεμένο” φορμαλισμό, ο οποίος μπορεί να αποφανθεί για την φυσική σε μη διαταρακτικό επίπεδο, εκεί που ο κοινός φορμαλισμός δεν έχει και πολλά να πει. Επίσης, καθώς ο φορμαλισμός αυτό είναι άμεσα μεταφράσιμος στην γλώσσα των υπολογιστών, το “οπτικό πεδίο” της θεωρίας έχει γίνει πολύ μεγάλο. Οι σύγχρονοι υπολογιστές, ακόμα και οι προσωπικοί, έχουν τέτοια υπολογιστική ισχύ που επιτρέπουν τον υπολογισμό μοντέλων σε αρκετά μικρούς χρόνους. Εκτός όμως από τέτοιους υπολογιστικούς πόρους, που πλέον κάθε φυσικός έχει στην διάθεσή του, την σήμερα ημέρα εμφανίζεται μια καινούργια καινοτομία στον χώρο του scientific

computing. Η καινοτομία αυτή είναι το Grid, και ανεβάζει σε τελείως άλλη κλίμακα την δυνατότητα κατανεμημένων υπολογισμών, που μέχρι τώρα δεν την απολάμβαναν όλοι οι ερευνητές. Με το Grid, ο φυσικός έχει πλέον στην διάθεσή του ένα τεράστιο πλήθος υπολογιστικών και αποθηκευτικών πόρων τους οποίους μπορεί να χρησιμοποιήσει και να περάσει την έρευνά του σε άλλο επίπεδο.

Κατά την εκπόνηση αυτής της εργασίας ήμασταν αρκετά τυχεροί ώστε να έχουμε στην διάθεσή μας αυτή την καινούργια τεχνολογία, η οποία πραγματικά ελάττωσε σημαντικά τους χρόνους εκτέλεσης των προγραμμάτων δίνοντας την δυνατότητα εξερεύνησης μεγαλύτερων όγκων στο πλέγμα και ανώτερων διαστάσεων. Φυσικά, κάτι τέτοιο, χωρίς την τεχνολογία του Grid, θα ήταν απλά αδύνατο. Χαρακτηριστικά, ο συνολικός υπολογιστικός χρόνος για την διεξαγωγή της εργασίας υπολογίζεται ότι ξεπέρασε τα πέντε χρόνια συνεχούς λειτουργίας ενός απλού επεξεργαστή!

Η παρούσα εργασία αποτελείται από πέντε μέρη. Στο πρώτο παρουσιάζεται μία σύντομη εισαγωγή σε βασικές έννοιες της στατιστικής μηχανικής, πάνω στις οποίες χτίζεται το οικοδόμημα των κβαντικών θεωριών πεδίου σε πλέγμα. Στο δεύτερο μέρος δίνεται μια συνεκτική εισαγωγή στον φορμαλισμό της συμπαγούς κβαντικής θεωρίας πεδίου $U(1)$ χωρίς φερμιόνια καθώς και ο τρόπος με τον οποίον η θεωρία μεταφέρεται στον υπολογιστή για να προσομοιωθεί κατά Monte Carlo. Στο τρίτο μέρος γίνεται μια διεξοδική αναφορά στην τεχνολογία του Grid όπως αυτή χρησιμοποιήθηκε στα πλαίσια της εργασίας. Στο τέταρτο μέρος παρουσιάζεται ο αλγόριθμος των πλακετών, τόσο με μια μικρή αναφορά σε θεωρητική υποστήριξή του, όσο και με την υλοποίησή του σε υπολογιστικό περιβάλλον και την μετέπειτα σύγκρισή του, όσον αφορά τις επιδόσεις, με απλούς αλγόριθμους, και τέλος στο πέμπτο μέρος παρουσιάζονται τα αποτελέσματα που έδωσε ο αλγόριθμος για την συμπαγή τετραδιάστατη θεωρία βαθμίδας $U(1)$ χωρίς φερμιόνια και συγκρίνονται με αντίστοιχα αποτελέσματα άλλων αλγορίθμων από την σχετική βιβλιογραφία.

Κεφάλαιο 1

Στοιχεία στατιστικής μηχανικής

Στο κεφάλαιο αυτό παρουσιάζονται κάποια βασικά θέματα της στατιστικής μηχανικής. Καθώς η θεωρία πεδίου σε πλέγμα τελικά καταλήγει στην μελέτη ενός στατιστικού φυσικού συστήματος, είναι σκόπιμο να αναφερθούμε στις βασικές έννοιες της θεωρίας στατιστικών φυσικών συστημάτων καθώς και στους αλγόριθμους που χρησιμοποιούνται για την μελέτη τέτοιων συστημάτων.

Η στατιστική μηχανική κυρίως ενδιαφέρεται για τον υπολογισμό ιδιοτήτων συστημάτων συμπυκνωμένης ύλης. Η βασική δυσκολία που συνδέεται με τέτοια συστήματα είναι ότι αποτελούνται από πάρα πολλά συνιστώσα στοιχεία, τυπικά άτομα ή μόρια. Αυτά τα στοιχεία συνήθως είναι όλα τα ίδια ή ενός μικρού αριθμού διαφορετικών ειδών και συχνά υπακούουν σε σχετικά απλές εξισώσεις κίνησης, έτσι ώστε η συμπεριφορά ολόκληρου του συστήματος να μπορεί να εκφραστεί μαθηματικά με έναν άμεσο τρόπο. Όμως ο τεράστιος αριθμός των εξισώσεων κάνει αδύνατη την ακριβή επίλυση των σχετιζομένων μαθηματικών. Παρόλο όμως, που δεν είναι εφικτή η επίλυση των εξισώσεων του Hamilton για αυτά τα συστήματα, διότι απλά είναι πάρα πολλές, μπορούμε να παρατηρήσουμε μακροσκοπικές ιδιότητες του υλικού, οι οποίες είναι καλά ορισμένες και προβλέψιμες. Προφανώς, υπάρχει κάτι ξεχωριστό στην συμπεριφορά των λύσεων αυτών των υπεράριθμων εξισώσεων το οποίο αναδεικνύεται σαν μέσος όρος για να μας δώσει την προβλέψιμη συμπεριφορά όλου του συστήματος. Η στατιστική μηχανική προσπαθεί να παρακάμψει το πρόβλημα της επίλυσης των εξισώσεων κίνησης και πάει απευθείας στον υπολογισμό μακροσκοπικών ιδιοτήτων μεγάλων συστημάτων μεταχειριζόμενη αυτά με πιθανοκρατικό τρόπο. Αντί να ψάχνουμε για ακριβείς λύσεις, έχουμε να κάνουμε με τις πιθανότητες του συστήματος να βρίσκεται στην μία κατάσταση ή την άλλη. Τέτοιοι πιθανολογικοί ισχυρισμοί αποδεικνύεται ότι είναι ιδιαίτερα χρήσιμοι, διότι συνήθως βλέπουμε ότι για μεγάλα συστήματα το πεδίο των συμπεριφορών του συστήματος, οι οποίες δεν είναι πιθανές είναι πολύ μικρό. Όλες οι λογικά πιθανές συμπεριφορές εμπίπτουν σε μια στενή περιοχή, το οποίο μας επιτρέπει με αρκετά μεγάλη σιγουριά να δηλώνουμε ότι το σύστημα θα δείξει κάποια συμπεριφορά εντός αυτής της περιοχής.

Το τυπικό παράδειγμα τέτοιων μακροσκοπικών συστημάτων είναι αυτό ενός συστήματος που ορίζεται από μια χαμιλτονιανή συνάρτηση H , η οποία μας δίνει την ολική ενέργεια του συστήματος σε μια συγκεκριμένη κατάσταση. Ας υποθέσουμε ότι το σύστημα αυτό είναι στην κατάσταση μ . Ας ορίσουμε την πιθανότητα το σύστημα να είναι στην κατάσταση ν μετά από χρόνο dt ως $R(\mu \rightarrow \nu)dt$. Αυτή η ποσότητα ονομάζεται ο ρυθμός μετάβασης (transition rate) για την μετάβαση του συστήματος από την κατάσταση μ στην ν . Ο ρυθμός μετάβασης συνήθως θεωρείται ότι είναι χρονοανεξάρτητος, κάτι το οποίο θα υποθέ-

σουμε και στα παρακάτω, όταν οι αλληλεπιδράσεις δεν εξαρτώνται από το χρόνο. Αυτοί οι ρυθμοί μετάβασης είναι συνήθως τα μόνα πράγματα που ξέρουμε σε σχέση με την δυναμική. Αυτό σημαίνει ότι ακόμη και αν γνωρίζουμε την αρχική κατάσταση του συστήματος, αρκεί μόνο να περιμένουμε ικανό χρονικό διάστημα και μετά το σύστημα θα μπορεί να είναι σε οποιαδήποτε από έναν πάρα πολύ μεγάλο αριθμό πιθανών καταστάσεων. Εδώ λοιπόν έρχεται να αναλάβει δράση η πιθανοθεωρητική αντιμετώπιση του προβλήματος. Ορίζουμε έναν σύνολο βαρών $w_\mu(t)$ τα οποία αντιπροσωπεύουν την πιθανότητα το σύστημα να είναι στην κατάσταση μ τον χρόνο t . Μπορούμε να γράψουμε μια κύρια εξίσωση για την εξέλιξη των $w_\mu(t)$ σε όρους των ρυθμών μετάβασης $R(\mu \rightarrow \nu)$:

$$\frac{dw_\mu}{dt} = \sum_\nu [w_\nu(t)R(\nu \rightarrow \mu) - w_\mu(t)R(\mu \rightarrow \nu)]. \quad (1.1)$$

Ο πρώτος όρος του δεξιού μέλους αυτής της εξίσωσης αντιπροσωπεύει τον ρυθμό με τον οποίον το σύστημα μεταβαίνει στην κατάσταση μ , ο δεύτερος όρος είναι ο ρυθμός με τον οποίον το σύστημα μεταβαίνει από την κατάσταση μ σε άλλες καταστάσεις. Οι πιθανότητες $w_\mu(t)$ πρέπει επίσης να υπακούουν τον αθροιστικό κανόνα:

$$\sum_\mu w_\mu(t) = 1 \quad (1.2)$$

για κάθε t , καθώς το σύστημα πρέπει πάντα να είναι σε κάποια κατάσταση. Η λύση της εξίσωσης (1.1), που υπακούει στον περιορισμό (1.2), μας λέει πώς τα βάρη w_μ διακυμαίνονται στον χρόνο.

Με ποιόν λοιπόν τρόπο συνδέονται τα βάρη αυτά με τις μακροσκοπικές ιδιότητες του υπό εξέταση συστήματος; Αν ενδιαφερόμαστε για κάποια ποσότητα Q , η οποία παίρνει την τιμή Q_μ στην κατάσταση μ , τότε μπορούμε να ορίσουμε την εκτίμηση της Q στον χρόνο t του συστήματός μας ως εξής:

$$\langle Q \rangle = \sum_\mu Q_\mu w_\mu(t). \quad (1.3)$$

1.1 Η συνάρτηση επιμερισμού

Ας κοιτάξουμε πάλι την κύρια εξίσωση (1.1). Αν το σύστημά μας κάποια στιγμή φτάσει σε κάποια κατάσταση στην οποία οι δύο όροι του δεξιού μέλους απαλείφουν ο ένας τον άλλον για όλα τα μ , τότε οι ρυθμοί μεταβολής dw_μ/dt θα μηδενίζονται και όλα τα βάρη θα παίρνουν σταθερές τιμές για τον υπόλοιπο χρόνο. Αυτή είναι μια κατάσταση ισορροπίας. Οι ρυθμοί μετάβασης $R(\mu \rightarrow \nu)$ που εμφανίζονται στην κύρια εξίσωση (1.1) δεν παίρνουν οποιεσδήποτε τιμές. Παίρνουν συγκεκριμένες τιμές οι οποίες υπαγορεύονται από την θερμοκή φύση της αλληλεπίδρασης μεταξύ του συστήματος και του θερμικού περιβάλλοντος (thermal reservoir). Το σημαντικό σημείο είναι ότι ξέρουμε εκ των προτέρων ποιές είναι οι τιμές ισορροπίας για τα βάρη w_μ του συστήματός μας. Καλούμε αυτές τις τιμές ισορροπίας, πιθανότητες κατάληψης ισορροπίας (equilibrium occupation probabilities) και τις συμβολίζουμε ως:

$$p_\mu = \lim_{t \rightarrow \infty} w_\mu(t). \quad (1.4)$$

Ο Gibbs έδειξε το 1902 ότι για ένα σύστημα σε θερμική ισορροπία με μια θερμική δεξαμενή σε θερμοκρασία T , οι πιθανότητες κατάληψης ισορροπίας είναι

$$p_\mu = \frac{1}{Z} e^{-E_\mu/kT}. \quad (1.5)$$

Εδώ E_μ είναι η ενέργεια της καταστάσεως μ και k είναι η σταθερά του Boltzmann. Επίσης, σύμφωνα με τις συμβάσεις, συμβολίζουμε την ποσότητα $(kT)^{-1}$ με το σύμβολο β . Η Z είναι, σε πρώτη φάση, μια σταθερά κανονικοποίησης, της οποίας η τιμή δίνεται από

$$Z = \sum_{\mu} e^{-E_\mu/kT} = \sum_{\mu} e^{-\beta E_\mu}. \quad (1.6)$$

Η δε κατανομή πιθανότητας (1.5) είναι γνωστή σαν κατανομή Boltzmann.

Η Z είναι κυρίως όμως γνωστή ως η *συνάρτηση επιμερισμού* (partition function), και παίζει πολύ πιο σημαντικό ρόλο στον μαθηματικό φορμαλισμό της στατιστικής μηχανικής από μια κοινή σταθερά κανονικοποίησης. Αποδεικνύεται ότι η γνώση της διακύμανσης της Z με την θερμοκρασία και όσων άλλων παραμέτρων επηρεάζουν το σύστημα¹ μπορεί να μας πεί πρακτικά τα πάντα για την μακροσκοπική συμπεριφορά του συστήματος!

Από τις εξισώσεις (1.3), (1.4) και (1.5) η εκτίμηση μιας ποσότητας Q για ένα σύστημα σε ισορροπία είναι

$$\langle Q \rangle = \sum_{\mu} Q_{\mu} p_{\mu} = \frac{1}{Z} \sum_{\mu} Q_{\mu} e^{-\beta E_{\mu}}. \quad (1.7)$$

Έτσι λοιπόν, η αναμενόμενη τιμή της ενέργειας $\langle E \rangle$, η οποία στην θερμοδυναμική ταυτίζεται με την εσωτερική ενέργεια U , δίνεται από

$$U = \frac{1}{Z} \sum_{\mu} E_{\mu} e^{-\beta E_{\mu}}. \quad (1.8)$$

Από την εξίσωση (1.6) βλέπουμε ότι αυτή μπορεί επίσης να γραφεί με όρους παραγώγων της συνάρτησης επιμερισμού:

$$U = -\frac{1}{Z} \frac{\partial Z}{\partial \beta} = -\frac{\partial \log Z}{\partial \beta}. \quad (1.9)$$

Έτσι, ένα άλλο μέγεθος, η ειδική θερμότητα C , που δίνεται από την παράγωγο της εσωτερικής ενέργειας, είναι:

$$C = \frac{\partial U}{\partial T} = -k\beta^2 \frac{\partial U}{\partial \beta} = k\beta^2 \frac{\partial^2 \log Z}{\partial \beta^2}. \quad (1.10)$$

Όμως, ξέρουμε επίσης από την θερμοδυναμική ότι η ειδική θερμότητα σχετίζεται με την εντροπία S :

$$C = T \frac{\partial S}{\partial T} = -\beta \frac{\partial S}{\partial \beta}, \quad (1.11)$$

¹Όπως για παράδειγμα ο όγκος στον οποίον περιορίζεται το σύστημα, ή το μαγνητικό πεδίο αν έχουμε να κάνουμε με μαγνητικά υλικά.

και έτσι, εξισώνοντας αυτές τις δύο εκφράσεις για την C και ολοκληρώνοντας ως προς β , παίρνουμε την ακόλουθη έκφραση για την εντροπία:

$$S = -k\beta \frac{\partial \log Z}{\partial \beta} + k \log Z. \quad (1.12)$$

Τέλος μπορούμε να γράψουμε μια έκφραση για την ελεύθερη ενέργεια F του συστήματος:

$$F = U - TS = -kT \log Z. \quad (1.13)$$

Δείξαμε λοιπόν πώς τα μεγέθη U , C , S και F μπορούν όλα να υπολογιστούν απευθείας από την συνάρτηση επιμερισμού Z . Η τελευταία εξίσωση μας λέει επίσης πώς μπορούμε να εργαστούμε και με άλλες παραμέτρους που επιδρούν στο σύστημα. Στην κλασική θερμοδυναμική, παράμετροι ή δεσμοί ή πεδία που αλληλεπιδρούν με το σύστημα έχουν συζυγείς μεταβλητές που αντιπροσωπεύουν την απόκριση του συστήματος σε μια διαταραχή της αντίστοιχης παραμέτρου. Για παράδειγμα, η ανταπόκριση ενός αερίου συστήματος σε ένα δοχείο σε μια μεταβολή του όγκου του είναι μια μεταβολή στην πίεση του αερίου. Η πίεση p λοιπόν είναι η συζυγής μεταβλητή της παραμέτρου V . Η θερμοδυναμική μας λέει ότι μπορούμε να υπολογίσουμε την τιμή των συζυγών μεταβλητών από τις παραγώγους της ελεύθερης ενέργειας:

$$p = -\frac{\partial F}{\partial V}. \quad (1.14)$$

Έτσι έχοντας υπολογίσει την συνάρτηση επιμερισμού και κατ' επέκταση την ελεύθερη ενέργεια βάση της εξίσωσης (1.13), μπορούμε να υπολογίσουμε και τις συνέπειες μεταβολών των παραμέτρων.

1.2 Προσομοιώσεις Monte Carlo

Πρακτικά υπάρχει μόνο μία γνωστή αριθμητική μέθοδος για τον υπολογισμό της συνάρτησης επιμερισμού ενός μοντέλου σαν το Ising σε ένα μεγάλο πλέγμα, και αυτή η μέθοδος είναι η προσομοίωση Monte Carlo. Η βασική ιδέα πίσω από την προσομοίωση Monte Carlo είναι να προσομοιώσουμε την τυχαία θερμική διακύμανση του συστήματος από κατάσταση σε κατάσταση κατά την διάρκεια ενός πειράματος. Για πάρα πολλούς λόγους, πρέπει εδώ να αναφέρουμε ότι είναι προτιμότερο να θεωρούμε τον υπολογισμό της αναμενόμενης τιμής σαν έναν χρονικό μέσο όρο πάνω στις καταστάσεις από τις οποίες περνά το σύστημα. Στον υπολογισμό Monte Carlo προσομοιώνουμε απευθείας αυτή την διαδικασία: δημιουργούμε ένα μοντέλο στον υπολογιστή και το αφήνουμε να περάσει από μια σειρά καταστάσεων με τέτοιο τρόπο ώστε η πιθανότητά του να βρίσκεται σε μια συγκεκριμένη κατάσταση μ σε έναν δεδομένο χρόνο t να είναι ίση με το βάρος $w_\mu(t)$, το οποίο θα είχε η κατάσταση σε ένα πραγματικό σύστημα. Για να το επιτύχουμε αυτό, πρέπει να διαλέξουμε μια δυναμική για την προσομοίωσή μας² η οποία θα οδηγεί στην εμφάνιση κάθε καταστάσεως με ακριβώς την πιθανότητα που της αντιστοιχεί. Διαλέγουμε λοιπόν ένα σύνολο ρυθμών $R(\mu \rightarrow \nu)$ για τις μεταβάσεις από την μία κατάσταση στην άλλη και τις διαλέγουμε με τέτοιο τρόπο ώστε η λύση ισορροπίας στην αντίστοιχη κύρια εξίσωση να είναι ακριβώς η κατανομή Boltzmann

²Έναν τρόπο, δηλαδή, αλλαγής από την μία κατάσταση στην άλλη κατά την διάρκεια της προσομοίωσης.

(1.5). Μετά, χρησιμοποιούμε αυτούς τους ρυθμούς για να διαλέξουμε τις καταστάσεις από τις οποίες περνά το προσομοιούμενο σύστημά μας κατά την διάρκεια της προσομοίωσης. Από αυτές τις καταστάσεις τελικά κάνουμε εκτιμήσεις για οποιοδήποτε παρατηρήσιμο μέγεθος ενδιαφερόμαστε.

1.2.1 Η εκτιμήτρια

Ο συνήθης στόχος μιας προσομοίωσης Monte Carlo ενός θερμικού συστήματος είναι ο υπολογισμός της αναμενόμενης τιμής $\langle Q \rangle$ μια παρατηρήσιμης ποσότητας Q . Ο ιδανικός τρόπος για τον υπολογισμό μιας τέτοιας αναμενόμενης τιμής είναι η εύρεση της μέσης τιμής της ποσότητας πάνω σε όλες τις καταστάσεις μ του συστήματος με βάρος την αντίστοιχη πιθανότητα Boltzmann

$$\langle Q \rangle = \frac{\sum_{\mu} Q_{\mu} e^{-\beta E_{\mu}}}{\sum_{\mu} e^{-\beta E_{\mu}}} \quad (1.15)$$

Ο τρόπος αυτός όμως είναι εφικτός μόνο για τα μικρότερα των συστημάτων. Σε μεγαλύτερα συστήματα το καλύτερο που μπορούμε να κάνουμε είναι να πάρουμε τους μέσους όρους πάνω σε κάποιο υποσύνολο των καταστάσεων, παρόλο που έτσι αναγκαία εισάγουμε κάποια ανακρίβεια στον υπολογισμό. Οι τεχνικές Monte Carlo δουλεύουν διαλέγοντας ένα υποσύνολο των καταστάσεων τυχαία από κάποια κατανομή πιθανότητας p_{μ} την οποία ορίζουμε. Έστω, λοιπόν, ότι διαλέγουμε M τέτοιες καταστάσεις $\{\mu_1 \dots \mu_M\}$. Η καλύτερη εκτίμησή μας για την ποσότητα Q έτσι θα είναι

$$Q_M = \frac{\sum_{i=1}^M Q_{\mu_i} p_{\mu_i}^{-1} e^{-\beta E_{\mu_i}}}{\sum_{j=1}^M p_{\mu_j}^{-1} e^{-\beta E_{\mu_j}}} \quad (1.16)$$

Το Q_{μ} καλείται *εκτιμήτρια* της ποσότητας Q . Έχει δε την ιδιότητα, όσο ο αριθμός M των καταστάσεων του δείγματος αυξάνει, γίνεται όλο και πιο ακριβής εκτίμηση της $\langle Q \rangle$, και όταν $M \rightarrow \infty$ έχουμε $Q_M = \langle Q \rangle$.

Το ερώτημα που αναδεικνύεται τώρα είναι πώς θα πρέπει κάποιος να διαλέξει τις M καταστάσεις έτσι ώστε η Q_M να είναι μια ακριβής εκτίμηση της $\langle Q \rangle$; Με άλλα λόγια, πώς θα διαλέξουμε την κατανομή πιθανότητας p_{μ} ; Εδώ θα πρέπει, λοιπόν, να παρατηρήσουμε ότι το ένα ή και τα δύο αθροίσματα που εμφανίζονται στην εξίσωση (1.15) κυριαρχούνται συνήθως από έναν μικρό αριθμό καταστάσεων, με όλες τις υπόλοιπες καταστάσεις να συνεισφέρουν μόνο ένα αμελητέο ποσό τελικά. Αυτό το σύνολο καταστάσεων εξαρτάται συνήθως από την παράμετρο β .

Επομένως, αν είχαμε κάποιον τρόπο να ξέρουμε ποιές καταστάσεις έχουν σημαντικές συνεισφορές στα αθροίσματα της εξίσωσης (1.15) και αν μπορούσαμε να επιλέξουμε το δείγμα των M καταστάσεών μας για μόνο αυτές τις καταστάσεις και να αμελούσαμε όλες τις άλλες, θα είχαμε μια αρκετά καλή εκτίμηση της $\langle Q \rangle$ με ένα μικρό αριθμό όρων. Αυτή είναι και η ουσία πίσω από τις μεθόδους Monte Carlo. Η τεχνική για να επιλέγουμε τις σημαντικές καταστάσεις από το τεράστιο σύνολο όλων των πιθανών καταστάσεων λέγεται *δειγματοληψία σημαντικότητας* (importance sampling).

1.2.2 Δειγματοληψία σημαντικότητας

Συνήθως προσπαθούμε να πάρουμε ένα δείγμα καταστάσεων του συστήματος στο οποίο η πιθανότητα εμφάνισης καθεμίας είναι ανάλογη του αντίστοιχου βάρους Boltzmann της κατάστασης. Η στρατηγική είναι η εξής: αντί να παίρνουμε τις M καταστάσεις με τέτοιο τρόπο ώστε όλες οι καταστάσεις του συστήματος να είναι ισοπίθανες, τις διαλέγουμε έτσι ώστε η πιθανότητα μια συγκεκριμένη κατάσταση να επιλεγεί να είναι

$$p_{\mu} = Z^{-1} e^{-\beta E_{\mu}}. \quad (1.17)$$

Τότε η εκτιμήτρια της $\langle Q \rangle$ γίνεται απλά

$$Q_M = \frac{1}{M} \sum_{i=1}^M Q_{\mu_i}. \quad (1.18)$$

Το μόνο ερώτημα που απομένει είναι πώς ακριβώς επιλέγουμε τις καταστάσεις ώστε κάθε μια να εμφανίζεται με την σωστή αντίστοιχη πιθανότητα Boltzmann. Η καθιερωμένη λύση σε αυτό το πρόβλημα είναι να κάνουμε χρήση των Μαρκοβιανών διαδικασιών.

1.2.3 Διαδικασίες Markov

Το δύσκολο σημείο στην εκτέλεση μιας προσομοίωσης Monte Carlo είναι η δημιουργία ενός κατάλληλου τυχαίου συνόλου καταστάσεων σύμφωνα με την κατανομή πιθανότητας Boltzmann. Όλα τα σχήματα Monte Carlo βασίζονται στις μαρκοβιανές διαδικασίες σαν τον μηχανισμό παραγωγής του συνόλου των καταστάσεων που θα χρησιμοποιηθούν.

Μια μαρκοβιανή διαδικασία είναι ένας μηχανισμός ο οποίος, δεδομένου ενός συστήματος σε μια κατάσταση μ , παράγει μια καινούργια κατάσταση του συστήματος ν . Αυτό το κάνει με τυχαίο τρόπο· δεν θα γεννήσει την ίδια καινούργια κατάσταση κάθε φορά που ξεκινά από την αρχική κατάσταση μ . Η πιθανότητα να παραχθεί η κατάσταση ν δεδομένης της μ ονομάζεται πιθανότητα μετάβασης (transition probability) $P(\mu \rightarrow \nu)$ για την μετάβαση από την μ στην ν , και για μια πραγματική διαδικασία Markov όλες οι πιθανότητες μετάβασης θα πρέπει να ικανοποιούν τις ακόλουθες δύο συνθήκες: (1) δεν θα πρέπει να κυμαίνονται με τον χρόνο, και (2) θα πρέπει να εξαρτώνται μόνο από τις ιδιότητες των καταστάσεων μ και ν , και από καμία άλλη κατάσταση από την οποία πέρασε το σύστημα. Οι πιθανότητες μετάβασης $P(\mu \rightarrow \nu)$ θα πρέπει επίσης να ικανοποιούν τον περιορισμό

$$\sum_{\nu} P(\mu \rightarrow \nu) = 1. \quad (1.19)$$

Σε μια προσομοίωση Monte Carlo χρησιμοποιούμε μαρκοβιανές διαδικασίες επανηλειμμένα για να δημιουργήσουμε μια μαρκοβιανή αλυσίδα (Markov chain) καταστάσεων. Ξεκινώντας από μια κατάσταση μ , χρησιμοποιούμε την διαδικασία για να γεννήσουμε μια καινούργια κατάσταση ν , και μετά τροφοδοτούμε αυτή την κατάσταση στην διαδικασία για να γεννήσουμε μία άλλη, την λ , και ούτω καθ' εξής. Η μαρκοβιανή διαδικασία επιλέγεται έτσι ώστε όταν έχει εκτελεστεί για αρκετό χρόνο ξεκινώντας από οποιαδήποτε κατάσταση του συστήματος θα δημιουργήσει τελικά μια διαδοχή καταστάσεων που θα εμφανίζονται με πιθανότητες που δίνονται από την κατανομή Boltzmann.

1.2.4 Εργοδικότητα

Η συνθήκη της εργοδικότητας είναι η απαίτηση του να είναι δυνατό η μαρκοβιανή μας διαδικασία να περνά από οποιαδήποτε αρχική κατάσταση του συστήματός μας σε όλες τις άλλες, αν εκτελεστεί για αρκετό χρόνο. Η συνθήκη αυτή μας λέει ότι μπορούμε να κάνουμε μερικές πιθανότητες μετάβασης της μαρκοβιανής διαδικασίας μας μηδέν, αλλά θα πρέπει να υπάρχει τουλάχιστον ένα μονοπάτι μη μηδενικής πιθανότητας μετάβασης μεταξύ οποιωνδήποτε δύο καταστάσεων. Επειδή πρακτικά όλοι οι αλγόριθμοι Monte Carlo θέτουν σχεδόν όλες τις πιθανότητες μετάβασης ίσες με μηδέν, θα πρέπει να είμαστε προσεκτικοί ώστε όταν το κάνουμε αυτό ο αλγόριθμος να μην παραβιάζει την εργοδικότητα.

1.2.5 Λεπτομερής ισορροπία

Η άλλη συνθήκη που υποβάλλουμε στην μαρκοβιανή μας διαδικασία είναι η συνθήκη της λεπτομερούς ισορροπίας (detailed balance). Αυτή η συνθήκη κάνει σίγουρο ότι η κατανομή πιθανότητας την οποία παράγουμε όταν σύστημά μας έχει έρθει σε ισορροπία είναι η κατανομή Boltzmann και όχι οποιαδήποτε άλλη κατανομή.

Αυτό μεταφράζεται μαθηματικά ότι οι πιθανότητες μετάβασής μας θα πρέπει να ικανοποιούν την σχέση

$$\frac{P(\mu \rightarrow \nu)}{P(\nu \rightarrow \mu)} = \frac{p_\nu}{p_\mu} = e^{-\beta(E_\nu - E_\mu)}. \quad (1.20)$$

Αυτή η εξίσωση μαζί με την εξίσωση (1.19) είναι οι περιορισμοί που επιβάλλουμε στην επιλογή των πιθανοτήτων μετάβασης $P(\mu \rightarrow \nu)$. Αν τις ικανοποιήσουμε αυτές, καθώς και την συνθήκη της εργοδικότητας, τότε η κατανομή καταστάσεων στην ισορροπία στην διαδικασία Markov θα είναι η κατανομή Boltzmann.

1.2.6 Λόγοι αποδοχής

Η παραπάνω μικρή εισαγωγή, δυστυχώς, δίνει την λάθος εντύπωση ότι η κατασκευή ενός αλγορίθμου Monte Carlo είναι εύκολη υπόθεση. Δεδομένου ενός επιθυμητού συνόλου πιθανοτήτων μετάβασης $P(\mu \rightarrow \nu)$ που ικανοποιούν τις εξισώσεις (1.19) και (1.20), λέμε, ότι απλά σκεφτόμαστε κάποια διαδικασία Markov, η οποία δημιουργεί τις καταστάσεις με ακριβώς αυτές τις πιθανότητες μετάβασης, και ως δια μαγείας παράγουμε μία αλυσίδα γεγονότων με ακριβώς την σωστή κατανομή Boltzmann. Συχνά όμως δεν είναι προφανές ποιά είναι η κατάλληλη διαδικασία Markov η οποία έχει τις απαραίτητες πιθανότητες μετάβασης, και το να την βρούμε είναι μια επίπονη διαδικασία. Για κάποια προβλήματα, μπορούμε να χρησιμοποιήσουμε γνωστούς αλγόριθμους, όπως η μέθοδος Metropolis, αλλά για πολλά άλλα προβλήματα οι καθιερωμένες μέθοδοι δεν είναι καθόλου βολικές και θα ήταν πολύ καλύτερο αν μπορούσαμε να κατασκευάσουμε έναν αλγόριθμο για τις δικές μας ειδικές ανάγκες. Αλλά ενώ μπορεί να είμαστε σε θέση να προτείνουμε πολλές υποψήφιες διαδικασίες Markov, παρόλα αυτά μπορεί να μην μπορούμε να βρούμε κάποια που δίνει το σωστό σύνολο των πιθανοτήτων μετάβασης. Τα καλά νέα, όμως, είναι ότι δεν χρειάζεται να κάνουμε κάτι τέτοιο. Στην πραγματικότητα, αποδεικνύεται ότι μπορούμε να επιλέξουμε οποιονδήποτε αλγόριθμο μας αρέσει για την κατασκευή των καινούργιων καταστάσεων, αλλά να έχουμε και το επιθυμητό σύνολο των πιθανοτήτων μετάβασης, εισάγοντας κάτι που το ονομάζουμε λόγο αποδοχής. Η ιδέα πίσω από αυτό το τέχνασμα είναι η εξής:

Ξέρουμε ότι επιτρέπεται το σύστημα να έχει μια μη μηδενική πιθανότητα να παραμείνει στην

ίδια κατάσταση, δηλαδή η $P(\mu \rightarrow \mu)$ μπορεί να είναι μη μηδενική αν θέλουμε. Αν θέσουμε $\nu = \mu$ στην εξίσωση (1.20), παίρνουμε την απλή ταυτολογία $1 = 1$, το οποίο σημαίνει ότι η συνθήκη της λεπτομερούς ισορροπίας ικανοποιείται πάντα για $P(\mu \rightarrow \mu)$, ανεξάρτητα από την τιμή που επιλέγουμε γι'αυτό. Αυτό μας δίνει κάποια ελαστικότητα στο πώς να επιλέξουμε τις υπόλοιπες πιθανότητες μετάβασης στις οποίες $\mu \neq \nu$. Για αρχή, σημαίνει ότι μπορούμε να ρυθμίσουμε την τιμή οποιασδήποτε $P(\mu \rightarrow \nu)$ και να ικανοποιείται ο κανόνας άθροισης (1.2), απλά αναπληρώνοντας την μεταβολή που κάνουμε με μια ίση και αντίθετη μεταβολή στην $P(\mu \rightarrow \mu)$. Το μόνο πράγμα που θα πρέπει να προσέχουμε είναι η $P(\mu \rightarrow \nu)$ να μην γίνεται μικρότερη του μηδενός ή μεγαλύτερη του 1. Αν κάνουμε κάποια τέτοια μεταβολή στην $P(\mu \rightarrow \nu)$, μπορούμε να κανονίσουμε έτσι ώστε και η εξίσωση (1.20) να παραμένει ικανοποιημένη, κάνοντας συγχρόνως και μία μεταβολή στην $P(\nu \rightarrow \mu)$, έτσι ώστε ο λόγος των δύο να διατηρείται.

Καταλήγουμε έτσι στο γεγονός ότι αυτές οι θεωρήσεις μας δίνουν αρκετή ελευθερία ώστε να μπορούμε να θέτουμε στις πιθανότητες μετάβασης ό,τι τιμή θέλουμε πειράζοντας τις τιμές των πιθανοτήτων $P(\mu \rightarrow \mu)$. Για να το δούμε αυτό καθαρά, σπάμε την πιθανότητα μετάβασης σε δύο μέρη:

$$P(\mu \rightarrow \nu) = g(\mu \rightarrow \nu)A(\mu \rightarrow \nu). \quad (1.21)$$

Η ποσότητα $g(\mu \rightarrow \nu)$ είναι η πιθανότητα επιλογής, η οποία είναι η πιθανότητα, δεδομένης μιάς κατάστασης μ , ο αλγόριθμός μας να γεννήσει την νέα κατάσταση ν , και $A(\mu \rightarrow \nu)$ είναι ο λόγος αποδοχής (συχνά καλείται και πιθανότητα αποδοχής). Ο λόγος αποδοχής μας λέει ότι αν ξεκινήσουμε με μια κατάσταση μ και ο αλγόριθμός μας γεννήσει από αυτήν την καινούργια κατάσταση ν , θα πρέπει να δεχόμαστε την καινούργια κατάσταση και να μεταβάλλουμε το σύστημά μας σε αυτήν με πιθανότητα $A(\mu \rightarrow \nu)$. Αλλιώς απλά παραμένουμε στην κατάσταση μ . Είμαστε ελεύθεροι να επιλέξουμε τον λόγο αποδοχής να είναι οποιοσδήποτε αριθμός μεταξύ του μηδέν και του ένα· διαλέγοντάς τον να είναι μηδέν για όλες τις μεταβάσεις είναι ταυτόσημο με την επιλογή $P(\mu \rightarrow \mu) = 1$, που είναι η μέγιστη τιμή που μπορεί να πάρει, και σημαίνει ότι δεν θα φύγουμε ποτέ από την κατάσταση μ (Κάτι όχι και τόσο επιθυμητό).

Τα παραπάνω μας δίνουν απόλυτη ελευθερία στο πώς να διαλέξουμε τις πιθανότητες επιλογής $g(\mu \rightarrow \nu)$, αφού η εξίσωση (1.20) καθορίζει μόνο τον λόγο

$$\frac{P(\mu \rightarrow \nu)}{P(\nu \rightarrow \mu)} = \frac{g(\mu \rightarrow \nu)A(\mu \rightarrow \nu)}{g(\nu \rightarrow \mu)A(\nu \rightarrow \mu)}. \quad (1.22)$$

Ο λόγος

$$\frac{A(\mu \rightarrow \nu)}{A(\nu \rightarrow \mu)} \quad (1.23)$$

μπορεί να πάρει οποιαδήποτε τιμή μεταξύ του μηδέν και του άπειρου, που σημαίνει ότι και οι $g(\mu \rightarrow \nu)$ και $g(\nu \rightarrow \mu)$ μπορούν να πάρουν ό,τι τιμή θέλουμε.

Ο άλλος μας περιορισμός, ο κανόνας άθροισης της εξίσωσης (1.19), ικανοποιείται ακόμα, αφού το σύστημα πρέπει να καταλήγει σε κάποια κατάσταση μετά από κάθε βήμα της αλυσίδας Markov, ακόμα και αν αυτή η κατάσταση είναι η ίδια με αυτή με την οποία ξεκινήσαμε. Έτσι, για να κατασκευάσουμε τον αλγόριθμο Monte Carlo αυτό που πρακτικά κάνουμε είναι να σκεφτούμε έναν αλγόριθμο που γεννά τυχαία νέες καταστάσεις ν από τις παλιές μ , με κάποιο σύνολο πιθανοτήτων $g(\mu \rightarrow \nu)$, και μετά δεχόμαστε ή απορρίπτουμε αυτές

τις καταστάσεις με λόγους αποδοχής $A(\mu \rightarrow \nu)$ τους οποίους επιλέγουμε έτσι ώστε να ικανοποιούν την εξίσωση (1.22). Αυτό τότε θα ικανοποιεί όλες τις απαιτήσεις για τις πιθανότητες μετάβασης, και έτσι θα δημιουργήσει μία αλυσίδα καταστάσεων οι οποίες, όταν ο αλγόριθμος φτάσει στην ισορροπία, θα εμφανίζονται με την σωστή τους πιθανότητα Boltzmann.

1.3 Η μέθοδος Metropolis και το μοντέλο Ising

Το μοντέλο Ising είναι το μοντέλο ενός μαγνήτη. Η βασική ιδέα πίσω από το μοντέλο, και πίσω από πολλά μαγνητικά μοντέλα, είναι ότι ο μαγνητισμός του υλικού είναι αποτέλεσμα του συνδυασμού της μαγνητικής διπολικής ροπής πολλών ατομικών σπίν μέσα στο υλικό. Το μοντέλο προϋποθέτει ένα πλέγμα με ένα μαγνητικό δίπολο ή σπίν σε κάθε κόμβο. Στο μοντέλο Ising αυτά τα σπίν παίρνουν την απλούστερη δυνατή μορφή τους, αυτή βαθμωτών μεταβλητών s_i οι οποίες παίρνουν μόνο δύο τιμές ± 1 , που αντιπροσωπεύουν το πάνω και κάτω προσανατολισμό για ένα δίπολο μοναδιαίας τιμής. Σε ένα πραγματικό μαγνητικό υλικό τα σπίν αλληλεπιδρούν και έτσι το μοντέλο Ising μιμείται αυτή την αλληλεπίδραση συμπεριλαμβάνοντας όρους στην Χαμιλτονιανή ανάλογους των γινομένων $s_i s_j$ των σπίν. Στην απλούστερη περίπτωση, οι αλληλεπιδράσεις είναι όλες της ίδιας έντασης, που απεικονίζεται με J και έχει διαστάσεις ενέργειας, και γίνονται μόνο μεταξύ σπίν σε κόμβους που είναι άμεσοι γείτονες στο πλέγμα. Μπορούμε επίσης να εισάγουμε ένα εξωτερικό μαγνητικό πεδίο B που αλληλεπιδρά με τα σπίν. Η Χαμιλτονιανή λοιπόν παίρνει την μορφή

$$H = -J \sum_{\langle ij \rangle} s_i s_j - B \sum_i s_i, \quad (1.24)$$

όπου ο συμβολισμός $\langle ij \rangle$ δηλώνει ότι οι κόμβοι i και j που εμφανίζονται στο άθροισμα είναι άμεσοι γείτονες. Τα αρνητικά πρόσημα που εμφανίζονται είναι καθαρά συμβατικά και υπαγορεύουν απλά την επιλογή του προσήμου της παραμέτρου αλληλεπίδρασης J και του εξωτερικού πεδίου B . Με τα πρόσημα ως έχουν, μια θετική τιμή του J κάνει τα σπίν να θέλουν να ευθυγραμμιστούν το ένα με το άλλο, και όλα να θέλουν να ευθυγραμμιστούν με την διεύθυνση του εξωτερικού πεδίου.

Οι καταστάσεις του μοντέλου Ising είναι τα διαφορετικά σύνολα των τιμών που μπορούν να πάρουν τα σπίν. Καθώς κάθε σπίν μπορεί να πάρει δύο τιμές, υπάρχουν συνολικά 2^N καταστάσεις ενός πλέγματος με N σπίν πάνω του. Η συνάρτηση επιμερισμού του μοντέλου είναι το άθροισμα

$$Z = \sum_{s_1=\pm 1} \sum_{s_2=\pm 1} \cdots \sum_{s_N=\pm 1} \exp \left[\beta J \sum_{\langle ij \rangle} s_i s_j + \beta B \sum_i s_i \right]. \quad (1.25)$$

Για καθαρά οπτικούς λόγους η παραπάνω έκφραση μπορεί να γραφεί με την πιο συμπαγή μορφή

$$Z = \sum_{\{s_i\}} e^{-\beta H}. \quad (1.26)$$

Αν μπορούσαμε να κάνουμε αυτό το άθροισμα, είτε αναλυτικά είτε αριθμητικά με την βοήθεια υπολογιστή, τότε μπορούμε να υπολογίσουμε την εσωτερική ενέργεια, την εντροπία,

την ελεύθερη ενέργεια, την ειδική θερμότητα και τα λοιπά, όπως αναφέρθηκε σε προηγούμενη παράγραφο.

Εδώ έρχεται να βοηθήσει ο αλγόριθμος Metropolis. Για να εξάγουμε τον αλγόριθμο αυτόν, ακολουθούμε τα εξής βήματα:

Διαλέγουμε ένα σύνολο από πιθανότητες επιλογής $g(\mu \rightarrow \nu)$, μία για κάθε πιθανή μετάβαση από μία κατάσταση στην άλλη, $\mu \rightarrow \nu$, και τότε διαλέγουμε ένα σύνολο από πιθανότητες αποδοχής $A(\mu \rightarrow \nu)$ έτσι ώστε η εξίσωση (1.22) να ικανοποιεί την συνθήκη της λεπτομερούς ισορροπίας, (1.20). Ο αλγόριθμος δουλεύει διαλέγοντας συνεχώς μία κατάσταση ν , και μετά την αποδέχεται ή την απορρίπτει σύμφωνα με την επιλεγμένη πιθανότητα αποδοχής. Αν η κατάσταση γίνει αποδεκτή, ο υπολογιστής αλλάζει το σύστημα στην νέα κατάσταση ν . Αν όχι, απλά την αφήνει ως έχει. Έπειτα η διαδικασία επαναλαμβάνεται ξανά και ξανά.

Επειδή ένα πραγματικό σύστημα σπανίως κάνει μεταβάσεις που αλλάζουν δραματικά την ενέργεια του συστήματος, δεν χρειάζεται να καταναλώνουμε πολύ χρόνο από την προσομοίωσή μας αναλογιζόμενοι μεταβάσεις σε καταστάσεις των οποίων η ενέργεια είναι πολύ διαφορετική από την ενέργεια της παρούσας κατάστασης. Ο απλούστερος τρόπος για να το επιτύχουμε αυτό στο μοντέλο Ising είναι να αναλογιστούμε καταστάσεις που διαφέρουν από την παρούσα με το "αναποδογύρισμα" ενός σπίν. Ένας αλγόριθμος που ακολουθεί αυτή την πρακτική λέγεται ότι έχει την δυναμική αλλαγής-μοναδικού-σπίν.

Η χρήση της δυναμικής αλλαγής-μοναδικού-σπίν μας εγγυάται ότι η καινούργια κατάσταση θα έχει ενέργεια E_ν που θα διαφέρει από την παρούσα ενέργεια E_μ το πολύ $2J$ για κάθε σύνδεσμο μεταξύ του σπίν που αλλάζουμε και των γειτόνων του.

Στον αλγόριθμο Metropolis η πιθανότητες επιλογής $g(\mu \rightarrow \nu)$ για κάθε μία από τις δυνατές καταστάσεις ν επιλέγονται να είναι ίσες. Οι πιθανότητες επιλογής για όλες τις υπόλοιπες καταστάσεις τίθενται ίσες με το μηδέν. Αυτό, για το μοντέλο Ising με την δυναμική αλλαγής-μοναδικού-σπίν, σημαίνει ότι η επιλογή οποιουδήποτε ενός σπίν από το πλέγμα είναι ισοπίθανη με την επιλογή οποιουδήποτε άλλου. Αντίθετα η επιλογή περισσοτέρων του ενός σπίν είναι μηδέν. Ας υποθέσουμε ότι προσομοιώνουμε ένα σύστημα με N σπίν. Με την δυναμική αλλαγής-μοναδικού-σπίν, υπάρχουν N διαφορετικά σπίν τα οποία μπορούμε να αλλάξουμε και επομένως N διαφορετικές πιθανές καταστάσεις ν στις οποίες μπορούμε να οδηγηθούμε από μία συγκεκριμένη κατάσταση μ . Έτσι υπάρχουν N πιθανότητες επιλογής $g(\mu \rightarrow \nu)$ οι οποίες είναι μη μηδενικές, και κάθε μία από αυτές παίρνει την τιμή

$$g(\mu \rightarrow \nu) = \frac{1}{N}. \quad (1.27)$$

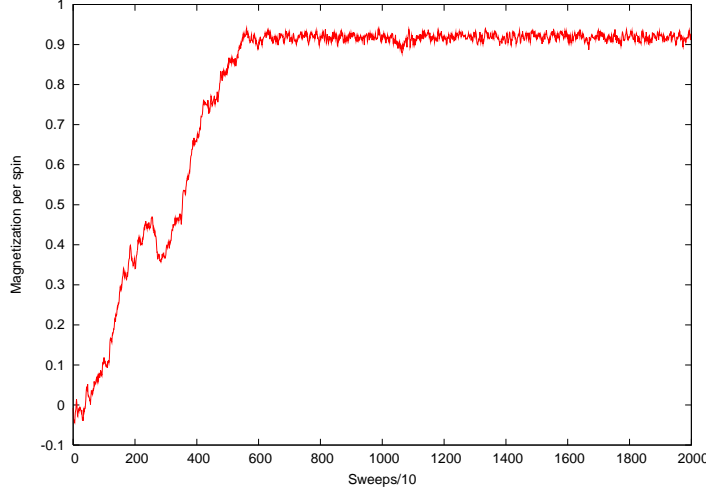
Με αυτές τις πιθανότητες επιλογής, η συνθήκη της λεπτομερούς ισορροπίας παίρνει την μορφή

$$\frac{P(\mu \rightarrow \nu)}{P(\nu \rightarrow \mu)} = \frac{g(\mu \rightarrow \nu)A(\mu \rightarrow \nu)}{g(\nu \rightarrow \mu)A(\nu \rightarrow \mu)} = \frac{A(\mu \rightarrow \nu)}{A(\nu \rightarrow \mu)} = e^{-\beta(E_\nu - E_\mu)}. \quad (1.28)$$

Τώρα θα πρέπει να διαλέξουμε τους λόγους αποδοχής $A(\mu \rightarrow \nu)$ ώστε να ικανοποιούν αυτή την εξίσωση. Μία δυνατότητα είναι να διαλέξουμε

$$A(\mu \rightarrow \nu) = A_0 e^{-\frac{1}{2}\beta(E_\nu - E_\mu)}. \quad (1.29)$$

Στην εξίσωση (1.29) έχουμε υποθέσει μία συγκεκριμένη συναρτησιακή μορφή για τον λόγο



Σχήμα 1.1: Η εξέλιξη της τιμής της μαγνήτισης ανά σπίν για ένα πλέγμα 100×100 κατά την διάρκεια της εκτέλεσης του αλγορίθμου Metropolis.

αποδοχής, αλλά η συνθήκη της λεπτομερούς ισορροπίας (1.28), δεν υπαγορεύει κάποια τέτοια μορφή. Η εξίσωση (1.28) απλά ορίζει τον λόγο ζευγών πιθανοτήτων αποδοχής, το οποίο μας δίνει αρκετό χώρο για χειρισμούς. Ο πιο αποδοτικός τρόπος να χειριστούμε λοιπόν τις πιθανότητες αποδοχής είναι αν δόσουμε στην μεγαλύτερη από τις δύο την τιμή 1 και να ρυθμίσουμε την άλλη ώστε να υπακούει στον περιορισμό. Για να δούμε πώς αυτό λειτουργεί στην περίπτωσή μας, ας υποθέσουμε ότι από τις δύο καταστάσεις μ και ν που θεωρούμε εδώ, η ν έχει την μικρότερη ενέργεια και η μ την μεγαλύτερη: $E_\mu < E_\nu$. Τότε η μεγαλύτερη από τις δύο πιθανότητες αποδοχής είναι η $A(\nu \rightarrow \mu)$ και επομένως την θέτουμε ίση με 1. Για να υπακούει στην εξίσωση (1.28) η $A(\mu \rightarrow \nu)$ πρέπει τότε να λάβει την τιμή $e^{-\beta(E_\nu - E_\mu)}$. Έτσι ο αλγόριθμος γίνεται

$$A(\mu \rightarrow \nu) = \begin{cases} e^{-\beta(E_\nu - E_\mu)} & E_\nu - E_\mu > 0 \\ 1, & \text{αλλιού.} \end{cases} \quad (1.30)$$

Με άλλα λόγια, αν επιλέξουμε κάποια καινούργια κατάσταση που έχει ενέργεια μικρότερη ή ίση με την παρούσα, θα πρέπει πάντα να αποδεχόμαστε την μετάβαση σε αυτή την κατάσταση. Αν έχει μεγαλύτερη ενέργεια τότε μπορεί να την επιλέξουμε με την πιθανότητα που δίνεται από την παραπάνω εξίσωση.

Τα παραπάνω δίνουν μία εικόνα των αρχών και των διαδικασιών που ακολουθούνται για την προσομοίωση ενός στατιστικού συστήματος. Αυτές οι αρχές ακολουθήθηκαν κατά γράμμα στην ανάπτυξη και εφαρμογή του αλγορίθμου των πλακετών, που παρουσιάζεται σε αυτή την εργασία. Ενδεικτικά, στο γράφημα (1.1) φαίνεται η εξέλιξη της τιμής της μαγνήτισης ανά σπίν ενός τέτοιου συστήματος Ising κατά την διάρκεια της εκτέλεσης του αλγορίθμου Metropolis με την δυναμική αλλαγής-μοναδικού-σπίν.

Η μαγνήτιση του συστήματος στην κατάσταση μ είναι απλά το μέγεθος

$$M_\mu = \sum_i s_i^\mu, \quad (1.31)$$

όπου το άθροισμα τρέχει πάνω σε όλα τα σπίν, και προφανώς η μαγνήτιση ανά σπίν είναι το παραπάνω μέγεθος διαιρεμένο με το πλήθος των σπίν του πλέγματος. Στο γράφημα αυτό

φαίνεται η τιμή της μαγνήτισης ανά σπίν για κάθε 10 sweeps. Ένα sweep είναι μία πλήρης διάσχιση του πλέγματος. Κάθε δέκατη δηλαδή φορά που έχουμε διασχίσει το πλέγμα με τον αλγόριθμό μας, παίρνουμε μια μέτρηση της μαγνήτισης ανά σπίν. Επίσης, χαρακτηριστικό είναι να σημειωθεί ότι η αρχική κατάσταση από την οποία ξεκίνησε η προσομοίωσή μας, και όπως φαίνεται και από το γράφημα, είναι η “παγωμένη”. Η κατάσταση αυτή είναι η κατάσταση κατά την οποία όλα τα σπίν του πλέγματος έχουν την τιμή κάτω. Τέλος, η προσομοίωση αυτή έγινε για τιμή της παραμέτρου β ίση με 1.

Κεφάλαιο 2

Κβαντική Θεωρία πεδίου σε πλέγμα και προσομοιώσεις Monte Carlo

2.1 Περί πλεγμάτων

Όπως αναφέρθηκε και στον πρόλογο, η καλύτερη ένδειξη που έχουμε για “περιορισμό” (confinement) σε μια μη Αβελιανή θεωρία βαθμίδας των ισχυρών αλληλεπιδράσεων πηγάζει από τον φορμαλισμό του Wilson σε ένα χωροχρονικό πλέγμα. Σε πρώτη όψη αυτή η συνταγή φαίνεται λίγο παράξενη διότι το κενό δεν είναι κρύσταλλος. Πράγματι, οι πειραματικοί φυσικοί δουλεύουν καθημερινά με σχετικιστικά σωματίδια που δεν δείχνουν καμία απόκλιση από τις συνεχείς συμμετρίες της ομάδας Lorentz. Γιατί, λοιπόν, οι θεωρητικοί τα τελευταία χρόνια δαπανούν τόσο χρόνο να περιγράφουν κβαντικές θεωρίες υπό την οπτική ενός χωροχρονικού πλέγματος;

Το πλέγμα καταρχήν αντιπροσωπεύει ένα μαθηματικό τρίκ. Παρέχει μία αποκοπή (cutoff) που απομακρύνει τις υπερίσχεις απειρίες που μαστίζουν την κβαντική θεωρία πεδίου. Όπως με οποιονδήποτε ρυθμιστή (regulator), θα πρέπει να απομακρύνεται μετά από την επανακανονικοποίηση (renormalization). Η φυσική μπορεί να εξαχθεί στον συνεχές όριο, όπου το βήμα του πλέγματος τείνει στο μηδέν.

Αλλά οι απειρίες και η συνεπαγόμενη ανάγκη για επανακανονικοποίηση βρίσκονται μαζί μας από τις απαρχές της σχετικιστικής κβαντομηχανικής. Το πρόγραμμα της κβαντικής ηλεκτροδυναμικής έχει τεράστιες επιτυχίες χωρίς την ανάγκη καταφυγής σε διακριτό χωροχρόνο. Γιατί, λοιπόν, να απορρίψουμε τις ήδη επιτυχείς διαταρακτικές επανακανονικοποιητικές διαδικασίες σε βάρος ενός νέου σχήματος αποκοπής;

Οδηγούμαστε στο πλέγμα από το μοναδικό φαινόμενο που εμφανίζεται στις ισχυρές αλληλεπιδράσεις, το φαινόμενο του περιορισμού. Το φαινόμενο αυτό είναι εγγενώς μη διαταρακτικό. Η ελεύθερη θεωρία με μία σταθερά σύζευξης που μηδενίζεται δεν έχει κάποια ομοιότητα στον παρατηρούμενο φυσικό κόσμο. Επιχειρήματα, που δεν είναι της παρούσης, απαιτούν την παρουσία σημαντικών ουσιαστικών απειρισμών όταν οι αδρονικές ιδιότητες εκφραστούν σαν συναρτήσεις της σταθεράς σύζευξης. Αυτό αντιτίθεται ισχυρά με τις μεγάλες επιτυχίες της κβαντικής ηλεκτροδυναμικής, όπου η θεωρία διαταραχών παίζει κεντρικό ρόλο. Τα περισσότερα συνήθη σχήματα ρύθμισης βασίζονται στην ανάπτυξη Feynman: κάποια διαδικασία υπολογίζεται μέχρι να παρατηρηθεί απειρισμός σε κάποιο διάγραμμα και μετά αυτός ο απειρισμός απομακρύνεται. Για να πάει κάποιος πέρα από την διαγραμματική προσέγγιση, χρειάζεται μία μη διαταρακτική αποκοπή. Εδώ έγκειται η κύρια αρετή

του πλέγματος, το οποίο απευθείας εξαλείφει όλα τα μήκη κύματος που είναι μικρότερα από το διπλάσιο της απόστασης του πλέγματος. Αυτό μάλιστα συμβαίνει πριν ξεκινήσουν οποιοδήποτε είδους αναπτύξεις ή προσεγγίσεις.

Στο πλέγμα, η θεωρία πεδίου γίνεται καλά ορισμένη και μπορεί να μελετηθεί με πολλούς τρόπους. Η θεωρία διαταραχών σε πλέγμα, αν και κάπως περίεργη, δίνει όλα τα συμβατικά αποτελέσματα που δίνουν τα άλλα σχήματα ρύθμισης. Ο διακριτός χωροχρόνος όμως είναι ιδιαίτερα κατάλληλος για ανάπτυξη στην ισχυρή σταθερά σύζευξης (strong coupling expansion). Είναι αξιοσημείωτο ότι ο περιορισμός είναι αυτόματος σε αυτό το όριο όπου η θεωρία καταλήγει σε ένα κουάρκ στην κάθε άκρη μιας χορδής με πεπερασμένη ενέργεια ανά μονάδα μήκους.

Ο φορμαλισμός σε πλέγμα αναδεικνύει τις συνδέσεις μεταξύ της θεωρίας πεδίου και της στατιστικής μηχανικής. Οι εγγενείς δεσμοί μεταξύ αυτών των δύο γνωστικών πεδίων γίνονται εμφανείς στον φορμαλισμό της κβαντικής μηχανικής από τον Feynman με το συναρτησιακό ολοκλήρωμα (path integral). Στον Ευκλείδιο χώρο, το συναρτησιακό ολοκλήρωμα είναι ισοδύναμο με μία συνάρτηση επιμερισμού (partition function) για ένα αντίστοιχο στατιστικό σύστημα. Το τετράγωνο της σταθεράς σύζευξης της θεωρίας πεδίου βρίσκει άμεση ανταπόκριση στην θερμοκρασία του αντίστοιχου στατιστικού συστήματος. Έτσι ο φυσικός στοιχειωδών σωματιδίων έχει εύκαιρη όλη την τεχνολογία του στατιστικού μηχανικού.

2.2 Συναρτησιακό ολοκλήρωμα και στατιστική μηχανική

Ο φορμαλισμός της κβαντικής μηχανικής με το συναρτησιακό ολοκλήρωμα του Feynman αναδεικνύει τις βαθιές συνδέσεις με την στατιστική μηχανική. Η παράγραφος αυτή ασχολείται με αυτές τις συνδέσεις για την απλή περίπτωση ενός μη-σχετικιστικού σωματιδίου μέσα σε ένα δυναμικό. Ξεκινώντας με την συνάρτηση επιμερισμού που αντιπροσωπεύει ένα συναρτησιακό ολοκλήρωμα σε ένα πλέγμα με φανταστικό χρόνο, θα δείξουμε πώς ο φορμαλισμός με τον πίνακα μεταφοράς (transfer matrix) απλοποιεί το πρόβλημα στην διαγωνοποίηση ενός τελεστή στον κανονικό κβαντομηχανικό χώρο Hilbert των τετραγωνικά ολοκληρώσιμων συναρτήσεων. Στο συνεχές όριο του χρονικού πλέγματος παίρνουμε την κανονική Χαμιλτονιανή.

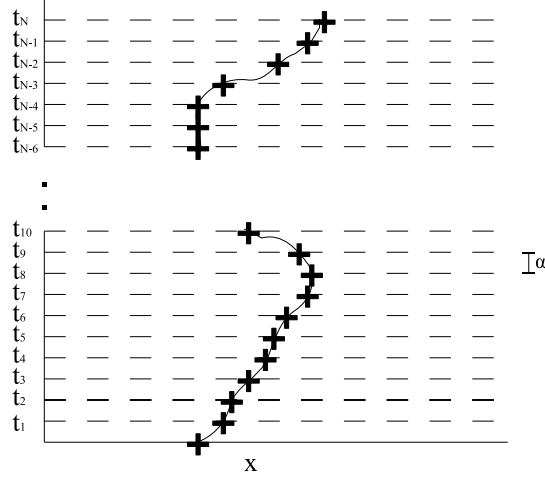
Ξεκινούμε με την Λαγκραντζιανή ενός ελεύθερου σωματιδίου μάζας m που κινείται σε δυναμικό $V(x)$

$$L(x, \dot{x}) = K(\dot{x}) + V(x),$$

$$K(\dot{x}) = \frac{1}{2}m\dot{x}^2, \tag{2.1}$$

όπου \dot{x} είναι η χρονική παράγωγος της συντεταγμένης x . Σημειώστε το μη συμβατικό θετικό πρόσημο μεταξύ των δύο όρων στην εξίσωση (2.1). Αυτό υπάρχει διότι κατασκευάζουμε το συναρτησιακό ολοκλήρωμα κατευθείαν στον φανταστικό χρόνο¹. Αυτή η πρακτική βελτιώνει την μαθηματική σύγκλιση και μας αφήνει την συνήθη Χαμιλτονιανή για την

¹Η μεταφορά σε φορμαλισμό με φανταστικό χρόνο γίνεται με τη λεγόμενη στροφή Wick, η οποία κάνει τον χωροχρόνο Ευκλείδειο. Η στροφή αυτή γίνεται με την αλλαγή μεταβλητής $t \rightarrow -it$ και στροφή του πεδίου ολοκλήρωσης στο μιγαδικό επίπεδο. Η διαδικασία αυτή γίνεται διότι, όπως θα δούμε, έχουμε να υπολογίζουμε ολοκληρώματα της μορφής $\int [dx] \exp(-iS)$, τα οποία στον κανονικό χρόνο είναι ολοκληρώματα



Σχήμα 2.1: Μια διαδρομή σε ένα πλέγμα με διακριτό χρόνο.

διαγωνοποίηση.

Για οποιαδήποτε τροχιά έχουμε την δράση

$$S = \int dt L(\dot{x}(t), x(t)), \quad (2.2)$$

η οποία εμφανίζεται στο συναρτησιακό ολοκλήρωμα

$$Z = \int [dx(t)] e^{-S}. \quad (2.3)$$

Εδώ το ολοκλήρωμα είναι πάνω σε όλες τις τροχιές $x(t)$. Όπως εμφανίζεται, όμως, η εξίσωση είναι σχετικά άσχημα ορισμένη. Για να χαρακτηρίσουμε τις πιθανές τροχιές εισάγουμε μία αποκοπή (cutoff) στην μορφή ενός χρονικού πλέγματος. Θέτοντας το σύστημά μας σε ένα χρονικό κουτί συνολικού μήκους τ , χωρίζουμε αυτό το διάστημα σε

$$N = \frac{\tau}{\alpha}, \quad (2.4)$$

διακριτά χρονικά τμήματα, όπου α είναι το χρονοειδές βήμα του πλέγματος. Σχετιζόμενη με το i -οστό τέτοιο χρονικό τμήμα είναι η συντεταγμένη x_i . Αυτή η κατασκευή απεικονίζεται στο σχήμα (2.1). Αντικαθιστώντας την χρονική παράγωγο με την διαφορά πλησιέστερου γείτονα, μειώνουμε την δράση στο άθροισμα

$$S = \alpha \sum_i \left[\frac{1}{2} m \left(\frac{x_{i+1} - x_i}{\alpha} \right)^2 + V(x_i) \right]. \quad (2.5)$$

Το ολοκλήρωμα στην εξίσωση (2.3) ορίζεται τώρα σαν ένα ολοκλήρωμα πάνω σε όλες τις συντεταγμένες

$$Z = \int \left(\prod_i dx_i \right) e^{-S}. \quad (2.6)$$

εναλλασσόμενων συναρτήσεων, τα οποία δεν συγκλίνουν. Με την στροφή Wick όμως, αυτά τα ολοκληρώματα παίρνουν την μορφή $\int [dx] \exp(-S)$, που είναι ολοκληρώματα αποσβεννόμενων συναρτήσεων, τα οποία και συγκλίνουν.

Η εξίσωση (2.6) έχει ακριβώς την μορφή μιας συνάρτησης επιμερισμού για ένα στατιστικό σύστημα. Έχουμε και εδώ μια μονοδιάστατη αλυσίδα συντεταγμένων x_i . Η δράση αντιπροσωπεύει την αντίστροφη θερμοκρασία επί την Χαμιλτονιανή του θερμικού αναλόγου. Θα δείξουμε τώρα ότι ο υπολογισμός αυτής της συνάρτησης επιμερισμού είναι ισοδύναμος με την διαγωνοποίηση μιας κβαντομηχανικής Χαμιλτονιανής που λαμβάνεται από αυτή την δράση με κανονικές μεθόδους (canonical methods). Αυτό γίνεται μέσω του πίνακα μεταφοράς.

Το κλειδί στην ανάλυση με την συνάρτηση μεταφοράς είναι να σημειώσουμε ότι η τοπική φύση της δράσης στην εξίσωση (2.2) μας επιτρέπει να γράψουμε την συνάρτηση επιμερισμού στην μορφή ενός γινομένου πινάκων

$$Z = \int \prod_i dx_i T_{x_{i+1}, x_i}, \quad (2.7)$$

όπου τα στοιχεία του πίνακα μεταφοράς είναι

$$T_{x', x} = \exp \left[-\frac{m}{2\alpha} (x' - x)^2 - \frac{\alpha}{2} (V(x') + V(x)) \right]. \quad (2.8)$$

Αυτός ο τελεστής δρά στον χώρο Hilbert τετραγωνικά ολοκληρώσιμων συναρτήσεων, όπου του εσωτερικό γινόμενο είναι το σύννηθες

$$\langle \psi' | \psi \rangle = \int dx \psi'^*(x) \psi(x). \quad (2.9)$$

Εισάγουμε τις μη κανονικοποιήσιμες καταστάσεις βάσεις $\{|\psi\rangle\}$ έτσι ώστε

$$\begin{aligned} |\psi\rangle &= \int dx \psi(x) |x\rangle, \\ \langle x' | x \rangle &= \delta(x' - x), \\ 1 &= \int dx |x\rangle \langle x|. \end{aligned} \quad (2.10)$$

Οι κανονικά συζυγείς τελεστές \hat{p} και \hat{x} ικανοποιούν τις

$$\begin{aligned} \hat{x} |x\rangle &= x |x\rangle, \\ [\hat{p}, \hat{x}] &= -i, \\ e^{-i\hat{p}\Delta} |x\rangle &= |x + \Delta\rangle. \end{aligned} \quad (2.11)$$

Σε αυτό τον χώρο Hilbert ο τελεστής T ορίζεται μέσω των στοιχείων πίνακα

$$\langle x' | T | x \rangle = T_{x', x}, \quad (2.12)$$

όπου ο $T_{x', x}$ δίνεται από την εξίσωση (2.8). Με περιοδικές συνοριακές συνθήκες για το σύστημά μας με N κόμβους, το συναρτησιακό ολοκλήρωμα γράφεται συμπαγώς

$$Z = \text{Tr} (T^N). \quad (2.13)$$

Ο τελεστής T γράφεται εύκολα με όρους των συζυγών μεταβλητών \hat{p} και \hat{x}

$$T = \int d\Delta \exp\left(-\alpha \frac{V(\hat{x})}{2}\right) \exp\left(-\frac{\Delta^2 m}{2\alpha} - i\hat{p}\Delta\right) \exp\left(-\alpha \frac{V(\hat{x})}{2}\right). \quad (2.14)$$

Το ολοκλήρωμα πάνω στο Δ είναι Γκαουσιανό και δίνει

$$T = \sqrt{\frac{2\pi\alpha}{m}} \exp\left(-\frac{1}{2}\alpha V(\hat{x})\right) \exp\left(-\frac{1}{2}\alpha \frac{\hat{p}^2}{m}\right) \exp\left(-\frac{1}{2}\alpha V(\hat{x})\right). \quad (2.15)$$

Η σύνδεση με την συνήθη κβαντομηχανική Χαμιλτονιανή εμφανίζεται στο όριο που το βήμα του πλέγματος γίνεται μικρό. Όταν το α λοιπόν είναι μικρό, τα εκθετικά στην εξίσωση (2.15) συνδυάζονται για να δώσουν

$$T = \sqrt{\frac{2\pi\alpha}{m}} e^{-\alpha H + O(\alpha^3)}, \quad (2.16)$$

όπου

$$H = \frac{\hat{p}^2}{2m} + V(\hat{x}). \quad (2.17)$$

Αυτή είναι ακριβώς η κανονική Χαμιλτονιανή που αντιστοιχεί στην Λαγκραντζιανή της εξίσωσης (2.1).

Η διαδικασία της μετάβασης από ένα συναρτησιακό ολοκλήρωμα σε έναν φορμαλισμό της κβαντικής μηχανικής σε κάποιο χώρο Hilbert αποτελείται από τρία βήματα. Πρώτον ορίζουμε το συναρτησιακό ολοκλήρωμα σε ένα χρονικό πλέγμα. Μετά κατασκευάζουμε τον πίνακα μεταφοράς και τον χώρο Hilbert στον οποίο λειτουργεί. Και τελικά, παίρνουμε τον λογάριθμο του πίνακα μεταφοράς και αναγνωρίζουμε το αρνητικό του συντελεστή του γραμμικού όρου ως προς την απόσταση του πλέγματος σαν την Χαμιλτονιανή. Φυσικά, ο πίνακας μεταφοράς διαδίδει το σύστημα από έναν χρόνο σε κάποιο άλλο. Τέτοιες μεταφορές δημιουργούνται από την Χαμιλτονιανή. Δηλώνοντας την i -οστή ιδιοτιμή του πίνακα μεταφοράς με λ_i , η εξίσωση γίνεται

$$Z = \sum_i \lambda_i^N. \quad (2.18)$$

Καθώς ο αριθμός των χρονικών διαστημάτων πάει προς το άπειρο, αυτή η έκφραση κυριαρχείται από την μεγαλύτερη ιδιοτιμή λ_0

$$Z = \lambda_0^N \times [1 + O(\exp[1 - N \ln(\lambda_0/\lambda_1)])]. \quad (2.19)$$

Έτσι στην στατιστική μηχανική οι θερμοδυναμικές ιδιότητες ενός συστήματος απορρέουν από αυτή την μεγαλύτερη ιδιοτιμή. Στην συνηθισμένη κβαντική μηχανική το αντίστοιχο ιδιοδιάνυσμα είναι η χαμηλότερη ιδιοκατάσταση της Χαμιλτονιανής· είναι η κατώτερη κατάσταση (ground state) ή, στην θεωρία πεδίου, το κενό. Σημειώστε ότι στην παρούσα συζήτηση η σχέση μεταξύ του φανταστικού και πραγματικού χρόνου είναι στοιχειώδης. Είτε ο γεννήτορας των χρονικών μεταφορών είναι ο H είτε ο iH , έχουμε πάντα τον ίδιο τελεστή να διαγωνοποιήσουμε.

2.3 Αβελιανά πεδία βαθμίδας και Κβαντική Ηλεκτροδυναμική

Το 1971 ο F. Wegner μελέτησε μία κλάση μοντέλων Ising, όπου η ολική συμμετρία $Z(2)$ της Χαμιλτονιανής μετέπιπτε σε τοπική συμμετρία. Παρόλο που τα μοντέλα δεν είχαν κάποια τοπική παράμετρο τάξης (local order parameter) παρουσίαζαν αλλαγή φάσης. Κατά την κατασκευή των μοντέλων, ο Wegner παρουσίασε έναν αριθμό από σημαντικές ιδέες οι οποίες αποδείχθηκε ότι παίζουν σπουδαίο ρόλο στην διατύπωση των θεωριών πεδίων βαθμίδας όπως η Κβαντική Ηλεκτροδυναμική και Χρωμοδυναμική. Συγκεκριμένα οδηγήθηκε στην κατασκευή μιας μη-τοπικής παραμέτρου τάξης που είναι αναλλοίωτη σε μετασχηματισμούς βαθμίδας, της οποίας το ανάλογο για την Κβαντική Χρωμοδυναμική εισήχθη αργότερα από τον Wilson και παρείχε ένα κριτήριο περιορισμού. Στην Κβαντική Χρωμοδυναμική αυτή η παράμετρος τάξης είναι γνωστή με τον όνομα βρόχος Wilson.

Ένα κοινό χαρακτηριστικό όλων των προαναφερθέντων θεωριών είναι ότι έχουν μια τοπική συμμετρία. Στην περίπτωση της Κβαντικής Ηλεκτροδυναμικής και Κβαντικής Χρωμοδυναμικής η ομάδα τοπικής συμμετρίας είναι συνεχής. Η δράση σε αυτές τις θεωρίες κατασκευάζεται “βαθμίζοντας” (μετατρέποντας δηλαδή σε τοπική) την ολική συμμετρία της ελεύθερης φερμιονικής δράσης και προσθέτοντας έναν κινητικό όρο για τα πεδία βαθμίδας. Στην συνεχή διατύπωση η αρχή κατασκευής είναι καλά γνωστή, για λόγους πληρότητας όμως θα την επαναλάβουμε παρακάτω για την περίπτωση της Κβαντικής Ηλεκτροδυναμικής. Η εκδοχή της δράσης για το πλέγμα μπορεί να κατασκευαστεί ακολουθώντας τον ίδιο τρόπο σκέψης, αλλά θα διαφέρει σε μερικές σημαντικές λεπτομέρειες από την απλοϊκή διακριτοποίηση.

Ας δούμε όμως πώς κάποιος φτάνει σε μια δράση που είναι αναλλοίωτη σε μετασχηματισμούς βαθμίδας στην συνεχή Κβαντική Ηλεκτροδυναμική. Το σημείο από το οποίο αρχίζουμε είναι η δράση του ελεύθερου πεδίου Dirac:

$$S_F^{(0)} = \int d^4x \bar{\psi}(x)(i\gamma^\mu \partial_\mu - M)\psi(x), \quad (2.20)$$

όπου γ^μ είναι ο 4×4 πίνακας Dirac με $\{\gamma_\mu, \gamma_\nu\} = g_{\mu\nu}$.

Η δράση αυτή είναι αναλλοίωτη κάτω από τον μετασχηματισμό

$$\psi(x) \rightarrow G\psi(x) \quad (2.21)$$

$$\bar{\psi}(x) \rightarrow \bar{\psi}(x)G^{-1} \quad (2.22)$$

όπου G είναι ένα στοιχείο της αβελιανής ομάδας $U(1)$, για παράδειγμα

$$G = e^{i\Lambda}, \quad (2.23)$$

με το Λ να είναι ανεξάρτητο του x . Το επόμενο βήμα αποτελείται από την απαίτηση η δράση να είναι αναλλοίωτη και κάτω από τοπικούς μετασχηματισμούς $U(1)$ με τα ψ να μετασχηματίζονται ανεξάρτητα σε διαφορετικά χωροχρονικά σημεία. Αυτό επιτυγχάνεται με την εισαγωγή ενός τετραδιανυσματικού δυναμικού $A_\mu(x)$ και αντικαθιστώντας την συνήθη παράγωγο ∂_μ με την συναλλοίωτη παράγωγο D_μ , η οποία ορίζεται ως

$$D_\mu = \partial_\mu + ie\Lambda_\mu. \quad (2.24)$$

Η νέα δράση που προκύπτει

$$S_F = \int d^4x \bar{\psi} (i\gamma^\mu D_\mu - M) \psi, \quad (2.25)$$

είναι τότε αναλλοίωτη από το παρακάτω σύνολο τοπικών μετασχηματισμών

$$\psi(x) \rightarrow G(x)\psi(x), \quad (2.26)$$

$$\bar{\psi}(x) \rightarrow \bar{\psi}(x)G^{-1}(x), \quad (2.27)$$

$$A_\mu(x) \rightarrow G(x)A_\mu(x)G^{-1}(x) - \frac{i}{e}G(x)\partial_\mu G^{-1}(x), \quad (2.28)$$

όπου

$$G(x) = e^{i\Lambda(x)}. \quad (2.29)$$

Η παρούσα περίπτωση είναι απλά ένας άλλος τρόπος για την γραφή του γνωστού νόμου μετασχηματισμού,

$$A_\mu \rightarrow A_\mu - \frac{i}{e}\partial_\mu \Lambda. \quad (2.30)$$

Εφόσον τα A_μ και G μετατίθενται, θα μπορούσαμε να έχουμε γράψει και A_μ αντί για $GA_\mu G^{-1}$. Η βασική ιδιότητα που βεβαιώνει την αναλλοιώτητα βαθμίδας της δράσης (2.25) είναι ότι, παρόλο που το A_μ μετασχηματίζεται μη ομογενώς, ο νόμος μετασχηματισμού για την συναλλοίωτη παράγωγο είναι ομογενής:

$$D_\mu \rightarrow GD_\mu G^{-1}. \quad (2.31)$$

Έχοντας εξασφαλίσει την τοπική αναλλοιώτητα βαθμίδας για την δράση εισάγωντας ένα τετραδιανυσματικό πεδίο A_μ , πρέπει τώρα να προσθέσουμε στην έκφραση (2.25) έναν κινητικό όρο ο οποίος επιτρέπει στο πεδίο A_μ να διαδίδεται. Αυτός ο όρος θα πρέπει να είναι επίσης αναλλοίωτος στους τοπικούς μετασχηματισμούς βαθμίδας (2.26-2.28), και δίνεται από την γνώριμη έκφραση

$$S_G = -\frac{1}{4} \int d^4x F_{\mu\nu} F^{\mu\nu}, \quad (2.32)$$

όπου $F_{\mu\nu} = \partial_\mu A_\nu - \partial_\nu A_\mu$ είναι ο αναλλοίωτος σε μετασχηματισμούς βαθμίδας τανυστής του ηλεκτρομαγνητικού πεδίου. Η πλήρης αναλλοίωτη σε μετασχηματισμούς βαθμίδας δράση που περιγράφει την δυναμική του ψ , $\bar{\psi}$ και A_μ είναι επομένως

$$S_{QED} = -\frac{1}{4} \int d^4x F^{\mu\nu} F_{\mu\nu} + \int d^4x \bar{\psi} (i\gamma^\mu D_\mu - M) \psi. \quad (2.33)$$

2.4 Κβαντική θεωρία πεδίου στο πλέγμα

Σε αυτό το κεφάλαιο εισάγουμε τον φορμαλισμό του Wilson ενός πεδίου βαθμίδας σε ένα χωροχρονικό πλέγμα. Η ιδέα αυτή είχε σαν κίνητρο την θεώρηση του πεδίου βαθμίδας σαν ένα παράγοντα φάσης εξαρτώμενο από την διαδρομή. Οι βασικοί βαθμοί ελευθερίας είναι τα στοιχεία της ομάδας που σχετίζονται με τους συνδέσμους που συνδέδουν τους πλησιέστερους γείτονες στους κόμβους του πλέγματος. Το στοιχείο ομάδας που σχετίζεται με ένα τυχαίο μονοπάτι που συνδέει μία σειρά γειτονικών κόμβων είναι το γινόμενο αυτών των στοιχειωδών μεταβλητών.

Θεωρώντας μία γενική ομάδα βαθμίδας G , σχετίζουμε ένα ανεξάρτητο στοιχείο της G με κάθε ζευγάρι πλησιέστερων γειτόνων των κόμβων του πλέγματος (i, j)

$$U_{ij} \in G. \quad (2.34)$$

Οι δείκτες i και j σηματοδοτούν τους κόμβους του πλέγματος στα πέρατα των δεσμών από τους οποίους ορίζεται το U_{ij} (δες σχήμα 2.2). Οι δείκτες που οφείλονται στο γεγονός ότι γενικά το U_{ij} είναι ένας πίνακας δεν γράφονται εδώ. Κατά την διάσχιση του δεσμού κατά την αντίθετη φορά, παίρνουμε το αντίστροφο στοιχείο

$$U_{ji} = (U_{ij})^{-1}. \quad (2.35)$$

Μπορούμε να ορίσουμε ένα διανυσματικό δυναμικό γράφοντας

$$U_{ij} = e^{ig_0 A_\mu \alpha}. \quad (2.36)$$

Εδώ α είναι το βήμα του πλέγματος και ο δείκτης Lorentz μ είναι η κατεύθυνση του δεδομένου δεσμού. Χρησιμοποιούμε τον συμβολισμό πινάκων για το A_μ , το οποίο είναι ένα στοιχείο της άλγεβρας Lie της ομάδας βαθμίδας. Η συντεταγμένη x_μ που σχετίζεται με το A_μ θα πρέπει να είναι στην περιοχή του υπό θεώρηση δεσμού· για ευκολία την παίρνουμε να βρίσκεται στην μέση κατά μήκος του δεσμού

$$x_\mu = \frac{1}{2}\alpha(i_\mu + j_\mu). \quad (2.37)$$

Στο συνεχές όριο, αυτή η επιλογή καθώς και το γεγονός ότι το U_{ij} είναι προσανατολισμένο κατά μήκος του δεσμού γίνονται άσχετες συμβάσεις.

Χρειαζόμαστε, λοιπόν, μια δράση για να προσδιορίσουμε την δυναμική αυτών των μεταβλητών πεδίου. Η Λαγκραντζιανή θα πρέπει να καταλήγει στο συνεχές όριο στην κλασική θεωρία Yang-Mills. Ξέρουμε μάλιστα ότι ο τανυστής του πεδίου $F_{\mu\nu}$ είναι ένας γενικευμένος στροβιλισμός του δυναμικού. Αυτό προτείνει την χρήση ολοκληρωμάτων των A_μ γύρω από μικρές κλειστές διαδρομές. Με αυτό το κίνητρο, ο Wilson πρότεινε η δράση να είναι το άθροισμα πάνω σε όλα τα στοιχειώδη τετράγωνα του πλέγματος

$$S = \sum_{\square} S_{\square}, \quad (2.38)$$

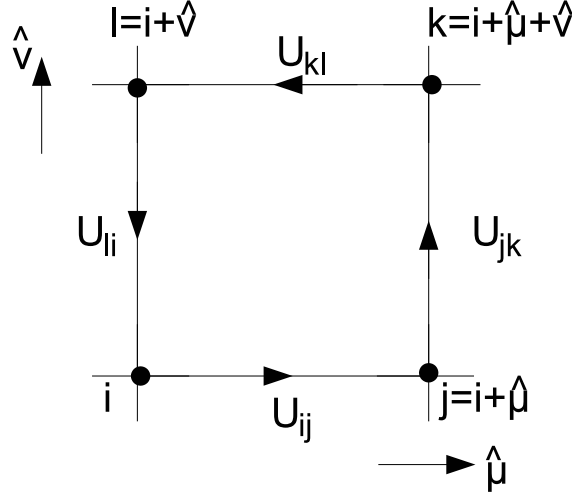
όπου ο συμβολισμός \square δηλώνει το στοιχειώδες τετράγωνο στο πλέγμα.

Η δράση για κάθε τέτοιο τετράγωνο ή πλακέτα, όπως θα το αναφέρουμε από εδώ και πέρα, είναι το ίχνος του γινομένου των στοιχείων ομάδας που σχηματίζουν την πλακέτα

$$S_{\square} = \beta \left[1 - \left(\frac{1}{n} \right) \text{ReTr}(U_{ij}U_{jk}U_{kl}U_{li}) \right]. \quad (2.39)$$

Εδώ οι κόμβοι περιστοιχίζουν την εν λόγω πλακέτα, όπως δείχνει το σχήμα (2.2). Ο παράγοντας $-1/n$ υπάρχει για την κανονικοποίηση και τις συμβάσεις προσήμων· n είναι η διάσταση των πινάκων της ομάδας. Ο παράγοντας κανονικοποίησης β θα οριστεί σε λίγο. Η προσθετική σταθερά στην εξίσωση (2.39) επιλέγεται ώστε να κάνει την δράση να μηδενίζεται όταν τα στοιχεία ομάδας πλησιάζουν την μονάδα.

Η επαλήθευση ότι αυτή η δράση καταλήγει στην συνήθη θεωρία Yang-Mills ξεκινάει από



Σχήμα 2.2: Μια πλακέτα στο πλέγμα.

την εξίσωση (2.36) για το U_{ij} σε όρους του διανυσματικού δυναμικού. Ας θεωρήσουμε, για παράδειγμα, μία πλακέτα με κέντρο στο x_μ και προσανατολισμένη στο επίπεδο $(\mu\nu) = (1, 2)$. Γράφοντας την εξίσωση (2.39) παίρνουμε

$$\begin{aligned}
 S_\square = \beta \left(1 - \left(\frac{1}{n} \right) \text{ReTr} \left(\exp \left(ig_0 A_1 \left(x_\mu - \frac{1}{2} \alpha \delta_{\mu 2} \right) \right) \right. \right. \\
 \times \exp \left(ig_0 A_2 \left(x_\mu + \frac{1}{2} \alpha \delta_{\mu 1} \right) \right) \\
 \times \exp \left(-ig_0 A_1 \left(x_\mu + \frac{1}{2} \alpha \delta_{\mu 2} \right) \right) \\
 \left. \left. \times \exp \left(-ig_0 A_2 \left(x_\mu - \frac{1}{2} \alpha \delta_{\mu 1} \right) \right) \right) \right). \quad (2.40)
 \end{aligned}$$

Τώρα θεωρούμε διανυσματικά δυναμικά αρκετά λεία ώστε να μπορούμε να τα αναπτύξουμε κατά Taylor γύρω από το x . Λίγη άλγεβρα, που δεν παρουσιάζεται εδώ, δίνει τελικά

$$S_\square = \beta \left(1 - \left(\frac{1}{n} \right) \text{ReTr} \left(\exp \left(ig_0 \alpha^2 F_{12} + O(\alpha^4) \right) \right) \right). \quad (2.41)$$

Εδώ το F_{12} είναι ο ταχυστής του πεδίου. Αναπτύσσοντας το δυναμικό βρίσκουμε

$$S_\square = \left(\frac{\beta g_0^2}{2n} \right) \alpha^4 \text{Tr}(F_{12}^2) + O(\alpha^6). \quad (2.42)$$

Ο όρος τάξεως α^2 εξαφανίζεται εξαιτίας του ότι οι μοναδιαίες ομάδες (unitary groups) είναι Ερμιτιανές. Τώρα προσεγγίζουμε το άθροισμα πάνω σε όλες τις πλακέτες με ένα χωροχρονικό ολοκλήρωμα για να πάρουμε

$$S = \frac{\beta g_0^2}{2n} \int \frac{1}{2} \text{Tr}(F_{\mu\nu} F_{\mu\nu}) d^4x + O(\alpha^6). \quad (2.43)$$

Ο παράγοντας του ενός δευτέρου μέσα στο ολοκλήρωμα προκύπτει από την συμμετρία κάτω από την αλλαγή των δεικτών μ και ν . Έτσι παίρνουμε την συνήθη δράση της θεωρίας βαθμίδας αν κάνουμε την αντιστοίχιση

$$\beta = \frac{2n}{g_0^2}. \quad (2.44)$$

Οι όροι ανώτερης τάξης του βήματος του πλέγματος στην εξίσωση (2.43) εξαφανίζονται στο κλασικό συνεχές όριο.

Έχουμε λοιπόν τις μεταβλητές μας και την Λαγκραντζιανή. Για να προχωρήσουμε στην κβαντική θεωρία, βάζουμε την δράση στο ολοκλήρωμα Feynman

$$Z = \int (dU) e^{-S(U)}. \quad (2.45)$$

Εδώ ολοκληρώνουμε πάνω σε όλες τις δυνατές τιμές των μεταβλητών βαθμίδας.

Τελικά η εξίσωση (2.45) ορίζει την συνάρτηση επιμερισμού για ένα στατιστικό σύστημα. Αν A είναι κάποια συνάρτηση των μεταβλητών πεδίου U , τότε η μέση τιμή της ορίζεται ως

$$\langle A \rangle = Z^{-1} \int (dU) A(U) e^{-S(U)}. \quad (2.46)$$

Στον κβαντομηχανικό χώρο Hilbert, αυτή είναι η αναμενόμενη τιμή του αντίστοιχου τελεστή στο κενό.

2.5 Ο κλασικός αλγόριθμος πάνω στους συνδέσμους

Σε αυτή την παράγραφο θα περιγράψουμε τον κλασικό αλγόριθμο που στηρίζεται στους συνδέσμους για μια τετραδιάστατη συμπαγή θεωρία βαθμίδας $U(1)$ σε πλέγμα χωρίς φερμιόνια.

Η δράση στο πλέγμα είναι η:

$$S = \beta \sum_{\mu, \nu} (1 - \text{Re}(U_{\mu\nu}(x))) \quad (2.47)$$

όπου οι αντίστοιχες μεταβλητές πλακετών είναι:

$$U_{\mu\nu}(x) = U_\mu(x) U_\nu(x + \alpha\hat{\nu}) U_\mu^\dagger(x + \alpha\hat{\nu}) U_\nu^\dagger(x) \quad (2.48)$$

όπου α είναι το βήμα του πλέγματος.

Ξεκινάμε λοιπόν από τον πρώτο κόμβο του πλέγματος και διαλέγουμε έναν σύνδεσμο που θέλουμε να αναεώσουμε, έστω τον U_1 . Τότε δημιουργούμε τις πλακέτες που περιέχουν τον συγκεκριμένο σύνδεσμο και μετά κατασκευάζουμε την σημαντική δράση² σύμφωνα με τον τύπο (2.48). Παράγουμε τώρα έναν τυχαίο αριθμό στο διάστημα $(-\pi/2, \pi/2)$, ο οποίος θα

²Ως σημαντική δράση ορίζουμε το τμήμα εκείνο της συνολικής δράσης που επηρεάζεται άμεσα από την διαταραχή, διότι το υπόλοιπο κατά τον υπολογισμό της διαφοράς απαλείφεται, δεδομένου ότι τελικά μας ενδιαφέρει η διαφορά της παλιάς από την καινούργια δράση.

είναι η διαταραχή μας και δημιουργούμε την καινούργια σημαντική δράση προσθέτοντας την διαταραχή στον υπό θεώρηση σύνδεσμο και κατασκευάζουμε ξανά τις σχετικές πλακέτες. Υπολογίζουμε την διαφορά στην δράση μας ΔS , και μετά παράγουμε έναν τυχαίο αριθμό r από το μηδέν έως το ένα και κάνουμε τον εξής έλεγχο:

$$r < e^{-\Delta S}. \quad (2.49)$$

Αν ο έλεγχος είναι θετικός, τότε κάνουμε την αλλαγή στον σύνδεσμο, προσθέτωντας την διαταραχή που διαλέξαμε, αλλιώς ο σύνδεσμος παραμένει ως έχει.

Το παραπάνω σενάριο επαναλαμβάνεται μετά για τους συνδέσμους με την ίδια κατεύθυνση για κάθε δεύτερο κόμβο³, μετά για τους κόμβους που παραλήψαμε και μετά όλα αυτά πάλι από την αρχή για τους συνδέσμους των άλλων κατευθύνσεων. Αφού τέλος έχουμε κάνει όλα τα παραπάνω, έχουμε επισκεφτεί όλους τους κόμβους και όλους τους συνδέσμους στο πλέγμα και λέμε ότι έχουμε κάνει ένα "πέρασμα". Επαναλαμβάνουμε την όλη διαδικασία για πολλά περάσματα πάνω στο πλέγμα μέχρι να έχουμε ένα καλό στατιστικό δείγμα της εξέλιξης του πλέγματος.

Αυτός είναι ο αλγόριθμος της προσομοίωσης Monte Carlo της συμπαγούς θεωρίας βαθμίδας $U(1)$ χωρίς φερμιόνια με βάση τους συνδέσμους.

³Το σχήμα κατά το οποίο επισκεπτόμαστε πρώτα τους μονούς και μετά τους ζυγούς κόμβους, θυμίζει σκακιέρα και χρησιμοποιείται ώστε να αποφύγουμε την επίσκεψη γειτονικών κόμβων και έτσι μεγαλύτερη στατιστική ανεξαρτησία.

Κεφάλαιο 3

Το υπολογιστικό περιβάλλον του Grid

3.1 Τι είναι το Grid;

Το υπολογιστικό περιβάλλον του Grid προέκυψε σαν ένα καινούργιο πεδίο, διαφοροποιημένο από τα συμβατικά κατανεμημένα υπολογιστικά συστήματα ως προς την σημασία που δίνει στον μεγάλης κλίμακας διαμοιρασμό πόρων, τις καινοτόμες εφαρμογές και σε μερικές περιπτώσεις ως προς τον προσανατολισμό σε υψηλές επιδόσεις. Ο όρος Grid εμφανίστηκε στα μέσα της δεκαετίας του 1990 για να σηματοδοτήσει μία κατανεμημένη υπολογιστική υποδομή για την προηγμένη επιστήμη και μηχανική. Σημαντική πρόοδος έχει συντελεστεί από τότε στην κατασκευή μιας τέτοιας υποδομής, αλλά ο όρος Grid έχει επίσης διευρυνθεί, τουλάχιστον στην δημόσια αντίληψη, ώστε να περικλείει τα πάντα από τα προηγμένα και ευφυή δίκτυα έως την τεχνητή νοημοσύνη. Κάποιος μάλιστα θα μπορούσε να αναρωτηθεί αν ο όρος έχει κάποια πραγματική ουσία και σημασία. Υπάρχει πράγματι κάποιο “Grid πρόβλημα” και επομένως κάποια ανάγκη για καινούργιες τεχνολογίες τύπου Grid; Αν ναι, ποιά είναι η φύση αυτών των τεχνολογιών και ποιό είναι το πεδίο στο οποίο θα εφαρμοστούν; Ενώ πολυάριθμες ομάδες ενδιαφέρονται για τις έννοιες του Grid και μοιράζονται, σε μεγάλο βαθμό, ένα κοινό όραμα για την αρχιτεκτονική του Grid, δεν παρατηρείται κάποια συναίνεση στις απαντήσεις των παραπάνω ερωτήσεων.

Το πραγματικό και καλά ορισμένο πρόβλημα που βρίσκεται κάτω από την ιδέα του Grid είναι ο συντεταγμένος διαμοιρασμός πόρων και η λύση προβλημάτων σε ένα περιβάλλον δυναμικών, ετερογενών εικονικών οργανισμών. Ο διαμοιρασμός για τον οποίο ενδιαφερόμαστε δεν είναι απλός διαμοιρασμός αρχείων αλλά κυρίως η απευθείας προσπέλαση σε υπολογιστικά συστήματα, λογισμικό, δεδομένα και άλλους πόρους όπως απαιτείται από ένα μεγάλο εύρος συνεργατικών στρατηγικών επίλυσης προβλημάτων και διαχείρισης πόρων που αναδεικνύονται στην βιομηχανία, την επιστήμη και την μηχανική. Αυτός ο διαμοιρασμός είναι απαραίτητα αυστηρά ελεγχόμενος, με τους παροχείς και χρήστες των πόρων να καθορίζουν καθαρά και προσεκτικά τί είναι αυτό που θα διαμοιραστεί, ποιός επιτρέπεται να το διαμοιραστεί και τις συνθήκες κάτω από τις οποίες αυτός ο διαμοιρασμός θα συμβεί. Ένα σύνολο ιδιωτών ή/και ινστιτούτων που καθορίζονται από τέτοιους κανόνες διαμοιρασμού καλείται ένας Εικονικός Οργανισμός (Virtual Organization - VO).

Οι Εικονικοί Οργανισμοί ποικίλουν σημαντικά στον σκοπό, τα ενδιαφέροντα, το μέγεθος, την διάρκεια, την δομή και την κοινωνιολογία τους. Παρόλα αυτά, προσεκτική μελέτη των

τεχνολογικών απαιτήσεων έχει οδηγήσει στην αναγνώριση ενός ευρέους συνόλου κοινών ανησυχιών και απαιτήσεων. Συγκεκριμένα, παρατηρείται η ανάγκη για πολύ ευέλικτες σχέσεις διαμοιρασμού, που κυμαίνονται από σχέσεις client-server σε peer-to-peer· η ανάγκη για εξελιγμένους και ακριβείς τρόπους ελέγχου πάνω στον τρόπο χρήσης των διαμοιραζομένων πόρων· η ανάγκη διαμοιρασμού ποικίλων και ετερογενών πόρων όπως λογισμικό, αρχεία και δεδομένα σε υπολογιστές, αισθητήρες και δίκτυα· και η ανάγκη για διαφορετικούς τρόπους χρήσης, που κυμαίνονται από τον απλό χρήστη στους πολλαπλούς χρήστες και από την ευαίσθητη σε επίδοση στην ευαίσθητη σε κόστος χρήση.

Οι υπάρχουσες τεχνολογίες καταμετρημένων υπολογιστικών περιβάλλοντων δεν ανταποκρίνονται στις απαιτήσεις που μόλις αναφέρθηκαν. Για παράδειγμα, οι υπάρχουσες τεχνολογίες Διαδικτύου ανταποκρίνονται σε απαιτήσεις επικοινωνίας και μεταφοράς πληροφοριών μεταξύ υπολογιστών αλλά δεν παρέχουν κάποια ολοκληρωμένη προσέγγιση στην συντεταγμένη χρήση πόρων από πολλαπλούς παροχείς.

Εδώ λοιπόν μπαίνουν στο παιχνίδι οι τεχνολογίες Grid. Τα τελευταία χρόνια προσπάθειες έρευνας και ανάπτυξης στην κοινότητα του Grid έχουν ως αποτέλεσμα την δημιουργία πρωτοκόλλων, υπηρεσιών και εργαλείων τα οποία ανταποκρίνονται ακριβώς τις προκλήσεις που αναδεικνύονται όταν προσπαθούμε να “χτίσουμε” κλιμακούμενους Εικονικούς Οργανισμούς. Αυτές οι τεχνολογίες συμπεριλαμβάνουν λύσεις ασφάλειας οι οποίες υποστηρίζουν την διαχείριση πιστοποιητικών και αρχών όταν οι υπολογιστικές προσπάθειες γίνονται από πολλά διαφορετικά ινστιτούτα, υποστηρίζουν πρωτόκολλα διαχείρισης πόρων και υπηρεσίες που υποστηρίζουν την ασφαλή απομακρυσμένη προσπέλαση σε υπολογιστικούς και αποθηκευτικούς πόρους και τον εντοπισμό πολλαπλών πόρων, υποστηρίζουν πρωτόκολλα και υπηρεσίες επερωτήσεων οι οποίες παρέχουν πληροφορίες κατάστασης και διαμόρφωσης για πόρους, οργανισμούς και υπηρεσίες και υποστηρίζουν υπηρεσίες διαχείρισης δεδομένων που εντοπίζουν και μεταφέρουν σύνολα δεδομένων μεταξύ συστημάτων αποθήκευσης και εφαρμογών.

3.2 Το LCG

Ο επιταχυντής LHC, που κατασκευάζεται στο Ευρωπαϊκό Κέντρο Σωματιδιακής Φυσικής CERN, θα είναι ο μεγαλύτερος και ισχυρότερος επιταχυντής του κόσμου. Θα ξεκινήσει την λειτουργία του το 2007 και τα πειράματα που θα τον χρησιμοποιήσουν θα παράγουν τεράστιες ποσότητες δεδομένων. Η επεξεργασία αυτών των δεδομένων θα απαιτήσει τεράστιους υπολογιστικούς και αποθηκευτικούς πόρους.

Η αποστολή του Προγράμματος του Υπολογιστικού Grid του LHC - LCG - (LHC Computing Grid Project) είναι να προετοιμάσει την υποδομή για την προσομοίωση, την επεξεργασία και ανάλυση των δεδομένων που θα παραχθούν από τα τέσσερα πειράματα του LHC: το ATLAS, το CMS, το ALICE και το LHCb. Αυτό συμπεριλαμβάνει και την κοινή υποδομή των βιβλιοθηκών, εργαλείων και υπολογιστικών πλαισίων που απαιτούνται για την υποστήριξη του λογισμικού των εφαρμογών φυσικής, και την ανάπτυξη και λειτουργία των υπολογιστικών υπηρεσιών για την αποθήκευση και επεξεργασία δεδομένων, παρέχοντας διαδραστικές και μη υπηρεσίες για την παγκόσμια κοινότητα των φυσικών που ασχολούνται με το LHC.

Οι απαιτήσεις για την διαχείριση των δεδομένων του LHC είναι τεράστιες, σε όρους υπολογιστικής ισχύος, αποθηκευτικού χώρου, καλής διαθεσιμότητας δεδομένων και των σχετικών ανθρώπινων πόρων για την λειτουργία και συντήρηση. Δεν θεωρείται εφικτό

να χρηματοδοτηθούν όλοι αυτοί οι πόροι σε ένα γεωγραφικό σημείο και έτσι έχει συμφωνηθεί ότι η υπηρεσία LCG θα υλοποιηθεί σαν ένα γεωγραφικά καταναμημένο υπολογιστικό Grid. Αυτό σημαίνει ότι η υπηρεσία θα χρησιμοποιεί υπολογιστικούς και αποθηκευτικούς πόρους, εγκατεστημένους σε έναν μεγάλο αριθμό γεωγραφικών σημείων σε πολλές διαφορετικές χώρες, που θα συνδέονται με ταχεία δίκτυα. Ειδικό λογισμικό, που αναφέρεται συνήθως σαν μεσισμικό (middleware), θα κρύψει πολλή από την πολυπλοκότητα αυτού του περιβάλλοντος από τον χρήστη, δίνοντάς του την εντύπωση ότι όλοι οι πόροι είναι διαθέσιμοι σε ένα συνεκτικό εικονικό υπολογιστικό κέντρο.

Το πρόγραμμα LCG είναι μία από τις δύο πιλοτικές εφαρμογές που έχουν επιλεγεί για να καθοδηγήσουν την υλοποίηση και να επικυρώσουν την λειτουργικότητα και αποδοτικότητα της ανερχόμενης Ευρωπαϊκής υποδομής Grid. Η άλλη εφαρμογή είναι τα Βιοϊατρικά Grid. Αυτό γίνεται σαν τμήμα του προγράμματος EGEE (Enabling Grids for E-science in Europe), το οποίο στοχεύει να ενσωματώσει τις παρούσες κρατικές, τοπικές και θεματικές προσπάθειες δημιουργίας Grid, με τελικό σκοπό να δημιουργήσει μία ενιαία Ευρωπαϊκή υποδομή Grid για την υποστήριξη της Ευρωπαϊκής έρευνας.

Από εδώ και εμπρός, συχνά οι όροι LCG και Grid θα χρησιμοποιούνται χωρίς διάκριση.

3.3 Η αρχιτεκτονική του LCG

3.3.1 Γενικά

Το πρόγραμμα LCG είναι οργανωμένο σε Εικονικούς Οργανισμούς (Virtual Organizations) - VO: δυναμικές συλλογές ιδιωτών και ινστιτούτων που μοιράζονται πόρους με έναν εύκαμπτο, ασφαλή και συντεταγμένο τρόπο. Σε τέτοια περιβάλλοντα, συναντούμε μοναδικές προκλήσεις σε θέματα πιστοποίησης, αδειοδότησης, προσπέλασης πόρων, εύρεσης πόρων και λοιπά.

Πρίν ο χρήστης μπορεί να χρησιμοποιήσει τους πόρους του LCG απαιτείται να καταγράψει μερικά προσωπικά δεδομένα και πληροφορίες για τον Εικονικό Οργανισμό στον οποίον ανήκει στον Εξυπηρετητή Εγγραφών του LCG. Το CERN παρέχει μια τέτοια υπηρεσία, που συλλέγει πληροφορίες για όλους τους χρήστες του LCG.

Η Υποδομή Ασφάλειας του Grid (Grid Security Infrastructure - GSI) στο LCG κάνει δυνατή την ασφαλή πιστοποίηση και επικοινωνία μέσω ενός ανοικτού δικτύου. Το GSI βασίζεται σε κρυπτογράφηση δημόσιου κλειδιού, σε πιστοποιητικά X.509, και το πρωτόκολλο επικοινωνίας Secure Sockets Layer - SSL. Για να μπορεί ένας χρήστης να προσπελάσει στους πόρους του Grid, θα πρέπει να έχει ένα ψηφιακό πιστοποιητικό X.509 από κάποια Υπηρεσία Πιστοποίησης (Certification Authority - CA) που εμπιστεύεται το LCG.

3.3.2 Η διεπαφή του χρήστη

Το αρχικό σημείο εισόδου στο LCG είναι μία Διεπαφή Χρήστη (User Interface - UI). Αυτή είναι ένα μηχανήμα στο οποίο οι χρήστες του LCG έχουν έναν προσωπικό λογαριασμό και όπου το πιστοποιητικό χρήστη είναι εγκατεστημένο. Αυτό είναι η πύλη προς τις υπηρεσίες του Grid. Από την Διεπαφή Χρήστη, ο χρήστης μπορεί να πιστοποιηθεί και να εξουσιοδοτηθεί να χρησιμοποιήσει τους πόρους του Grid, και μπορεί να προσπελάσει στις λειτουργίες που παρέχονται από τις Υπηρεσίες Διαχείρισης Πληροφοριών, Εργασιών

και Δεδομένων. Η Διεπαφή Χρήστη παρέχει καταρχήν μία Διεπαφή Γραμμής Εντολών (Command Line Interface - CLI) στην οποία ο χρήστης μπορεί να εκτελέσει κάποιες βασικές λειτουργίες στο Grid:

- να υποβάλλει μία εργασία προς εκτέλεση σε κάποιο Υπολογιστικό Στοιχείο,
- να απαριθμήσει όλους τους πόρους που είναι κατάλληλοι για την εκτέλεση μιας δεδομένης εργασίας,
- να αντιγράψει αρχεία,
- να σταματήσει μία ή περισσότερες εργασίες,
- να ανακτήσει την έξοδο μίας ή περισσότερων εργασιών που έχουν τερματίσει,
- να δει την φάση εκτέλεσης μίας ή περισσότερων εργασιών.

3.3.3 Υπολογιστικά Στοιχεία και Στοιχεία Αποθήκευσης

Ένα Υπολογιστικό Στοιχείο (Computing Element - CE) ορίζεται ως μία υπολογιστική ουρά στο Grid και το αναγνωριστικό του είναι το ζεύγος `<hostname>:<port>/<batch_queue_name>`.

Είναι σημαντικό να καταλάβουμε ότι σύμφωνα με αυτό τον ορισμό, διαφορετικές ουρές που ορίζονται με το ίδιο hostname θεωρούνται διαφορετικά CEs. Παραδείγματα ονομάτων CE είναι:

```
ce01.isabella.grnet.gr:2119/jobmanager-pbs-long
ce01.isabella.grnet.gr:2119/jobmanager-pbs-short
```

Ένα Υπολογιστικό Στοιχείο είναι χτισμένο σε μία ομογενή φάρμα από υπολογιστικούς κόμβους που ονομάζονται Κόμβοι Εργασίας (Worker Node - WN) και από έναν κόμβο που λειτουργεί σαν Πύλη του Grid (Grid Gate - GG) ή σαν διεπαφή με το υπόλοιπο Grid. Ενώ όλοι οι Κόμβοι Εργασίας χρειάζονται μόνο συνδέσεις προς τα έξω, ο Κόμβος Πύλης πρέπει να είναι προσπελάσιμος και εκτός του τόπου. Η GG είναι υπεύθυνη για την αποδοχή εργασιών και την αποστολή τους για εκτέλεση στους Κόμβους Εργασίας. Η GG παρέχει μία ομοιόμορφη διεπαφή στους υπολογιστικούς πόρους που διαχειρίζεται. Κάθε τύπος LCG έχει τουλάχιστον ένα CE και μία φάρμα από WN από πίσω.

Ένα Στοιχείο Αποθήκευσης (Storage Element - SE) παρέχει ομοιόμορφη προσπέλαση και υπηρεσίες σε μεγάλους αποθηκευτικούς χώρους. Το SE μπορεί να ελέγχει μεγάλες σειρές δίσκων, συστήματα μαζικής αποθήκευσης (Mass Storage Systems - MSS) ή/και ανάλογα συστήματα. Κάθε τύπος LCG παρέχει ένα ή περισσότερα SE.

3.3.4 Το Σύστημα Πληροφόρησης

Το Σύστημα Πληροφόρησης (Information System - IS) παρέχει πληροφορίες για τους πόρους του LCG και την κατάσταση τους. Η προσφερόμενη πληροφορία συμφωνεί με το σχήμα GLUE - Grid Laboratory for a Uniform Environment. Το σχήμα αυτό σκοπεύει να ορίσει ένα κοινό μοντέλο δεδομένων το οποίο θα χρησιμοποιείται για τον έλεγχο και την ανάκτηση πόρων του Grid.

Στο LCG, σαν παροχέας της Υπηρεσίας Πληροφόρησης έχει επιλεγεί η Υπηρεσία Ελέγχου

και Εύρεσης (Monitoring and Discovery Service - MDS) από το Globus. Υλοποιεί το σχήμα GLUE χρησιμοποιώντας OpenLDAP, που είναι μία έκδοση ανοιχτού λογισμικού του πρωτοκόλλου LDAP - Lightweight Directory Access Protocol.

Λόγω της δυναμικής φύσης του Grid, οι Εξυπηρετητές GIS - Grid Index Information Server μπορεί να έχουν την πληροφορία για τους πόρους οι οποίοι είναι πραγματικά διαθέσιμοι στο Grid αλλά, για κάποιους λόγους, δεν μπορούν να μεταφέρουν ενημερωμένη πληροφορία για την κατάστασή τους στους εξυπηρετητές αυτούς. Λόγω αυτού, έχει εισαχθεί το BDII - Berkeley DB Information Index. Το BDII κάνει ερωτήσεις στους GIS και λειτουργεί σαν μία αποθήκη στην οποία αποθηκεύονται πληροφορίες για την κατάσταση του Grid.

3.3.5 Διαχείριση δεδομένων

Οι υπηρεσίες Διαχείρισης Δεδομένων παρέχονται από στο Σύστημα Διαχείρισης Αντιγράφων (Replica Management System - RMS) του Ευρωπαϊκού DataGrid (EDG) και τα εργαλεία διαχείρισης δεδομένων του LCG. Στο περιβάλλον του Grid, τα αρχεία δεδομένων αντιγράφονται, πιθανόν σε προσωρινή βάση, σε πολλές διαφορετικές τοποθεσίες ανάλογα με το πού ζητούνται τα δεδομένα. Οι χρήστες ή οι εφαρμογές δεν χρειάζεται να γνωρίζουν πού βρίσκονται φυσικά τα δεδομένα. Χρησιμοποιούν λογικά ονόματα για τα αρχεία και οι υπηρεσίες διαχείρισης δεδομένων είναι υπεύθυνες για τον εντοπισμό και την προσπέλαση των δεδομένων.

Τα αρχεία στο Grid σηματοδοτούνται με διαφορετικά ονόματα: το Grid Unique Identifier - GUID, το Logical File Name - LFN, το Storage URL και το Transport URL. Ενώ το GUID ή το LFN αναφέρονται σε αρχεία και όχι σε αντίγραφα, και δεν λένε τίποτα για τις τοποθεσίες, τα SURL και TURL δίνουν πληροφορίες για το πού βρίσκεται φυσικά κάποιο αντίγραφο.

Ένα αρχείο μπορεί πάντα να αναγνωριστεί από το GUID του· αυτό παραχωρείται κατά την δήλωση των δεδομένων και βασίζεται στο πρότυπο UUID το οποίο εξασφαλίζει μία μοναδική ταυτότητα. Ένα GUID είναι της μορφής `guid:<unique_string>`. Όλα τα αντίγραφα του αρχείου θα μοιράζονται το ίδιο GUID. Για να εντοπίσει ένα αρχείο στο Grid κάποιος χρήστης θα χρησιμοποιήσει ένα LFN. Τα LFN είναι πιο εύχρηστα και ευανάγνωστα από τον χρήστη καθώς ορίζονται από αυτόν σαν ψευδώνυμο του GUID. Η μορφή τους είναι `lfn:<any_alias>`.

Το SURL χρησιμοποιείται από το Σύστημα Διαχείρισης Αντιγράφων (RMS) για να βρει πού ένα αντίγραφο είναι φυσικά αποθηκευμένο, και από κάποιο Στοιχείο Αποθήκευσης για να το εντοπίσει. Τα SURL είναι της μορφής `sfn:<SE_hostname>/<local_string>`, όπου το `<local_string>` χρησιμοποιείται εσωτερικά από το SE για τον εντοπισμό του αρχείου. Τέλος, το TURL δίνει την απαραίτητη πληροφορία για τον εντοπισμό του φυσικού αντιγράφου, συμπεριλαμβανομένων του ονόματος hostname, του πρωτοκόλλου και της θύρας, έτσι ώστε μία εφαρμογή να μπορεί να το εντοπίσει και να το ανοίξει.

Οι κύριες υπηρεσίες που παρέχονται από έναν RMS είναι: η Υπηρεσία Εντοπισμού Αντιγράφων (Replica Location Service - RLS) και ο Κατάλογος Μεταδεδομένων Αντιγράφων (Replica Metadata Catalog). Η RLS διατηρεί πληροφορία για την φυσική τοποθεσία των αντιγράφων. Συντίθεται από διάφορους Τοπικούς Καταλόγους Αντιγράφων (Local Replica Catalog - LRC) οι οποίοι έχουν την πληροφορία για τα αντίγραφα ενός και μόνου VO. Ο RMC αποθηκεύει την αντιστοίχιση μεταξύ των GUID και των αντίστοιχών τους ψευδ-

ωνύμων LFN και διατηρεί και άλλες πληροφορίες μεταδεδομένων (πληροφορίες μεγέθους, ημερομηνιών, κυριότητας, κ.λ.π.).

3.3.6 Διαχείριση εργασιών

Οι υπηρεσίες του Συστήματος Διαχείρισης Εργασιών (Workload Management System - WMS) ευθύνονται για την αποδοχή των αιτήσεων για εργασίες και την αποστολή αυτών των εργασιών στο κατάλληλο CE, ανάλογα με τις συγκεκριμένες απαιτήσεις της εργασίας και τους διαθέσιμους πόρους. Για αυτό το λόγο, το WMS πρέπει να λάβει πληροφορία από το BDII και το RLS. Το μηχάνημα στο οποίο τρέχουν οι υπηρεσίες του WMS ονομάζεται Resource Broker - RB. Αυτές οι υπηρεσίες είναι:

- ο Εξυπηρετητής Δικτύου (Network Server - NS), ο οποίος αποδέχεται τις εισερχόμενες αιτήσεις για εργασίες από το UI, και παρέχει υποστήριξη για τον έλεγχο των εργασιών,
- ο Διαχειριστής Εργασιών (Workload Manager), το οποίο είναι το βασικό τμήμα του συστήματος,
- ο Match-Maker, του οποίου δουλειά είναι να βρίσκει τον καταλληλότερο πόρο που ανταποκρίνεται στις απαιτήσεις μίας συγκεκριμένης εργασίας,
- ο Ρυθμιστής Εργασίας (Job Adapter), ο οποίος προετοιμάζει το περιβάλλον για την εργασία και την τελική περιγραφή της, πριν την περάσει στη Υπηρεσία Ελέγχου Εργασίας,
- η Υπηρεσία Ελέγχου Εργασίας (Job Control Service - JCS), η οποία τελικά εκτελεί της λειτουργίες διαχείρισης της εργασίας (όπως υποβολή, ακύρωση...)

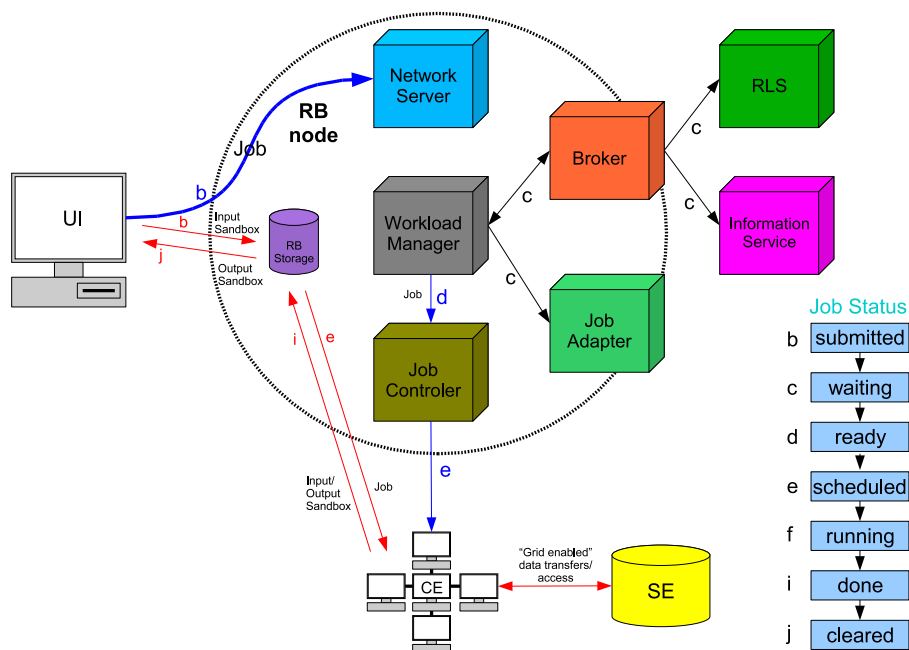
Επίσης, η Υπηρεσία Logging and Bookkeeping - LB τρέχει και αυτή στο μηχάνημα του RB. Η LB κρατάει ημερολόγιο για όλα τα γεγονότα διαχείρισης κάποιας εργασίας από το Grid, το οποίο μπορεί μετά να ανακτηθούν από τον χρήστη ή τους διαχειριστές του συστήματος για έλεγχο ή επίλυση προβλημάτων.

3.3.7 Η ροή των εργασιών

Αυτή η παράγραφος περιγράφει σύντομα τί συμβαίνει όταν ένας χρήστης υποβάλλει κάποια εργασία το Grid για την επεξεργασία κάποιων δεδομένων και πώς τα διαφορετικά συστατικά αλληλεπιδρούν.

Το σχήμα (3.1) απεικονίζει την διαδικασία την οποία ακολουθεί μια εργασία η οποία υποβλήθηκε στο Grid. Τα ανεξάρτητα βήματα περιγράφονται παρακάτω:

- a. Αφού ο χρήστης πάρει ένα ψηφιακό πιστοποιητικό από κάποια Πιστοποιητική Αρχή, εγγραφεί στο LCG, εγγραφεί σε κάποιον VO και φτιάξει λογαριασμό σε κάποιο UI του LCG (κάτι που κάνει άπαξ), είναι έτοιμος να χρησιμοποιήσει το Grid. Εγγράφεται λοιπόν στο μηχάνημα UI και δημιουργεί ένα πιστοποιητικό proxy το οποίο τον πιστοποιεί σε κάθε ασφαλή λειτουργία και έχει περιορισμένο χρόνο ζωής.



Σχήμα 3.1: Η ροή μιας εργασίας στο LCG.

- b. Ο χρήστης υποβάλλει την εργασία από το UI στην WMS, όπου η εργασία θα εκτελεστεί σε κάποιο υπολογιστικό στοιχείο. Ο χρήστης μπορεί να ορίσει στο αρχείο περιγραφής της εργασίας ότι ένα ή περισσότερα αρχεία θα αντιγραφούν από το UI στον κόμβο RB· αυτό το σύνολο αρχείων ονομάζεται Input Sandbox. Το γεγονός καταγράφεται στο LB και η κατάσταση της εργασίας είναι **SUBMITTED**.
- c. Το WMS και συγκεκριμένα το συστατικό Match-Maker, ψάχνουν για το καταλληλότερο διαθέσιμο CE το οποίο μπορεί να εκτελέσει την εργασία. Για να το κάνει αυτό, ο Match-Maker ρωτά το BDII να ζητήσει την κατάσταση των υπολογιστικών και αποθηκευτικών πόρων και το RLS να βρει την τοποθεσία των δεδομένων. Το γεγονός καταγράφεται στο LB και η κατάσταση της εργασίας είναι **WAIT**.
- d. Ο Ρυθμιστής Εργασιών του WMS προετοιμάζει την εργασία για εκτέλεση δημιουργώντας ένα script το οποίο προωθείται, μαζί με άλλες παραμέτρους, στο JCS για υποβολή στο επιλεγμένο CE. Το γεγονός καταγράφεται στο LB και η κατάσταση της εργασίας είναι **READY**.
- e. Ο Gatekeeper του Globus στο CE λαμβάνει την αίτηση και στέλνει την εργασία προς εκτέλεση στο LRMS. Το γεγονός καταγράφεται στο LB και η κατάσταση της εργασίας είναι **SCHEDULED**.
- f. Το LRMS χειρίζεται την εκτέλεση της εργασίας στην διαθέσιμη τοπική φάρμα από κόμβους εργασίας. Τα αρχεία του χρήστη αντιγράφονται από τον RB στον κόμβο εργασίας όπου η εργασία εκτελείται. Το γεγονός καταγράφεται στο LB και η κατάσταση της εργασίας είναι **RUNNING**.
- g. Όσο η εργασία εκτελείται, αρχεία του Grid μπορούν να προσπελαστούν σε κάποιο κοντινό SE χρησιμοποιώντας είτε το πρωτόκολλο RFIO ή τοπική προσπέλαση αν

τα αρχεία έχουν αντιγραφεί στο τοπικό σύστημα του WN. Για να μπορεί η εργασία να βρεί ποιό είναι ένα κοντινό Στοιχείο Αποθήκευσης, ο WMS παράγει ένα αρχείο πληροφοριών και αποστέλεται μαζί με το αρχείο της εργασίας. Το αρχείο αυτό είναι γνωστό ως `.BrokerInfo`.

- h. Η εργασία μπορεί να παράξει καινούργια δεδομένα εξόδου, τα οποία μπορούν να μεταφερθούν στο Grid και να είναι διαθέσιμα προς χρήση από άλλους χρήστες του Grid. Η μεταφορά ενός αρχείου στο Grid σημαίνει την αντιγραφή του σε ένα SE και την καταχώρηση της τοποθεσίας του και μεταπληροφοριών στον RMS. Συγχρόνως, κατά την διάρκεια της εκτέλεσης της εργασίας ή μέσω της διεπαφής χρήστη, αρχεία δεδομένων μπορούν να αντιγραφούν μεταξύ δύο SE χρησιμοποιώντας τα εργαλεία Διαχείρισης Αρχείων.
- i. Αν η εργασία φτάσει στο τέλος της χωρίς σφάλματα, η έξοδος (αρχεία τα οποία έχουν οριστεί στο λεγόμενο Output Sandbox) μεταφέρεται στον κόμβο RB. Το γεγονός καταγράφεται στο LB και η κατάσταση της εργασίας είναι DONE.
- j. Σε αυτό το σημείο ο χρήστης μπορεί να ανακτήσει την έξοδο της εργασίας του από το UI χρησιμοποιώντας το CLI ή το API του WMS. Το γεγονός καταγράφεται στο LB και η κατάσταση της εργασίας είναι CLEARED.
- k. Επερωτήσεις για την κατάσταση της εργασίας απευθύνονται στην βάση δεδομένων LB από το μηχάνημα UI. Επίσης από το UI είναι δυνατόν να ρωτήσουμε το BDII για την κατάσταση των πόρων.
- l. Αν η τοποθεσία στην οποία τρέχει η εργασία "πέσει", η εργασία θα σταλεί αυτόματα σε κάποιο άλλο CE το οποίο είναι ανάλογο με το προηγούμενο και φυσικά συμφωνεί με τις ίδιες απαιτήσεις που έθεσε ο χρήστης. Στην περίπτωση που αυτή η νέα υποβολή δεν είναι εφικτή, η εργασία θα σημειωθεί ως ματαιωμένη. Οι χρήστες μπορούν να πάρουν πληροφορία για το τί έγινε θέτωντας ερωτήσεις στην υπηρεσία LB.

3.4 Τα πιστοποιητικά proxy certificates

Η πρώτη απαίτηση την οποία πρέπει να ικανοποιεί ένας χρήστης του Grid είναι να είναι κάτοχος ενός πιστοποιητικού X.509 το οποίο έχει εκδοθεί από κάποια Αρχή Πιστοποίησης (Certification Authority - CA). Ο ρόλος της CA είναι να εγγυάται ότι ο χρήστης είναι αυτός που ισχυρίζεται και ότι δικαιούται το πιστοποιητικό του. Εναπόκειται στον χρήστη να ανακαλύψει με ποιά CA θα πρέπει να επικοινωνήσει. Γενικά οι CA είναι οργανωμένες γεωγραφικά και με βάση τα ερευνητικά κέντρα. Κάθε CA έχει την δικιά της διαδικασία για την έκδοση των πιστοποιητικών.

Μία δεύτερη απαίτηση για τον χρήστη του Grid είναι να ανήκει σε κάποιον Εικονικό Οργανισμό. Ένας VO είναι μία οντότητα, η οποία συνήθως ανταποκρίνεται σε κάποιον οργανισμό ή ομάδα ανθρώπων στον πραγματικό κόσμο. Η συμμετοχή σε έναν VO δίνει κάποια προτερήματα στον χρήστη. Για παράδειγμα, ένας χρήστης που ανήκει στον VO του ATLAS θα μπορεί να έχει το δικαίωμα να διαβάζει αρχεία του ATLAS ή να έχει πρόσβαση σε πόρους που έχουν κρατηθεί συγκεκριμένα για την ομάδα του ATLAS.

Συνήθως η είσοδος σε κάποιον VO απαιτεί την συμμετοχή σε κάποια συνεργασία· ο χρήστης θα πρέπει να είναι συμμορφωμένος με τους κανόνες του VO που τον αφορούν για να

μπορεί να απαιτήσει την συμμετοχή του.

Δεν είναι δυνατό να προσπελάσει κάποιος το LCG Grid χωρίς να είναι μέλος κάποιου VO. Κάθε χρήστης απαιτείται να επιλέξει τον VO όταν καταχωρείται στο LCG και η παρεχόμενη πληροφορία αποστέλεται στον διαχειριστή του VO και στους παροχείς πόρων για πιστοποίηση πριν η διαδικασία της καταχώρησης ολοκληρωθεί.

Σε αυτό το σημείο ο χρήστης μπορεί να δημιουργήσει ένα πιστοποιητικό proxy. Ένα πιστοποιητικό proxy είναι ένα διαπιστευτήριο εξουσιοδότησης χρήστη το οποίο πιστοποιεί τον χρήστη σε κάθε ασφαλή αλληλεπίδραση με το Grid και έχει ένα περιορισμένο χρόνο ζωής.

Για να δημιουργήσει κάποιος ένα πιστοποιητικό proxy, δίνει την ακόλουθη εντολή:

```
$ grid-proxy-init
```

Θα του ζητηθεί ο κωδικός χρήστη και αν όλα πάνε καλά, η έξοδος της διαδικασίας θα δείχνει κάπως έτσι:

```
Your identity: /C=GR/O=HellasGrid/OU=central.ntua.gr/CN=Dimitris Skipis
Enter GRID pass phrase for this identity:
Creating proxy ..... Done
Your proxy is valid until: Tue Jan 31 00:52:48 2006
```

Ένα πιστοποιητικό proxy έχει χρόνο ζωής 12 ωρών.

Είναι δυνατό, επίσης, για τον χρήστη να τυπώσει πληροφορίες για έναν υπάρχον πιστοποιητικό proxy ή να το καταστρέψει πριν λήξει ο χρόνος ζωής του.

Για να τυπώσει πληροφορίες ο χρήστης δίνει την εντολή:

```
$ grid-proxy-info
```

Αν το πιστοποιητικό υπάρχει, η έξοδος της εντολής θα δείχνει κάπως έτσι:

```
subject : /C=GR/O=HellasGrid/OU=central.ntua.gr/CN=Dimitris Skipis/CN=proxy
issuer  : /C=GR/O=HellasGrid/OU=central.ntua.gr/CN=Dimitris Skipis
identity : /C=GR/O=HellasGrid/OU=central.ntua.gr/CN=Dimitris Skipis
type    : full legacy globus proxy
strength : 512 bits
path    : /tmp/x509up_u1010
timeleft : 11:59:56
```

Για την καταστροφή ενός υπάρχοντος πιστοποιητικού, ο χρήστης αρκεί να δώσει την εντολή:

```
$ grid-proxy-destroy
```

Το πιστοποιητικό proxy, όπως περιγράφηκε παραπάνω έχει όμως ένα πρόβλημα: αν η εργασία δεν τερματίσει πριν λήξει το πιστοποιητικό τότε ματαιώνεται. Αυτό είναι προφανώς ένα πρόβλημα αν, για παράδειγμα, ο χρήστης θέλει να υποβάλλει εργασίες που η εκτέλεσή

τους διαρκεί αρκετό χρόνο. Τότε θα πρέπει λοιπόν να δημιουργήσει ένα πιστοποιητικό proxy το οποίο έχει μεγάλη διάρκεια ζωής. Αυτό όμως αυξάνει τους κινδύνους ασφάλειας. Για να υπερβεί αυτό το εμπόδιο, χρησιμοποιείται μία αποθήκη πιστοποιητικών proxy, η οποία επιτρέπει στον χρήστη να δημιουργεί και να αποθηκεύει ένα πιστοποιητικό μακράς διάρκειας σε έναν αφιερωμένο εξυπηρετητή (Proxy Server). Ο WMS θα μπορεί τότε να χρησιμοποιεί αυτό το μακράς διάρκειας πιστοποιητικό για να ανανεώνει περιοδικά το proxy για μία υποβεβλημένη εργασία πριν λήξει και μέχρι να τερματίσει η εργασία (ή να λήξει ο μακράς διάρκειας proxy).

Για να δημιουργήσει και αποθηκεύσει ένα μακράς διάρκειας πιστοποιητικό proxy ο χρήστης δίνει την εντολή:

```
$ myproxy-init -s <host_name> -d -n
```

όπου το `-s <host_name>` δηλώνει το όνομα της μηχανής στην οποία τρέχει ο Proxy Server, η επιλογή `-d` καθοδηγεί τον εξυπηρετητή να χρησιμοποιήσει το αντικείμενο του πιστοποιητικού σαν το όνομα χρήστη, και η επιλογή `-n` αποτρέπει την χρήση κωδικού χρήστη για την πρόσβαση στο μακράς διάρκειας proxy, έτσι ώστε ο WMS να μπορεί να κάνει τις ανανεώσεις αυτόματα.

Η έξοδος της εντολής θα δείχνει κάπως έτσι:

```
Your identity: /C=GR/O=HellasGrid/OU=central.ntua.gr/CN=Dimitris Skipis
Enter GRID pass phrase for this identity:
Creating proxy ..... Done
Proxy Verify OK
Your proxy is valid until: Mon Feb 6 12:53:48 2006
A proxy valid for 168 hours (7.0 days) for user
/C=GR/O=HellasGrid/OU=central.ntua.gr/CN=Dimitris Skipis
now exists on myproxy.grid.auth.gr.
```

Για να λάβει πληροφορία για ένα proxy μακράς διάρκειας το οποίο είναι αποθηκευμένο σε έναν Proxy Server, ο χρήστης αρκεί να δώσει την εντολή:

```
$ myproxy-info -d
```

Η έξοδος θα δείχνει κάπως έτσι:

```
username: /C=GR/O=HellasGrid/OU=central.ntua.gr/CN=Dimitris Skipis
owner: /C=GR/O=HellasGrid/OU=central.ntua.gr/CN=Dimitris Skipis
timeleft: 167:59:07 (7.0 days)
```

Για να καταστρέψει ένα proxy μακράς διάρκειας, ο χρήστης δίνει την εντολή:

```
$ myproxy-destroy -d
```

Θα πρέπει να σημειωθεί ότι ο χρήστης θα πρέπει να έχει ένα έγκυρο πιστοποιητικό proxy στο UI, το οποίο έχει δημιουργηθεί με την εντολή `grid-proxy-init`, για να μπορέσει να

αλληλεπιδράσει με το πιστοποιητικό μακράς διάρκειας στον Proxy Server.

3.5 Διαχείριση Εργασιών

Στο LCG Grid, ο χρήστης μπορεί να υποβάλλει και να ακυρώσει εργασίες, να ρωτήσει για την κατάστασή τους, και να ανακτήσει την έξοδό τους. Αυτές οι διαδικασίες βρίσκονται κάτω από το όνομα Διαχείριση Εργασιών (Workload Management). Το LCG παρέχει δύο διαφορετικές διεπαφές για να μπορεί κάποιος να εκτελέσει αυτές τις διαδικασίες. Η μία είναι η Διεπαφή Γραμμής Εντολών (Command Line Interface - CLI) και η άλλη είναι μία Γραφική Διεπαφή.

Ανεξάρτητα, όμως, από το ποιά διεπαφή χρησιμοποιηθεί για την υποβολή μιας εργασίας, η περιγραφή των χαρακτηριστικών και των απαιτήσεών της θα πρέπει να αποσταλούν μαζί της, έτσι ώστε να ευρεθεί ένας κατάλληλος προορισμός για την εκτέλεσή της. Η γλώσσα που χρησιμοποιείται για την περιγραφή της εργασίας καλείται Job Description Language (JDL), και περιγράφεται συνοπτικά στην επόμενη παράγραφο.

3.5.1 Η γλώσσα ορισμού εργασιών JDL

Στο LCG για την περιγραφή των εργασιών προς εκτέλεση στο Grid χρησιμοποιούνται τα αρχεία ορισμού εργασιών (.jdl αρχεία). Αυτά τα αρχεία γράφονται χρησιμοποιώντας την γλώσσα ορισμού εργασιών JDL. Η JDL χρησιμοποιείται στο LCG για να καθοριστούν τα επιθυμητά χαρακτηριστικά και οι περιορισμοί μιας εργασίας, τα οποία θα χρησιμοποιηθούν μετά κατά την διαδικασία του match-making για την επιλογή των πόρων που θα απαιτήσει η εργασία.

Η σύνταξη της JDL αποτελείται από εκφράσεις που τελειώνουν με ελληνικό ερωτηματικό, όπως:

```
attribute = value;
```

Αλφαριθμητικά, χαρακτήρες ή συμβολοσειρές που θα χρησιμοποιηθούν ως τιμές, πρέπει να περικλείονται από διπλά εισαγωγικά. Για ειδικούς χαρακτήρες, όπως για παράδειγμα τον &, θα πρέπει να τους εισάγουμε ως \&. Τα σχόλια θα πρέπει να προσημαίνονται από τον χαρακτήρα #. Επίσης η JDL είναι ευαίσθητη στο κενό χαρακτήρα και το tab.

Σε ένα αρχείο ορισμού εργασίας μερικά χαρακτηριστικά είναι υποχρεωτικά ενώ μερικά όχι. Συγκεκριμένα, ο χρήστης θα πρέπει τουλάχιστον να ορίσει το όνομα του εκτελέσιμου, τα αρχεία στα οποία θα γραφεί η έξοδος και τα σφάλματα της εργασίας. Για παράδειγμα:

```
Executable = "text.sh";
StdOutput = "std.out";
StdError = "std.err";
```

Αν χρειαστεί, μπορούν να εισαχθούν ορίσματα για το εκτελέσιμο αρχείο:

```
Arguments = "some_argument another_argument"
```

Τότε, τα αρχεία, τα οποία θα μεταφερθούν από το UI στον WN πριν και μετά την εκ-

τέλεση της εργασίας μπορούν να οριστούν ως:

```
InputSandbox = {"test.sh"};
OutputSandbox = {"std.out", "std.err"};
```

Είναι πολύ σημαντικό εδώ να αναφερθεί ότι ο χαρακτήρας ενός αρχείου ως εκτελέσιμου δεν διατηρείται για τα αρχεία τα οποία βρίσκονται στο Input Sandbox και μεταφέρονται στον WN. Γι'αυτό το λόγο, για κάθε αρχείο το οποίο θέλει ο χρήστης να έχει άδεια εκτέλεσης, θα πρέπει να προηγηθεί η διαδικασία `chmod +x` στο script που είναι ορισμένο σαν Executable στο αρχείο JDL.

3.5.2 Υποβολή εργασίας

Για να υποβληθεί μία εργασία στο LCG Grid, ο χρήστης θα πρέπει καταρχήν να έχει ένα έγκυρο πιστοποιητικό proxy στο μηχάνημα UI. Αν κάτι τέτοιο συμβαίνει, τότε ο χρήστης θα μπορεί να δώσει την ακόλουθη εντολή:

```
$ edg-job-submit <jdl_file>
```

όπου <jdl_file> είναι το αρχείο το οποίο περιέχει την περιγραφή της εργασίας. Ένα απλό παράδειγμα είναι ένα αρχείο `test.jdl` με τα εξής περιεχόμενα:

```
Executable = "test.sh";
StdOutput = "std.out";
StdError = "std.err";
InputSandbox = {"test.sh"};
OutputSandbox = {"std.out", "std.err"};
```

Το αρχείο αυτό περιγράφει μία απλή εργασία, η οποία θα εκτελεσθεί από το αρχείο `test.sh`. Η στάνταρντ έξοδος και τα σφάλματα θα γραφούν στα αρχεία `std.out` και `std.err` αντίστοιχα, τα οποία μετά το πέρας της εκτέλεσης θα μεταφερθούν πίσω στο UI καθώς περιέχονται στο Output Sandbox.

Η εργασία, λοιπόν, υποβάλλεται με την εντολή:

```
$ edg-job-submit --vo see -o JOBIDs test.jdl
```

όπου η επιλογή `--vo see` δηλώνει τον VO στον οποίο ανήκουμε (στην περίπτωση μας τον SEE), και η επιλογή `-o JOBIDs` δηλώνει ότι θέλουμε το ID να αποθηκευθεί στο αρχείο JOBIDs.

Αν η υποβολή είναι επιτυχής, τότε η έξοδος θα δείχνει κάπως έτσι:

```
Selected Virtual Organisation name (from --vo option): see
Connecting to host rb.isabella.grnet.gr, port 7772
Logging to host rb.isabella.grnet.gr, port 9002
```

```
===== edg-job-submit Success=====
```

The job has been successfully submitted to the Network Server.
Use `edg-job-status` command to check job current status. Your job identifier (`edg_jobId`) is:

- <https://rb.isabella.grnet.gr:9000/riFWbic7k8RaAMrMxhev4Q>

The `edg_jobId` has been saved in the following file:
`/home/dskipis/Skipis/QU1RK/4D/JOBIDS`

=====

Είναι επίσης δυνατό να δει ο χρήστης ποιά Υπολογιστικά Στοιχεία είναι κατάλληλα να εκτελέσουν την εργασία που περιγράφεται από ένα αρχείο JDL χρησιμοποιώντας την εντολή `edg-job-list-match`:

```
$ edg-job-list-match --vo see test.jdl
```

```
Selected Virtual Organisation name (from --vo option): see  
Connecting to host rb.isabella.grnet.gr, port 7772
```

```
*****  
COMPUTING ELEMENT IDs LIST
```

```
The following CE(s) matching your job requirements have been found:
```

```
*CEId*
```

```
ce01.isabella.grnet.gr:2119/jobmanager-pbs-long
```

```
*****
```

3.5.3 Λειτουργίες εργασίας

Αφού υποβληθεί κάποια εργασία, είναι δυνατό ο χρήστης να δει την κατάσταση της και την ιστορία της, να ανακτήσει τα αρχεία εξόδου της εργασίας μόλις αυτή τερματίσει και να την ακυρώσει πριν αυτή τερματίσει.

Δεδομένης μίας υποβεβλημένης εργασίας της οποίας το αναγνωριστικό είναι `<jobId>`, ο χρήστης μπορεί να δει την κατάσταση της με την εντολή:

```
$ edg-job-status <jobId>
```

```
*****  
BOOKKEEPING INFORMATION:
```

```
Status info for the Job :
```

```
https://rb.isabella.grnet.gr:9000/Ibdrj1L.HiyM462L8rG9Pg
```

```
Current Status: Running
```

```
Status Reason: Job successfully submitted to Globus
```

```
Destination: ce01.isabella.grnet.gr:2119/jobmanager-pbs-long
```

```
reached on: Fri Jan 27 10:22:24 2006
```

όπου φαίνεται η παρούσα κατάσταση της εργασίας μαζί με τον χρόνο κατά τον οποίο η εργασία μπήκε σε αυτή την κατάσταση και τον λόγο για τον οποίο μπήκε.
Μία εργασία μπορεί να ακυρωθεί πριν φτάσει στον τερματισμό της απλά με την εντολή `edg-job-cancel`:

```
$ edg-job-cancel <jobId>
```

```
Are you sure you want to remove specified job(s)? [y/n]n :y
```

```
===== edg-job-cancel Success =====
```

```
The cancellation request has been successfully submitted for the following job(s):
```

```
- https://rb.isabella.grnet.gr:9000/riFWbic7k8RaAMrMxhev4Q
```

```
=====
```

Επίσης, αφού μία εργασία τερματίσει, ο χρήστης μπορεί να αντιγράψει την έξοδο της εργασίας στο UI με την εντολή:

```
$ edg-job-get-output --dir ./Results <jobId>
```

```
Retrieving files from host: rb.isabella.grnet.gr ( for https://rb.isabella.grnet.gr:9000/5KX_as4PQZEEq-EvN3pZzg )
```

```
JOB GET OUTPUT OUTCOME
```

```
Output sandbox files for the job:
```

```
- https://rb.isabella.grnet.gr:9000/5KX_as4PQZEEq-EvN3pZzg have been successfully retrieved and stored in the directory: /home/dskipis/Skipis/QU1RK/4D/Results/dskipis_5KX_as4PQZEEq-EvN3pZzg
```

Τέλος, με την εντολή `edg-get-logging-info` μπορεί να θέσει ερώτηση στην βάση δεδομένων του LB για την πληροφορία καταχώρησης κάποιας εργασίας που έχει υποβληθεί με την εντολή `edg-job-submit`. Αυτό είναι ιδιαίτερα χρήσιμο για την απασφαλμάτωση στην περίπτωση μη επιτυχών υποβολών. Ένα απλό παράδειγμα είναι:

```
$ edg-job-get-logging-info -o logfile.txt <jobId>
```

όπου με την επιλογή `-o logfile.txt` ορίζουμε ότι η έξοδος θα εγγραφεί στο αρχείο `logfile`. Μία ενδεικτική έξοδος είναι:

LOGGING INFORMATION:

Printing info for the Job :
https://rb.isabella.grnet.gr:9000/5KX_as4PQZEQq-EvN3pZzg

Event: RegJob
- source = UserInterface
- timestamp = Sat Jan 28 11:46:09 2006

Event: Transfer
- destination = NetworkServer
- result = START
- source = UserInterface
- timestamp = Sat Jan 28 11:46:51 2006

Event: Transfer
- destination = NetworkServer
- result = OK
- source = UserInterface
- timestamp = Sat Jan 28 11:46:54 2006

Event: Accepted
- source = NetworkServer
- timestamp = Sat Jan 28 11:46:52 2006

Event: EnQueued
- result = OK
- source = NetworkServer
- timestamp = Sat Jan 28 11:46:54 2006

...

Κεφάλαιο 4

Ο αλγόριθμος των πλακετών

Πρίν αναφερθούμε στην θεωρητική υποστήριξη του αλγόριθμου, καλό είναι να παρουσιαστεί πώς καταλήξαμε στον αλγόριθμο ξεκινώντας από τον απλό αλγόριθμο πάνω στους συνδέσμους.

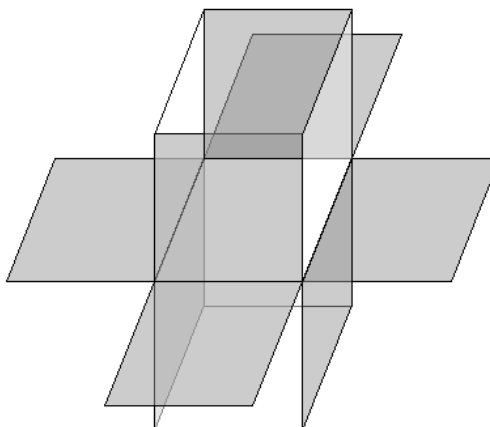
Η αφορμή για την δημιουργία του αλγόριθμου των πλακετών ήταν καταρχήν καθαρά υπολογιστική. Το πρώτο που βλέπει κανείς όταν χειρίζεται την συμπαγή θεωρία βαθμίδας $U(1)$ χωρίς φερμιόνια με δράση την

$$S = \beta \sum_P (1 - \cos P) \quad (4.1)$$

όπου P η πλακέτα και το άθροισμα να τρέχει πάνω σε όλες τις πλακέτες του πλέγματος, είναι ότι πραγματικά δεν χρειάζεται τους συνδέσμους μεταξύ των κόμβων του κρυστάλλου. Στον απλό αλγόριθμο, ο φυσικός κατασκευάζει για κάθε κόμβο τις αντίστοιχες πλακέτες του συνδέσμου που θέλει να ανανεώσει. Το μειονέκτημα σε αυτή την διαδικασία είναι ότι χρειάζεται κάθε φορά να κατασκευάσει τις πλακέτες τις οποίες θα χειριστεί, και μάλιστα θα χρειαστεί να κατασκευάσει τις ίδιες πλακέτες αρκετές φορές. Πρώτο βήμα λοιπόν είναι να θεωρήσουμε ένα πλέγμα όχι με τέσσερις συνδέσμους σε κάθε κόμβο, για την τετραδιάστατη περίπτωση, αλλά ένα πλέγμα με έξι πλακέτες σε κάθε κόμβο. Η πλακέτα πλέον είναι η βασική οντότητα στο πλέγμα και η έννοια των συνδέσμων δεν υφίσταται!

Πώς λοιπόν θα ανανεώνουμε τις πλακέτες κατ'αντίστοιχο τρόπο με αυτόν στον αλγόριθμο των συνδέσμων; Ας δούμε πάλι τί κάνει ο αλγόριθμος με τους σύνδεσμους. Επιλέγει έναν κόμβο και στον κόμβο αυτό επιλέγει έναν σύνδεσμο. Μετά για αυτόν τον σύνδεσμο κατασκευάζει όλες τις πλακέτες που τον περιέχουν και έτσι χτίζει τις πλακέτες με τις οποίες θα ασχοληθεί. Καθώς μία πλακέτα έχει τέσσερις πλευρές, η κάθε πλακέτα θα κατασκευαστεί τέσσερις φορές κατά την διάσχιση του πλέγματος. Τώρα όμως με τις πλακέτες ήδη έτοιμες, αυτό δεν χρειάζεται. Κάθε φορά που επιλέγεται ένας κόμβος, επιλέγεται μία πλακέτα του και μετά βρίσκεται η γειτονιά της, δηλαδή όλες οι πλακέτες που συνορεύουν μαζί της. Αυτές θα φτιάξουν την σημαντική δράση, όπως αυτή ορίστηκε σε προηγούμενο κεφάλαιο.

Τίθενται καταρχήν λοιπόν δύο ζητήματα: πώς μπορεί να βρεί κάποιος την γειτονιά μιας πλακέτας και δεύτερον πώς θα κατασκευάσει το τμήμα της δράσης που χρειάζεται. Η απάντηση στο πρώτο ερώτημα είναι σχετικά απλή καθώς το όλο πρόβλημα είναι καθαρά γεωμετρικό: έχοντας έναν υπερκύβο τεσσάρων διαστάσεων, ποιές πλευρές συνορεύουν με ποιές. Η απάντηση σε αυτό το πρόβλημα χρειάζεται να δοθεί για ένα συγκεκριμένο πλέγμα μόνο μία φορά με την κατασκευή ενός χάρτη του πλέγματος. Αυτός ο χάρτης δείχνει

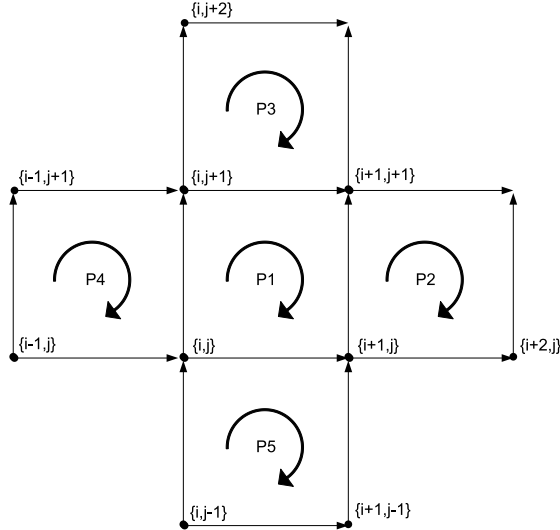


Σχήμα 4.1: Η γειτονιά μιας πλακέτας στην τριδιάστατη περίπτωση με χρωματισμό που υποδηλώνει το πρόσημο με το οποίο προστίθεται η διαταραχή στην αντίστοιχη πλακέτα. Σε ανώτερες διαστάσεις, αυτό που αλλάζει είναι ένα “κουτί” από πλακέτες πάνω και ένα κάτω από την πλακέτα.

ακριβώς ποιές πλακέτες συνορεύουν με ποιές με βάση ένα σχήμα αρίθμησής τους. Για την απάντηση του δεύτερου ερωτήματος, που έχει αντίκτυπο και στην κατασκευή του χάρτη, αρκεί να θυμηθούμε ότι όταν έχουμε το πλέγμα με τους συνδέσμους, αυτοί είναι προσανατολισμένοι. Έτσι, στον παλιό αλγόριθμο αν συναντούσαμε έναν σύνδεσμο κατά την φορά του, τότε έμπαινε με θετικό πρόσημο στην πλακέτα, ενώ αν τον συναντούσαμε αντίρροπα, τότε έμπαινε με αρνητικό. Τώρα, στην κατασκευή του τμήματος της δράσης που θα ανανεώσουμε, και το οποίο αποτελείται από τις πλακέτες της γειτονιάς μίας πλακέτας, μπαίνουν με πρόσημα τα οποία βασίζονται σε ένα σχήμα το οποίο θα δώσει τα ίδια αποτελέσματα όταν μιλάμε με την γλώσσα των συνδέσμων. Τα πρόσημα αυτά μπαίνουν στον χάρτη του πλέγματος και ξεπερδύουμε με αυτά σχετικά γρήγορα. Μιά γειτονιά μιας πλακέτας μαζί με τα πρόσημά της φαίνεται στο σχήμα (4.1) για την τριδιάστατη περίπτωση. Στο σχήμα αυτό οι άσπρες πλακέτες φέρουν θετικό πρόσημο ενώ οι σκουρόχρωμες αρνητικό. Έως τώρα, έχουμε λοιπόν κατασκευάσει την βασική δομή πάνω στην οποία θα δουλέψει ο αλγόριθμός μας. Για να γίνουν τελείως προφανή τα πράγματα, θα εφαρμόσουμε τον παλιό και τον καινούργιο αλγόριθμο πάνω σε μία πλακέτα για την διδιάστατη περίπτωση. Το σχετικό σχήμα είναι το (4.2). Με τον παλιό αλγόριθμο για την ανανέωση και των τεσσάρων πλευρών της πλακέτας $P1$ θα κάναμε το εξής:

1. Επιλέγουμε τον σύνδεσμο $\{i, j\}_i^1$ προς ανανέωση. Άρα κατασκευάζουμε τις πλακέτες $P1$ και $P5$, από το άθροισμα $\{i, j\}_j + \{i, j + 1\}_i - \{i + 1, j\}_j - \{i, j\}_i$ και $\{i, j\}_i - \{i + 1, j - 1\}_j - \{i, j - 1\}_i + \{i, j - 1\}_j$ αντίστοιχα, και έτσι φτιάχνουμε την παλιά σημαντική δράση υπολογίζοντας τα συνημίτονα για κάθε πλακέτα που δημιουργήσαμε. Για την καινούργια δράση προσθέτουμε στην $P5$ μία διαταραχή δ και αφαιρούμε από την $P1$ την δ και υπολογίζουμε τα συνημίτονα. Μετά κάνουμε ή όχι την αλλαγή.
2. Επιλέγουμε τον σύνδεσμο $\{i, j\}_j$ προς ανανέωση. Άρα κατασκευάζουμε τις πλακέτες $P1$ και $P4$, από το άθροισμα $\{i, j\}_j + \{i, j + 1\}_i - \{i + 1, j\}_j - \{i, j\}_i$ και $\{i - 1, j\}_j + \{i -$

¹Ο συμβολισμός $\{i, j\}_i$ δηλώνει τον σύνδεσμο στην κατεύθυνση i του κόμβου με συντεταγμένες i, j στο πλέγμα.



Σχήμα 4.2: Η γειτονιά μιας πλακέτας στην διδιάστατη περίπτωση, όπου φαίνονται οι κόμβοι, οι πλακέτες και οι φορές διαγραφής των πλακετών.

$1, j + 1\}_i - \{i, j\}_j - \{i - 1, j\}_i$ αντίστοιχα, και έτσι φτιάχνουμε την παλιά σημαντική δράση υπολογίζοντας τα συνημίτονα για κάθε πλακέτα που δημιουργήσαμε. Για την καινούργια σημαντική δράση προσθέτουμε στον σύνδεσμο $\{i, j\}_j$ μία διαταραχή δ , το οποίο σημαίνει ότι προσθέτουμε στην $P1$ μία διαταραχή δ και αφαιρούμε από την $P4$ την δ και υπολογίζουμε τα συνημίτονα. Μετά κάνουμε ή όχι την αλλαγή.

3. Επιλέγουμε τον σύνδεσμο $\{i, j + 1\}_i$ προς ανανέωση. Άρα κατασκευάζουμε τις πλακέτες P1 και P3, από το άθροισμα $\{i, j\}_j + \{i, j + 1\}_i - \{i + 1, j\}_j - \{i, j\}_i$ και $\{i, j + 1\}_j + \{i, j + 2\}_i - \{i + 1, j + 1\}_j - \{i, j + 1\}_i$ αντίστοιχα, και έτσι φτιάχνουμε την παλιά σημαντική δράση υπολογίζοντας τα συνημίτονα για κάθε πλακέτα που δημιουργήσαμε. Για την καινούργια σημαντική δράση προσθέτουμε στην $P1$ μία διαταραχή δ και αφαιρούμε από την $P3$ την δ και υπολογίζουμε τα συνημίτονα. Μετά κάνουμε ή όχι την αλλαγή.
4. Επιλέγουμε τον σύνδεσμο $\{i + 1, j\}_j$ προς ανανέωση. Άρα κατασκευάζουμε τις πλακέτες P1 και P2, από το άθροισμα $\{i, j\}_j + \{i, j + 1\}_i - \{i + 1, j\}_j - \{i, j\}_i$ και $\{i + 1, j\}_j + \{i + 1, j + 1\}_i - \{i + 2, j\}_j - \{i + 1, j\}_i$ αντίστοιχα, και έτσι φτιάχνουμε την παλιά σημαντική δράση υπολογίζοντας τα συνημίτονα για κάθε πλακέτα που δημιουργήσαμε. Για την καινούργια δράση προσθέτουμε στην $P2$ μία διαταραχή δ και αφαιρούμε από την $P1$ την δ και υπολογίζουμε τα συνημίτονα. Μετά κάνουμε ή όχι την αλλαγή.

Με αυτόν τον τρόπο λοιπόν έχουμε επισκεφτεί και τις τέσσερις πλευρές μίας πλακέτας με τον παλιό αλγόριθμο. Προσέψτε ότι κατασκευάσαμε την πλακέτα $P1$ και υπολογίσαμε το συνημίτονό της τέσσερις φορές!

Ας δούμε λοιπόν τί θα έκανε ο καινούργιος αλγόριθμος:

Η παλιά σημαντική δράση είναι η $\cos(P1) + \cos(P2) + \cos(P3) + \cos(P4) + \cos(P5)$, όπου πλέον τα P_i είναι απλοί αριθμοί, και η καινούργια είναι η $\cos(P1 + \delta) + \sum_i \cos(P_{i \neq 1} + \text{sign}_i \delta / 4.0)$. Το sign_i δηλώνει το πρόσημο με το οποίο υπολογίζεται η συνεισφορά της διαταραχής στην κάθε πλακέτα, όπως αυτό προκύπτει από τον χάρτη του πλέγματος. Αν

η ανανέωση γίνει, τότε η αλλαγή που γίνεται είναι στην πλακέτα $P1 \rightarrow P1 + \delta$ και στις άλλες πλακέτες $Pi_{i \neq 1} \rightarrow Pi_{i \neq 1} + \text{sign}_i \delta / 4.0$. Δηλαδή δίνεται ολόκληρη η διαταραχή στην πλακέτα που ανανεώνουμε και το ένα τέταρτο στις πλακέτες της γειτονιάς της ανάλογα με το πρόσημο που εμφανίζεται στον χάρτη.

Η βελτίωση είναι προφανής!

Το δεύτερο βήμα στην κατασκευή του αλγόριθμου ήταν η διαπίστωση ότι για την κατασκευή της δράσης χρειαζόμασταν τα συνημίτονα των πλακετών και τα συνημίτονα των πλακετών σύν την διαταραχή για την καινούργια δράση. Ο υπολογισμός των συνημιτόνων κοστίζει αρκετά οπότε θα ήταν πολύ επιθυμητό να μπορούσαμε να τους αποφύγουμε. Η ιδέα για να τους αποφύγουμε είναι πολύ απλή και στηρίζεται στους εξής τριγωνομετρικούς τύπους:

$$\cos(a + b) = \cos a \cos b - \sin a \sin b \quad (4.2)$$

$$\sin(a + b) = \sin a \cos b + \cos a \sin b \quad (4.3)$$

Αν λοιπόν έχουμε αποθηκεύσει κάπου το ημίτονο και το συνημίτονο της πλακέτας, τότε απλά υπολογίζουμε το ημίτονο και το συνημίτονο της διαταραχής και από εκεί και πέρα η κατασκευή της δράσης αλλά και η ανανέωση έχουν μόνο προσθέσεις!

Συμμαζεύοντας τα παραπάνω καταλήγουμε στον αλγόριθμο των πλακετών:

- Κατασκευάζουμε ένα πλέγμα, σε κάθε κόμβο του οποίου έχουμε $N = \sum_{i=1}^{D-1} i$ ζεύγη ημιτόνων και συνημιτόνων της γωνίας για κάθε πλακέτα $(\sin P, \cos P)$, όπου D είναι η διάσταση του προβλήματός μας.
- Κατασκευάζουμε τον χάρτη του πλέγματός μας, που δείχνει ποιές πλακέτες γειτνιάζουν με ποιές και με τί πρόσημο θα δεχθούν την ενδεχόμενη διαταραχή.
- Για κάθε πλακέτα τρέχουμε μια διαδικασία Monte Carlo, όπου η παλιά σημαντική δράση είναι το άθροισμα των συνημιτόνων των πλακετών και η καινούργια σημαντική δράση δίνεται από τον τύπο $\cos(P + \delta) = \sin P \cos \delta + \cos P \sin \delta$ για κάθε πλακέτα.
- Ανάλογα με το τί θα αποφανθεί ο έλεγχος του Monte Carlo, αν κάνουμε την αλλαγή, στην πλακέτα που ανανεώνουμε προσθέτουμε την διαταραχή, ενώ στις γειτονικές της προσθέτουμε ή αφαιρούμε, ανάλογα με το πρόσημο στον χάρτη, το ένα τέταρτο της διαταραχής.

Τα παραπάνω βήματα αποτελούν τον αλγόριθμο των πλακετών. Είναι αξιοσημείωτο να παρατηρήσουμε την ομοιότητα του προβλήματος, έτσι όπως αντιμετωπίζεται από τον αλγόριθμο, με ένα σπίν σύστημα, όπου σε κάθε κόμβο του κρυστάλλου έχουμε $N = \sum_{i=1}^{D-1} i$ σπίν, με D την διάσταση του χώρου μας.

4.1 Θεωρητική υποστήριξη

Αφού κατασκευάστηκε ο αλγόριθμος των πλακετών, με αφορμή καθαρά υπολογιστικούς λόγους, όπως αναφέρθηκε προηγουμένως, αναζητήθηκαν στην συνέχεια εργασίες στην διεθνή βιβλιογραφία, που μπορεί να ήταν σχετικές με το θέμα κυρίως σε θεωρητικό επίπεδο. Οι κυριώτερες τέτοιες εργασίες που βρέθηκαν ήταν η “String, corner, and plaquette formulation of finite lattice gauge theory” των G. Batrouni και M. Halpern [4] και η “Bianchi Identity for non-Abelian lattice gauge fields” του J. Kiskis [19].

Και οι δύο εργασίες είναι εξαιρετικά τεχνικές και η αναπαραγωγή τους σε αυτή την εργασία δεν κρίνεται αναγκαία. Αυτό που πρέπει να σημειωθεί, είναι ότι στην πρώτη εργασία κατασκευάζεται ένας φορμαλισμός ο οποίος “δουλεύει” με τις πλακέτες και δίνεται μια αρκετά αυστηρή μεθοδολογία για αυτό τον σκοπό. Έτσι η χρήση των πλακετών πλέον δείχνεται ότι μπορεί να γίνει σε φορμαλιστικό επίπεδο, οπότε ο αλγόριθμος που τις χειρίζεται σε υπολογιστικό επίπεδο έχει ένα σταθερό θεωρητικό έρεισμα.

Κατά την ανάγνωση και μελέτη των παραπάνω εργασιών, ένα αξιοσημείωτο γεγονός είναι η ύπαρξη των τοπικών και ολικών ταυτοτήτων Bianchi, οι οποίες προκύπτουν σε θεωρητικό επίπεδο. Για παράδειγμα, η ολική ταυτότητα Bianchi, απαιτεί το γινόμενο όλων των πλακετών στο πλέγμα να είναι μονάδα. Το γεγονός αυτό προκύπτει κατά τον υπολογισμό της συνάρτησης επιμερισμού:

$$Z = \int D\mathcal{P}_{\mu\nu} I \exp \left(\frac{\beta}{4} \sum_{\text{plaq}} \text{Tr}[\mathcal{P}_{\mu\nu}(r) + \mathcal{P}_{\mu\nu}^\dagger(r)] \right), \quad (4.4)$$

όπου

$$I = \delta \left(\prod_{x'=L_x-1}^0 \prod_{y'=0}^{L_y-1} \mathcal{P}_{xy}(x', y') - 1 \right). \quad (4.5)$$

Η συνάρτηση δέλτα που εμφανίζεται είναι η ολική ταυτότητα Bianchi.

Το εντυπωσιακό λοιπόν είναι ότι όλες αυτές οι ταυτοότητες, που υπαγορεύονται από θεωρητικά επιχειρήματα, ικανοποιούνται εκ κατασκευής στον αλγόριθμο των πλακετών! Χωρίς λοιπόν να καταφύγουμε σε περισσότερα τεχνικά ζητήματα, θα αρκестούμε στο ότι οι παραπάνω εργασίες, καθώς και μερικές ακόμη, παρουσιάζουν ένα θεωρητικό πλαίσιο πάνω στο οποίο μπορεί να στηριχθεί ο αλγόριθμος των πλακετών. Η πιστοποίηση του αλγόριθμου αφήνεται επομένως να γίνει σε υπολογιστικό επίπεδο, με την σύγκριση των αποτελεσμάτων που δίνει με άλλα αποτελέσματα της διεθνούς βιβλιογραφίας.

4.2 Η αρχή της λεπτομερούς ισορροπίας

Στο σημείο αυτό, θα πρέπει να αναφερθούμε σε μια πολύ βασική αρχή, αυτή της λεπτομερούς ισορροπίας. Η αρχή αυτή παρουσιάστηκε στο πρώτο κεφάλαιο. Με απλά λόγια η αρχή αυτή απαιτεί τον υπολογισμό της πιθανότητας να μεταβεί το σύστημα από την μία κατάσταση σε μία άλλη, αλλά και από τον υπολογισμό της πιθανότητας της αντίστροφης μετάβασης, έτσι ώστε να είμαστε σίγουροι ότι ο αλγόριθμός μας δίνει την σωστή κατανομή για τις καταστάσεις στην θερμική ισορροπία.

Στην περίπτωση του αλγόριθμου των συνδέσμων, οι δύο αυτές πιθανότητες είναι ίσες. Αυτό είναι σχετικά προφανές:

η ευθεία μετάβαση είναι να επιλέξουμε έναν σύνδεσμο και μετά μία διαταραχή από ένα συμμετρικό διάστημα $(-\delta, \delta)$. Η αντίστροφη μετάβαση είναι να επιλέξουμε τον ίδιο σύνδεσμο και μετά την αντίθετη διαταραχή ώστε το σύστημα να επανέλθει στην αρχική κατάσταση. Δεδομένου ότι το διάστημα από το οποίο επιλέγουμε τις διαταραχές είναι συμμετρικό και η κατανομή πιθανότητας να επιλέξουμε ένα στοιχείο είναι η ομοιόμορφη, η πιθανότητα επιλογής μια διαταραχής δ και της αντίθετης της $-\delta$ είναι η ίδια. Έτσι συνολικά η πιθανότητα της ευθείας και της αντίστροφης μετάβασης είναι η ίδια.

Στον αλγόριθμο των πλακετών ισχύει ακριβώς το ίδιο. Η ευθεία μετάβαση είναι να επιλέξουμε μία πλακέτα και μετά μία διαταραχή από ένα συμμετρικό διάστημα $(-\delta, \delta)$. Η διαταραχή αυτή θα μοιραστεί ολόκληρη στην επιλεγμένη πλακέτα και κατά το ένα τέταρτο στις γειτονικές ανάλογα με το πρόσημο που προκύπτει από το πλέγμα. Η αντίστροφη μετάβαση θα πρέπει να μπορεί να αναιρέσει την μεταβολή που κάναμε στο πλέγμα κατά την ευθεία μετάβαση. Δεδομένου του τρόπου που μοιράζουμε την διαταραχή στους γείτονες της επιλεγμένης πλακέτας, είναι φανερό ότι ο μόνος τρόπος να αναιρέσουμε την προηγούμενη αλλαγή, είναι να επιλέξουμε την ίδια πλακέτα και να επιλέξουμε την αντίθετη διαταραχή. Η διαταραχή αυτή θα αναιρέσει την αλλαγή και στην επιλεγμένη πλακέτα και στις γειτονικές, επαναφέροντας το πλέγμα στην αρχική κατάσταση. Πάλι, αφού η κατανομή πιθανότητας που αφορά την επιλογή της διαταραχής από το συμμετρικό διάστημα είναι η ομοιόμορφη, η πιθανότητα επιλογής μια διαταραχής δ και της αντίθετης της $-\delta$ είναι η ίδια. Συνολικά λοιπόν, και στην περίπτωση του αλγόριθμου των πλακετών η αρχή της λεπτομερούς ισορροπίας ισχύει.

4.3 Υλοποίηση και βελτιστοποιήσεις

Στο κεφάλαιο αυτό παρουσιάζονται θέματα υλοποίησης του αλγόριθμου των πλακετών και διαμόρφωσης του περιβάλλοντος του Grid και την εκτέλεση των σχετικών προγραμμάτων καθώς και δύο βελτιστοποιήσεις που θεωρήθηκαν σε υπολογιστικό επίπεδο.

4.3.1 Θέματα υλοποίησης

Η υλοποίηση του αλγόριθμου των πλακετών αποτελείται από δύο κύρια μέρη, το πρόγραμμα κατασκευής του χάρτη πλέγματος και το πρόγραμμα της διαδικασίας Monte Carlo, καθώς και μία σειρά δευτερευόντων βοηθητικών προγραμμάτων, που περιγράφονται παρακάτω.

Το πρόγραμμα κατασκευής του χάρτη πλέγματος

Το πρόγραμμα αυτό κατασκευάζει τον χάρτη του πλέγματος, υλοποιήθηκε με την γλώσσα προγραμματισμού C++ και ονομάζεται Mapper. Η δημιουργία του χάρτη του πλέγματος γίνεται δημιουργώντας πρώτα και αριθμοδοτώντας όλες τις πλακέτες που εμφανίζονται στο πλέγμα από τους συνδέσμους που τις αποτελούν, και μετά ελέγχοντας ποιές πλακέτες έχουν κοινούς σύνδεσμους. Το πρόγραμμα αυτό υπάρχει σε ξεχωριστές εκδόσεις για κάθε μήκος πλέγματος και διάσταση της θεωρίας. Η εκτέλεσή του γίνεται μία φορά στην αρχή και μετά αναλαμβάνει το πρόγραμμα της προσομοίωσης. Ο πηγαίος κώδικας υπάρχει στο παράρτημα Γ'.

Το πρόγραμμα Monte Carlo

Το πρόγραμμα αυτό τρέχει την προσομοίωση Monte Carlo, υλοποιήθηκε με την γλώσσα προγραμματισμού C και ονομάζεται Sweeper. Ο πηγαίος κώδικας υπάρχει στο παράρτημα Δ'. Οι παράμετροι που δέχεται είναι ο αριθμός των διασχίσεων του πλέγματος που πρέπει να εκτελεστούν συνολικά, ο αριθμός των διασχίσεων του πλέγματος που θα θεωρούνται θέρμανση (thermalization) και κατά τις οποίες δεν θα λαμβάνονται μετρήσεις, η τιμή της παραμέτρου β καθώς και το όνομα του αρχείου που περιέχει τον χάρτη του πλέγματος.

Το πρόγραμμα αυτό έχοντας δημιουργήσει στην μνήμη μια δομή όλων των πλακετών του πλέγματος, επισκέπτεται μία μία τις πλακέτες και με βάση την γειτονία τους, την οποία την βρίσκει από τον χάρτη του πλέγματος, κατασκευάζει την διαφορά της παλιάς σημαντικής δράσης από την καινούργια και με βάση αυτή μεταδίδει την διαταραχή στο πλέγμα ή όχι.

Το πρόγραμμα αρχικοποίησης του πλέγματος

Το πρόγραμμα αυτό απλά κατασκευάζει ένα πλέγμα το οποίο έχει όλες τις διπλέτες των πλακετών ίσες με $(1,0)$, δηλαδή το συνημίτονο ήταν μονάδα και το ημίτονο ήταν μηδέν. Το πλέγμα αυτό είναι η αρχική κατάσταση από την οποία θα ξεκινάει κάθε προσομοίωση. Το πρόγραμμα αυτό επειδή είναι τετριμένο, δεν παρουσιάζεται ο πηγαίος κώδικάς του στο παρόν κείμενο.

Τα προγράμματα ανάλυσης

Τα προγράμματα ανάλυσης εφαρμόζονται στα αρχεία μετρήσεων που προκύπτουν από την προσομοίωση. Υπολογίζουν την μέση τιμή, την επιδεκτικότητα και τα σχετιζόμενα σφάλματα ή κατασκευάζουν τα σχετικά ιστογράμματα. Ο πηγαίος κώδικας του προγράμματος που υπολογίζει την επιδεκτικότητα και μερικές άλλες στατιστικές συναρτήσεις υπάρχει στο παράρτημα Ε'.

Τα scripts για το Grid

Η εκτέλεση του αλγόριθμου στο Grid γίνεται με δύο τρόπους και περιγράφηκε σε scripts της γλώσσας Python. Με τον πρώτο τρόπο, εκτελούμε τον αλγόριθμο για μια συγκεκριμένη τιμή της παραμέτρου β και μετά υπολογίζουμε διάφορες στατιστικές παραμέτρους πάνω στο αρχείο των μετρήσεων και δημιουργούμε και το ιστόγραμμα των κατανομών των μετρήσεων. Για την εκτέλεση της παραπάνω διαδικασίας στο Grid χρησιμοποιήθηκε το εξής script:

```
#!/usr/bin/env python

import os, locale

os.system('chmod a+x LatticeMap4D_06')
os.system('chmod a+x InitLatGen')
os.system('chmod a+x Sweeper06')
os.system('chmod a+x Bin2txt')
os.system('chmod a+x Cumulants')
os.system('chmod a+x Binner')
os.system('./LatticeMap4D_06')
os.system('./InitLatGen')
os.system('./Sweeper06 1000000 10000 1.0099 4D_6.map')
os.system('./Bin2Txt MeanCosPlaquette.dat')
os.system('./Cumulants MeanCosPlaquette.tsv 1.0099 6')
os.system('./Binner MeanCosPlaquette.dat 100')
```

Παρατηρήστε στην αρχή του script την εκτέλεση των εντολών `chmod a+x` πάνω στα αρχεία προς εκτέλεση. Όπως αναφέρθηκε και στο τρίτο κεφάλαιο, οι εντολές αυτές είναι απαραίτητες διότι ο χαρακτήρας του εκτελέσιμου δεν διατηρείται κατά την μεταφορά των αρχείων στον κόμβο εκτέλεσης.

Με τον δεύτερο τρόπο, εκτελούμε τον αλγόριθμο σε ένα εύρος τιμών της παραμέτρου β με ένα σταθερό βήμα και για έναν σχετικά μικρό αριθμό επαναλήψεων. Μετά υπολογίζουμε την μέση τιμή των μετρήσεων για κάθε β και αποθηκεύουμε το ζευγάρι β -μέση τιμή σε ένα αρχείο. Για την εκτέλεση της παραπάνω διαδικασίας χρησιμοποιήθηκε το εξής script:

```
#!/usr/bin/env python

import os, locale

os.system('chmod a+x LatticeMap4D_06')
os.system('chmod a+x InitLatGen')
os.system('chmod a+x Sweeper06')
os.system('chmod a+x MeanCos')
os.system('./LatticeMap4D_06')
os.system('./InitLatGen 6 0')
Beta=0.98
for i in xrange(50):
    os.system('./Sweeper06 10000000 10000 '+str(Beta)+' 4D_6.map')
    os.system('./MeanCos MeanCosPlaquette.dat '+str(Beta))
    os.system('rm MeanCosPlaquette.dat')
    Beta=Beta+0.001
```

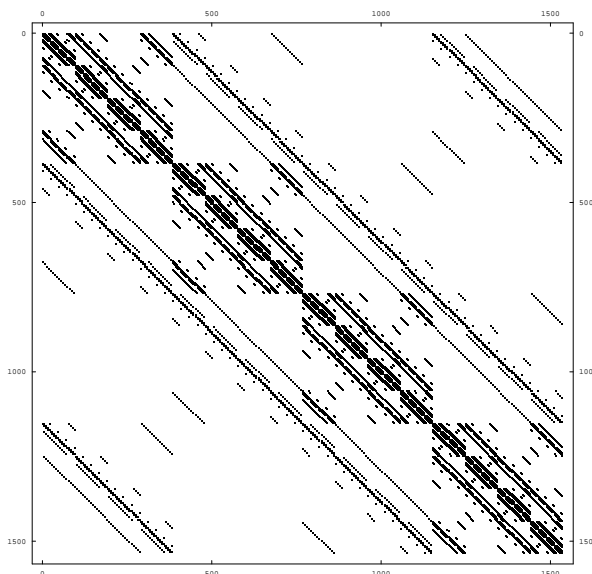
4.3.2 Βελτιστοποιήσεις

Κατά την υλοποίηση του αλγόριθμου θεωρήθηκαν δύο βασικές βελτιστοποιήσεις. Η μία αφορούσε την δημιουργία του χάρτη πλέγματος που τελικά δεν υλοποιήθηκε, και η άλλη αφορούσε τον έλεγχο που κάνει η διαδικασία Monte Carlo.

Βελτιστοποίηση στον χάρτη πλέγματος

Πρέπει να σημειωθεί ότι αυτή η βελτιστοποίηση δεν υλοποιήθηκε για την τετραδιάστατη θεωρία πεδίου αλλά καθώς ο αλγόριθμος έτρεξε και πάνω στην εξαδιάστατη θεωρία, της οποίας η πολυπλοκότητα είναι πολλές τάξεις μεγαλύτερη, η βελτιστοποίηση μελετήθηκε διεξοδικά και υλοποιήθηκε για τις έξι διαστάσεις.

Ένα γνωστό πρόβλημα, για όλους τους αλγόριθμους που δρουν πάνω σε κάποιο πλέγμα, είναι ότι η απλή αριθμοδότηση και αποθήκευση της δομής σε έναν μονοδιάστατο πίνακα είναι πολύ μη αποδοτική. Αυτό συμβαίνει διότι όταν ο υπολογιστής θέλει να εργαστεί με κάποιο στοιχείο και τους γείτονές του στο πλέγμα, θα αναζητήσει το στοιχείο και τους γείτονες στην μονοδιάστατη δομή, και θα φέρει στην μνήμη cache τις σελίδες από την μνήμη που περιέχουν τα εν λόγω στοιχεία. Με την απλή αριθμοδότηση, δυστυχώς, η απόσταση των γειτονικών στοιχείων στην γραμμική δομή είναι πολύ μεγάλη! Έτσι ο υπολογιστής αναγκάζεται να κάνει πολλές μεταφορές και εγγραφές σελίδων στην μνήμη, οι οποίες περ-



Σχήμα 4.3: Ο πίνακας γειτνίασης για ένα χάρτη ενός πλέγματος 6^4 .

ιέχουν πολύ λίγα οφέλιμα στοιχεία.

Το καλύτερο σενάριο είναι η γειτνίαση των στοιχείων στο πλέγμα να σημαίνει και την γειτνιάσή τους στην μονοδιάστατη δομή αποθήκευσης του πλέγματος. Αυτό το πρόβλημα μπορεί να αντιμετωπιστεί με τρεις τρόπους: ο πρώτος είναι να βρούμε έναν έξυπνο ευριστικό τρόπο αριθμοδότησης πάνω στο πλέγμα ώστε να επιτύχουμε αυτό που θέλουμε. Δυστυχώς η εύρεση ενός τέτοιου τρόπου είναι αρκετά δύσκολη και εξαρτάται απόλυτα από το πρόβλημα που καλούμαστε να λύσουμε στο πλέγμα. Γι'αυτό τον λόγο έχουν προταθεί σενάρια αριθμοδότησης που στατιστικά περιμένουμε να φέρουν αρκετά κοντά τους γείτονες στην γραμμική δομή. Τέτοια είναι σενάρια αριθμοδότησης με καμπύλες Hilbert πάνω στο πλέγμα.

Ο δεύτερος τρόπος είναι να χρησιμοποιήσουμε πλήρη δυαδικά δένδρα για την αρίθμηση του πλέγματος. Για την κατασκευή ενός πλήρους δυαδικού δένδρου σε ένα πλέγμα τριών διαστάσεων κάνουμε τα εξής:

χωρίζουμε το κυβικό πλέγμα σε οκτώ υποκύβους και αυτούς σε οκτώ υποκύβους τον καθένα και ούτω καθεξής μέχρι οι υποκύβοι μας να είναι οι κόμβοι του πλέγματος. Έχουμε λοιπόν ένα δένδρο με ρίζα το κύβο όλου του πλέγματος, οκτώ παιδιά στο πρώτο επίπεδο, εξήντα τέσσερα παιδιά στο δεύτερο και τα λοιπά, και φύλλα τους κόμβους του πλέγματος. Μετά αριθμούμε τα φύλλα του δένδρου από τα αριστερά προς τα δεξιά.

Το παραπάνω σχήμα είναι αρκετά αποδοτικό αφού καταφέρνει να μαζέψει κοντά την γειτονιά κάθε στοιχείου και δίνει 77% επιτυχία εύρεσης στην μνήμη cache σε αντίθεση με το 55% στην περίπτωση της απλής αριθμοδότησης. Ο μόνος περιορισμός της μεθόδου είναι ότι πρέπει οι πλευρές του πλέγματος με το οποίο θα ασχοληθούμε να είναι δύναμη του δύο.

Ο τρίτος τρόπος είναι να θεωρήσουμε τον πίνακα γειτνίασης του πλέγματος. Αριθμούμε με τον απλοϊκό τρόπο το πλέγμα και μετά κατασκευάζουμε τον πίνακα γειτνίασης, ο οποίος είναι ένας διδιάστατος πίνακας $N \times N$, όπου N το πλήθος των στοιχείων του πλέγματος, και ο οποίος σε κάθε γραμμή του έχει μονάδες για όλους τους γείτονες του στοιχείου που έχει τον ίδιο αριθμό με την γραμμή και μηδέν σε όλα τα άλλα στοιχεία. Ο πίνακας αυτός είναι αραιός, και ένας τέτοιος πίνακας εμφανίζεται στο σχήμα (4.3) για ένα πλέγμα

6⁴. Για να έχουμε συνθήκες καλής γειτνίασης στην γραμμική δομή, είναι προφανές ότι θα θέλαμε τα στοιχεία του πίνακα γειτνίασης να είναι όσο το δυνατόν περισσότερο συγκεντρωμένα περί την διαγώνιο. Αυτό το πρόβλημα μπορούμε να το αντιμετωπίσουμε σαν ένα καθαρό πρόβλημα βελτιστοποίησης όπου έχουμε έναν διδιάστατο αραιό πίνακα από bits και θέλουμε με επαναριθμοδότηση των δεικτών του πίνακα να συγκεντρώσουμε τις μονάδες όσο το δυνατόν περισσότερο γύρω από την διαγώνιο του πίνακα.

Τόσο ο πρώτος όσο και ο τρίτος τρόπος αρίθμησης είναι ακρετά αποδοτικοί, όμως το πρόβλημα είναι ότι ο πρώτος δεν είναι εύκολο να υλοποιηθεί και εξαρτάται πάρα πολύ από το μήκος του πλέγματος και την διάστασή του. Ο τρίτος, από την άλλη πλευρά, είναι και δύσκολο σχετικά να υλοποιηθεί αλλά κυρίως είναι χρονοβόρος στην εκτέλεσή του. Έτσι τελικά συμφέρει η υλοποίηση του δεύτερου τρόπου αρίθμησης με τα δυαδικά δένδρα, αφού και εύκολη είναι η υλοποίησή του και η εκτέλεσή του είναι γρήγορη.

Βελτιστοποίηση στο έλεγχο του Monte Carlo

Η διαδικασία Monte Carlo κάνει τον έλεγχο

$$r > \ln(\Delta S) \quad (4.6)$$

όπου r είναι ένα τυχαίος αριθμός και ΔS η διαφορά που γίνεται στην δράση με την διαταραχή που ελέγχουμε.

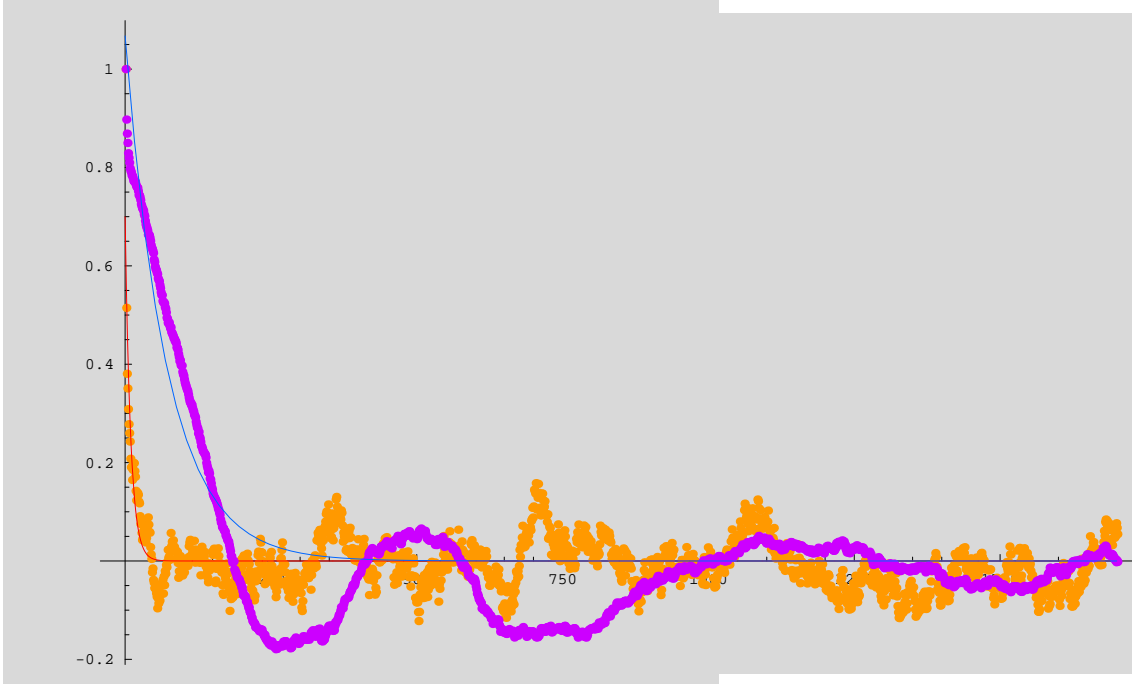
Ο απλούστερος τρόπος, προφανώς, είναι να κατασκευάζουμε τον τυχαίο αριθμό, να υπολογίζουμε τον λογάριθμο του ΔS και να κάνουμε τον έλεγχο της ανισότητας. Όμως ο υπολογισμός του λογαρίθμου κοστίζει πολύ! Ο υπολογιστής καλείται να υπολογίσει όρους της σειράς του λογαρίθμου μέχρι να επιτύχει την ακρίβειά του (περίπου 16 σημαντικά ψηφία). Αν σκεφτούμε λίγο όμως, αυτό είναι περιττό λόγω του ελέγχου.

Το σχήμα που προτείνουμε είναι το εξής: κατασκευάζουμε τον τυχαίο αριθμό, μετά υπολογίζουμε τον πρώτο όρο της σειράς του λογαρίθμου και συγκρίνουμε τους δύο αριθμούς. Αν ο έλεγχος είναι αρνητικός τότε σταματάμε εδώ. Αν ο έλεγχος ικανοποιείται, τότε υπολογίζουμε και τον δεύτερο όρο της σειράς του λογαρίθμου και συγκρίνουμε τον τυχαίο αριθμό με τους δύο πρώτους όρους της σειράς. Αν ο δεύτερος έλεγχος αποτύχει τότε σταματάμε εδώ. Αν ο δεύτερος έλεγχος ικανοποιηθεί προχωράμε και στον επόμενο όρο της σειράς του λογαρίθμου και ούτω καθεξής, μέχρι κάποιον επιλεγμένο όρο. Αν ο τυχαίος αριθμός έχει περάσει όλους τους ελέγχους μέχρι εκεί, τότε υπολογίζουμε τον λογάριθμο ακριβώς. Αυτή η διαδικασία φυσικά είναι αποδοτική στους αποτυχημένους ελέγχους της διαδικασίας Monte Carlo διότι σε αυτή την περίπτωση ο υπολογιστής δεν χρειάζεται να υπολογίσει ολόκληρο τον λογάριθμο όταν φτάνουν μόνο κάποιοι όροι της σειράς για τον έλεγχο.

Ο τροποποιημένος έλεγχος γίνεται από την συνάρτηση `check` και φαίνεται στη αρχή του πηγαίου κώδικα του προγράμματος `sweeper` στο παράρτημα Β'.

4.4 Σύγκριση χρόνων αυτοσυσχέτισης

Μία πολύ βασική έννοια είναι αυτή του χρόνου αυτοσυσχέτισης. Κατά την διάρκεια της προσομοίωσης του υπό μελέτη συστήματος στον υπολογιστή, παράγονται, σε κάθε σάρωση, συλλογές (configurations) που η κάθε μια συνίσταται από το σύνολο των τιμών σε κάθε πλακέτα των μετρούμενων ποσοτήτων. Ακόμα και όταν έχουμε υποβάλλει το σύστημα



Σχήμα 4.4: Οι συναρτήσεις αυτοσυσχέτισης για τους αλγόριθμους των πλακετών (πορτοκαλί) και των συνδέσμων (μώβ).

σε μια πολύ ικανοποιητική θέρμανση, ώστε να φτάσει στην θερμική ισορροπία, θα πρέπει κάποιος να είναι πολύ αισιόδοξος για να πιστεύει πως η μία συλλογή είναι ανεξάρτητη της άλλης. Στην πραγματικότητα υπάρχει μία έντονη συσχέτιση οφειλόμενη στο γεγονός πως η κάθε συλλογή επηρεάζεται από τις προηγούμενες και με την σειρά της επηρεάζει τις επόμενες. Κανείς πρέπει να μπορεί να είναι σε θέση να εκτιμήσει το βαθμό συσχέτισης των συλλογών ώστε να είναι σε θέση να υπολογίσει και το σφάλμα στις μετρήσεις που επιχειρεί και να έχει μια ένδειξη της ποιότητας των μετρήσεών του. Ο χρόνος αυτοσυσχέτισης λοιπόν είναι ένας καλός δείκτης για την ποιότητα του καινούργιου αλγόριθμου των πλακετών, και έτσι συγκρίθηκε με τον αντίστοιχο του αλγόριθμου των συνδέσμων.

Ορίζοντας ως αυτοσυσχέτιση για τις μετρήσεις στις συλλογές ενός τυχαίου μεγέθους A το ακόλουθο μέγεθος

$$(A_\nu A_{\nu+\tau}) = \langle A_\nu A_{\nu+\tau} \rangle - \langle A_\nu \rangle \langle A_{\nu+\tau} \rangle \quad (4.7)$$

μπορεί να φθάσει κανείς, με στατιστική ανάλυση των μετρήσεων, στον γενικό ορισμό του ολοκληρωμένου χρόνου αυτοσυσχέτισης (integrated autocorrelation time) που δίνεται από την σχέση:

$$\tau_{int,A} = \frac{1}{2} \sum_{\tau=-\infty}^{\infty} \frac{(A_\nu A_{\nu+\tau})}{(\langle A^2 \rangle - \langle A \rangle^2)}. \quad (4.8)$$

Η ποσότητα $\tau_{int,A}$ έχει την έννοια του υπολογιστικού χρόνου που χωρίζει την δημιουργία δύο στατιστικώς ανεξάρτητων συλλογών του μετρούμενου μεγέθους A .

Στο σχήμα (4.4) φαίνονται οι συναρτήσεις αυτοσυσχέτισης για τον αλγόριθμο των πλακετών και για τον αλγόριθμο των συνδέσμων. Η τιμή του χρόνου αυτοσυσχέτισης για τον αλγόριθμο των πλακετών υπολογίστηκε στο 10 περίπου ενώ του αλγόριθμου των συνδέσμων στο 72.

Βλέπουμε λοιπόν ότι ο καινούργιος αλγόριθμος των πλακετών δίνει ποιοτικά καλύτερες μετρήσεις καθώς αυτές είναι λιγότερο συσχετισμένες από αυτές του αλγόριθμου των συνδέσμων. Το γεγονός αυτό, κατά πάσα πιθανότητα οφείλεται στην σχετικά μη-τοπική φύση του καινούργιου αλγόριθμου.

4.5 Σύγκριση επιδόσεων ως προς τον χρόνο

Όταν πλέον η υλοποίηση του αλγόριθμου είχε υποστηριχθεί με όλες τις δυνατές βελτιστοποιήσεις, έγιναν κάποια συγκριτικά με μια βελτιστοποιημένη υλοποίηση του απλού αλγόριθμου με τους συνδέσμους. Και οι δύο αλγόριθμοι εκτελέστηκαν στον ίδιο υπολογιστή και στο ίδιο υπολογιστικό περιβάλλον για μια σειρά μηκών πλέγματος και για την τετραδιάστατη και εξαδιάστατη περίπτωση. Ο χρόνος εκτέλεσης του αλγόριθμου των πλακετών έδειχνε να είναι συνέχεια μεγαλύτερος από τον χρόνο εκτέλεσης του αλγόριθμου των συνδέσμων. Χαρακτηριστικά, για ένα εξαδιάστατο πλέγμα και μήκος πλέγματος 4, ο αλγόριθμος των πλακετών ήταν τρεις σχεδόν φορές πιο αργός από τον αλγόριθμο των συνδέσμων.

Κεφάλαιο 5

Αποτελέσματα και ανάλυση αποτελεσμάτων

Στο κεφάλαιο αυτό παρουσιάζονται τα αποτελέσματα, που έδωσε η προσομοίωση Monte Carlo για την τετραδιάστατη συμπαγή θεωρία βαθμίδας $U(1)$ χωρίς φερμιόνια με την μέθοδο των πλακετών, και συγκρίνονται με αποτελέσματα που εμφανίστηκαν σε εργασίες της διεθνούς βιβλιογραφίας.

Οι προσομοιώσεις που έγιναν αφορούσαν μήκη πλέγματος 6, 8, 10 και 12 και δεκαπέντε τιμές της παραμέτρου β ανομοιογενώς κατανεμημένων στην περιοχή (0.998-1.14) για κάθε μήκος πλέγματος. Ο τρόπος με τον οποίο κατανεμήθηκαν οι τιμές, στις οποίες έγιναν οι εξομοιώσεις, ήταν τέτοιος ώστε να υπάρξουν αρκετές μετρήσεις γύρω από την θέση όπου αναμενόταν το κρίσιμο σημείο. Θα πρέπει να αναφερθεί ότι συνολικά ο χρόνος εκτέλεσης των προσομοιώσεων, αν αυτές γινόντουσαν σε έναν επεξεργαστή θα ήταν περίπου ένας χρόνος συνεχούς λειτουργίας!

5.1 Παρατηρήσιμες ποσότητες

Καταρχήν ας ορίσουμε την μέση πλακέτα ως τον μέσο όρο πάνω στον όγκο:

$$\hat{P} \equiv \frac{1}{6V} \sum (P_1), \quad (5.1)$$

όπου V ο όγκος του πλέγματος και P_1 το “πάνω” μέρος των διπλετών που αφορούν την πλακέτα και είναι ουσιαστικά τα συνημίτονα της για κάθε επίπεδο.

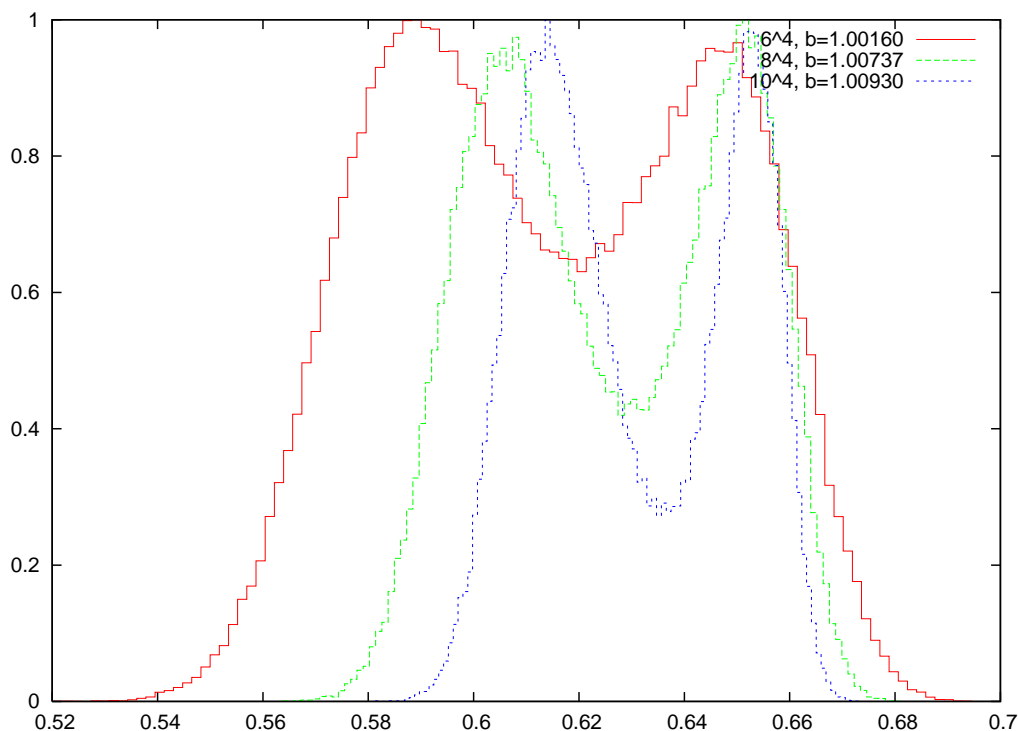
Οι παρατηρήσιμες ποσότητες που θεωρήθηκαν είναι:

1. Η χρονική μέση τιμή της πλακέτας:

$$P \equiv \langle \hat{P} \rangle, \quad (5.2)$$

2. Η επιδεκτικότητα (susceptibility):

$$S = V \left(\langle \hat{P}^2 \rangle - \langle \hat{P} \rangle^2 \right) \quad (5.3)$$



Σχήμα 5.1: Τα ιστογράμματα της κατανομής της μέσης τιμής στον όγκο των πλακετών για διάφορους όγκους.

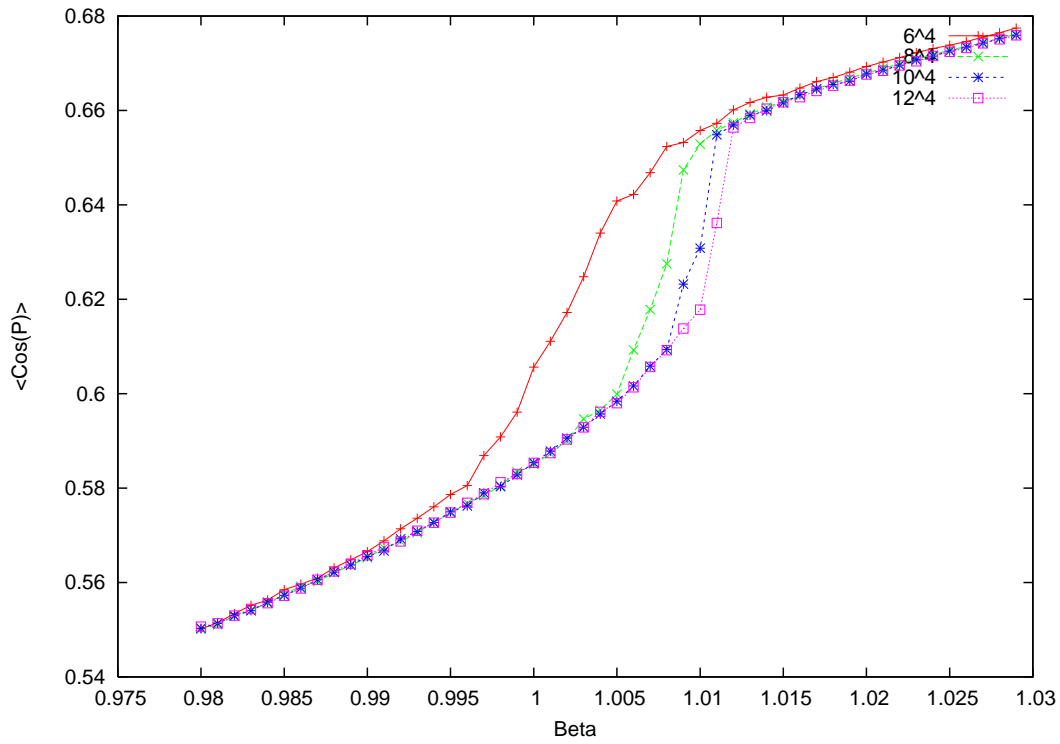
Ένα πρώτο αποκαλυπτικό γράφημα είναι αυτό του ιστογράμματος της κατανομής του μέσου όρου των πλακετών στον όγκο. Όπως φαίνεται και στο γράφημα (5.1), είναι εμφανής η δομή των δύο ξεχωριστών κορυφών, που είναι μια πρώτη ένδειξη αλλαγής φάσης πρώτης τάξης. Το συγκεκριμένο γράφημα δείχνει τα ιστογράμματα των κατανομών για τους όγκους 6^4 , 8^4 και 10^4 με τιμές της παραμέτρου β 1.0016, 1.00737 και 1.0093 αντίστοιχα.

Ένα άλλο σημαντικό γράφημα είναι αυτό της καμπύλης της μέσης τιμής των πλακετών συναρτήσει της παραμέτρου β , το οποίο καθαρά φαίνεται ότι γίνεται όλο και πιο απότομο καθώς αυξάνει ο όγκος του πλέγματος. Το γεγονός αυτό είναι άλλη μία ένδειξη ότι έχουμε να κάνουμε με αλλαγή φάσης πρώτης τάξης. Το γράφημα αυτό είναι το (5.2).

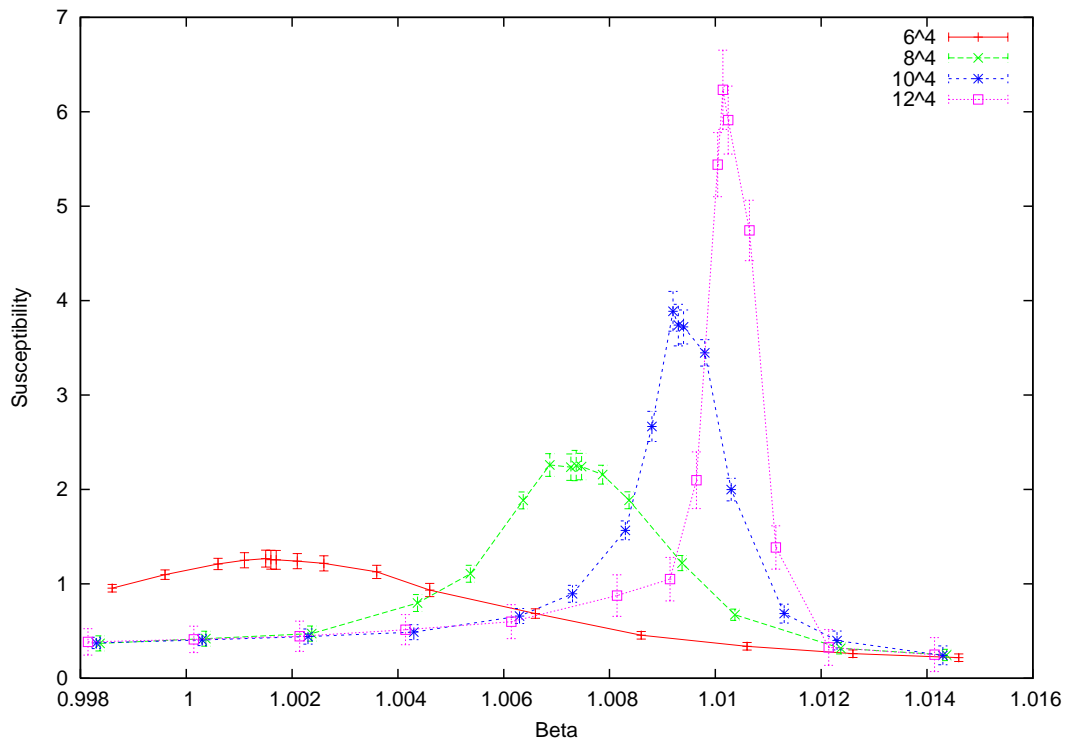
Η πιο βασική παρατηρήσιμη ποσότητα όμως για την ανάλυσή μας ήταν η επιδεικτικότητα. Το γράφημα (5.3) δείχνει την επιδεικτικότητα υπολογισμένη σε διάφορες τιμές της παραμέτρου β και για μήκη πλέγματος 6, 8, 10 και 12. Είναι γνωστό ότι για αλλαγές φάσης πρώτης τάξης το μέγιστο της επιδεικτικότητας μεταβάλλεται σύμφωνα με τον όγκο του πλέγματος, ενώ για μία αλλαγή φάσης δεύτερης τάξης το μέγιστο μεταβάλλεται σύμφωνα με κάποια δύναμη του όγκου μικρότερη του ένα. Οι λόγοι των όγκων είναι λοιπόν $\frac{10^4}{8^4} \simeq 2.44$ και $\frac{12^4}{8^4} \simeq 5.06$ ενώ οι αντίστοιχοι λόγοι των μεγίστων είναι $\frac{3.74077}{2.25907} \simeq 1.65589$ και $\frac{6.232680}{2.25907} \simeq 2.75896$. Μια γραμμική προσαρμογή της εξάρτησης από τον όγκο των μεγίστων της επιδεικτικότητας φαίνεται στο σχήμα (5.4). Μπορούμε λοιπόν να μιλάμε με κάποια σιγουριά για αλλαγή φάσης πρώτης τάξης.

Τα σφάλματα που φαίνονται υπολογίστηκαν με την μέθοδο jackknife¹.

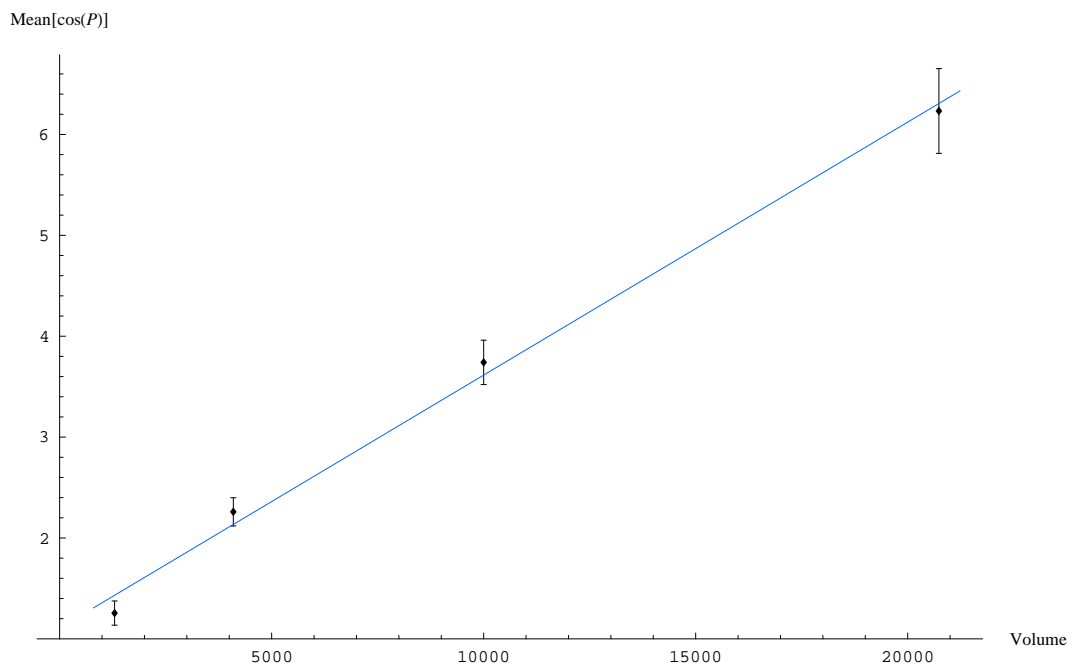
¹Η μέθοδος jackknife παρουσιάζεται στο παράρτημα Β'.



Σχήμα 5.2: Η κρίσιμη περιοχή για τους όγκους 6^4 , 8^4 , 10^4 και 12^4 .



Σχήμα 5.3: Η επιδεκτικότητα για τους όγκους 6^4 , 8^4 , 10^4 και 12^4 .



Σχήμα 5.4: Γραμμική προσαρμογή των μεγίστων της επιδεικτικότητας.

5.2 Συγκριτικά αποτελεσμάτων

Από την προηγούμενη ανάλυση μπορούμε πλέον να υποθέσουμε ότι έχουμε να κάνουμε όντως με αλλαγής φάσης πρώτης τάξης. Έτσι προχωράμε σε ανάλυση πεπερασμένης κλιμάκωσης (finite scaling analysis), όπως αυτή περιγράφεται στην εργασία των G. Arnold, Th. Lippert και K. Schilling [2].

Από τις τιμές της επιδεικτικότητας, υπολογίστηκε το ψευδοκρίσιμο σημείο για κάθε όγκο, το σημείο δηλαδή που εμφανίζεται το μέγιστό της. Το μέγιστο της επιδεικτικότητας υπολογίστηκε με βάση την μέθοδο multihistogram² σε τέσσερις εκτελέσεις της προσομοίωσης σε τιμές της παραμέτρου κοντά στο σημείο όπου αναμενώταν το μέγιστο.

Από τα ψευδοκρίσιμα σημεία υπολογίσαμε το κρίσιμο σημείο με βάση τον τύπο

$$\beta_c(V) = \beta_\infty + \sum_{k=1}^{\infty} \frac{A_k}{V^k} \quad (5.4)$$

και την τιμή του διακένου με βάση τον τύπο

$$\frac{\beta^2 S_{max}(P_s)}{V} = \frac{1}{4} G^2 + \sum_{k=1}^{\infty} \frac{B_k}{V^k} \quad (5.5)$$

όπου G είναι η τιμή του διακένου της ενέργειας σε άπειρο όγκο και β_∞ είναι το κρίσιμο σημείο του συστήματος για άπειρο όγκο.

Χρησιμοποιώντας λοιπόν τους τύπους (5.4) και (5.5) και κρατώντας μέχρι και όρους V^{-3} υπολογίσαμε την τιμή του διακένου και του κρίσιμου σημείου για το όριο του άπειρου όγκου. Ακολουθούν οι τιμές που υπολογίσθηκαν στις εργασίες των G. Arnold, Th. Lippert,

²Η μέθοδος multihistogram παρουσιάζεται στο παράρτημα Α'

K. Schilling, των Σ. Βρέντζου και Κ. Φαράκου και στην παρούσα αντίστοιχα.

L	β_c	$C_{u,max}/6V \times 10^{-4}$
6	1.001794	9.7728
8	1.007413	5.5428
10	1.009383	3.8535
12	1.010229	2.9813

$$\beta_\infty = 1.011128$$

$$G = 0.02698$$

(από την εργασία των G. Arnold, Th. Lippert, K. Schilling.)

L	β_c	$C_{u,max}/6V \times 10^{-4}$
6	1.0018	8.711
8	1.0073	5.524
10	1.009	3.858
12	1.0099	2.975

$$\beta_\infty = 1.0106$$

$$G = 0.0306$$

(από την υπό έκδοση εργασία των Σ. Βρέντζου και Κ. Φαράκου)

L	β_c	$C_{u,max}/6V \times 10^{-4}$
6	1.001590	9.74528
8	1.007436	5.56124
10	1.00937	3.86652
12	1.010188	3.03328

$$\beta_\infty = 1.01102$$

$$G = 0.0291456$$

Βλέπουμε την καλή συμφωνία με τις μετρήσεις που έχουν γίνει στις άλλες εργασίες.

Κεφάλαιο 6

Συμπεράσματα

Ανακεφαλαιώνοντας, στο παρόν κείμενο παρουσιάσαμε έναν καινούργιο αλγόριθμο για προσομοίωση Monte Carlo μιας συμπαγούς θεωρίας βαθμίδας $U(1)$ χωρίς φερμιόνια. Ο αλγόριθμος αυτός αναδεικνύει έναν ομομορφισμό του προβλήματος, με το παλιό πλέγμα με τους σύνδεσμους, με ένα πρόβλημα ενός σπίν συστήματος. Αντί να έχουμε να κάνουμε με μεταβλητές συνδέσμων και να κατασκευάζουμε τον βρόχο Wilson πλέον έχουμε να κάνουμε με διπλές σπίν, οι οποίες αντιπροσωπεύουν το πραγματικό και μιγαδικό μέρος του βρόχου.

Ο καινούργιος αλγόριθμος, αν και αναμενόταν να είναι πιο γρήγορος από τον προηγούμενο, τελικά αποδείχθηκε ότι σε αρκετά μεγάλα προβλήματα (μεγάλης διάστασης) είναι πιο αργός στην παρούσα υλοποίησή του. Αν και δεν μελετήθηκε διεξοδικά γιατί συμβαίνει αυτό, μια πρώτη ένδειξη είναι ότι η γειτονιά κάθε σπίν μεγαλώνει πάρα πολύ με την διάσταση του πλέγματος. Χαρακτηριστικά, ενώ στον αλγόριθμο των συνδέσμων στο εξαδιάστατο πλέγμα χρειαζόμαστε 10 πλακέτες για κάθε κόμβο, στην περίπτωση του αλγόριθμου των πλακετών χρειαζόμαστε 37!

Από την άλλη πλευρά, λόγω της μη-τοπικότητας του αλγόριθμου, ο χρόνος αυτοσυσχέτισης των μετρήσεων είναι αρκετά μικρότερος από τον αντίστοιχο του παλιού αλγόριθμου. Αυτό σημαίνει ότι η ποιότητα των μετρήσεων είναι ανώτερη.

Τέλος, τα αποτελέσματα που έδωσε ο αλγόριθμος για την τετραδιάστατη περίπτωση συμφωνούν απόλυτα με τα αποτελέσματα που υπάρχουν στην βιβλιογραφία.

Παράρτημα Α΄

Η μέθοδος multihistogram

Είναι προφανές πως η διαδικασία της προσομοίωσης ενός φυσικού συστήματος στον υπολογιστή απαιτεί όσο το δυνατό περισσότερες μετρήσεις ώστε να είμαστε σε θέση να αποφανθούμε με μεγαλύτερη βεβαιότητα για τη συμπεριφορά του συστήματος. Ακόμα, είναι περισσότερο σημαντικό να διαθέτουμε πλήθος μετρήσεων στις περιοχές όπου το σύστημα εκδηλώνει μετάβαση φάσης έτσι ώστε να είμαστε ικανοί να προσδιορίσουμε με ακρίβεια από τις τιμές των μετρούμενων ποσοτήτων και τις κρίσιμες τιμές των παραμέτρων το είδος της μετάβασης φάσης που λαμβάνει χώρα. Ωστόσο, είναι φανερό πως το τίμημα για μεγαλύτερη ακρίβεια είναι πολύτιμος χρόνος λειτουργίας του υπολογιστή.

Η μέθοδος multihistogram, η οποία είναι γνωστή και σαν μέθοδος Ferrenberg - Swendsen, συμβάλλει στην εξοικονόμηση υπολογιστικού χρόνου καθώς δίνει την δυνατότητα προσέγγισης των κρίσιμων τιμών από ένα σύνολο σχετικά λίγων τιμών περί του σημείου μετάβασης φάσης. Καταρχήν, θα παρουσιάσουμε την μέθοδο για την περίπτωση του μοντέλου Ising με μαγνητικό πεδίο ακολουθώντας πιστά την αναφορά των Ferrenberg και Swendsen.

Η χαμιλτονιανή του συγκεκριμένου μοντέλου είναι

$$-\beta\mathcal{H} = K \sum_{\langle ij \rangle} \sigma_i \sigma_j + h \sum_i \sigma_i = KS + hM, \quad (\text{A'.1})$$

όπου K είναι η αδιάστατη σταθερά σύζευξης και h είναι το εφαρμοζόμενο μαγνητικό πεδίο. Η κατανομή πιθανότητας των S, M για τυχαίες τιμές των παραμέτρων K, h δίνεται από:

$$P_{(K,h)}(S, M) = \frac{1}{Z(K, h)} N(S, M) \exp(KS + hM) \quad (\text{A'.2})$$

όπου $N(S, M)$ είναι ο αριθμός των συλλογών στο σημείο (S, M) του χώρου των φάσεων και $Z(K, h)$ είναι η συνάρτηση επιμερισμού που δίνεται από:

$$Z(K, h) = \sum_{S, M} N(S, M) \exp(KS + hM). \quad (\text{A'.3})$$

Το ιστόγραμμα των τιμών (S, M) που δημιουργείται από την προσομοίωση Monte Carlo είναι ανάλογο της πιθανότητας $P_{K,h}(S, M)$ και μπορούμε να το χρησιμοποιήσουμε για να παράγουμε την κανονικοποιημένη κατανομή πιθανότητας που αντιστοιχεί σε ελαφρά διαφορετικές τιμές των παραμέτρων.

Έτσι παίρνουμε

$$P_{K',h'}(S, M) = \frac{P_{K,h}(S, M) \exp[(K' - K)S + (h' - h)M]}{\sum_{S, M} P_{K,h}(S, M) \exp[(K' - K)S + (h' - h)M]}. \quad (\text{A'.4})$$

Με υπολογισμένη την ανωτέρω κατανομή μπορεί κάποιος να προσδιορίσει τις τιμές των φυσικών μεγεθών που τον ενδιαφέρουν για νέες τιμές των παραμέτρων.

Παράρτημα Β'

Η μέθοδος jackknife

Η μέθοδος jackknife εισήχθη από τους Quenouille και Tukey στα τέλη της δεκαετίας του '50. Σκοπός τους ήταν να επιλύσουν το πρόβλημα της αξιοπιστίας των διαφορετικών εκτιμητήριων των στατιστικών δειγμάτων. Με άλλα λόγια, ο στόχος ήταν να επιχειρηθεί μια απάντηση στο ερώτημα πόσο αξιόπιστη και ακριβής μπορεί να είναι μια μέση τιμή ή μια πιθανότητα που αποδίδεται σε ένα στατιστικό δείγμα. Και επιπλέον να δοθεί μια εκτίμηση για το σχετικό σφάλμα.

Ας υποθέσουμε πως έχουμε ένα δείγμα τυχαίων μεταβλητών X_1, \dots, X_n ενός μεγέθους x , οι οποίες λαμβάνουν τις τιμές x_1, \dots, x_n και ανήκουν σε ένα πληθυσμό ο οποίος υπακούει σε μια κατανομή $f(x)$. Υποθέτουμε, επίσης, πως η αληθινή μέση τιμή είναι X . Καθεμία μεταβλητή X_i , $i = 1, \dots, n$ υπακούει στην ίδια κατανομή και έτσι η αναμενόμενη τιμή τους είναι:

$$E(X_i) \equiv X \quad (\text{B'.1})$$

ενώ η διασπορά ορίζεται ως:

$$\text{Var}(X_i) = E[(X_i - X)^2] \equiv \sigma^2. \quad (\text{B'.2})$$

Η ανεξαρτησία των τυχαίων X_i σημαίνει πως για $i \neq j$

$$E[(X_i - X)(X_j - X)] = 0. \quad (\text{B'.3})$$

Ας λάβουμε ως εκτιμητήρια της αληθινής τιμής X την:

$$\hat{X} = \frac{1}{n} \sum_{i=1}^n X_i, \quad (\text{B'.4})$$

τότε

$$E(\hat{X}) = E\left[\frac{1}{n} \sum_{i=1}^n X_i\right] = \frac{1}{n} \sum_{i=1}^n E(X_i) \equiv X \quad (\text{B'.5})$$

το οποίο σημαίνει πως η εκτιμητήρια είναι απροκατάληπτη (unbiased). Για τη διασπορά έχουμε:

$$\text{Var}(\hat{X}) \equiv E[(\hat{X} - X)^2] = \quad (\text{B'.6})$$

$$= \frac{1}{n^2} E\left[\sum_{i=1}^n (X_i - X)^2 + 2 \sum_{i \neq j} (X_i - X)(X_j - X)\right] = \quad (\text{B'.7})$$

$$= \frac{1}{n^2} \sum_{i=1}^n E[(X_i - X)^2] \equiv \frac{\sigma^2}{n}. \quad (\text{B'.8})$$

Ας δούμε τώρα τί γίνεται όταν μελετήσουμε στατιστικά μια ποσότητα της οποίας η αληθινή μέση τιμή είναι m . Υποθέτουμε επίσης πως αυτή εξαρτάται από το μέγεθος x έτσι ώστε να είναι $m(X)$. Οπότε είναι και $m_i(X_i)$ για κάθε X_i . Ας εργαστούμε συγκεκριμένα με την ποσότητα του τετραγωνικού σφάλματος της μέσης τιμής (mean square error). Έχουμε:

$$\text{MSE}(\hat{m}) \equiv \text{E}[(\hat{m} - m)^2] = \tag{B'.9}$$

$$= \text{E}[(\hat{m} - \text{E}(\hat{m}) + \text{E}(\hat{m}) - m)^2] = \tag{B'.10}$$

$$= \text{E}[(\hat{m} - \text{E}(\hat{m}))^2] + (\text{E}(\hat{m}) - m)^2 \equiv \tag{B'.11}$$

$$\equiv \text{Var}(\hat{m}) + \text{B}^2(\hat{m}), \tag{B'.12}$$

όπου η διασπορά ορίζεται ως

$$\text{Var}(\hat{m}) = \text{E}[(\hat{m} - \text{E}(\hat{m}))^2] = \text{E}(\hat{m}^2) - \text{E}^2(\hat{m}) \tag{B'.13}$$

και η προκατάληψη ως

$$\text{B}(\hat{m}) = \text{E}(\hat{m}) - m. \tag{B'.14}$$

Με την μέθοδο jackknife, λοιπόν, σκοπεύουμε να μειώσουμε την προκατάληψη $\text{B}(\hat{m})$ χωρίς να μεταβληθεί η διασπορά Var .

Σύμφωνα με την μέθοδο, σχηματίζουμε ένα νέο δείγμα από τις n τιμές που η καθεμία έχει προέλθει αφαιρώντας κάθε φορά μια από το αρχικό δείγμα και υπολογίζοντας την μέση τιμή των υπολοίπων. Βάσει του νέου δείγματος υπολογίζουμε τις αντίστοιχες ποσότητες m_i για $i = 1, \dots, n$ των οποίων η μέση τιμή δίνεται από:

$$\hat{m}_J = \frac{1}{n} \sum_{i=1}^n m_i. \tag{B'.15}$$

Μπορεί να αποδειχθεί, τότε, ότι η προκατάληψη δίνεται από την σχέση:

$$\hat{\text{B}} = (n - 1)(\hat{m}_J - m) \tag{B'.16}$$

και πως η νέα εκτιμήτρια για το τετραγωνικό σφάλμα της μέσης τιμής παρέχεται, πλέον, από:

$$\text{MSE}_J(\hat{m}) = \sqrt{\frac{n-1}{n} \sum_{i=1}^n (\hat{m}_i - m)^2} \tag{B'.17}$$

από την οποία λαμβάνουμε τις τιμές των σφαλμάτων μας.

Στην παρούσα εργασία εμπιστευτήκαμε την μέθοδο jackknife στον υπολογισμό του σφάλματος των μέσων τιμών της επιδεκτικότητας για τον απλό αλλά σημαντικό λόγο πως οι παραγόμενες από την προσομοίωση αναπαραστάσεις (συνεπώς και οι μετρήσεις) δεν είναι ανεξάρτητες μεταξύ τους λόγω του τρόπου δημιουργίας τους. Έτσι, σε κάθε περίπτωση μέτρησης υπολογίσαμε τον χρόνο αυτοσυσχέτισης τ_{int} . Έπειτα, κατασκευάσαμε ένα σύνολο μετρήσεων που η καθεμία να αντιστοιχεί σε αναπαραστάσεις που τις χώριζε "απόσταση", τουλάχιστον, ίση με τ_{int} και κατόπιν εφαρμόσαμε την μέθοδο jackknife για τον προσδιορισμό του σφάλματος.


```

loops[a][3] = index2int(i,j,k+1,l);
a++;
//i-1 loop
loops[a][0] = index2int(i,j,k,l);
loops[a][1] = index2int(i+1,j,k,l);
loops[a][2] = index2int(i+1,j,k,l+1);
loops[a][3] = index2int(i,j,k,l+1);
a++;
//j-k loop
loops[a][0] = index2int(i,j,k,l);
loops[a][1] = index2int(i,j+1,k,l);
loops[a][2] = index2int(i,j+1,k+1,l);
loops[a][3] = index2int(i,j,k+1,l);
a++;
//j-1 loop
loops[a][0] = index2int(i,j,k,l);
loops[a][1] = index2int(i,j+1,k,l);
loops[a][2] = index2int(i,j+1,k,l+1);
loops[a][3] = index2int(i,j,k,l+1);
a++;
//k-1 loop
loops[a][0] = index2int(i,j,k,l);
loops[a][1] = index2int(i,j,k+1,l);
loops[a][2] = index2int(i,j,k+1,l+1);
loops[a][3] = index2int(i,j,k,l+1);
a++;
    }
}
}
}

for(int i=0; i<N_LOOPS; i++)
    adjLoops[i][0] = 1;
for(int i=0; i<N_LOOPS; i++){
    for(int j=i+1; adjLoops[i][0]<N_ADJ+1; j++){
        int linkStatus = 0;
        int p1n = -1;
        int p2n = -1;
        int p1m = -1;
        int p2m = -1;
        int dpn;
        int dpm;
        int i1, i2;
        bool sortn,sortm;
        for(int n=0; n<4; n++){
            i1 = loops[i][n];
            for(int m=0; m<4; m++){
                i2 = loops[j][m];
                if(i1==i2){
                    if(p1n==-1){
                        p1n = n;
                        p1m = m;
                    }
                    else{
                        p2n = n;
                        p2m = m;
                    }
                }
            }
        }
        if(p1n==-1||p2n==-1) continue;
        else{
            dpn = p2n-p1n;
            dpm = p2m-p1m;
            sortn = (dpn==1||dpm==3);
            sortm = (dpm==1||dpm==3);
            if((sortm&&sortn)||(!sortn&&!sortm))
                linkStatus = 1;
            else linkStatus = -1;
            adjLoops[i][adjLoops[i][0]] = linkStatus*(j+1);
            adjLoops[i][0] = adjLoops[i][0]+1;
        }
    }
}

```

```
        adjLoops[j][adjLoops[j][0]] = linkStatus*(i+1);
        adjLoops[j][0] = adjLoops[j][0]+1;
    }
}

int loop;
for(int i=0; i<N_LOOPS; i++){
    for(int j=1; j<N_ADJ+1; j++){
        loop = adjLoops[i][j];
        outputFile.write((char *)&loop,sizeof(loop));
    }
}

outputFile.close();
}
```

Παράρτημα Δ'

Πηγαίος κώδικας του προγράμματος sweeper

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>

#define frand() ((double) rand()/(RAND_MAX+1.0))

const double Pi = 3.1415926535897932385;
const int LatticeLength = 6;
const int AllLoops = 7777; // 6*L4+1
const int AllNeighbors = 155540; // 20*AllLoops

inline int check(double r, double Acceptance){
    double x, xp, Ln, L, Tn;
    int n;
    x=1-r;
    xp=1;
    Ln=0;
    L=1.0/r;
    for(n=1;n<=4;n++){
        xp*=x;
        Tn=(double) xp/(double)n;
        if(Acceptance<=(Ln-L*Tn)){
            return 0;
        }
        Ln-=Tn;
        if(Ln<=Acceptance){
            return 1;
        }
    }
    if(log(r)<Acceptance)
        return 1;
    return 0;
}

int main(int argc, char* argv[0]){

    int Sweeps, index, index1, index2, sweep, hit,
    TransitionCounter, i, j, tempindex, MapIndex;
    double Beta, TransitionRate, DTheta, CosDTheta,
    SinDTheta, DThetaQuota, CosDThetaQuota, SinDThetaQuota,
    scale, DS, DS1, DS2, Acceptance, r, MeanCosPlaquette,
    Coefficient, Sign, OldLatticeCos, OldLatticeSin, temp1, temp2;
    char *filename;
    FILE *MapFile, *LatCos, *LatSin, *outMCP, *newLatCos, *newLatSin;
    int Map[AllNeighbors];
    double LatticeCos[AllLoops], LatticeSin[AllLoops];
```

```

printf("*****\n");
printf("* *\n");
printf("* S W E E P E R *\n");
printf("* *\n");
printf("* version 1.7 16/12/2005 *\n");
printf("* *\n");
printf("*-----*\n");
printf("* Runs the Monte Carlo algorithm on *\n");
printf("* the lattice *\n");
printf("*****\n\n");

Sweeps=atoi(argv[1]);
Beta=atof(argv[2]);

printf("> Reading previous snapshot...");

filename=argv[3];
MapFile=fopen(filename,"rb");

if(MapFile==NULL){
    printf(" ERROR!\n Unable to open map file.  Exiting....\n");
    exit(1);
}

LatCos=fopen("LatCos.qrk","rb");

if(LatCos==NULL){
    printf(" ERROR!\n Unable to open LatCos.qrk.  Exiting....\n");
    exit(1);
}

LatSin=fopen("LatSin.qrk","rb");

if(LatSin==NULL){
    printf(" ERROR!\n Unable to open LatSin.qrk.  Exiting....\n");
    exit(1);
}

printf(" DONE\n\n");

printf("> Creating the map...");

for(index=20;index<AllNeighbors;index++){
    fread(&Map[index],sizeof(int),1,MapFile);
}

printf(" DONE\n\n");

printf("> Creating the lattice...");

for(index=1;index<AllLoops;index++){
    fread(&LatticeCos[index],sizeof(double),1,LatCos);
}

for(index=1;index<AllLoops;index++){
    fread(&LatticeSin[index],sizeof(double),1,LatSin);
}

fclose(MapFile);
fclose(LatCos);
fclose(LatSin);

printf(" DONE\n\n");

printf("Parameters:\n");
printf("+ Lattice side:  %d\n",LatticeLength);
printf("+ Beta value:   %lg\n",Beta);
printf("+ Thermalization sweeps:  %d\n",Sweeps);

printf("> Running Metropolis...\n");

```

```

outMCP=fopen("MeanCosPlaquette.dat","wb");

scale=0.5;

for(sweep=0; sweep<Sweeps; sweep++){
  Coefficient=scale*Pi;
  MeanCosPlaquette=0;
  TransitionCounter=0;

  for(i=1;i<AllLoops;i++){
    tempindex=20*i;
    for(hit=0;hit<5;hit++){
      DTheta=(frand()-0.5)*Coefficient;
      CosDTheta=cos(DTheta);
      SinDTheta=sin(DTheta);
      DThetaQuota=DTheta/4.0;
      CosDThetaQuota=cos(DThetaQuota);
      SinDThetaQuota=sin(DThetaQuota);

      DS=LatticeCos[i]*(CosDTheta-1.0)-LatticeSin[i]*SinDTheta;
      DS1=0;
      DS2=0;
      for(j=0;j<20;j++){
        index1=j+tempindex;
        MapIndex=abs(Map[index1]);
        if(Map[index1]>=0)
          Sign=1;
        else
          Sign=-1;
        DS1+=LatticeCos[MapIndex];
        DS2+=LatticeSin[MapIndex]*Sign;
      }
      DS+=(DS1*(CosDThetaQuota-1.0)-DS2*SinDThetaQuota);

      if(DS>=0)
        Acceptance = 1;
      else
        Acceptance = Beta * DS;
      r = frand();
      if(check(r,Acceptance)==1){
        OldLatticeCos=LatticeCos[i];
        OldLatticeSin=LatticeSin[i];
        LatticeCos[i]=OldLatticeCos*cosDTheta-OldLatticeSin*sinDTheta;
        LatticeSin[i]=OldLatticeSin*cosDTheta+OldLatticeCos*sinDTheta;
        for(j=0;j<20;j++){
          index2=j+tempindex;
          if(Map[index2]>=0)
            Sign=1;
          else
            Sign=-1;
          temp1=LatticeCos[abs(Map[index2])];
          temp2=LatticeSin[abs(Map[index2])];
          LatticeCos[abs(Map[index2])]=temp1*cosDThetaQuota-temp2*Sign*sinDThetaQuota;
          LatticeSin[abs(Map[index2])]=temp2*cosDThetaQuota+temp1*Sign*sinDThetaQuota;
        }
        TransitionCounter++;
      }
    }
  }
}

TransitionRate=(double)TransitionCounter/(double)AllLoops/5.0;

if(scale>0.9)
  scale=0.9;
else
  if(TransitionRate<0.5)
    scale*=0.95;
  else
    if(TransitionRate>0.7)
      scale*=1.05;

```



```

if(sweep%8==0){
    for(i=0;i<AllLoops;i++)
        MeanCosPlaquette+=LatticeCos[i];

    MeanCosPlaquette/=AllLoops;

    fwrite(&MeanCosPlaquette,sizeof(double),1,outMCP);
}

if(sweep%1024==0)
    printf("Reached sweep:  %d\n",sweep);
}

fclose(outMCP);

printf("> Running Metropolis DONE\n\n");

printf("> Saving new snapshot...");

newLatCos=fopen("LatCos.qrk","wb");
newLatSin=fopen("LatSin.qrk","wb");

for(i=1;i<AllLoops;i++){
    fwrite(&LatticeCos[i],sizeof(double),1,newLatCos);
}
for(i=1;i<AllLoops;i++){
    fwrite(&LatticeSin[i],sizeof(double),1,newLatSin);
}

fclose(newLatCos);
fclose(newLatSin);

printf(" DONE\n\n");
return 0;
}

```

Παράρτημα Ε΄

Πηγαίος κώδικας του προγράμματος cumulants

```
#include <iostream>
#include <cstdlib>
#include <cstdio>
#include <fstream>
#include <cmath>
#include <string>

using namespace std;

int main(int argc, char* argv[0]){

    cout << "*****\n";
    cout << "* C U M U L A N T S 4 D *\n";
    cout << "*                               *\n";
    cout << "* version 0.1 11/13/2005 *\n";
    cout << "*-----*\n";
    cout << "* Calculates various cumulants of *\n";
    cout << "* the given data file *\n";
    cout << "*****\n\n";

    ifstream inMCP;
    inMCP.open(argv[1]);

    string fileName(argv[1]);

    if(fileName.compare("-h")==0){
        cout << " cumulants4d calculates various cumulants of a
                given data file produced by the 4D sweeper module.\n";
        cout << " Usage:  cumulants4d someFileName.dat\n\n";
        exit(1);
    }

    double MeanCosPlaquette;
    double MeanCos;
    double MeanSquareCos;
    double MeanCosSquare;
    double Mean4Cos;
    double MeanCos4;
    double Susceptibility;
    double Beta;
    char ParameterName[20];
    int counter=0;
    double LatticeLength;
    double Volume;
    double C;
```

```

double U2;
double U4;

Beta=atof(argv[2]);
LatticeLength=atof(argv[3]);

Volume=pow(LatticeLength,4);

while(!inMCP.eof()){
    inMCP >> MeanCosPlaquette;
    MeanCos+=MeanCosPlaquette;
    MeanSquareCos+=pow(MeanCosPlaquette,2);
    Mean4Cos+=pow(MeanCosPlaquette,4);
    counter++;
}

MeanCos/=(double)counter;
MeanCosSquare=pow(MeanCos,2);
MeanCos4=pow(MeanCosSquare,2);

MeanSquareCos/=(double)counter;
Mean4Cos/=(double)counter;

C=(MeanSquareCos-MeanCosSquare)*6*Volume*Beta*Beta;
Susceptibility=(MeanSquareCos-MeanCosSquare)*Volume;
U2=1-(MeanSquareCos/MeanCosSquare);
U4=(1-(Mean4Cos/MeanCos4))/3;

fstream outSc;
outSc.open("Cumulants4D.tsv",ios_base::out | ios_base::app);
outSc.width(16);
outSc.precision(16);

outSc << "Beta: " << Beta << endl;
outSc << "Susceptibility: " << Susceptibility << endl;
outSc << "C cumulant: " << C << endl;
outSc << "U2 cumulant: " << U2 << endl;
outSc << "U4 cumulant: " << U4 << endl;

cout << "> Cumulants calculated: " << endl;
cout << "\tSusceptibility: " << Susceptibility << endl;
cout << "\tC: " << C << endl;
cout << "\tU2: " << U2 << endl;
cout << "\tU4: " << U4 << endl;
cout << "> Result saved in file: Cumulants4D.tsv" << endl;
cout << endl;

inMCP.close();
outSc.close();

return 0;
}

```

Βιβλιογραφία

- [1] Abers, E.S., Lee, B.W., **Gauge Theories**, Phys.Rev. C9, No.1, (1973)
 - [2] Arnold, G., Bunk, B., Lippert, Th., Schilling, K., **Compact QED under scrutiny: it's first order**, arXiv:hep-lat/0210010, 7 Oct 2002
 - [3] Arnold, G., Lippert, Th., Schilling, K., **Finite size scaling analysis of compact QED**, arXiv:hep-lat/0011058, 13 Nov 2000
 - [4] Batrouni, G.G., Halpern, M.B., **String, corner, and plaquette formulation of finite lattice gauge theory**, Phys.Rev. D30, No.8, (1984)
 - [5] Caldi, D.G., **How well do Monte Carlo simulations distinguish between U(1) and SU(2)?, A study of the string tension in U(1) lattice gauge theory**, Nuclear Physics B220[FS8], (1983)
 - [6] Creutz, M., **Quarks, gluons and lattices**, Cambridge Monographs on Mathematical Physics, Cambridge University Press, (1983)
 - [7] Creutz, M., **Overrelaxation and Monte Carlo simulation**, Phys.Rev. D36, No.2, (1987)
 - [8] Creutz, M., **Lattice gauge theory - A retrospective**, arXiv:hep-lat/0010047, 26 Oct 2000
 - [9] Creutz, M., Jacobs, L., Rebbi, C., **Monte Carlo study of Abelian lattice gauge theories**, Phys.Rev. D20, (1979)
 - [10] Dimopoulos, P., Farakos, K., Kehagias, A., Koutsoumbas, G., **Lattice Evidence for Gauge Field Localization on a Brane**, arXiv:hep-th/0007079, 10 Jul 2000
 - [11] Δημόπουλος Πέτρος, **Μελέτη του Αβελιανού Πρότυπου Higgs στο Χωροχρονικό Πλέγμα σε Πεπερασμένη Θερμοκρασία**, Διδακτορική Διατριβή
 - [12] Ferrante, D.D., **Lattice Quantum Field Theory - A constructive approach**, (2001)
 - [13] Ferrenberg, A.M., Swendsen, R.H., Phys.Rev.Lett. 61 (1988) 2635
 - [14] Ferrenberg, A.M., Swendsen, R.H., Phys.Rev.Lett. 63 (1989) 1195
 - [15] Foster, I., Kesselman, C., Tuecke, S., **The Anatomy of the Grid - Enabling Scalable Virtual Organizations**, Intl J. Supercomputer Applications, (2001)
-

- [16] Huang, K., **Quarks, Leptons & Gauge Fields**, World Scientific, (1992)
- [17] Huang, S., Moriarty, K.J., Myers, E., Potvin, J., Phys., C50, 221 (1991)
- [18] Jersák, J., **Study of compact abelian lattice gauge theories**, arXiv:hep-lat/0010014, 9 Oct 2000
- [19] Kiskis, J. **Bianchi Identity for non-Abelian lattice gauge fields**, Phys. Rev. D26, (1982)
- [20] Kleinert, H., **Path Integrals in Quantum Mechanics, Statistics, Polymer Physics, and Financial Markets**, World Scientific, (2004)
- [21] Kogut, J.B., **An introduction to gauge theory and spin systems**, Reiview of Modern Physics, Vol.51, No.4, (1979)
- [22] Loan, M., Brunner, M., Sloggett, C., Hammer, C., **Path Integral Monte Carlo approach to the U(1) lattice gauge theory in 2+1 dimensions**, Phys.Rev.Let. D68, 034503(2003)
- [23] Moriarty, K.J.M., **Monte Carlo study of compact U(1) four-dimensional lattice gauge theory**, Phys.Rev. D25, (1982)
- [24] Mitrjushkin, V., **Jackknife Method**, (unpublished preprint)
- [25] Newman, M.E.J., Barkema, G.T. **Monte Carlo Methods in Statistical Physics**, Oxford University Press, (1999)
- [26] Reif, F., **Fundamentals of Statistical and Thermal Physics**, McGraw-Hill International Editions, Physics Series, (1985)
- [27] Rothe, H.J., **Lattice Gauge Theories, An Introduction**, World Scientific, (1992)
- [28] Smit, J., **Introduction to Quantum Fields on a Lattice**, Cambridge Lecture Notes in Physics 15, Cambridge University Press, (2002)
- [29] Ukawa, A., UTHEP-302, (1993)
- [30] Zee, A., **Quantum Field Theory in a Nutshell**, Princeton University Press, (2003)