

2011

Προσομοίωση Κβαντικής Μεταφοράς σε Ημιαγώγιμες Διατάξεις Κλίμακας Νανομέτρου

Πανεπιστήμιο Θεσσαλίας- Τμήμα Μηχανικών
Η/Υ, Τηλεπικοινωνιών & Δικτύων

Διπλωματική εργασία

Επιβλέποντες καθηγητές:

Ευμορφόπουλος Νέστορας
Σταμούλης Γεώργιος

Λέκτορας
Καθηγητής

Σκορδίλης Ερωτόκριτος

27/5/2011



Προσομοίωση κβαντικής μεταφοράς σε ημιαγώγιμες διατάξεις κλίμακας νανομέτρου

Επιβλέποντες καθηγητές:

Ευμορφόπουλος Νέστορας

Σταμούλης Γεώργιος

Λέκτορας

Καθηγητής

Ευχαριστίες

Αρχικά θα ήθελα να ευχαριστήσω τον υπεύθυνο καθηγητή μου, κύριο Ευμορφόπουλο Νέστορα, ο οποίος αποτέλεσε την πηγή έμπνευσης για το εν λόγω θέμα, αλλά και έναν σημαντικό οδηγό για την συγγραφή της παρούσας εργασίας και η οποία χωρίς την πολύτιμη συμβολή του δεν θα ήταν δυνατόν να γίνει. Ο ίδιος με τις πολύτιμες, έγκυρες και έγκαιρες συμβουλές του βοήθησε ώστε η συγγραφή αυτής της εργασίας να τελειώσει σε εύλογο χρονικό διάστημα. Επίσης θα ήθελα να ευχαριστήσω και τον 2^ο υπεύθυνο καθηγητή μου, κύριο Σταμούλη Γεώργιο.

Ένα μεγάλο ευχαριστώ πρέπει επίσης να πω και στους γονείς μου, που τα τελευταία 7 χρόνια μου στάθηκαν αρωγοί σε όλη την φοιτητική μου πορεία και με την υπομονή και την οικονομική τους στήριξη με βοήθησαν σημαντικά στην πορεία μου μέχρι τώρα.

Τέλος ένα μεγάλο ευχαριστώ σε όλους τους φίλους που απέκτησα όλα αυτά τα χρόνια, χωρίς την παρουσία των οποίων η ζωή μου θα ήταν κενή και άνευ περιεχομένου.

*Ειδική μνεία θα ήθελα να κάνω στην πραγματική φίλη και συνάδελφο Μαρίνα Μ. Ιωαννίτη χάρις την οποία μου άνοιξαν νέοι δρόμοι γνώσης και γι' αυτό την ευχαριστώ πάρα πολύ.

Στην Γωγώ...

Περιεχόμενα

Ευχαριστίες.....	3
1. Εισαγωγή.....	6
1.1 Κλιμάκωση συσκευών CMOS.....	6
1.2 Κατηγοριοποίηση των μοντέλων μεταφοράς.....	8
1.3 Σχεδιασμός νανοσυσκευών-μια θεωρητική προσέγγιση.....	10
1.4 Περιγραφή του φορμαλισμού NEGF.....	12
2. Περιγραφή του προβλήματος.....	15
2.1 Φορμαλισμός.....	15
2.2 Διακριτοποιημένες εξισώσεις Green: G^r και $G^<$	19
2.3 Self energies (Σ^r , $\Sigma^<$, Σ).....	23
2.4 Χρήση των διαγώνιων στοιχείων του A'	26
2.4.1 Επίδραση στην περιοχή απογύμνωσης πολυπυριτίου...27	
2.4.2 I_d vs V_d	28
2.4.3 Ισοτροπικές vs ανισοτροπικές ενεργές μάζες.....	30
3. Περιγραφή των αλγόριθμων.....	32
3.1 Περιγραφή του 1 ^{ου} αλγόριθμου.....	32
3.2 Περιγραφή του 2 ^{ου} αλγόριθμου.....	35
4. Κώδικας MATLAB.....	39
4.1 Μεταβλητές και σταθερές.....	39
4.2 Υπολογισμοί πινάκων και τελικά αποτελέσματα.....	42
5. Αποτελέσματα προσομοιώσεων.....	65
Επίλογος.....	87
Αναφορές.....	88

1. Εισαγωγή

1.1 Κλιμάκωση συσκευών CMOS

Τα τελευταία χρόνια η συνεχιζόμενη μείωση του μεγέθους των τρανζίστορ που χρησιμοποιούνται από τις μεγάλες βιομηχανίες για τον σχεδιασμό ολοκληρωμένων κυκλωμάτων, επεξεργαστών και καρτών γραφικών οδήγησε τους σχεδιαστές αυτών των συστημάτων σε μεγάλες προκλήσεις όσον αφορά τον ακριβή σχεδιασμό και την αποτελεσματική προσομοίωση αυτών των συστημάτων. Εξαιτίας της συνεχιζόμενης σμίκρυνσης του μεγέθους των τρανζίστορ, οι μέχρι τώρα τεχνικές σχεδίασης δεν είναι πλέον αποδοτικές. Αυτό οδήγησε στην ανάπτυξη μοντέλων με μεγαλύτερη ακρίβεια, τα οποία βασίζονται στις θεμελιώδεις αρχές της κβαντικής μηχανικής. Τα μοντέλα τα οποία συμπεριλαμβάνουν αυτά τα κβαντομηχανικά φαινόμενα τείνουν να εξελίσσονται σε λύσεις πολύ μεγάλων συστημάτων από γραμμικές εξισώσεις. Αυτό οδηγεί σε υπερβολικές απαιτήσεις μνήμης και η προσομοίωση με αυτά τα μοντέλα οδηγεί σε απαγορευτικό μέγεθος υπολογισμών που πρέπει να γίνουν. (Stephen Cauley, 2007)

Μια ομάδα εταιριών πρωτοπόρων στην σχεδίαση MOSFET εξέδωσαν τις προβλέψεις τους για την επόμενη δεκαετία και όπως χαρακτηριστικά μπορούμε να δούμε στον παρακάτω πίνακα (Table 1) το μήκος της πύλης του τρανζίστορ αναμένεται να φτάσει στα 7 νανόμετρα το έτος 2018.

Year of production	2012	2013	2014	2015	2016	2017	2018
MPU half pitch (nm)	35	32	28	25	22	20	18
Physical gate length (nm)	14	13	11	10	9	8	7

Table 1: πρόβλεψη της International Technology Roadmap for Semiconductors (ITRS) για το χαρακτηριστικό μέγεθος και το φυσικό μήκος πύλης ανά έτος

Αρχικά η δυνατότητα 3-διάστατης σχεδίασης είναι πολύ χρήσιμη, εφόσον οι συμβατικές σχεδιάσεις μίας πύλης (single-gate) σταδιακά αντικαθίστανται με διάφορα είδη σχεδιάσεων με πολλαπλές πύλες (multiple gates) ούτως ώστε να περιορίσουν διάφορα φαινόμενα βραχέως καναλιού (drain-induced barrier lowering, threshold voltage roll-off). Επιπλέον όταν οι χαρακτηριστικές διαστάσεις των τρανζίστορ MOSFET αρχίζουν να είναι συγκρίσιμες με το μέγεθος κυματικών πακέτων ηλεκτρονίων, τότε πρέπει να συμπεριλάβουμε και τα διάφορα κβαντομηχανικά φαινόμενα στους υπολογισμούς μας για την μεταφορά διαμέσου του καναλιού του τρανζίστορ. Από την στιγμή που τα μοντέλα μεταφοράς που χρησιμοποιούμε σήμερα προέρχονται από την εξίσωση μεταφοράς Boltzmann, πρέπει να τα αναβαθμίσουμε ώστε να περιλαμβάνουν και κάποια σημαντικά χαρακτηριστικά των κβαντομηχανικών φαινομένων, ή να τα αντικαταστήσουμε με πιο ακριβή μοντέλα κβαντικής μεταφοράς. Επίσης όσο οι διαστάσεις των MOSFET μικραίνουν, ένα σημαντικό ποσοστό ηλεκτρονίων που εγχέονται από την πηγή προς το κανάλι δεν διασκορπίζονται πριν βγουν από το αυτό. Επομένως η μεταφορά είναι ημι-βαλλιστική και η συνάρτηση διανομής

μεταφοράς (carrier distribution function) μέσα στο κανάλι απέχει πολύ από συνθήκες ισορροπίας. Έτσι καταλήγουμε στο γεγονός ότι η πληροφορία που περιλαμβάνεται στις μακροσκοπικές μεταβλητές δεν επαρκεί για να αναπαράγουμε την μη-ισορροπημένη συνάρτηση διανομής που παρατηρείται σε ημι-βαλλιστική μεταφορά και είναι πολύ δύσκολο να αντλήσουμε ικανοποιητικά αποτελέσματα από τα μοντέλα μακροσκοπικής μεταφοράς. (Jin, 2006)

1.2 Κατηγοριοποίηση των μοντέλων μεταφοράς

Στην εικόνα που ακολουθεί (Figure 1) παραθέτουμε τα ήδη υπάρχοντα μοντέλα μεταφοράς ανάλογα με την εφαρμοσιμότητά τους και το αν αποκομίζονται από κβαντομηχανικό ή κλασσικό φορμαλισμό. Τυπικά, όταν το μήκος του καναλιού είναι μεγαλύτερο από 0.1 μm μπορούμε να εφαρμόσουμε μακροσκοπικά μοντέλα μεταφοράς που περιλαμβάνουν μακροσκοπικές μεταβλητές όπως η μέση πυκνότητα ηλεκτρονίων, ταχύτητα, ενέργεια και θερμοκρασία ηλεκτρονίων. Από τα ημι-κλασσικά μοντέλα μεταφοράς, τα μοντέλα drift-diffusion και υδροδυναμικής μπορούν να συμπεριληφθούν σε αυτή την κατηγορία, η οποία αποκομίζεται από την εξίσωση μεταφοράς του Boltzmann. Από την άλλη, κβαντομηχανικά μακροσκοπικά μοντέλα μπορούν να αποκομιστούν από την εξίσωση διανομής Wigner, που δίνει μοντέλα κβαντικής διόρθωσης όπως το μοντέλο density-gradient, η προσέγγιση effective potential και το μοντέλο κβαντικής υδροδυναμικής. Εάν το κανάλι είναι μικρότερο από 0.1 μm οι υποθέσεις των μακροσκοπικών μοντέλων δεν έχουν πλέον εφαρμογή, επομένως πρέπει να στηριχθούμε σε μοντέλα μικροσκοπικής μεταφοράς. Η εξίσωση μεταφοράς Boltzmann λύνεται με την μέθοδο Monte Carlo για να αποκομίσουμε την εξίσωση διανομής $f(\mathbf{r}, \mathbf{k}, t)$. Με κβαντομηχανική προσέγγιση μπορούμε να δούμε την πληροφορία που παρουσιάζεται από τον τελεστή πυκνότητας $\rho(\mathbf{r}_1, t_1; \mathbf{r}_2, t_2)$. Μπορούμε

επίσης να αποκομίσουμε την απαραίτητη πληροφορία από την συνάρτηση μη-ισορροπίας του Green (non-equilibrium Green's function or NEGF) ή την εξίσωση διανομής Wigner. Επίσης αν οι εξισώσεις του τελεστή πυκνότητας είναι διαγώνιοι σε συνδυασμό με την μη-αλληλεπιδραστική Χαμιλτονιανή, μπορούμε να χρησιμοποιήσουμε την κύρια εξίσωση Pauli που δίνει την πληροφορία για τα διαγώνια στοιχεία του τελεστή πυκνότητας. Εάν το μήκος καναλιού τώρα μειωθεί περαιτέρω σε μεγέθη κάτω των 10 nm, η επιρροή των μικροσκοπικών μηχανισμών διάχυσης γίνονται λιγότερο σημαντικοί. Σε αυτό το όριο μπορούμε να χρησιμοποιήσουμε μοντέλα βαλλιστικής μεταφοράς. Ημι-κλασικά μπορούμε να αποκομίσουμε αναλυτική λύση από την εξίσωση βαλλιστικής μεταφοράς Boltzmann (επίσης γνωστή ως εξίσωση Vlasov), ή μπορούμε να υπολογίσουμε στιγμιότυπα από την εξίσωση βαλλιστικής μεταφοράς Boltzmann. Για την κβαντομηχανική περιγραφή μπορούμε να εφαρμόσουμε τον φορμαλισμό NEGF ή απευθείας να λύσουμε την εξίσωση του Schroedinger με ανοιχτές οριακές συνθήκες για να αποκομίσουμε τις καταστάσεις μεταφοράς του συμφασικού ρεύματος. (Jin, 2006)

	$L > 0.1 \mu\text{m}$ Macroscopic variable	Intermediate Microscopic variable	$L < 10 \text{ nm}$ Microscopic variable
Semi-classical description	obtain n, v, E, T , etc. -drift-diffusion -hydrodynamic -six moments equation	obtain distribution $f(\mathbf{r}, \mathbf{k}, t)$ from BTE -MC simulation -direct calculation	obtain distribution $f(\mathbf{r}, \mathbf{k}, t)$ from ballistic BTE -analytic solution -moment equation
Quantum mechanical description	obtain n, v, E, T, V_g , etc. -density-gradient -effective potential	obtain mixed state $\rho(\mathbf{r}_1, t_1; \mathbf{r}_2, t_2)$ -NEGF -Wigner function -Pauli master equation	obtain pure state $\psi(\mathbf{r}, t)$ -NEGF (no scattering) -Schrödinger equation

Table 2: Κατηγοριοποίηση ήδη υπαρχόντων μοντέλων μεταφοράς ανάλογα με τις διαστάσεις των τρανζίστορ

1.3 Σχεδιασμός νανοσυσκευών-μια θεωρητική προσέγγιση

Τα τρανζίστορ MOS με μήκος καναλιού τουλάχιστον 10nm πλέον μελετώνται εντατικά, τόσο θεωρητικά όσο και πειραματικά. Την ίδια ώρα πρόσφατες επιδείξεις μοριακής μετατροπής (molecular switching) κάνει τις μοριακές ηλεκτρονικές συσκευές να πλησιάζουν πιο κοντά στην πραγματικότητα. Είναι πλέον ξεκάθαρο ότι εργαλεία ποσοτικής προσομοίωσης για αυτή την νέα γενιά συσκευών θα χρειαστεί ατομικού επιπέδου κβαντομηχανικά μοντέλα. Ο φορμαλισμός της συνάρτησης μη-ισορροπίας Green (η οποία καμιά φορά αναφέρεται σαν φορμαλισμός των Keldysh ή Kadanoff-Baym όπως αναφέρουμε παρακάτω) μας προμηθεύει με μια εξαιρετική βάση για την ανάπτυξη αυτού του νέου είδους προσομοιωτών. Μονοδιάστατες κβαντικές συσκευές όπως δίοδοι σήραγγας ή ηχηρές σήραγγας έχουν μοντελοποιηθεί ποσοτικά χρησιμοποιώντας την μέθοδο NEMO (R.Lake, 1997), η οποία βασίζεται στον φορμαλισμό NEGF. Παρόλο που τα θέματα μεταφοράς σε MOS τρανζίστορ ή μοριακά ηλεκτρονικά συστήματα είναι τελείως διαφορετικά, ο φορμαλισμός NEGF διαθέτει ένα κατάλληλο σχέδιο και για την δική τους ανάλυση. Παρ' όλα αυτά, αυτός ο φορμαλισμός βασίζεται σε ιδέες που είναι ξένες προς τους περισσότερους φυσικούς συσκευών και έτσι παραμένει σχετικά δυσνόητος παρόλη την προφανή αξία μιας βασικής προσέγγισης στην οποία μπορούμε να βασίσουμε πρακτικά εργαλεία προσομοίωσης για νανο-συσκευές.

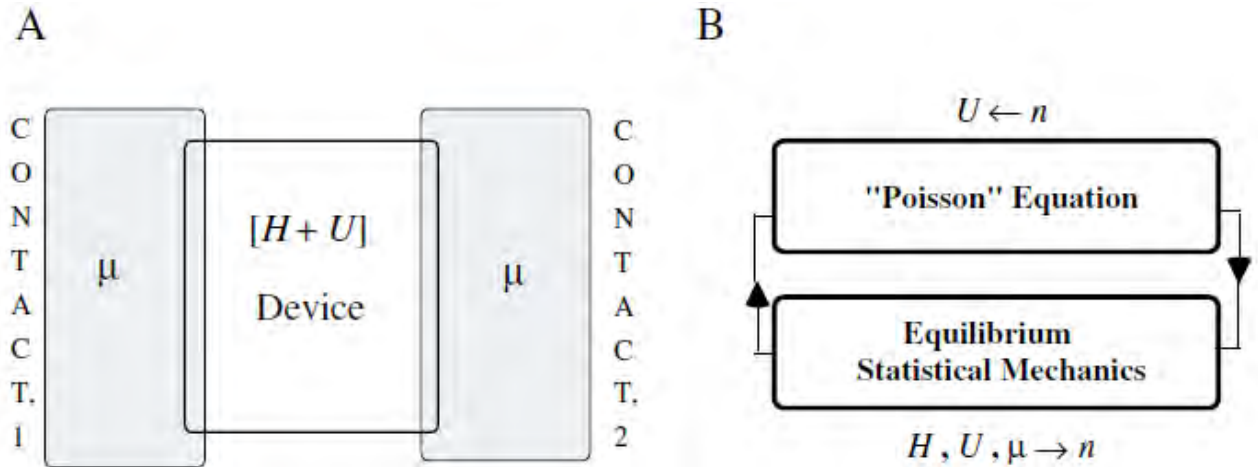


Figure 1: A, Συσσκευή σε ισορροπία. B, αυτό-σταθερή διαδικασία για την ανάλυση ηλεκτρονικών συσκευών σε ισορροπία. Το 'Poisson' γράφεται σε εισαγωγικά για να υπενθυμίζουμε ότι ίσως χρειαστεί να συμπληρωθεί με ένα δυναμικό ανταλλαγής-συσχετισμού

Οι περισσότεροι φυσικοί συσκευών είναι γνώστες της λύσης Schrödinger-Poisson. Ας ξεκινήσουμε λοιπόν με τον τρόπο που λειτουργεί η λύση αυτή για μια συσκευή σε ισορροπία. Το 1^ο βήμα είναι η εύρεση κατάλληλης Χαμιλτονιανής, H , που να παρέχει μια ικανή περιγραφή της απομονωμένης συσκευής. Όταν η συσκευή είναι συνδεδεμένη με τις επαφές, υπάρχει μεταφορά φορτίου μέσα ή έξω από την συσκευή, που δημιουργεί μια διαφορά δυναμικού, $U(r)$, η οποία πρέπει να υπολογιστεί με αυτό-σταθερότητα. Η λύση αυτή δρα μεταξύ της εξίσωσης Poisson που μας δίνει το δυναμικό αυτό για μια δοσμένη πυκνότητα ηλεκτρονίου $n(r)$ που είναι σχετική με αυτή που χρειάζεται για τοπική ουδετερότητα φορτίου

$$\nabla \cdot (\epsilon \nabla U) = q^2 [N_D - n]$$

και του νόμου της ισορροπημένης στατιστικής μηχανικής που μας λέει ότι η πυκνότητα ηλεκτρονίου για δοσμένο δυναμικό δίνεται από

$$n(r) = \sum_{\alpha} |\Psi_{\alpha}(r)|^2 f_0(\epsilon_{\alpha} - \mu)$$

γемίζοντας εντελώς τις ιδιοκαταστάσεις $\Psi_{\alpha}(r)$ της εξίσωσης Schrödinger

$$[H + U]\Psi_{\alpha}(r) = \epsilon_{\alpha}\Psi_{\alpha}(r)$$

σύμφωνα με την εξίσωση του Fermi

$$f_0(E - \mu) \equiv (1 + \exp[(E - \mu)/k_B T])^{-1}$$

(μ είναι το επίπεδο Fermi).

Αυτή είναι η βασική προσέγγιση που χρησιμοποιείται ευρύτατα για τον σχεδιασμό πυκνωτών MOS.

Γενικά η αυτό-σταθερή προσέγγιση που περιγράφουμε (NEGF) μας προμηθεύει με ένα ικανοποιητικό μοντέλο ισορροπίας για νανοσυσκευές, εκτός κι αν αυτές βρίσκονται υπό καθεστώς «αποκλεισμού Coulomb», η περιγραφή του οποίου θα ξέφευγε από το αντικείμενο της παρούσας εργασίας οπότε δεν θα επεκταθούμε σε αυτό.

1.4 Περιγραφή του φορμαλισμού NEGF

Η προσέγγιση NEGF που χρησιμοποιούμε σε αυτή την εργασία, θεωρείται το καλύτερο εργαλείο που υπάρχει αυτή την στιγμή για να προβλέψουμε την απόδοση και για τον σχεδιασμό των συσκευών σε νανοκλίμακα. Η ανάπτυξη πολυδιαστατικών εξομοιωτών που βασίζονται στην NEGF είναι κρίσιμη για την απόδοση τόσο των κβαντομηχανικών φαινομένων όσο και για τα αποτελέσματα διάχυσης μετά από συγκρούσεις ηλεκτρονίων με πλεγματικές θέσεις (phonons) και άλλα ηλεκτρόνια. Παρά το γεγονός ότι η μεταφορά έχει πολλές διαφορές μεταξύ νανο-τρανζίστορ, νανοκαλωδίων και μοριακών ηλεκτρονικών συσκευών, μπορεί να υπολογιστεί με τον κοινό φορμαλισμό που μας παρέχει η NEGF. Αυτό προκύπτει από την συζευγμένη λύση των εξισώσεων του Schroedinger και του Poisson. (S.Li, 2008)

Το δικό μας μοντέλο κβαντικής μεταφοράς βασίζεται στον φορμαλισμό NEGF, ο οποίος αναπτύχθηκε πρώτα την δεκαετία του 1960 από 2 Σοβιετικούς επιστήμονες, τους Keldysh και Kadanoff, ανεξάρτητα ο ένας

από τον άλλο. Παρόλο που οι διάφορες μέθοδοι επίλυσης των εξισώσεων Green μπορούν να επεκταθούν αποτελεσματικά σε 3 διαστάσεις, όπως τονίσαμε παραπάνω, οι μέθοδοι επίλυσης που παρουσιάζουμε εδώ θα επικεντρωθούν στην ακριβή και αποτελεσματική υλοποίηση της μεθόδου NEGF για MOSFET 2 διαστάσεων.

Ο πιο ακριβός σε χρόνο υπολογισμός που αντιμετωπίζουμε στον φορμαλισμό NEGF είναι η εύρεση μερικών και συγκεκριμένα των διαγώνιων στοιχείων του πίνακα G^r ο οποίος δίνεται από τον τύπο

$$G^r(E) = [EI - H - \Sigma]^{-1} = A^{-1} \text{ (retarded Green's function)}$$

καθώς και των αντίστοιχων στοιχείων του πίνακα $G^<$ που δίνεται από τον τύπο $G^<(E) = G^r \Sigma^< (G^r)'$ (less-than Green's function). Σε αυτές τις εξισώσεις, ο I είναι ο μοναδιαίος πίνακας και E το ενεργειακό επίπεδο, ενώ το «'» δηλώνει τον συζυγή ενός πίνακα. Ο πίνακας της Χαμιλτονιανής H περιγράφει το σύστημα το οποίο αναλύουμε (πχ νανο-τρανζίστορ). Συνήθως είναι αραιός (sparse) πίνακας στον οποίο η συνδεσιμότητα υπάρχει μόνο μεταξύ γειτονικών κόμβων στο πλέγμα, εκτός από τους κόμβους στα σύνορα της συσκευής, οι οποίοι μπορεί να έχουν μη-τοπική σύζευξη. Οι πίνακες Σ και $\Sigma^<$ αντιστοιχούν στις *self-energies* και, όπως θα δούμε και στην συνέχεια, είναι διαγώνιοι μπλοκ πίνακες.

Επειδή η αναλυτική λύση των εξισώσεων Green είναι ιδιαίτερα περίπλοκη, ειδικά από την στιγμή που δεν ξέρουμε αν όντως υπάρχει αναλυτική αριθμητική λύση, η μέθοδος των Klimeck και Svizhenko (A. Svizhenko, 2002) αποτελεί την βάση για την διακριτοποίηση των εξισώσεων Green σε μορφή πίνακα, ούτως ώστε να μην χρειαστεί η πλήρης λύση του συστήματος των εξισώσεων Green.

Στην παρούσα εργασία περιγράφεται η υλοποίηση ενός αλγόριθμου, του οποίου η λειτουργία είναι η εύρεση των διαγώνιων στοιχείων του αντίστροφου ενός οποιουδήποτε μπλοκ τρι-διαγώνιου πίνακα. Στην εργασία του ο Svizhenko (A. Svizhenko, 2002) δίνει μια πολύ καλή

μετατροπή των μερικών διαφορικών εξισώσεων Green σε μορφή πινάκων και τον τρόπο με τον οποίο αυτοί προκύπτουν. Ακολουθούμε την ίδια διαδικασία έτσι ώστε να φτάσουμε στον μπλοκ τρι-διαγώνιο πίνακα που χρειαζόμαστε. Έπειτα με την χρήση του αλγόριθμου που παρουσιάζουμε στην 3^η ενότητα εξάγουμε τα διαγώνια μπλοκ του αντίστροφου του πίνακα που προκύπτει με την παραπάνω μέθοδο. Όπως περιγράψαμε και πιο πάνω αυτά τα στοιχεία είναι τα σημαντικά για τον υπολογισμό των τιμών των G^r και $G^<$. Έχοντας υπολογίσει αυτές τις τιμές μπορούμε να υπολογίσουμε την πυκνότητα των ηλεκτρονίων που περιγράφονται στον φορμαλισμό NEGF.

Σκοπός της παρούσας εργασίας δεν είναι η αναλυτική περιγραφή της κβαντικής μεταφοράς σε κανάλι τρανζίστορ ή γενικά σε οποιαδήποτε νανο-ηλεκτρονική συσκευή, καθώς κάτι τέτοιο αποτελεί διδακτορική έρευνα, γι' αυτό και όπως θα παρατηρήσετε παρακάτω υπάρχουν κάποια δεδομένα τα οποία προεπιλεγμένα παίρνουν μηδενικές τιμές. Αυτά είναι οι ακριβείς τιμές των *self-energies* στις επαφές της συσκευής μας με την πηγή, την πύλη, την υποδοχή καθώς και την διάχυση των ηλεκτρονίων μέσα στην συσκευή λόγω σύγκρουσής των με πλεγματικές θέσεις και άλλα ηλεκτρόνια. Θα εξηγήσουμε στην ενότητα 3 τον τρόπο με τον οποίο υπολογίζονται οι τιμές για τις *self-energies* και κυρίως πως αυτές διακριτοποιούνται σε πίνακα. Κύριος σκοπός μας είναι α) η εύρεση των διαγωνίων στοιχείων του αντιστρόφου ενός μπλοκ-τριδιαγώνιου πίνακα (τα οποία αποτελούν τις τιμές των G^r και $G^<$) β) η εξαγωγή ενδεικτικών τιμών μέσω κατάλληλων προσομοιώσεων και γ) η παράθεση 2 αλγορίθμων εκ των οποίων ο πρώτος υλοποιείται σε κώδικα, για την πρακτική υλοποίηση της υπόθεσής μας, την εύρεση των διαγώνιων στοιχείων ενός μπλοκ-τριδιαγώνιου πίνακα.

2 Περιγραφή του προβλήματος

2.1 Φορμαλισμός

Όπως αναφέραμε παραπάνω οι διάφορες μέθοδοι για την επίλυση των εξισώσεων Green λειτουργούν καλά και στις 3 διαστάσεις. Εμείς όμως θα επικεντρωθούμε στην επίλυση των εξισώσεων για 2 διαστάσεων MOSFET. Έστω ότι έχουμε το 2-διάστατο μοντέλο ενός νανο-τρανζίστορ όπως φαίνεται στην εικόνα 2.

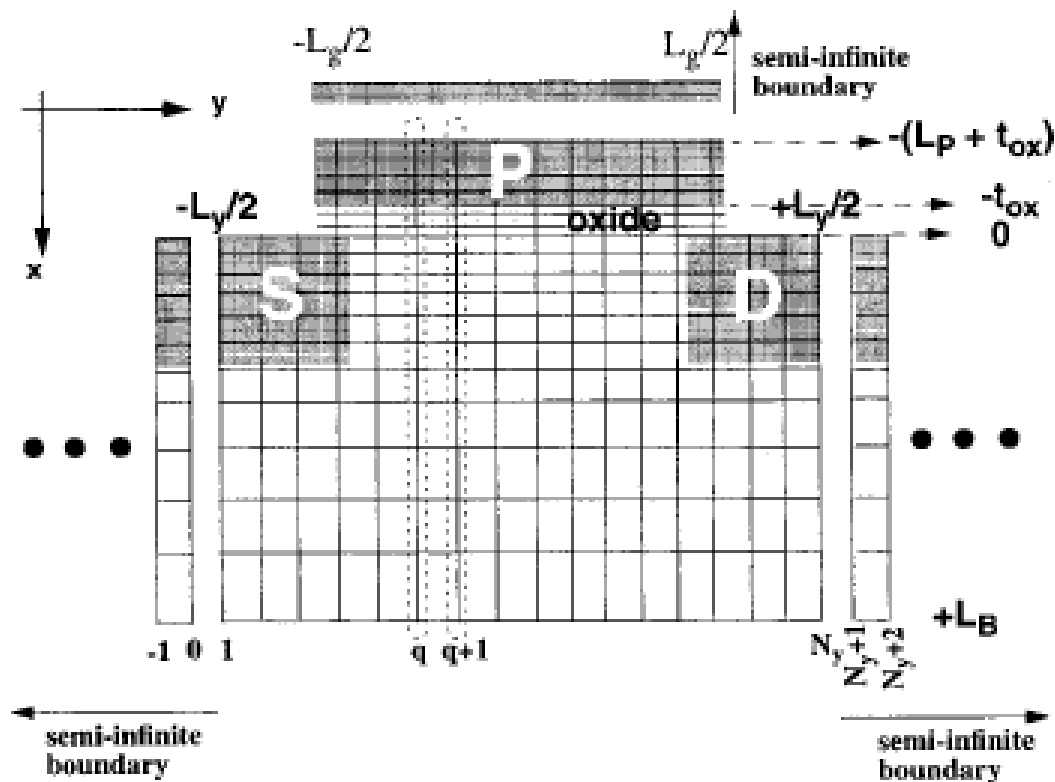


Figure 2: οι εξισώσεις λύνονται σε ένα 2-διάστατο μη-ομοιόμορφο χωρικό πλέγμα, με ημι-άπειρα όρια όπως βλέπουμε. Κάθε στήλη q αποτελεί τα διαγώνια στοιχεία των blocks των πινάκων που υλοποιούμε παρακάτω. Το ηλεκτροστατικό δυναμικό κρατιέται σταθερό στην αρχή των ημι-άπειρων περιοχών κοντά στην συσκευή μας.

Το σώμα του τρανζίστορ προβάλλεται πάνω σε ένα 2-διάστατο μη-ομοιόμορφο πλέγμα διαστάσεων $N_x \times N_y$ όπου το N_x περιγράφει τον αριθμό των στοιχείων κατά μήκος της συσκευής και το N_y περιγράφει τον αριθμό των στοιχείων κατά βάθος. Οι τιμές των N_x, N_y διαφέρουν ανάλογα με την γεωμετρία της κάθε συσκευής με δοσμένη πλεγματική ανάλυση. Αυξάνοντας τις τιμές θα βελτιώσουμε την ακρίβεια της προσομοίωσης καθορίζοντας ένα καλώς ορισμένο πλέγμα για την περιοχή που ενδιαφερόμαστε. Στην περίπτωση του νανο-τρανζίστορ, οι τυπικές τιμές των N_x και N_y μπορούν να οδηγήσουν σε προβλήματα των οποίων τα μεγέθη είναι της τάξης των εκατομμυρίων.

Στην συνέχεια θα δώσουμε την περιγραφή των κανόνων της φυσικής που σχετίζονται με την συσκευή που μας ενδιαφέρει. Η Χαμιλτονιανή μιας «κοιλιάδας» b ηλεκτρονίων, που να σχετίζεται με την συσκευή που μελετάμε έχει την εξής μορφή

$$H_b(\mathbf{r}) = -\frac{\hbar^2}{2} \left[\frac{d}{dx} \left(\frac{1}{m_x^b} \frac{d}{dx} \right) + \frac{d}{dy} \left(\frac{1}{m_y^b} \frac{d}{dy} \right) + \frac{d}{dz} \left(\frac{1}{m_z^b} \frac{d}{dz} \right) \right] + V(\mathbf{r}), \quad (1)$$

όπου (m_x^b, m_y^b, m_z^b) είναι τα συστατικά της ενεργής μάζας των ηλεκτρονίων στην κοιλιάδα b . Οι εξισώσεις κίνησης για τις G^r και $G^<$ είναι

$$[E - H_{b_1}(\mathbf{r}_1)] G_{b_1 b_2}^r(\mathbf{r}_1, \mathbf{r}_2, E) - \int d\mathbf{r} \Sigma_{b_1, b'}^r(\mathbf{r}_1, \mathbf{r}, E) G_{b', b_2}^r(\mathbf{r}, \mathbf{r}_2, E) = \delta_{b_1, b_2} \delta(\mathbf{r}_1, \mathbf{r}_2) \quad (2)$$

και

$$\begin{aligned}
& [E - H_{b_1}(\mathbf{r}_1)] G_{b_1, b_2}^<(\mathbf{r}_1, \mathbf{r}_2, E) \\
& - \int d\mathbf{r} \Sigma_{b_1, b'}^r(\mathbf{r}_1, \mathbf{r}, E) G_{b', b_2}^<(\mathbf{r}, \mathbf{r}_2, E) \\
& = \int d\mathbf{r} \Sigma_{b_1, b'}^<(\mathbf{r}_1, \mathbf{r}, E) G_{b', b_2}^a(\mathbf{r}, \mathbf{r}_2, E),
\end{aligned} \tag{3}$$

όπου G^a είναι η προχωρημένη εξίσωση Green (advanced Green's function). Στις παραπάνω εξισώσεις, οι συντεταγμένες αναφέρονται μόνο στην συσκευή μας. Η επιρροή των ημι-άπειρων περιοχών της πηγής, της πύλης και της υποδοχής, καθώς και μηχανισμοί διάχυσης (electron-phonon) συμπεριλαμβάνονται μέσω των όρων των *self-energies* $\Sigma_{b_1, b'}^r$ και $\Sigma_{b_1, b'}^<$. Υποθέτουμε ότι το φορτίο εισέρχεται ανεξάρτητα από την επαφή σε κάθε κοιλάδα. Τότε $\Sigma_{b_1, b_2, c}^r = \Sigma_{b_1, c}^r \delta_{b_1, b_2}$, όπου το C περιγράφει την *self-energy* εξ αιτίας των επαφών. Τέλος οι ζώνες οπών (hole bands) επεξεργάζονται χρησιμοποιώντας το μοντέλο drift-diffusion, το οποίο αναμένεται να είναι καλή προσέγγιση για n-MOSFET's.

Το ηλεκτροστατικό δυναμικό ποικίλει στο (x,y) επίπεδο και το σύστημα είναι σταθερό γύρω από τον άξονα z. Επομένως όλες οι ποσότητες $A(\mathbf{r}_1, \mathbf{r}_2, E)$ εξαρτώνται μόνο από την διαφορά συχνοτήτων $\mathbf{z}_1 - \mathbf{z}_2$. Χρησιμοποιώντας την σχέση

$$A(\mathbf{r}_1, \mathbf{r}_2, E) = \int \frac{dk_z}{2\pi} e^{ik_z(z_1 - z_2)} A(x_1, y_1, x_2, y_2, k_z, E), \tag{4}$$

οι εξισώσεις κίνησης για τις G^r και $G^<$ απλοποιούνται στην εξής μορφή

$$\left[E - \frac{\hbar^2 k_z^2}{2m_z} - H_b(\mathbf{r}_1) \right] G_b^r(\mathbf{r}_1, \mathbf{r}_2, k_z, E) - \int d\mathbf{r} \Sigma_b^r(\mathbf{r}_1, \mathbf{r}, k_z, E) G_b^r(\mathbf{r}, \mathbf{r}_2, k_z, E) = \delta(\mathbf{r}_1 - \mathbf{r}_2) \quad (5)$$

και

$$\left[E - \frac{\hbar^2 k_z^2}{2m_z} - H_b(\mathbf{r}_1) \right] G_b^r(\mathbf{r}_1, \mathbf{r}_2, k_z, E) - \int d\mathbf{r} \Sigma_b^r(\mathbf{r}_1, \mathbf{r}, k_z, E) G_b^<(\mathbf{r}, \mathbf{r}_2, k_z, E) = \int d\mathbf{r} \Sigma_b^<(\mathbf{r}_1, \mathbf{r}, k_z, E) G_b^a(\mathbf{r}, \mathbf{r}_2, k_z, E), \quad (6)$$

όπου $Z_b = Z_{b,b}$ και για την συνέχεια εφόσον περιοριζόμαστε σε 2 διαστάσεις $\mathbf{r} \rightarrow (x, y)$.

Η πυκνότητα καταστάσεων (density of states) $[N(\mathbf{r}, k_z, E)]$ και η πυκνότητα φορτίου $[\rho(\mathbf{r}, k_z, E)]$ είναι το άθροισμα της συνεισφοράς από τις ανεξάρτητες κοιλάδες

$$N(\mathbf{r}, k_z, E) = \sum_b N_b(\mathbf{r}, k_z, E) = -\frac{1}{\pi} \text{Im}[G_b^r(\mathbf{r}, \mathbf{r}, k_z, E)], \quad (7)$$

$$\rho(\mathbf{r}, k_z, E) = \sum_b \rho_b(\mathbf{r}, k_z, E) = -iG_b^<(\mathbf{r}, \mathbf{r}, k_z, E). \quad (8)$$

2.2 Διακριτοποιημένες εξισώσεις Green: G^r και $G^<$

Η σταθερή επίλυση της συνάρτησης Green και των εξισώσεων του Poisson απαιτεί επαναληπτικό υπολογισμό της μη-ισορροπημένης πυκνότητας φορτίου. Αυτός ο υπολογισμός είναι συχνά το πιο χρονοβόρο κομμάτι στον σχεδιασμό των ηλεκτρονικών χαρακτηριστικών των συσκευών.

Η κοινή διαδικασία για την εκτίμηση της πυκνότητας ηλεκτρονίου χρησιμοποιεί την εξής έκφραση,

$$\begin{aligned}\rho_b(\mathbf{r}, k_z, E) &= -iG_b^<(\mathbf{r}, \mathbf{r}, k_z, E) \\ &= -i \int d\mathbf{r}_1 d\mathbf{r}_2 G_b^r(\mathbf{r}, \mathbf{r}_1, k_z, E) \\ &\quad \times \Sigma_b^<(\mathbf{r}_1, \mathbf{r}_2, k_z, E) G_b^a(\mathbf{r}_2, \mathbf{r}, k_z, E)\end{aligned}\quad (9)$$

όπου $G^r(\mathbf{r}_1, \mathbf{r}_2, k_z, E)$ πρέπει να υπολογιστούν μεταξύ όλων των $N_x \times N_y$ στοιχείων του πλέγματος και εκείνων των πλεγματικών στοιχείων που συμπεριλαμβάνουν μια μη-μηδενική G^a . Ο απαιτούμενος χρόνος επίλυσης για όλα τα στοιχεία του G^r φτάνει μέχρι $(N_x \times N_y)^3$, έτσι όπως συμπεραίνουμε η χρήση της προηγούμενης εξίσωσης είναι πολύ ακριβή.

Σε μορφή πίνακα οι εξισώσεις (5) και (6) γράφονται ως:

$$A'G^r = \lambda \quad (10)$$

Και

$$A'G^< = \Sigma^<G^{\alpha} . \quad (11)$$

Οι *self-energies* που οφείλονται στην πηγή, την υποδοχή και την πύλη (S,D,P) είναι μη-μηδενικές μόνο κατά μήκος των γραμμών $y = y_S = y_1$, $y = y_D = y_{N_y}$ και $x = x_P$. Ο πίνακας A' έχει διαστάσεις $N_x N_y * N_x N_y$ και είναι έτσι διατεταγμένος ώστε όλα τα πλεγματικά στοιχεία τα οποία είναι σε συγκεκριμένη συντεταγμένη y να αντιστοιχούν στα διαγώνια στοιχεία του. Η συμβολογραφία που έχουμε υιοθετήσει είναι ότι το $A'_{j_1,j_2}(i, i')$ αναφέρεται σε είσοδο που δεν ανήκει στην κύρια διαγώνιο και το οποίο αντιστοιχεί στα πλεγματικά στοιχεία (x_i, y_{j_1}) και $(x_{i'}, y_{j_2})$.

Τα μη-μηδενικά στοιχεία των διαγωνίων στοιχείων του πίνακα A' δίνονται από τις εξής σχέσεις:

$$\begin{aligned} A'_{j,j}(i, i) = & E' - V_{i,j} - T_{j,j}(i+1, i) - T_{j,j}(i-1, i) \\ & - T_{j+1,j}(i, i) - T_{j-1,j}(i, i) - \sum_S^r(x_i, x_i) \delta_{j,1} \\ & - \sum_D^r(x_i, x_i) \delta_{j, N_y} - \sum_P^r(y_j, y_j) \delta_{i,1} \\ & - \sum^r(x_i, y_j; x_i, y_j), \end{aligned} \quad (12)$$

$$\begin{aligned} A'_{j,j}(i \pm 1, i) = & T_{j,j}(i \pm 1, i) - \sum_S^r(x_{i \pm 1}, x_i) \delta_{j,1} \\ & - \sum_D^r(x_{i \pm 1}, x_i) \delta_{j, N_y} - \sum^r(x_{i \pm 1}, y_j; x_i, y_j) \end{aligned} \quad (13)$$

$$A'_{j,j}(i, i') = - \sum_S^r(x_i, x_{i'}) \delta_{j,1} - \sum_D^r(x_i, x_{i'}) \delta_{j, N_y} \quad (14)$$

για $i' \neq i, i \neq 1$,

όπου $E' = E - \hbar_{bar}^2 k_z^2 / 2m_z$ και $V_{i,j} = V(x_i, y_j)$. Τα άνω και κάτω διαγώνια μπλοκ του πίνακα δίνονται από τις εξής σχέσεις:

$$A'_{j \pm 1, j}(i, i) = T_{j \pm 1, j}(i, i) - \sum_P' (y_j, y_{j \pm 1}) \delta_{i,1}$$

$$A'_{j, j'}(i, i') = 0 \quad \text{για} \quad j' \neq j, j \neq 1 \quad (15)$$

Όπως παρατηρούμε παρότι ο πίνακας είναι μπλοκ-τριδιαγώνιος και έτσι θα τον χειριστούμε, υπάρχουν στοιχεία των διαγώνιων μπλοκ που δεν ανήκουν στις τριδιαγώνιους των μπλοκ αυτών. Αυτά τα στοιχεία αποτελούνται μόνο από τιμές *self-energy* και θα τα θεωρήσουμε μηδενικά για όλο το υπόλοιπο της εργασίας (Λεπτομέρειες για τα *self-energies* παρακάτω στο ίδιο κεφάλαιο).

Τα μη-μηδενικά στοιχεία του πίνακα T καθορίζονται από τις εξής σχέσεις:

$$T_{j, j}(i \pm 1, i) = \frac{\hbar^2}{2m^{\pm x}} \frac{2}{x_{i+1} - x_{i-1}} \frac{1}{|x_{i \pm 1} - x_i|} \quad (16)$$

$$T_{j \pm 1, j}(i, i) = \frac{\hbar^2}{2m^{\pm y}} \frac{2}{y_{j+1} - y_{j-1}} \frac{1}{|y_{j \pm 1} - y_j|} \quad (17)$$

όπου $m^{\pm x} = 2/(m_{i \pm 1, j} + m_{i, j})$ και $m^{\pm y} = 2/(m_{i, j \pm 1} + m_{i, j})$.

Τα μη-μηδενικά στοιχεία της $\Sigma_P^r(y_j, y'_j)$, όπου $j' \neq j$ θεωρούνται αμελητέα για να σιγουρέψουμε το γεγονός ότι ο A' είναι μπλοκ-τριδιαγώνιος (εφόσον έχουμε τονίσει ότι οι αλγόριθμοι που χρησιμοποιούμε για την εύρεση των τιμών των G^r και $G^<$ βασίζονται στην μπλοκ-τριδιαγώνια μορφή του A').

Το λ που εμφανίζεται στην (10) αντιστοιχεί στην συνάρτηση δ της εξίσωσης (5). Ο λ είναι διαγώνιος πίνακας διαστάσεων $N_x N_y * N_x N_y$ του οποίου τα στοιχεία δίνονται από την εξής σχέση:

$$\lambda_{jj}^{(i,i)} = \frac{4}{(x_{i+1} - x_{i-1})(y_{i+1} - y_{i-1})} \quad (18)$$

Ο φορμαλισμός που περιγράψαμε μόλις τώρα θα αποτελέσει την βάση στην οποία υλοποιούμε τους αλγόριθμούς μας.

Προχωρώντας τον φορμαλισμό μας, για τον υπολογισμό των τιμών της G^r θα πολλαπλασιάσουμε την εξίσωση (10) με τον αντίστροφο του λ (λ^{-1}). Η εξίσωση που προκύπτει πλέον είναι η εξής:

$$AG^r = I,$$

όπου ο πίνακας A είναι συμμετρικός πίνακας για ομογενή και μη-ομογενή ορθογώνια πλέγματα (το αναφέρουμε αυτό επειδή ο A' είναι συμμετρικός μόνο σε ομογενή πλέγματα).

Προτού προχωρήσουμε στην περιγραφή του αλγόριθμου που χρησιμοποιήσαμε για τον υπολογισμό των διαγωνίων μπλοκ του πίνακα A θα αναφέρουμε επιγραμματικά κάποια στοιχεία για τα *self-energies* διότι μπορεί σε αυτή την εργασία να θεωρούμε τις τιμές τους μηδενικές αλλά δεν παύουν να παίζουν σημαντικό ρόλο στην κβαντική μεταφορά ηλεκτρονίων διαμέσου καναλιών τρανζίστορ ή νανο-καλωδίων. (A. Svizhenko, 2002)

2.3 Self-energies ($\Sigma^r, \Sigma^<, \Sigma$)

Σε αυτή την παράγραφο θα δώσουμε μια σύντομη περιγραφή για το πώς προκύπτουν θεωρητικά οι *self-energies* και πως αυτές οι τιμές εισέρχονται στην επίλυση των εξισώσεων *Green*.

Για την ιδιοενέργεια κάθε επίπεδου κύματος E_{kj} η μορφή της *retarded Green* εξίσωσης που σχετίζεται με την μεταφορά σε 2 διαστάσεις είναι:

$$G(E) = [EI - [h(x, z) + E_{kj}I]]^{-1} = [E(k_x, k_z)I - h(x, z)]^{-1}$$

όπου η ενέργεια του επιπέδου ορίζεται ως $E(k_x, k_z) = E - E_{kj}$. Έπειτα λαμβάνουμε υπόψη τις άπειρες επαφές, στις οποίες εφάπτεται το τρανζίστορ, με την χρήση μιας κατάλληλης εξίσωσης *self-energy*. Η *self-energy* (Σ) περιγράφει τα φαινόμενα στην Χαμιλτονιανή του τρανζίστορ, τα οποία οφείλονται στις εξερχόμενες κυματικές εξισώσεις που προκαλούνται από διεγέρσεις μέσα στο τρανζίστορ. Αποτελεί μια πολύ χρήσιμη παράμετρο καθώς μας επιτρέπει να εξαλείψουμε τις άπειρες πηγή (S) και υποδοχή (D) και να δουλέψουμε μόνο με τον υποχώρο του τρανζίστορ του οποίου οι διαστάσεις είναι αρκετά μικρότερες και σαφώς υπολογίσιμες. Το μέγεθος του πίνακα της Σ είναι αντίστοιχο με τις διαστάσεις του πίνακα A' που υπολογίσαμε πιο πάνω δηλαδή $N_x N_y * N_x N_y$. Συμπεριλαμβάνοντας τις *self-energies*, η τελική μορφή του πίνακα της εξίσωσης *Green* θα είναι:

$$G[E(k_x, k_z)] = [E(k_x, k_z)I - h(x, z) - \Sigma_S - \Sigma_D]^{-1}$$

Οι εξισώσεις *self-energy* εκτός από τις επαφές του τρανζίστορ περιγράφουν και την διάχυση στις συγκρούσεις των ηλεκτρονίων με τις πλεγματικές θέσεις (phonons) καθώς και με το οξείδιο που χωρίζει το κανάλι του τρανζίστορ από την πύλη του. Οι τελευταίες συνήθως αγνοούνται και δεν λαμβάνονται υπ' όψιν στις μελέτες για την κβαντική μεταφορά. Στην παρούσα εργασία θα θεωρήσουμε αμελητέες και τις τιμές των *self-energies* που οφείλονται στις επαφές του τρανζίστορ με την πύλη, την πηγή και την υποδοχή. Παρ' όλα αυτά θα περιγράψουμε περιληπτικά πως προκύπτουν οι τιμές τους.

Η άπειρη Χαμιλτονιανή και η αντίστοιχη εξίσωση Green μπορούν να κερματισθούν ως εξής:

$$\begin{bmatrix} G_{device} & G_{device,D} \\ G_{D,device} & G_D \end{bmatrix} = \begin{bmatrix} E(k_x, k_z)I - h(x, z)_{device} & -\beta \\ -\beta & E(k_x, k_z)I - h(x, z)_D \end{bmatrix}$$

Στην προκειμένη περίπτωση το D συμβολίζει την υποδοχή (drain). Το μπλοκ του πίνακα που μας ενδιαφέρει είναι το G_{device} καθώς δεν ενδιαφερόμαστε για την εξίσωση Green στην υποδοχή. Το G_{device} μπορεί να εκφραστεί με γνωστές τιμές ως:

$$G_{device} [E(k_x, k_z)] = [E(k_x, k_z)I - h(x, z)_{device} - \Sigma_D]^{-1}$$

όπου ο πίνακας για την *self-energy* της υποδοχής είναι:

$$\Sigma_D = \begin{bmatrix} 0 & 0 & \dots \\ 0 & 0 & \dots \\ -\beta & 0 & \dots \end{bmatrix} \left[\begin{array}{c|ccc} E(k_x, k_z)I - \alpha_{N_X+1} & -\beta & 0 & \dots \\ \hline -\beta & E(k_x, k_z)I - \alpha_{N_X+2} & -\beta & 0 \\ 0 & -\beta & \dots & \dots \\ \dots & 0 & \dots & \dots \end{array} \right]^{-1} \\ \times \begin{bmatrix} \dots & 0 & -\beta \\ \dots & 0 & 0 \\ \dots & \dots & \dots \end{bmatrix}$$

Για να υπολογίσουμε τις τιμές του προηγούμενου πίνακα χρειαζόμαστε μόνο το 1ο μπλοκ από τον αντίστροφο του άπειρου πίνακα που σχετίζεται με την υποδοχή. Επίσης πρέπει να σημειώσουμε ότι τα διαγώνια μπλοκ αυτού του άπειρου πίνακα επαναλαμβάνονται εξαιτίας της αναλλοιότητας μέσα στην υποδοχή. Χρησιμοποιώντας αυτή την έκφραση και κερματίζοντας τον πίνακα όπως δείξαμε προκύπτει μια έκφραση για το 1ο μπλοκ (g_D) του αντίστροφου του άπειρου πίνακα,

$$I = g_D [E(k_x, k_z)I - \alpha_{N_X+1} - \beta g_D \beta]$$

Όταν υπολογιστεί το g_D θα έχουμε:

$$\Sigma_D = \begin{bmatrix} 0 & \dots & 0 \\ 0 & \dots & 0 \\ 0 & \dots & \beta g_D \beta \end{bmatrix}$$

Πρέπει να σημειώσουμε ότι μόνο η τελευταία κάθετη “γραμμή” από πλεγμιακά στοιχεία εφάπτεται στην υποδοχή, επομένως η *self-energy* για την υποδοχή θα έχει ένα μη-μηδενικό μπλοκ που θα «διαταράσσει» το τελευταίο διαγώνιο μπλοκ της Χαμιλτονιανής του τρανζίστορ. Αντίστοιχα βρίσκουμε και τις τιμές για την *self-energy* της πηγής.

Όσον αφορά την επαφή του πλέγματος με το οξείδιο της πύλης, ξέρουμε ότι η αντίστοιχη *self-energy* είναι μη-μηδενική μόνο σε αυτό το σημείο, επομένως ακολουθώντας μια διαδικασία αντίστοιχη με την εύρεση των τιμών της Σ για τις επαφές με την πηγή και την υποδοχή, θα βρούμε και τις τιμές για την επαφή με την πύλη. Είναι ασφαλές να υποθέσουμε ότι στον πίνακα της Σ για την πύλη τα μη-μηδενικά στοιχεία του θα είναι τα αρχικά στοιχεία κάθε διαγώνιου μπλοκ. (Venugopal, 2003)

2.4 Χρήση των διαγωνίων στοιχείων του A'

Με την παραπάνω διαδικασία περιγράφουμε την εξαγωγή των διαγωνίων αντιστρόφων στοιχείων του πίνακα A' τα οποία στην συνέχεια θα χρησιμοποιήσουμε για να υπολογίσουμε τις τιμές που μας ενδιαφέρουν στην προσομοίωσή μας. Πιο συγκεκριμένα τα στοιχεία μάς χρειάζονται για τον υπολογισμό των πυκνοτήτων φορτίο, ρεύματος και καταστάσεων. Στην συνέχεια αυτές οι τιμές χρησιμοποιούνται για τον ακριβή υπολογισμό διαφόρων χαρακτηριστικών του τρανζίστορ όπως ρεύματα και τάσεις.

Ο υπολογισμός αυτών των στοιχείων είναι πολύ σημαντικός όταν περιγράφουμε κβαντική μεταφορά επομένως πρέπει να είμαστε πολύ προσεκτικοί στον τρόπο που τα υπολογίζουμε καθώς και να ελέγχουμε συχνά τα αποτελέσματα των υπολογισμών ώστε να είμαστε σίγουροι ότι συνάδουν με την θεωρία στην οποία βασίζεται η εν λόγω διαδικασία μεταφοράς.

Τα χαρακτηριστικά σταθερής κατάστασης των τρανζίστορ MOSFET τα οποία έχουν κάποιο πρακτικό ενδιαφέρον είναι το ρεύμα οδήγησης, το

ρεύμα αποκοπής, η κλίση της χαρακτηριστικής ρεύματος-τάσης της υποδοχής και η τάση V_{th} . Από τα αποτελέσματα μπορούμε να συμπεράνουμε ότι οι κβαντομηχανικές προσομοιώσεις αποφέρουν αποτελέσματα πολύ διαφορετικά από τις μεθόδους κίνησης-διάχυσης. Αυτές οι διαφορές προκύπτουν εξαιτίας των παρακάτω κβαντομηχανικών χαρακτηριστικών:

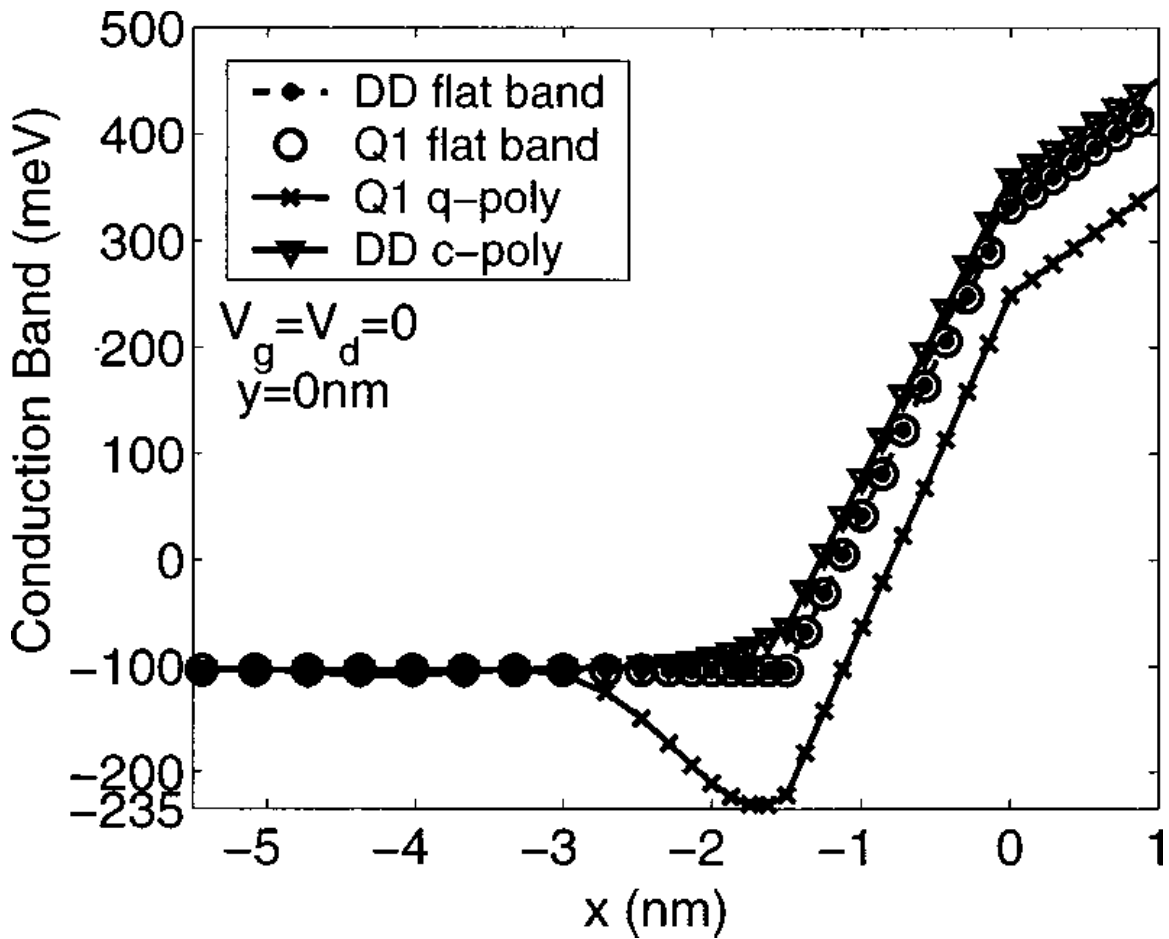
- Εξάντληση της πύλης με τρόπο διαφορετικό από την κλασική περίπτωση.
- Εξάρτηση των επιπέδων αντήχησης στο κανάλι.
- Φαινόμενο σύρραγγας του φορτίου κατά μήκος του οξειδίου της πύλης και από την πηγή προς την υποδοχή.
- Ημι-βαλλιστική ροή ηλεκτρονίων.

Όπως αναφέρουμε και στο κεφάλαιο 4, οι προσομοιώσεις που πραγματοποιούμε βασίζονται στο *MIT well-tempered 25nm transistor*.

Παρακάτω αναφέρουμε ενδεικτικά τα χαρακτηριστικά τα οποία χρησιμοποιούν τις τιμές που υπολογίσαμε, καθώς και παραθέτουμε και κάποια διαγράμματα τα οποία δείχνουν την διαφορά υπολογισμού ανάμεσα σε κβαντομηχανικές μεθόδους και μεθόδους κίνησης-διαχύσεως (A. Svizhenko, 2002).

2.4.1 Επίδραση στην περιοχή απογύμνωσης πολυπυριτίου: I_d vs V_g .

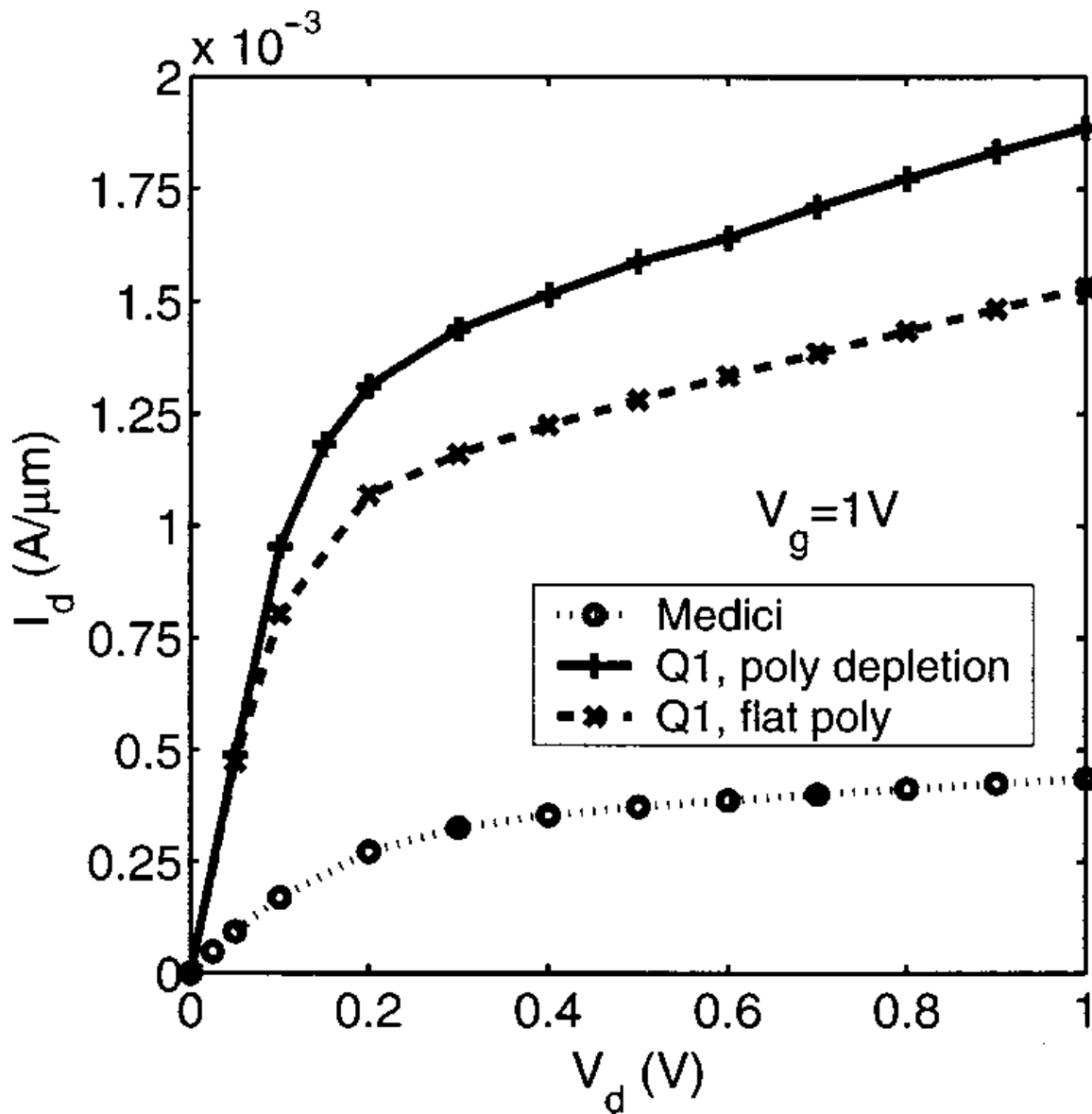
Η κβαντομηχανικά υπολογισμένη πυκνότητα ηλεκτρονίου κοντά στο όριο του διοξειδίου του πυριτίου στην περιοχή του πολυπυριτίου είναι μικρότερη από την ομογενή πυκνότητα υποβάθρου. Αυτό συμβαίνει επειδή η κυματική εξίσωση ηλεκτρονίων είναι μικρή κοντά στο όριο. Σαν αποτέλεσμα, η μπάντα αγωγιμότητας στην πύλη «κλίνει» προς μια κατεύθυνση αντίθετη από αυτή που υπολογίστηκε ημι-κλασικά (συγκρίνετε το x και το τρίγωνο στην παρακάτω εικόνα).



Η μπάντα που κάμπτεται στην πύλη παίζει σημαντικό ρόλο στον προσδιορισμό της τάσης κατωφλίου και του ρεύματος αποκοπής.

2.4.2 I_d vs V_d

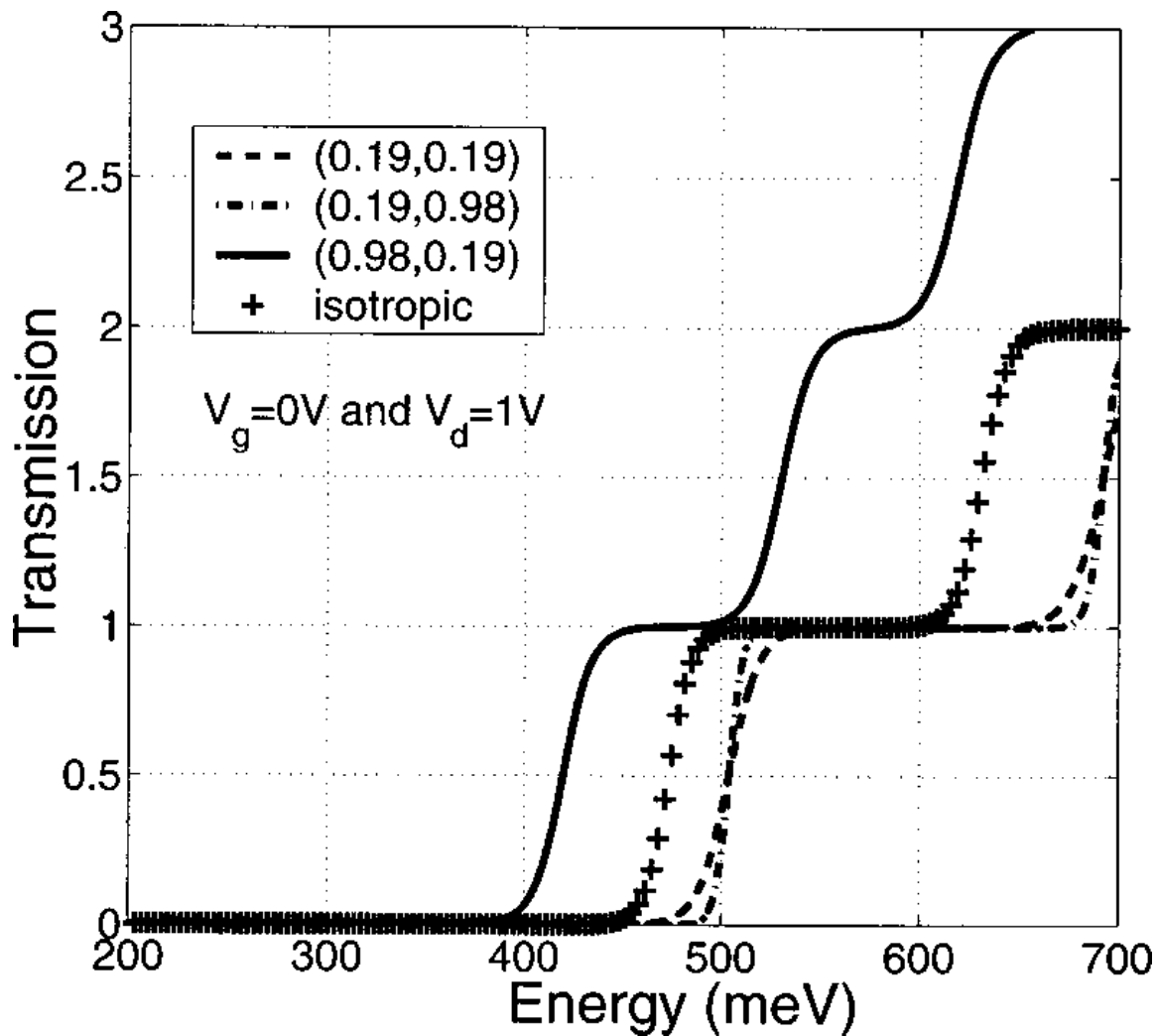
Οι τιμές dI_d/dV_d και το ρεύμα οδήγησης είναι πολύ σημαντικά σε εφαρμογές MOSFET επειδή καθορίζουν ταχύτητες εναλλαγής. Η παρακάτω εικόνα δείχνει την σύγκριση του ρεύματος υποδοχής με την αντίστοιχη τάση για $V_g = 0$ και $V_g = 1$.



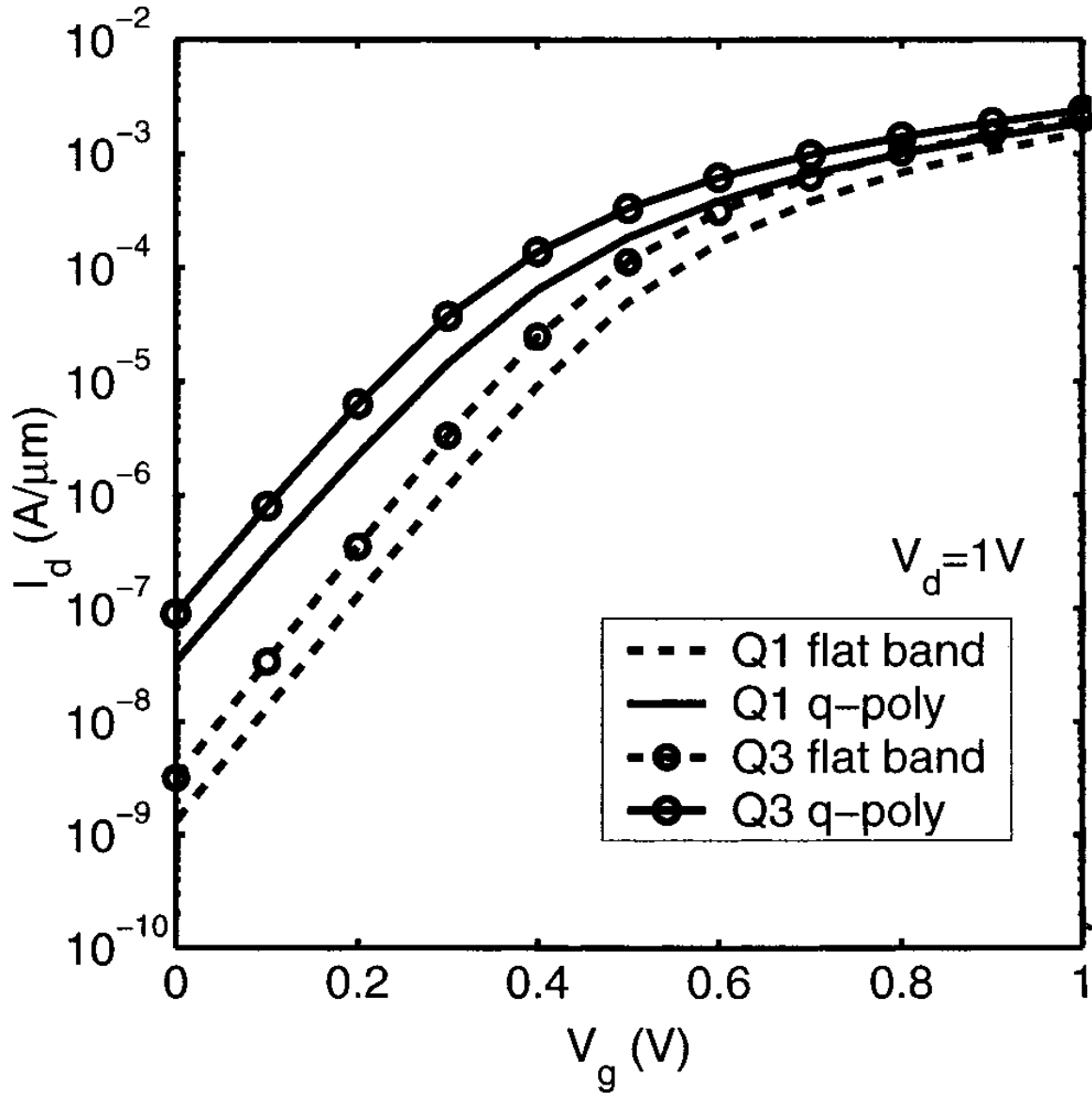
Το ρεύμα οδήγησης ($V_g = 1$) υπολογίστηκε χρησιμοποιώντας κβαντομηχανική προσομοίωση για ιστροπικές ενεργές μάζες σωματιδίων. Οι dI_d/dV_d στην γραμμική περιοχή είναι 3 φορές μεγαλύτερη σε αυτή την προσομοίωση. Το ρεύμα υποστρώματος υποδοχής είναι μικρότερο σε αυτή την προσομοίωση.

2.4.3 Ισοτροπικές vs ανισοτροπικές ενεργές μάζες

Η πρωταρχική επιρροή των ανισότροπων ενεργών μαζών είναι να επηρεάσουν την ενέργεια των υποστρωμάτων στο επίπεδο αντιστροφής. Κοιλάδες με τις μεγαλύτερες ενεργές μάζες κάθετες στο οξείδιο ($0.98m_0$) έχουν υποστρωματικές ενέργειες μικρότερες από την περίπτωση των ισοτροπικών ενεργών μαζών. Στην παρακάτω εικόνα βλέπουμε ότι οι κοιλάδες με ($m_x = 0.98m_0, m_y = m_z = 0.19m_0$) έχουν επίπεδα αντήρησης τα οποία είναι περισσότερο από 50meV χαμηλότερο σε ενέργεια από την περίπτωση των ισοτροπικών μαζών.



Το αντίστοιχο ρεύμα υποστρώματος είναι μερικές εκατοντάδες φορές μεγαλύτερο από την τιμή που προκύπτει από την περίπτωση ιστροπικών ενεργών μαζών (βλέπε επόμενη εικόνα).



3 Περιγραφή των αλγορίθμων

3.1 Περιγραφή του 1^{ου} αλγόριθμου

Αρχικά θα στρέψουμε την προσοχή μας στον πίνακα A που προέκυψε πιο πάνω όταν πολλαπλασιάσαμε την εξίσωση (10) με τον αντίστροφο του πίνακα λ .

Γενικά, όπως έχουμε ήδη προαναφέρει, το πρόβλημα του υπολογισμού των πυκνοτήτων των ηλεκτρονίων σε ένα νανο-τρανζίστορ μπορεί να επικεντρωθεί στην εύρεση των διαγώνιων μπλοκ του G^r , όπου $AG^r = I$ και ο A είναι ένας μπλοκ τριδιαγώνιος πίνακας της μορφής

$$A = \begin{pmatrix} A_1 & -B_1 & & & \\ -B_1^T & A_2 & -B_2 & & \\ & \ddots & \ddots & \ddots & \\ & & -B_{N_y-2}^T & A_{N_y-1} & -B_{N_y-1} \\ & & & -B_{N_y-1}^T & A_{N_y} \end{pmatrix} \quad (19)$$

όπου κάθε $A_i, B_i \in \mathbb{C}^{N_y N_x \times N_y N_x}$ με N_x διαγώνια μπλοκ μεγέθους N_y το καθένα. Όταν ο A είναι κανονικός, δηλαδή όταν ο B_i δεν είναι ιδιόμορφος (nonsingular), υπάρχουν 2 (μη μοναδικές) σειρές πινάκων $\{U_i\}$ και $\{V_i\}$ για τους οποίους για κάθε $j \geq 1$

$$(A^{-1})_{ij} = U_i V_j^T.$$

Έτσι ο A^{-1} μπορεί να γραφτεί ως:

$$A^{-1} = \begin{pmatrix} U_1 V_1^T & U_1 V_2^T & U_1 V_3^T & \cdots & U_1 V_{N_y}^T \\ V_2 U_1^T & U_1 V_2^T & U_2 V_3^T & \cdots & U_2 V_{N_y}^T \\ V_3 U_1^T & V_3 U_2^T & U_3 V_3^T & \cdots & U_3 V_{N_y}^T \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ V_{N_y} U_1^T & V_{N_y} U_2^T & V_{N_y} U_3^T & \cdots & U_{N_y} V_{N_y}^T \end{pmatrix} \quad (20)$$

Οι σειρές πινάκων $\{U_i\}$ και $\{V_i\}$ μπορούν να υπολογιστούν σε $O(N_y N_x^3)$ αριθμό πράξεων σύμφωνα με την παρακάτω διαδικασία:

$$\begin{aligned} U_1 &= I_{N_x, N_x}, \quad U_2 = B_1^{-1} A_1, \\ U_{i+1} &= B_i^{-1} (A_i U_i - B_{i-1}^T U_{i-1}), \quad i = 2, \dots, N_y - 1, \\ V_{N_y}^T &= (A_{N_y} U_{N_y} - B_{N_y-1}^T U_{N_y-1})^{-1}, \quad V_{N_y-1}^T = V_{N_y}^T A_{N_y} B_{N_y-1}^{-1} \\ V_i^T &= (V_{i+1}^T A_{i+1} - V_{i+2}^T B_{i+1}^T) B_i^{-1}, \quad i = N_y - 2, \dots, 1. \end{aligned}$$

Παρόλο που τα μαθηματικά τα οποία περιγράφονται για τον υπολογισμό του αντιστρόφου είναι θεωρητικά πολύ κομψά, ο υπολογισμός των παραμέτρων $\{U_i\}$ και $\{V_i\}$ βασανίζεται από αριθμητικά προβλήματα ακόμα και για μεσαίου μεγέθους προβλήματα. Το αίτιο βρίσκεται στο γεγονός ότι οι $\{U_i\}$ και $\{V_i\}$ αυξάνονται εκθετικά με το i . Πιο συγκεκριμένα για πίνακες με μέγεθος τόσο μικρό όσο $N_y N_x = 1000$, ο υπολογισμός των $\{U_i\}$ και $\{V_i\}$ γίνεται ασταθής (εξαιτίας της υπερχειλίσης και της υποχειλίσης) με υπολογιστές που χρησιμοποιούν αριθμητική διπλής ακρίβειας. Επομένως δεν μπορούμε να στηριχθούμε σε αυτή την μέθοδο για τους δικούς μας υπολογισμούς.

Αντίθετα η μέθοδος που θα χρησιμοποιήσουμε είναι η εξής:

Χρησιμοποιούμε λόγους ακολουθιακών στοιχείων των σειρών $\{U_i\}$ και $\{V_i\}$ για τον σταθερό υπολογισμό του A^{-1} . Αυτές οι ακολουθίες που τις ονομάζουμε $\{R_i\}$ και $\{S_i\}$ προσδιορίζονται ως εξής:

$$U_i = R_i U_{i+1}, \quad V_{i+1}^T = V_i^T S_i.$$

Οι σειρές $\{R_i\}$ και $\{S_i\}$ μπορούν να υπολογιστούν σε χρόνο $O(N_y N_x^3)$ με χρήση των παρακάτω αριθμητικώς σταθερών αναδρομών:

$$R_1 = A_1^{-1} B_1, \quad R_i = (A_i - B_{i-1}^T R_{i-1})^{-1} B_i, \quad i = 2, \dots, N_y - 1, \\ S_{N_y-1} = B_{N_y-1} A_{N_y}^{-1}, \quad S_i = B_i (A_{i+1} - S_{i+1} B_{i+1}^T)^{-1}, \quad i = N_y - 2, \dots, 1.$$

Μπορούμε πλέον να προσδιορίσουμε ότι τα διαγώνια στοιχεία του A^{-1} , τα οποία τα ονομάζουμε D_i , δίνονται από την αναδρομή

$$D_1 = (A_1 - B_1 S_1^T)^{-1}, \quad D_{i+1} = (A_{i+1} - B_{i+1} S_{i+1}^T)^{-1} (I + B_i^T D_i S_i), \quad i = 1, \dots, N_y - 2, \\ D_{N_y} = A_{N_y}^{-1} (I + B_{N_y-1}^T D_{N_y-1} S_{N_y-1}).$$

Έκτοτε, τα υπόλοιπα μπλοκ του πίνακα μπορούν να υπολογιστούν με έναν αριθμητικά σταθερό τρόπο όπως ακολουθεί:

$$A_{ij}^{-1} = D_i S_{i+1} S_{i+2} \dots S_j, \quad j > i, \quad A_{ij}^{-1} = R_{j+1} R_{j+2} \dots R_i D_i, \quad j < i.$$

Παρόλο που ο αλγόριθμος μας δίνει την δυνατότητα να υπολογίσουμε όλα τα μπλοκ του A^{-1} , δεν αποτελεί σκοπό μας στην εργασία οπότε δεν προχωράμε στην εύρεση των μη-διαγώνιων μπλοκ. Αλλά είναι σημαντικό να αναφέρουμε όλη την διαδικασία του αλγόριθμου για πιθανή μελλοντική χρήση. (Jitesh Jain, 2007)

Τέλος θα πρέπει να αναφέρουμε ότι ο παραπάνω αλγόριθμος είναι ενδεικτικός για την εύρεση των διαγωνίων στοιχείων ενός αντίστροφου μπλοκ-πίνακα. Στο διαδίκτυο υπάρχει πληθώρα δημοσιευμένων εργασιών και αλγόριθμων που ασχολούνται με αυτό το αντικείμενο, οι οποίοι είναι σταθεροί και διαρκώς συντομότεροι σε χρόνο υπολογισμού [1],[2],[5].

3.2 Περιγραφή του 2^{ου} αλγόριθμου

Όπως αναφέραμε και πιο πάνω θα παραθέσουμε και έναν 2^ο αλγόριθμο ο οποίος εκτελεί την ίδια διαδικασία με τον 1^ο, δηλαδή την εύρεση των διαγώνιων στοιχείων του αντιστρόφου ενός μπλοκ-τριδιαγώνιου πίνακα. Παρ' όλα αυτά δεν θα προχωρήσουμε στην υλοποίησή του σε κώδικα καθώς αποτελεί εξτρά φόρτο εργασίας, αλλά η υλοποίησή του θα πραγματοποιηθεί εκτός διπλωματικής.

Η λογική πάνω στην οποία υλοποιείται ο αλγόριθμος είναι η *διαίρει και βασίλευε*. Έστω ότι έχουμε έναν μη ιδιόμορφο συμμετρικό πίνακα $A \in \mathbb{C}^{NyNx \times NyNx}$ και ο οποίος μπορεί να αποσυντεθεί σε έναν 2x2 μπλοκ διαγώνιο πίνακα C και έναν χαμηλής στάθμης (low rank) πίνακα $-L$. Εάν η διάσταση του 1^{ου} μπλοκ του C είναι μεγέθους m και η τάξη του L προσδιορίζεται ως q μπορούμε να γράψουμε την εξής μορφή:

$$A = C - L, \quad C := \begin{pmatrix} C_1 & \\ & C_2 \end{pmatrix}, \quad L := EE^T$$

όπου $C_1 \in \mathbb{C}^{m \times m}, C_2 \in \mathbb{C}^{(n-m) \times (n-m)}$

και $E \in \mathbb{C}^{n \times q}$

Ο αντίστροφος του A μπορεί τώρα να εξαχθεί αφαιρώντας $Y := C$ και $Z := E$ στην εξίσωση των Sherman-Morrison-Woodbury. Αυτό αποφέρει την σχέση

$$A^{-1} = C^{-1} + UR^{-1}U^T$$

με $U \in \mathbb{C}^{n \times q}$ και $R \in \mathbb{C}^{q \times q}$ που ορίζονται σαν

$$U := C^{-1}E, \quad R := I_q - E^T U$$

όπου μπορούμε να υποθέσουμε ότι οι C, R είναι μη ιδιόμορφοι. Επομένως η διαγώνιος που ψάχνουμε, την οποία συμβολίζουμε $\mathcal{D}(A^{-1})$ μπορεί να βρεθεί από την σχέση

$$\mathcal{D}(A^{-1}) = \mathcal{D}(C^{-1}) + \mathcal{D}(UR^{-1}U^T)$$

Ο αλγόριθμος παρουσιάζεται επακριβώς με όλα του τα βήματα στον παρακάτω πίνακα:

Algorithm 1: A Divide-and-Conquer (D&C) method for computing $\mathcal{D}(A^{-1})$

input : $A \in \mathbb{C}^{n \times n}$

output: $\mathcal{D}(A^{-1}) \in \mathbb{C}^{n \times n}$

- 1 **if** $n < \rho$ for a small $\rho \in \mathbb{N}$ or A does not satisfy (2.1) **then**
 - 2 \lfloor Compute A^{-1} explicitly, and determine $\mathcal{D}(A^{-1})$
 - 3 **else**
 - 4 Choose $m \in \mathbb{N}$ such that $m < n$
 - 5 Partition A as in Eq. (2.1)
 - 6 Determine $C = \begin{pmatrix} C_1 & \\ & C_2 \end{pmatrix}$ and E
 - 7 Compute $U := C^{-1}E$ and $R := I_q - E^T U$
 - 8 Compute $W := R^{-1}U^T$ and $M_1 := \mathcal{D}(UW)$
 - 9 Find $M_2^{(1)} := \mathcal{D}(C_1^{-1})$ by applying Algorithm 1 to C_1
 - 10 Find $M_2^{(2)} := \mathcal{D}(C_2^{-1})$ by applying Algorithm 1 to C_2
 - 11 Set $M_2 := \mathcal{D}(C^{-1}) = \begin{pmatrix} M_2^{(1)} & \\ & M_2^{(2)} \end{pmatrix}$
 - 12 \lfloor Set $\mathcal{D}(A^{-1}) := M_1 + M_2$
-

Σε προβλήματα 2 διαστάσεων ο πίνακας A είναι αυτός που δείξαμε προηγουμένως στην ανάλυση του 1^{ου} αλγόριθμου. Ο πίνακας A μπορεί πλέον να πάρει την μορφή

$$A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} = \begin{pmatrix} A_{11} & \\ & A_{22} \end{pmatrix} + \begin{pmatrix} & A_{12} \\ A_{21} & \end{pmatrix}$$

όπου $A_{11} \in \mathbb{C}^{m \times m}$ και $A_{22} \in \mathbb{C}^{(n-m) \times (n-m)}$ με την ιδιότητα $m = \phi n_x$ ώστε να ισχύει $0 < \phi < n_y$.

Επίσης έχουμε

$$A_{12} = A_{21}^T = \begin{pmatrix} \\ Q \end{pmatrix} = -E_1 E_2^T$$

με $E_1 \in \mathbb{C}^{m \times n_x}$ και $E_2 \in \mathbb{C}^{(n-m) \times n_x}$ να προσδιορίζονται ως

$$E_1 := \begin{pmatrix} \\ Q_1 \end{pmatrix}, \quad E_2 := \begin{pmatrix} Q_2 \\ \end{pmatrix}$$

όπου $Q, Q_1, Q_2 \in \mathbb{C}^{n_x \times n_x}$ είναι διαγώνιοι πίνακες που υπακούν

στον νόμο $Q_1 Q_2 = -Q$.

Έτσι πλέον ο πίνακας A μπορεί να πάρει την μορφή

$$A = \begin{pmatrix} A_{11} + E_1 E_1^T & \\ & A_{22} + E_2 E_2^T \end{pmatrix} - \begin{pmatrix} E_1 E_1^T & E_1 E_2^T \\ E_2 E_1^T & E_2 E_2^T \end{pmatrix}$$

που είναι ακριβώς η μορφή που θέλουμε. Πιο συγκεκριμένα τα C_1, C_2 είναι:

$$C_1 := A_{11} + E_1 E_1^T, \quad C_2 := A_{22} + E_2 E_2^T$$

4 ΚΩΔΙΚΑΣ MATLAB

4.1 Μεταβλητές και σταθερές

Στην παρούσα ενότητα θα δώσουμε τον κώδικά μας για τον 1^ο αλγόριθμο καθώς και σχόλια πάνω σε αυτόν.

Ξεκινάμε με τα τεχνικά χαρακτηριστικά του *testbend* που χρησιμοποιήσαμε (MIT25):

Αρχικά δηλώσαμε τις διαστάσεις του πλέγματος, ορίζοντας ως μήκος το N_x και ύψος το N_y . Από αυτές τις τιμές προκύπτει ο αριθμός των πλεγματικών θέσεων (200 στο παράδειγμά μας), επομένως συμπεραίνουμε ότι οι μπλοκ πίνακες που εμφανίζονται στις προσομοιώσεις μας είναι όλοι μεγέθους 200×200 ($N_x N_y * N_x N_y$).

$N_x = 20;$

$N_y = 10;$

Συνεχίζουμε δηλώνοντας τα χαρακτηριστικά του MIT25 καθώς και τις σταθερές που θα χρησιμοποιήσουμε στην πορεία. Αυτές τις τιμές τις αναφέραμε και πιο πάνω στα χαρακτηριστικά του MIT25 και εδώ δείχνουμε πως τις δηλώσαμε στον κώδικά μας.

$t_{ox} = 1.5e-9;$

$h_{plnk} = 4.13566733e-15;$

Οι παρακάτω τιμές είναι όλες εκφρασμένες σε μέτρα. Συγκεκριμένα οι τιμές που έχουμε δώσει στις διαστάσεις του τρανζίστορ είναι όλες παρμένες από το <http://www-mtl.mit.edu/researchgroups/Well/> του εργαστηρίου μικροσυστημάτων του MIT.

$L_p = 6.0e-8;$

$L_b = 12.0e-8;$

$L_g = 5.0e-8;$

Στη συνέχεια δηλώνουμε τις ενεργές μάζες (*effective masses*) του διοξειδίου του πυριτίου στο κανάλι. Παρατηρούμε ότι έχουμε και την ενεργή μάζα που απευθύνεται στην 3^η διάσταση (z). Παρόλο που έχουμε διευκρινίσει επισταμένως ότι η προσομοίωση γίνεται στις 2 διαστάσεις, πρέπει να αναφέρουμε και την ενεργή μάζα της 3^{ης} διάστασης καθαρά για τυπικούς λόγους.

$m_x = 0.5;$

$m_y = 0.5;$

$m_z = 0.5;$

Παρακάτω παραθέτουμε πάλι τις τιμές 2 σταθερών και συγκεκριμένα της διηλεκτρικής σταθεράς του πυριτίου καθώς και το γινόμενο της σταθεράς του Boltzman με την απόλυτη θερμοκρασία εκφρασμένη σε τιμές ενέργειας (eV). Τέλος δίνουμε μια τυχαία τιμή για την ενέργεια που παρατηρείται κατά την διάρκεια της κβαντικής μεταφοράς.

$e_{Si} = 11.9;$

$kT = 0.02585;$

$E = 1e-15;$

Πληροφοριακά, χωρίς να μας χρησιμεύουν σε κάτι στην διάρκεια του κώδικα δίνουμε τιμές για τα ρεύματα και τις τάσεις στην υποδοχή και την πύλη αντίστοιχα.

$I_d = 10;$

$I_g = 5;$

$V_d = 1e-20;$

$V_g = 5e-21;$

Αυτές οι τιμές στην συγκεκριμένη εργασία δεν χρησιμοποιήθηκαν πουθενά αλλά τις αναφέρουμε σε περίπτωση που κάποιος θελήσει να κάνει κάποιες περαιτέρω προσομοιώσεις.

Επανερχόμενοι στο MIT25, δηλώνουμε το μήκος και το ύψος της βάσης του τρανζίστορ με τιμές όπως τις πήραμε από το αρχείο **sh25.doping** της ιστοσελίδας που προαναφέραμε. Στο συγκεκριμένο αρχείο βλέπουμε ότι οι

διαστάσεις του τρανζίστορ όσον αφορά το μήκος του, ξεκινάνε από την τιμή $-4.75e-8$ μέχρι την τιμή $4.75e-8$. Από αυτό συμπεραίνουμε ότι το τρανζίστορ «χωρίζεται» στην μέση και οι επιλύσεις των εξισώσεων NEGF και Poisson γίνονται σε διαστάσεις από $-L_y/2$ μέχρι $L_y/2$, ενώ αντιθέτως η διάσταση που μετράει το ύψος πηγαίνει από το 0 μέχρι την τιμή $1.2e-7$.

length_y = 9.5e-8;
length_x = 1.2e-7;

Στο αρχείο **sh25.doping** μπορούμε να συμπεράνουμε ότι η ερευνητική ομάδα που το έγραψε έχει χρησιμοποιήσει συγκεκριμένο μέγεθος βήματος ώστε να βρει τα πλεγματικά σημεία που την ενδιαφέρουν. Εμείς δεν χρησιμοποιήσαμε συγκεκριμένο αριθμό, αλλά αυτός υπολογίζεται δυναμικά ανάλογα με τον αριθμό των πλεγματικών στοιχείων που έχουμε βάλει και αποθηκεύεται στις μεταβλητές **step_h**, **step_v**, όπου δηλώνουμε το βήμα για την απόσταση των οριζόντιων και κάθετων στοιχείων αντίστοιχα.

step_h = length_y/Ny;
step_v = length_x/Nx;

Τέλος όσον αφορά τις συντεταγμένες των στοιχείων στο πλέγμα τις υπολογίζουμε και αυτές δυναμικά με χρήση των μεταβλητών που υπολογίσαμε προηγουμένως καθώς και των διαστάσεων μήκους, ύψους του τρανζίστορ που έχουμε.

x_coord = 0:step_h:length_x;
y_coord = -length_y/2:step_v:length_y/2;

Μια ακόμα μεταβλητή που χρειαζόμαστε κυρίως στον υπολογισμό των ενεργειών είναι και ο *κυματοαριθμός* που εκφράζει το ηλεκτρομαγνητικό κύμα μεταφοράς στο κανάλι.

```
kz = (2*E^2)/h_plnk;
```

4.2 Υπολογισμός πινάκων και τελικά αποτελέσματα

Σε αυτή την ενότητα θα παρουσιάσουμε τα μέρη του κώδικα που υπολογίζουν τους πίνακες τους οποίους χρησιμοποιήσαμε στην προσομοίωσή μας.

Αρχικά υπολογίσαμε τον πίνακα λ^{-1} . Ο κώδικας είναι:

```
L_inv = sparse(0,0);  
for j = 1:Nx  
  for i = 2:Ny  
  
    L_inv(((j-1)*Ny + i-1),((j-1)*Ny + i-1)) =  
    ((x_coord(i+1) - x_coord(i-1))*(y_coord(i+1) -  
    y_coord(i-1)))/4);  
  
    L_inv(((j-1)*Ny + Ny),((j-1)*Ny + Ny)) =  
    ((x_coord(Ny+1) - x_coord(Ny-1))*(y_coord(Ny+1) -  
    y_coord(Ny-1)))/4);  
  
  end  
end
```

Αρχικά δηλώσαμε έναν αραιό (sparse) πίνακα, κάτι το οποίο θα γίνεται πάντα από δω και πέρα με λίγες εξαιρέσεις.

Τα στοιχεία των πινάκων δηλώνονται ως εξής: χρησιμοποιούμε τον N_x για να δηλώσουμε το πλήθος των blocks, δηλαδή πόσους πίνακες θα έχει ο

μπλοκ πίνακας που φτιάχνουμε σε μια διάσταση και με N_y δηλώνουμε το πλήθος των στοιχείων του κάθε μπλοκ σε μια διάσταση. Στο παράδειγμά μας $N_x=20$ επομένως ο μπλοκ πίνακάς μας θα έχει διαστάσεις $20*20$ μπλοκ, ενώ $N_y=10$ επομένως το κάθε μπλοκ θα είναι μεγέθους $10*10$.

Έχοντας ορίσει τον τρόπο που εισάγουμε στοιχεία στους πίνακες και δεδομένου ότι οι πίνακες είναι τετραγωνικοί, βγαίνει το συμπέρασμα ότι και η διαγώνιος τόσο του μπλοκ πίνακα όσο και των επιμέρους πινάκων είναι μεγέθους 20 και 10 αντίστοιχα. Δεν πρέπει να ξεχνάμε ότι τα στοιχεία του πλέγματος του καναλιού αποτελούν τα διαγώνια στοιχεία των πινάκων μας.

Στην συνέχεια περιγράψουμε την κατασκευή του πίνακα T.

Εδώ πρέπει να επισημάνουμε ότι ο πίνακας T είναι φτιαγμένος σε 3 μέρη. Το 1^ο μέρος αφορά τα μπλοκ της κύριας διαγωνίου (T1), ενώ τα άλλα 2 μέρη αφορούν την κατασκευή της άνω (T_4) και κάτω (T_5) διαγωνίου.

Αυτό το κάναμε για να διευκολυνθούμε στις πράξεις στην συνέχεια για τον ορισμό του πίνακα A, διότι σε κανένα σημείο δεν χρειάζεται ολόκληρος ο πίνακας T παρά μόνο κάποια από τις διαγώνιους του.

Για την κατασκευή του T1 έχουμε τον παρακάτω κώδικα:

```
T1 = sparse(0,0);
```

```
for j = 1:Nx
```

```
  for i = 2:Ny
```

```
    T1(((j-1)*Ny + i-1),((j-1)*Ny + i-1)) = 0;
```

```
    T1(((j-1)*Ny + i-1),(j-1)*Ny + (i)) =
```

```
    (h_plnk^2/2*m)*(2/(x_coord(i+1) - x_coord(i-
```

```
    1)))*(1/abs((x_coord(i+1) - x_coord(i)))));
```

```
T1(((j-1)*Ny + (i)),((j-1)*Ny + i-1)) =  
(h_plnk^2/2*m)*(2/(x_coord(i+1) - x_coord(i-  
1)))*(1/abs((x_coord(i-1) - x_coord(i))));
```

```
T1(((j-1)*Ny + Ny-1),((j-1)*Ny + Ny-1)) = 0;
```

```
T1(((j-1)*Ny + Ny-1),(j-1)*Ny + Ny) =  
(h_plnk^2/2*m)*(2/(x_coord(Ny+1) - x_coord(Ny-  
1)))*(1/abs((x_coord(Ny+1) - x_coord(Ny))));
```

```
T1(((j-1)*Ny + Ny),((j-1)*Ny + Ny-1)) =  
(h_plnk^2/2*m)*(2/(x_coord(Ny+1) - x_coord(Ny-  
1)))*(1/abs((x_coord(Ny-1) - x_coord(Ny))));
```

```
end
```

```
end
```

ενώ για τους T_4, T_5 έχουμε:

```
T_4 = sparse(0,0);
```

```
T_5 = sparse(0,0);
```

```
for j = 1:Nx-1
```

```
    for i = 2:Ny
```

```
        T_4(((j-1)*Ny + i-1),((j-1)*Ny + i-1)) =  
        (h_plnk^2/2*m)*(2/(y_coord(i+1) - y_coord(i-  
1)))*(1/abs((y_coord(i+1) - y_coord(i))));
```

```
        T_4(((j-1)*Ny + Ny),((j-1)*Ny + Ny)) =  
        (h_plnk^2/2*m)*(2/(y_coord(Ny+1) - y_coord(Ny-  
1)))*(1/abs((y_coord(Ny+1) - y_coord(Ny))));
```

```
    end
```

```
end
```

```

for j = 1:Nx-1
  for i = 2:Ny
    T_5(((j-1)*Ny + i-1),((j-1)*Ny + i-1)) =
    (h_plnk^2/2*m)*(2/(y_coord(i+1) - y_coord(i-
    1)))*(1/abs((y_coord(i-1) - y_coord(i)))));

    T_5(((j-1)*Ny + Ny),((j-1)*Ny + Ny)) =
    (h_plnk^2/2*m)*(2/(y_coord(Ny+1) - y_coord(Ny-
    1)))*(1/abs((y_coord(Ny-1) - y_coord(Ny)))));
  end
end

```

Παρατηρούμε ότι ενώ ο T1 κατασκευάζεται και για τα 20 μπλοκ στοιχεία του, οι T_4, T_5 κατασκευάζονται για 19 μπλοκ κάτι το οποίο είναι απολύτως φυσιολογικό εφόσον αναφέρονται στις άνω και κάτω διαγώνιους.

Οι τιμές των *self-energies* όπως τονίσαμε στην 2^η ενότητα είναι μηδενικές. Παρ' όλα αυτά υποχρεούμαστε να συμπεριλάβουμε στον κώδικά μας τους πίνακες που θα έχουν τις τιμές τους, διότι σε μελλοντικές προσομοιώσεις έχοντας γνωστές τις τιμές τους, θα μπορούμε να τις χρησιμοποιήσουμε για την εξαγωγή πιο ρεαλιστικών αποτελεσμάτων.

Οι πίνακες για τις *self-energies* είναι:

```

Sigma_r_S = sparse(0,0);
Sigma_r_D = sparse(0,0);
Sigma_r_P = sparse(0,0);

```

```
Sigma_r = sparse(0,0);
```

```
for j = 1:Nx  
  for i = 1:Ny  
    if(j == 1)  
      Sigma_r_S((j-1)*Ny + i, (j-1)*Ny + i) = 0;  
    else  
      Sigma_r_S((j-1)*Ny + i, (j-1)*Ny + i) = 0;  
    end  
  end  
end
```

```
for j = 1:Nx  
  for i = 1:Ny  
    if(j == Nx)  
      Sigma_r_D((j-1)*Ny + i, (j-1)*Ny + i) = 0;  
    else  
      Sigma_r_D((j-1)*Ny + i, (j-1)*Ny + i) = 0;  
    end  
  end  
end
```

```
for j = 1:Nx  
  for i = 1:Ny  
    if(i == 1)  
      Sigma_r_P((j-1)*Ny + i, (j-1)*Ny + i) = 0;  
    else  
      Sigma_r_P((j-1)*Ny + i, (j-1)*Ny + i) = 0;  
    end  
  end  
end
```

```

for j = 1:Nx
  for i = 2:Ny
    Sigma_r((j-1)*Ny + i-1,(j-1)*Ny + i-1) = 0;
    Sigma_r((j-1)*Ny + i-1,(j-1)*Ny + i) = 0;
    Sigma_r((j-1)*Ny + i,(j-1)*Ny + i-1) = 0;
    Sigma_r((j-1)*Ny + Ny,(j-1)*Ny + Ny) = 0;
  end
end

```

Οι πίνακες που περιγράφουν τις *self-energies* στον κώδικά μας αφορούν τις *self-energies* που οφείλονται στις επαφές του τρανζίστορ μας με την πηγή, την υποδοχή και την πύλη καθώς και αυτές που οφείλονται στις διεγέρσεις μέσα στο τρανζίστορ. Όπως αναφέραμε και στην 2^η ενότητα στην παράγραφο για τις *self-energies*, αυτές είναι μη μηδενικές μόνο στο μπλοκ πίνακα που αντιστοιχεί στις πλεγματικές θέσεις του καναλιού που άπτεται στην συγκεκριμένη περιοχή. Επομένως ο μπλοκ πίνακας που αφορά την *self-energy* της πηγής θα έχει μη μηδενικά στοιχεία στο 1^ο μπλοκ, ο μπλοκ πίνακας της υποδοχής θα έχει μη μηδενικά στοιχεία στο τελευταίο (20^ο στο παράδειγμά μας) μπλοκ και ο πίνακας που αφορά την *self-energy* της επαφής του τρανζίστορ με το οξείδιο της πύλης θα έχει μη μηδενικά στοιχεία στα αρχικά σημεία κάθε μπλοκ. Τέλος όσον αφορά τον πίνακα για την *self-energy* που οφείλεται στις διεγέρσεις στο εσωτερικό του τρανζίστορ, παρατηρούμε ότι ο πίνακας είναι μπλοκ διαγώνιος.

Για την $\Sigma^<$ θα πούμε στην συνέχεια του κώδικα.

Έπειτα ακολουθεί ο πίνακας που περιγράφει την τιμή των τάσεων σε κάθε πλεγματικό στοιχείο. Στο παράδειγμά μας οι τιμές είναι τυχαίες, αλλά είναι φανερό ότι μπορούμε να δώσουμε όποια τιμή θέλουμε ανάλογα με την τάση που θέλουμε.

```
V = zeros(0,0);
```

```
for j = 1:Nx  
  for i = 1:Ny  
    V(i,j) = rand(1);  
  end  
end
```

Στην προκειμένη περίπτωση ο πίνακας των τάσεων ορίζεται ως μηδενικός, επειδή είναι πλήρης.

Η ενέργεια E' που χρησιμοποιούμε στον πίνακα A' δίνεται από τον εξής τύπο:

```
E_new = E - ((h_plnk^2)*(kz^2))/(2*mz);
```

Πλέον προχωράμε στην κατασκευή του πίνακα A' . Με το `KronD` συμβολίζουμε την συνάρτηση που υπολογίζει το δ του Kronecker.

```
A = sparse(0,0);
```

```
for j = 1:Nx  
  for i = 2:Ny  
    A(((j-1)*Ny + i-1),((j-1)*Ny + i-1)) = E_new - V(i,j) -  
    T1(i+1,i) - T1(i-1,i) - T_4(i,i) - T_5(i,i) -  
    Sigma_r_S(i,i)*KronD(j,1) -  
    Sigma_r_D(i,i)*KronD(j,Ny) -  
    Sigma_r_P(j,j)*KronD(i,1) - Sigma_r(i,j);  
    A(((j-1)*Ny + i-1),(j-1)*Ny + (i)) = T1(i+1,i) -  
    Sigma_r_S(i+1,i)*KronD(j,1) -  
    Sigma_r_D(i+1,i)*KronD(j,Ny) - Sigma_r(i,j);
```


A(((j-1)*Ny + (i)),((j-1)*Ny + i-1)) = T1(i-1,i)- Sigma_r(i,j);

**A(((j-1)*Ny + Ny),((j-1)*Ny + Ny)) = E_new - V(Ny,Nx)
- T1(Ny+1,i) - T1(Ny-1,i) - T_4(Ny,Ny) - T_5(Ny,Ny) -
Sigma_r_S(i,i)*KronD(j,1) -
Sigma_r_D(i,i)*KronD(j,Ny) -
Sigma_r_P(j,j)*KronD(i,1) - Sigma_r(i,j);**

**A(((j-1)*Ny + Ny-1),(j-1)*Ny + (Ny)) = T1(Ny+1,i) -
Sigma_r_S(i+1,i)*KronD(j,1) -
Sigma_r_D(i+1,i)*KronD(j,Ny) - Sigma_r(i,j);**

**A(((j-1)*Ny + (Ny)),((j-1)*Ny + Ny-1)) = T1(Ny-1,i)-
Sigma_r(i,j);**

**end
end**

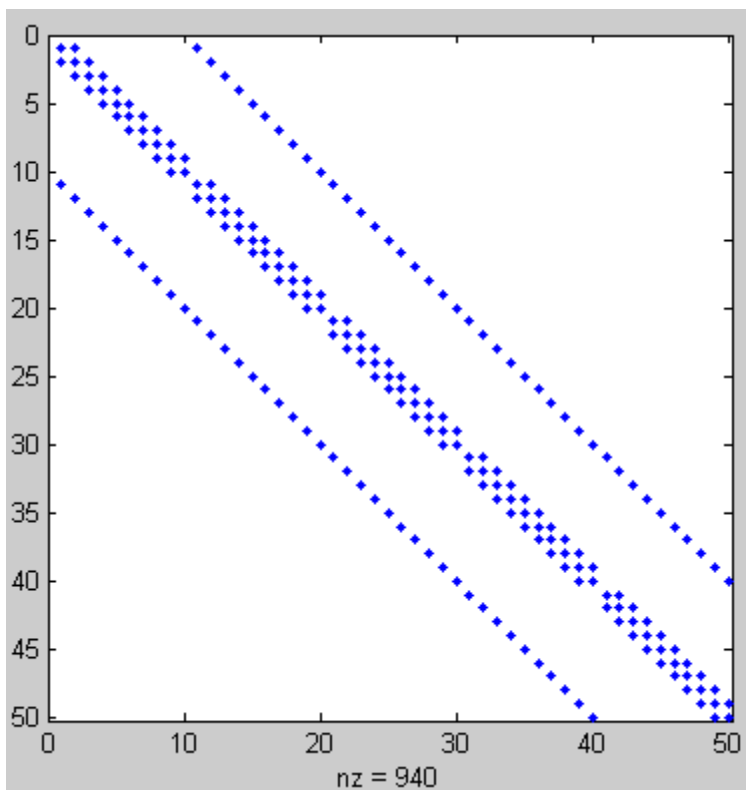
**for j = 1:Nx-1
for i = 1:Ny
A(((j-1)*Ny + i),(j*Ny + i)) = T_4(i,i) -
Sigma_r_P(j,j+1)*KronD(i,1);
end
end**

**for j = 2:Nx
for i = 1:Ny
A(((j-1)*Ny + i),((j-2)*Ny + i)) = T_5(i,i) - Sigma_r_P(j,j-
1)*KronD(i,1);**

end
end

Αρχικά δηλώνουμε τον πίνακα όπως είπαμε (sparse) και προχωράμε στην κατασκευή του. Χρησιμοποιούμε 3 δομές επανάληψης. Η 1^η είναι για την εισαγωγή στοιχείων στα μπλοκ της κυρίας διαγωνίου του πίνακα, όπου τα μπλοκ αυτά με την σειρά τους είναι τριδιαγώνιοι πίνακες. Στις επόμενες 2 δομές επανάληψης κατασκευάζουμε τις άνω και κάτω διαγώνιους αντίστοιχα. Η δεικτοδότηση των στοιχείων έχει γίνει έτσι ώστε να μην υπάρχουν προβλήματα όπως εμφάνιση θέσεων εκτός πίνακα.

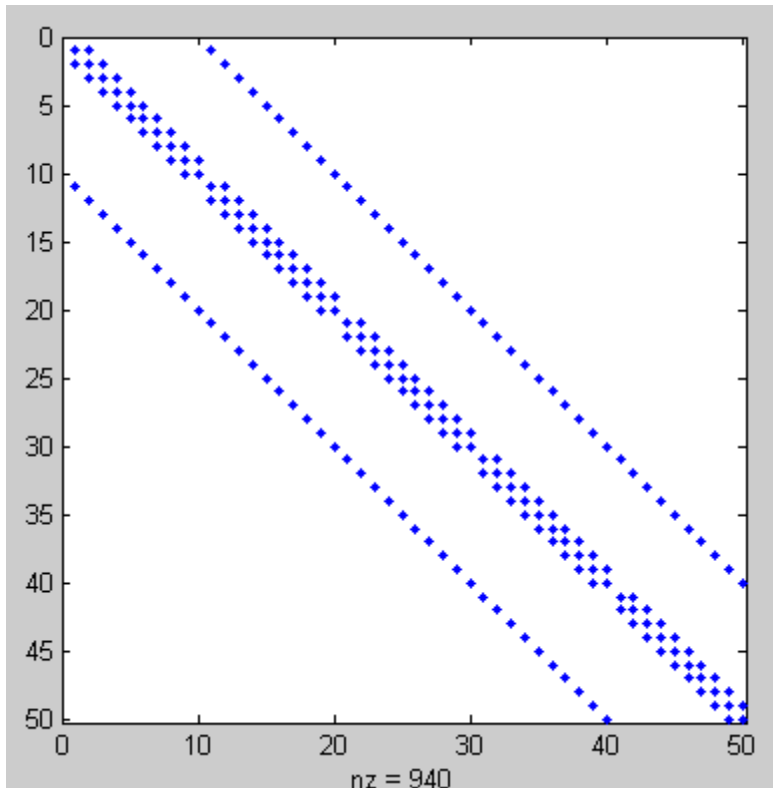
Παρακάτω βλέπουμε μια σχηματική απεικόνιση από ένα στιγμιότυπο του πίνακα A' για τα πρώτα 5 μπλοκ του πίνακα



Για να προχωρήσουμε στον υπολογισμό της $AG^T = I$ πρώτα πρέπει να υπολογίσουμε τον πίνακα A (ο οποίος είναι ο A του κώδικά μας πολλαπλασιασμένος με τον αντίστροφο του λ), τον οποίο για διευκόλυνση τον ονομάσαμε K .

$$\mathbf{K} = \mathbf{L}_{inv} * \mathbf{A};$$

Παρουσιάζουμε και ένα στιγμιότυπο του πίνακα K ο οποίος είναι επίσης μπλοκ τριδιαγώνιος, όμοιος με τον A .



Για να προχωρήσουμε στην υλοποίηση του αλγόριθμού μας πρέπει αρχικά να δηλώσουμε κάποιους πίνακες που θα μας χρησιμεύσουν στην πορεία. Οι πίνακες αυτοί, καθώς και κάποιες απαραίτητες μεταβλητές παρατίθενται στην επόμενη λίστα:

```
A1 = sparse(0,0);  
A1_inv = sparse(0,0);  
B1 = sparse(0,0);  
R1 = sparse(0,0);  
Bk_rev = sparse(0,0);  
Ak = sparse(0,0);  
Bk = sparse(0,0);  
R_table = sparse(0,0);
```

```
temp = zeros(Ny,Ny);  
B_k = sparse(0,0);  
Anx = sparse(0,0);  
SNy = sparse(0,0);  
B_l = sparse(0,0);  
B_l_rev = sparse(0,0);  
A_k = sparse(0,0);  
S_table = sparse(0,0);  
temp2 = zeros(Ny,Ny);  
A2k = sparse(0,0);  
B2k = sparse(0,0);  
S_table_rec = sparse(0,0);  
B2k_rev = sparse(0,0);  
D = sparse(0,0);  
D_inv = sparse(0,0);  
temp3 = sparse(0,0);  
D1 = sparse(0,0);  
ANy = sparse(0,0);  
ANy_rev = sparse(0,0);  
BNy = sparse(0,0);  
DNy = sparse(0,0);  
temp4 = sparse(0,0);  
temp5 = sparse(0,0);  
A1_inv = sparse(0,0);  
K_inv = sparse(0,0);  
G_r = sparse(0,0);  
k = 1;  
I2 = Nx-1;
```

Θα εξηγήσουμε στην πορεία την χρήση καθενός.

Πλέον έχουμε μπει στο κυρίως σώμα του αλγόριθμου, οπότε θα δούμε την υλοποίηση κάθε πίνακα ξεχωριστά.

Αρχικά θα παραθέσουμε την διαδικασία σύμφωνα με την οποία υπολογίζουμε τον πίνακα R_i .

Πρώτα κατασκευάζουμε τους πίνακες A_1 και B_1 , οι οποίοι είναι τα πρώτα μπλοκ των άνω και κάτω διαγώνιων του πίνακα K .

```
for i = 1:Ny
    A1(i,i) = K(i,i);
end

for i = 1:Ny-1
    A1(i,i+1) = K(i,i+1);
    A1(i+1,i) = K(i+1,i);
end

A1_inv = inv(A1);

for i = 1:Ny
    B1(i,i) = -K(i,Ny+i);
end
```

έπειτα υπολογίζουμε τον R_1 .

```
R1 = A1_inv.*B1;
```

Στην συνέχεια με επαναληπτική διαδικασία υπολογίζουμε τους υπόλοιπους R_i πάντα με βάση τον αλγόριθμο που περιγράψαμε στην 3^η ενότητα.

```
for l = 2:Nx-1
```

```
for i = 1:Ny  
    Bk_rev(i,i) = -K((l-1)*Ny+i,i);  
end
```

```
for i = 1:Ny  
    Ak(i,i) = K((l-1)*Ny+i,(l-1)*Ny+i);  
end
```

```
for i = 1:Ny-1  
    Ak(i+1,i) = K((l-1)*Ny+i+1,(l-1)*Ny+i);  
    Ak(i,i+1) = K((l-1)*Ny+i,(l-1)*Ny+i+1);  
end
```

```
for i = 1:Ny  
    Bk(i,i) = -K((l-1)*Ny+i,l*Ny+i);  
end
```

```
R_table = inv(Ak - Bk_rev.*R1).*Bk;  
temp = R_table;  
R1 = temp;  
end
```

Έπειτα προχωράμε μέσω μιας επαναληπτικής διαδικασίας στον υπολογισμό του πίνακα S . Αντίστοιχα με πριν υπολογίζουμε αρχικά το S_{Ny} και στην συνέχεια και τα υπόλοιπα μπλοκ S_i .

```
for l = Nx-1:-1:1
```

if(l == Nx-1)

for i = 1:Ny

B_k(i,i) = K((Nx-2)*Ny+i,(Nx-1)*Ny+i);

end

for i = 1:Ny

Anx(i,i) = K((Nx-1)*Ny+i,(Nx-1)*Ny+i);

end

for i = 1:Ny-1

Anx(i+1,i) = K((Nx-1)*Ny+i+1,(Nx-1)*Ny+i);

Anx(i,i+1) = K((Nx-1)*Ny+i,(Nx-1)*Ny+i+1);

end

SNy = B_k.*Anx;

else

for i = 1:Ny

B_l(i,i) = -K((l-1)*Ny+i,l*Ny+i);

end

for i = 1:Ny

A_k(i,i) = K(l*Ny+i,l*Ny+i);

end

```

for i = 1:Ny-1
    A_k(i+1,i) = K(l*Ny+i+1,k*Ny+i);
    A_k(i,i+1) = K(l*Ny+i,l*Ny+i+1);
end

```

```

for i = 1:Ny
    B_l_rev(i,i) = -K((l+1)*Ny+i,l*Ny+i);
end

```

```

S_table = B_l.*(inv(A_k - SNy.*B_l_rev));
temp2 = S_table;
SNy = temp2;

```

```

S_table_rec = S_func(l,B_l,A_k,SNy,B_l_rev,Nx);
end
end

```

Όπως θα δούμε προς το τέλος έχουμε 2 υπολογισμούς για τους πίνακες S_i , μία που τα υπολογίζει από N_y μέχρι 2 και μια αναδρομική που τα υπολογίζει από 2 μέχρι N_y . Αυτό το κάνουμε επειδή για να υπολογίσουμε τους D_i , όπως φαίνεται και από τον αλγόριθμο στην 3^η ενότητα, πρέπει να έχουμε τους πίνακες S_i να υπολογίζονται και ορθά και ανάποδα. Αυτό δεν μπορεί να γίνει μόνο με μια δομή επανάληψης οπότε χρησιμοποιούμε και την αναδρομή που εκτελείται με την συνάρτηση **S_func**.

Παραθέτουμε τον κώδικα της εν λόγω συνάρτησης:

```

function S = S_func(l,B_l,A_k,SNy,B_l_rev,Nx)
%Mathematical formulation of Purdue e-pubs, 6-3-2007

```



```
%S_func calculates S matrices that are necessary for  
making K_inv  
%recursion for calculation of D(i+1)
```

```
if(l == Nx-1)  
  S = SNy;  
else  
  S = B_l.*(inv(A_k - SNy.*B_l_rev));  
end
```

Όπως βλέπουμε και από τα σχόλια, οι υπολογισμοί της αναδρομής είναι για τον υπολογισμό των D_{i+1} .

Πλέον μπορούμε να υπολογίσουμε τον D_1 , ο οποίος δίνεται από την παρακάτω γραμμή κώδικα. Μαζί δείχνουμε πως ο D_1 εισχωρεί σε έναν μπλοκ διαγώνιο πίνακα, έτσι ώστε να μπορέσουμε να απεικονίσουμε στο τέλος τον μπλοκ διαγώνιο πίνακα που θα απεικονίζει τον G_r .

```
D1 = eye(size(A1-B1.*S_table'))/(A1-B1.*S_table');  
K_inv = blkdiag(K_inv,D1);
```

Στη συνέχεια βλέπουμε πως υπολογίζονται τα υπόλοιπα D_{i+1} στοιχεία-πίνακες μέσω μιας επαναληπτικής διαδικασίας με βάση πάντα τον αλγόριθμό μας.

```
for k = 1:Nx-2
```

```
  for i = 1:Ny  
    A2k(i,i) = K((k+1)*Ny+i,(k+1)*Ny+i);
```

end

for i = 1:Ny-1

A2k(i+1,i) = K((k+1)*Ny+i+1,(k+1)*Ny+i);

A2k(i,i+1) = K((k+1)*Ny+i,(k+1)*Ny+i+1);

end

for i = 1:Ny

B2k(i,i) = -K(k*Ny+i,(k+1)*Ny+i);

end

I = eye(Ny);

for i = 1:Ny

B2k_rev(i,i) = -K((k+1)*Ny+i,k*Ny+i);

end

**D_inv = eye(size(A2k -
B2k.*S_func(k+1,B_I,A_k,SNy,B_I_rev,Nx)))/(A2k -
B2k.*S_func(k+1,B_I,A_k,SNy,B_I_rev,Nx));**

Αφού έχουμε κάνει τους απαραίτητους υπολογισμούς,
υπολογίζουμε τον D και τον τοποθετούμε με την σειρά του στον
μπλοκ διαγώνιο πίνακα για τον G_r .

```

D = D_inv*(I +
B2k_rev*D1*S_func(k,B_I,A_k,SNy,B_I_rev,Nx));
K_inv = blkdiag(K_inv,D);

```

```

temp3 = D;
D1 = sparse(0,0);
D1 = temp3;

```

```

end

```

Τέλος θα δώσουμε τον κώδικα κατασκευής της τελευταίας συνάρτησης που έχουμε στον αλγόριθμό μας για τον υπολογισμό του D_{Ny} .

```

for i = 1:Ny
    ANy(i,i) = K((Nx-1)*Ny+i,(Nx-1)*Ny+i);
end

```

```

for i = 1:Ny-1
    ANy(i+1,i) = K((Nx-1)*Ny+i+1,(Nx-1)*Ny+i);
    ANy(i,i+1) = K((Nx-1)*Ny+i,(Nx-1)*Ny+i+1);
end

```

```

ANy_rev = eye(size(ANy))/ANy;

```

```

for i = 1:Ny
    BNy(i,i) = -K((Nx-1)*Ny+i,(Nx-2)*Ny+i);
end

```

Αφού έχουμε ακολουθήσει παρόμοια διαδικασία με τα προηγούμενα, υπολογίζουμε και το τελευταίο μπλοκ D_{Ny} και το τοποθετούμε με την σειρά του στον μπλοκ διαγώνιο πίνακα K_inv .

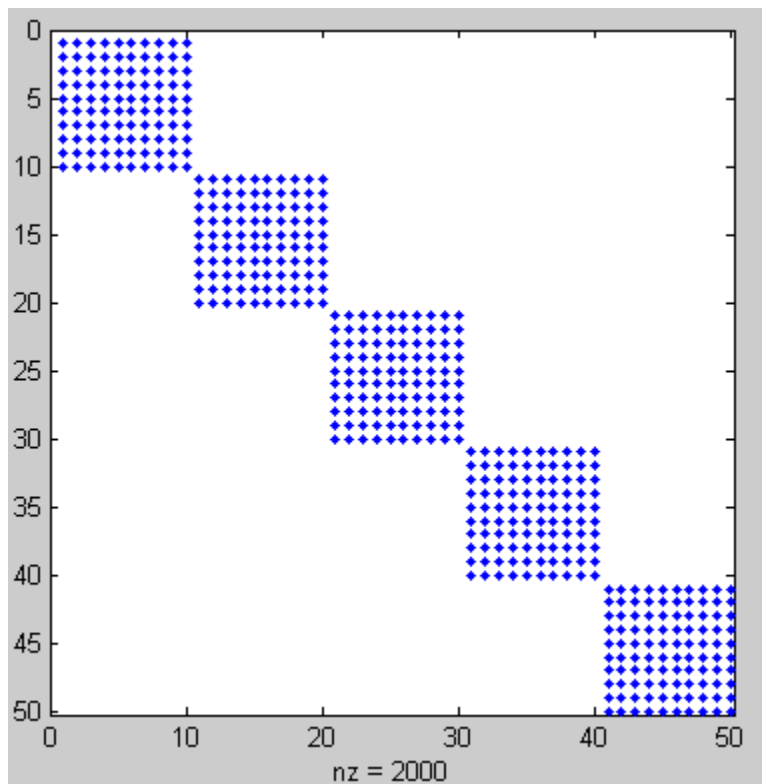
```
DNy = ANy_rev*(I + BNy*D*SNy);  
K_inv = blkdiag(K_inv,DNy);
```

Στην συνέχεια ο κώδικάς μας εκτελεί τις προσομοιώσεις των οποίων τα αποτελέσματα δώσαμε στην 4^η ενότητα.

Αρχικά ονομάζουμε τον **K_inv** σε **G_r** για να είμαστε συνεπείς με την ονοματοδοσία των εξισώσεων Green.

```
G_r = K_inv;
```

του οποίου ένα στιγμιότυπο είναι αυτό που ακολουθεί

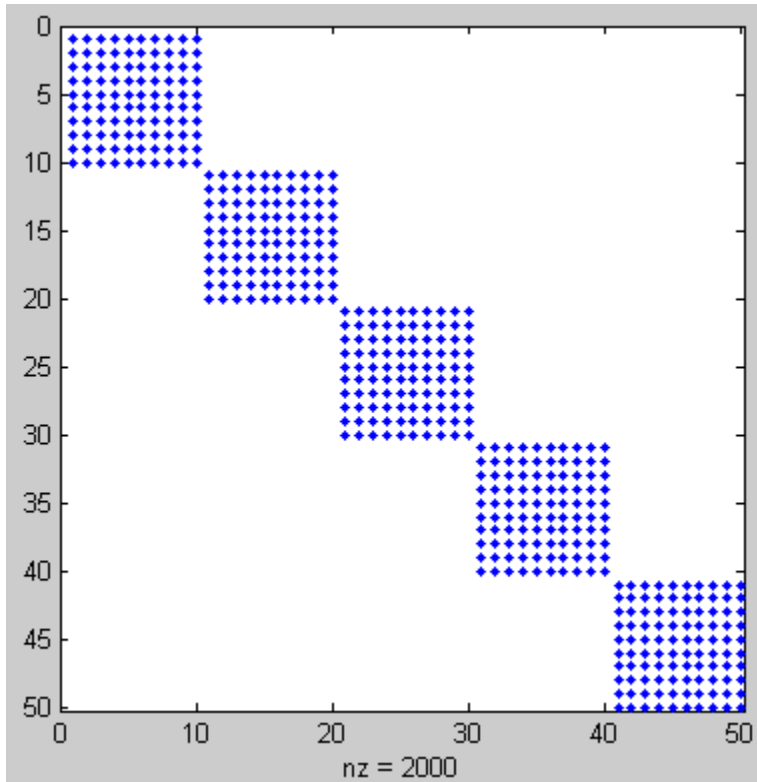


Παρατηρούμε όπως είναι λογικό ότι αυτά τα μπλοκ είναι πλήρη στοιχείων, εφόσον γνωρίζουμε ότι γενικά ο αντίστροφος πίνακας ενός μπλοκ τριδιαγώνιου είναι κι αυτός πλήρης.

Προχωράμε με τον υπολογισμό του G^α .

```
G_a = sparse(0,0);  
G_a = G_r';
```

Ένα στιγμιότυπο για την G^α είναι αυτό που ακολουθεί



Όπως δείξαμε και στην 4^η ενότητα ο G^α είναι ο ανάστροφος πίνακας του G_r .

Από δω και πέρα μπαίνουμε στις τυχαίες τιμές που δώσαμε στην $\Sigma^<$ για να κάνουμε τις απαραίτητες προσομοιώσεις. Από τον αλγόριθμο του A. Svizhenko ξέρουμε ότι η $\Sigma^<$ είναι μπλοκ τριδιαγώνιος πίνακας και σαν τέτοιον τον φτιάχνουμε γεμίζοντάς τον με μια τυχαία τιμή έστω $1e-5$ ώστε να μην υπάρχει πρόβλημα συμμετρίας.

```
S_less = sparse(0,0);
```

```

for j = 1:Nx
  for i = 2:Ny
    S_less(((j-1)*Ny + i-1),((j-1)*Ny + i-1)) = 1e-5;
    S_less(((j-1)*Ny + i-1),(j-1)*Ny + (i)) = 1e-5;
    S_less(((j-1)*Ny + (i)),((j-1)*Ny + i-1)) = 1e-5;
    S_less(((j-1)*Ny + Ny),((j-1)*Ny + Ny)) = 1e-5;
    S_less(((j-1)*Ny + Ny-1),(j-1)*Ny + (Ny)) = 1e-5;
    S_less(((j-1)*Ny + (Ny)),((j-1)*Ny + Ny-1)) = 1e-5;
  end
end

```

```

for j = 1:Nx-1
  for i = 1:Ny
    S_less(((j-1)*Ny + i),(j*Ny + i)) = 1e-5;
  end
end

```

```

for j = 2:Nx
  for i = 1:Ny
    S_less(((j-1)*Ny + i),((j-2)*Ny + i)) = 1e-5;
  end
end

```

Για να προχωρήσουμε στην όμως στις προσομοιώσεις πρέπει να υπολογίσουμε και τον $G^<$, τον οποίο υπολογίζουμε αφού πρώτα πολλαπλασιάσουμε την $\Sigma^<$ με τον αντίστροφο πίνακα του λ

```

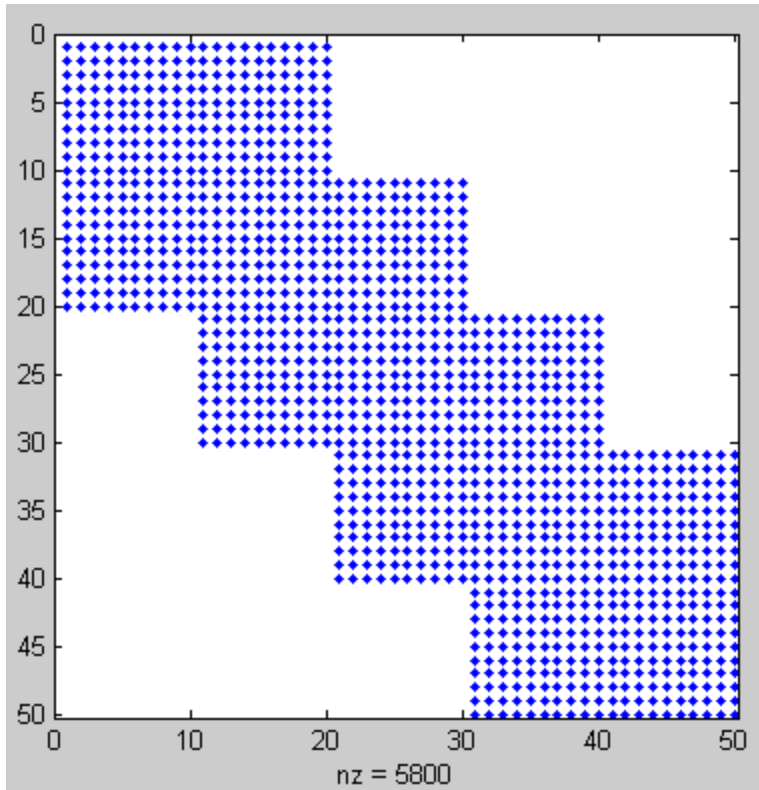
S_less2 = sparse(0,0);
S_less2 = L_inv*S_less;

```

και στην συνέχεια τον $G^<$.

```
G_less = sparse(0,0);  
G_less = G_r*S_less2*G_a;
```

Στιγμιότυπο του $G^<$ ακολουθεί



Παραθέτουμε στην συνέχεια τον κώδικα για τον υπολογισμό της πυκνότητας φορτίου

```
i=sqrt(-1);  
charge_density = sparse(0,0);  
charge_density = -i*G_less;
```

όπου πρώτα εξασφαλίζουμε ότι ο i είναι φανταστικός αριθμός.

Το τελευταίο μέρος του κώδικα αφορά τον υπολογισμό της πυκνότητας ρεύματος

Αρχικά δηλώνουμε τους πίνακες και τις μεταβλητές που θα χρησιμοποιήσουμε και έπειτα με μια απλή επαναληπτική διαδικασία υπολογίζουμε την πυκνότητα φορτίου μεταξύ όλων των διαγώνιων μπλοκ

```
J_scalar = 0;
```

```
J1 = 0;
```

```
J2 = 0;
```

```
J = sparse(0,0); %in A/m^2
```

```
for j = 1:Nx-1
```

```
  for i = 2:Ny
```

```
    J1 = (2*e_Si/h_plnk)*(T_4(((j-1)*Ny + i-1),((j-1)*Ny + i-1))*G_less(((j-1)*Ny + i-1),(j*Ny + i-1)) - T_5(((j-1)*Ny + i-1),((j-1)*Ny + i-1))*G_less((j*Ny + i-1),((j-1)*Ny + i-1))));
```

```
    J2 = (2*e_Si/h_plnk)*(T_4(((j-1)*Ny + Ny),((j-1)*Ny + Ny))*G_less(((j-1)*Ny + Ny),(j*Ny + Ny)) - T_5(((j-1)*Ny + Ny),((j-1)*Ny + Ny))*G_less((j*Ny + Ny),((j-1)*Ny + Ny))));
```

```
    J_scalar = J1 + J2;
```

```
  end
```

```
  J(j,1) = J_scalar;
```

```
end
```

Για αριθμητικά αποτελέσματα των προσομοιώσεων ανατρέξτε στην 4^η ενότητα.

Εδώ τελειώσαμε με την λεπτομερή επεξήγηση του κώδικα.

5 Αποτελέσματα προσομοιώσεων

Ο κώδικας που αναπτύξαμε υλοποιήθηκε σε γλώσσα προγραμματισμού MATLAB. Η έκδοση MATLAB που χρησιμοποιήσαμε είναι η R2009b.

Επίσης ο κώδικάς μας δοκιμάστηκε και στο open-source πρόγραμμα gnucOctave του Linux, για περαιτέρω σιγουριά κυρίως όσον αφορά τις πράξεις αντιστροφής των πινάκων, ώστε να μην προκύπτουν απειριζουσες τιμές.

Από τον αλγόριθμο που περιγράψαμε στην προηγούμενη ενότητα προκύπτουν τα στοιχεία ενός οποιουδήποτε μπλοκ-τριδιαγώνιου πίνακα, από τα οποία εμείς ενδιαφερόμαστε μόνο για τα διαγώνια. Ο παραπάνω αλγόριθμος μας έδωσε την ευχέρεια να μπορεί να υπολογίσει μόνο τα διαγώνια στοιχεία χωρίς να χρειαστεί να υπολογιστούν και τα υπόλοιπα, επομένως υλοποιήσαμε τον αλγόριθμο μέχρι το σημείο που υπολογίζονται τα D .

Ενδεικτικές τιμές της G^r :

$G_r =$

1.0e+018 *

(1,1) -0.1328

(2,1) -0.0000

(3,1) -0.0000

(4,1) -0.0000

(5,1)	-0.0000
(6,1)	-0.0000
(7,1)	-0.0000
(8,1)	-0.0000
(9,1)	-0.0000
(10,1)	-0.0000
(1,2)	-0.0000
(2,2)	-0.0243
(3,2)	-0.0000
(4,2)	-0.0000
(5,2)	-0.0000
(6,2)	-0.0000
(7,2)	-0.0000
(8,2)	-0.0000
(9,2)	-0.0000
(10,2)	-0.0000
(1,3)	-0.0000
(2,3)	-0.0000
(3,3)	-0.1590
(4,3)	-0.0000
(5,3)	-0.0000
(6,3)	-0.0000

(7,3)	-0.0000
(8,3)	-0.0000
(9,3)	-0.0000
(10,3)	-0.0000
(1,4)	-0.0000
(2,4)	-0.0000
(3,4)	-0.0000
(4,4)	-0.1493
(5,4)	-0.0000
(6,4)	-0.0000
(7,4)	-0.0000
(8,4)	-0.0000
(9,4)	-0.0000
(10,4)	-0.0000
(1,5)	-0.0000
(2,5)	-0.0000
(3,5)	-0.0000
(4,5)	-0.0000
(5,5)	-0.0274
(6,5)	-0.0000
(7,5)	-0.0000
(8,5)	-0.0000

(9,5)	-0.0000
(10,5)	-0.0000
(1,6)	-0.0000
(2,6)	-0.0000
(3,6)	-0.0000
(4,6)	-0.0000
(5,6)	-0.0000
(6,6)	-0.0534

Οι τιμές που προκύπτουν δεν είναι τίποτα άλλο από την λύση των εξισώσεων Green σε κάθε γραμμή του πλέγματος στο οποίο αναλύεται το κανάλι του τρανζίστορ ξεκινώντας από την γραμμή που εκφράζει την επαφή με την πηγή και τελειώνοντας στην γραμμή που εκφράζει την επαφή με την υποδοχή.

Έχοντας πλέον τις τιμές των D , οι οποίες ταυτίζονται με τις τιμές G^r , μπορούμε να προχωρήσουμε σε αριθμητική προσομοίωση του φορμαλισμού NEGF. Από μόνες τους όμως οι τιμές του G^r δεν βοηθάνε σε τίποτα. Για να προχωρήσουμε στην προσομοίωση των πυκνοτήτων ρεύματος και φορτίου θα πρέπει να υπολογίσουμε και τα στοιχεία του $G^<$. Αυτό γίνεται ως εξής:

Αρχικά θα πρέπει να βρούμε τις τιμές των στοιχείων της προηγμένης συνάρτησης Green (advanced Green's function) G^a . Η προηγμένη συνάρτηση Green ορίζεται ως

$$G^a = G^{r\dagger} = G^{a0} + G^{a0} U^\dagger G^a$$

$$= G^{a0} + G^a U^\dagger G^{a0}.$$

Ο αλγόριθμος που υλοποιήσαμε μας έδωσε τα στοιχεία του G^r , επομένως με έναν απλό υπολογισμό στο MATLAB μπορέσαμε και βρήκαμε και τον αντίστοιχο πίνακα για τον G^a .

Ενδεικτικές τιμές της G^a :

G_a =

1.0e+018 *

(1,1)	-0.1328
(2,1)	-0.0000
(3,1)	-0.0000
(4,1)	-0.0000
(5,1)	-0.0000
(6,1)	-0.0000
(7,1)	-0.0000
(8,1)	-0.0000
(9,1)	-0.0000
(10,1)	-0.0000
(1,2)	-0.0000

(2,2)	-0.0243
(3,2)	-0.0000
(4,2)	-0.0000
(5,2)	-0.0000
(6,2)	-0.0000
(7,2)	-0.0000
(8,2)	-0.0000
(9,2)	-0.0000
(10,2)	-0.0000
(1,3)	-0.0000
(2,3)	-0.0000
(3,3)	-0.1590
(4,3)	-0.0000
(5,3)	-0.0000
(6,3)	-0.0000
(7,3)	-0.0000
(8,3)	-0.0000
(9,3)	-0.0000
(10,3)	-0.0000
(1,4)	-0.0000
(2,4)	-0.0000
(3,4)	-0.0000

(4,4)	-0.1493
(5,4)	-0.0000
(6,4)	-0.0000
(7,4)	-0.0000
(8,4)	-0.0000
(9,4)	-0.0000
(10,4)	-0.0000
(1,5)	-0.0000
(2,5)	-0.0000
(3,5)	-0.0000
(4,5)	-0.0000
(5,5)	-0.0274
(6,5)	-0.0000
(7,5)	-0.0000
(8,5)	-0.0000
(9,5)	-0.0000
(10,5)	-0.0000
(1,6)	-0.0000
(2,6)	-0.0000
(3,6)	-0.0000
(4,6)	-0.0000
(5,6)	-0.0000

(6,6) -0.0534

Έχοντας πλέον τις τιμές και του G^r και του G^a , το μόνο που μας λείπει για να υπολογίσουμε τον $G^<$ και να μπορέσουμε να προχωρήσουμε στις προσομοιώσεις είναι οι τιμές της *self-energy* $\Sigma^<$ όπως φαίνεται και από τον παρακάτω τύπο:

$$AG^< = \Sigma^< G^a,$$

Σε αυτόν τον τύπο απλά πρέπει να επισημάνουμε ότι το $\Sigma^<$ που βλέπουμε δεν είναι ακριβώς η *self-energy* που δίνεται, αλλά η *self-energy* πολλαπλασιασμένη με τον αντίστροφο πίνακα του λ (λ^{-1}).

Όπως όμως αναφέραμε και πιο πάνω, στην παρούσα εργασία δεν θα προχωρήσουμε σε αναλυτικούς υπολογισμούς για την εύρεση των τιμών των *self-energies*. Επειδή όμως είναι απαραίτητες οι συγκεκριμένες τιμές στην προσομοίωση, στον κώδικα μας προσθέσαμε μια μικρή ενότητα (που περιγράφεται αναλυτικά μαζί με τον υπόλοιπο κώδικα στο 4^ο κεφάλαιο της εργασίας) στην οποία δίνουμε συμβολικά κάποιες τιμές στην $\Sigma^<$ (συγκεκριμένα όλες οι τιμές έχουν τιμή $1e-5$) έτσι ώστε να μπορέσουμε να προχωρήσουμε στην προσομοίωση. Εννοείται ότι αυτές οι τιμές επιλέχτηκαν τυχαία και δεν έχουν κάποια σημασία πέρα από το ότι μας βοηθάνε στις προσομοιώσεις που θα κάνουμε.

Έχοντας πλέον κάνει και αυτόν τον υπολογισμό μπορούμε πλέον πολύ εύκολα να υπολογίσουμε τις πυκνότητες φορτίου και ρεύματος (charge density, current density).

Προτού προχωρήσουμε στην περιγραφή των προσομοιώσεων πρέπει να αναφέρουμε μερικά πράγματα για το *testbench* πάνω στο οποίο θα κάνουμε τις προσομοιώσεις. Αυτό είναι το *MIT well-tempered 25nm transistor*, παρόλο που ο κώδικας που αναπτύχθηκε μπορεί να χρησιμοποιηθεί και για άλλες τοπολογίες, ανάλογα με τις ανάγκες της προσομοίωσης. Πιο κάτω θα δώσουμε και μια λίστα με τα χαρακτηριστικά της συγκεκριμένης συσκευής.

Έχοντας πλέον τις τιμές που προκύψαν για το G^r , την $\Sigma^<$ και το G^α μπορούμε πλέον να βρούμε τις τιμές που προκύπτουν για το $G^<$.

Ενδεικτικές τιμές της $G^<$:

$G_less =$

$1.0e+013 *$

(1,1)	1.0056
(1,2)	0.1838
(1,3)	0.0000
(1,4)	0.0000
(1,5)	0.0000
(1,6)	0.0000
(1,7)	0.0000
(1,8)	0.0000
(1,9)	0.0000
(1,10)	0.0000
(1,11)	0.3613
(1,12)	0.0000
(1,13)	0.0000
(1,14)	0.0000
(1,15)	0.0000

(1,16) 0.0000

(1,17) 0.0000

(1,18) 0.0000

(1,19) 0.0000

(1,20) 0.0000

Στην θεωρητική περιγραφή του φορμαλισμού NEGF στην 2^η ενότητα είχαμε πει ότι η πυκνότητα φορτίου δίνεται από την σχέση

$$\rho(\mathbf{r}, k_z, E) = \sum_b \rho_b(\mathbf{r}, k_z, E) = -iG_b^<(\mathbf{r}, \mathbf{r}, k_z, E).$$

Όπου b είναι οι ανεξάρτητες «κοιλάδες» ηλεκτρονίων. Πλέον αυτές οι κοιλάδες διακριτοποιήθηκαν στους πίνακες που χρησιμοποιήσαμε επομένως με έναν απλό πολλαπλασιασμό του πίνακα $G^<$ που υπολογίσαμε με τον φανταστικό αριθμό i βρίσκουμε τις τιμές της πυκνότητας φορτίου.

Χρησιμοποιώντας τις τιμές που υπολογίσαμε στον αλγόριθμό μας και τις οποίες παρουσιάσαμε πιο πάνω, προκύπτουν

Ενδεικτικές τιμές της ρ :

charge_density =

1.0e+013 *

(1,1) 0 - 1.0056i

(1,2) 0 - 0.1838i

(1,3)	0 - 0.0000i
(1,4)	0 - 0.0000i
(1,5)	0 - 0.0000i
(1,6)	0 - 0.0000i
(1,7)	0 - 0.0000i
(1,8)	0 - 0.0000i
(1,9)	0 - 0.0000i
(1,10)	0 - 0.0000i
(1,11)	0 - 0.3613i
(1,12)	0 - 0.0000i
(1,13)	0 - 0.0000i
(1,14)	0 - 0.0000i
(1,15)	0 - 0.0000i
(1,16)	0 - 0.0000i
(1,17)	0 - 0.0000i
(1,18)	0 - 0.0000i
(1,19)	0 - 0.0000i
(1,20)	0 - 0.0000i

Όπως βλέπουμε καθαρά οι τιμές είναι απλώς τα γινόμενα των τιμών του $G^<$ με τον φανταστικό αριθμό i .

Στη συνέχεια θα δώσουμε τις τιμές που προκύπτουν για την πυκνότητα του ρεύματος μεταξύ 2 γειτονικών μπλοκ του πίνακα (ουσιαστικά πρόκειται για 2 γειτονικές κοιλάδες ηλεκτρονίων).

Η πυκνότητα ρεύματος που ρέει μεταξύ δύο γειτονικών μπλοκ (έστω q και $q+1$) δίνεται από τον τύπο

$$\begin{aligned}
 J(q \rightarrow q+1, k_z, E) &= \sum_b J_b(q \rightarrow q+1, k_z, E) \\
 &= \frac{2e}{h} \sum_b \text{Tr} [T_{q,q+1} G_{b;q,q+1}^<(k_z, E) \\
 &\quad - T_{q+1,q} G_{b;q+1,q}^<(k_z, E)],
 \end{aligned}$$

όπου ο πίνακας T έχει υπολογιστεί στις εξ. (16) και (17), ενώ οι τιμές της $G^<$ είναι γνωστές από πριν.

Οι ενδεικτικές τιμές του πίνακα T και της πυκνότητας ρεύματος J δίνονται στην συνέχεια.

Ενδεικτικές τιμές του πίνακα T :

$T_{44} =$

1.0e-012 *

(1,1)	0.4751
(2,2)	0.4751
(3,3)	0.4751
(4,4)	0.4751
(5,5)	0.4751

(6,6) 0.4751
(7,7) 0.4751
(8,8) 0.4751
(9,9) 0.4751
(10,10) 0.4751
(11,11) 0.4751
(12,12) 0.4751
(13,13) 0.4751
(14,14) 0.4751
(15,15) 0.4751
(16,16) 0.4751
(17,17) 0.4751
(18,18) 0.4751
(19,19) 0.4751
(20,20) 0.4751
(21,21) 0.4751
(22,22) 0.4751
(23,23) 0.4751
(24,24) 0.4751
(25,25) 0.4751
(26,26) 0.4751
(27,27) 0.4751

(28,28) 0.4751

(29,29) 0.4751

(30,30) 0.4751

T_5 =

1.0e-012 *

(1,1) 0.4751

(2,2) 0.4751

(3,3) 0.4751

(4,4) 0.4751

(5,5) 0.4751

(6,6) 0.4751

(7,7) 0.4751

(8,8) 0.4751

(9,9) 0.4751

(10,10) 0.4751

(11,11) 0.4751

(12,12) 0.4751

(13,13) 0.4751

(14,14) 0.4751
(15,15) 0.4751
(16,16) 0.4751
(17,17) 0.4751
(18,18) 0.4751
(19,19) 0.4751
(20,20) 0.4751
(21,21) 0.4751
(22,22) 0.4751
(23,23) 0.4751
(24,24) 0.4751
(25,25) 0.4751
(26,26) 0.4751
(27,27) 0.4751
(28,28) 0.4751
(29,29) 0.4751
(30,30) 0.4751

(στον αλγόριθμό μας συμβολίζουμε με T_4 , T_5 τις άνω και κάτω διαγώνιους αντίστοιχα του πίνακα T , διότι είναι οι μόνες που μας χρειάζονται για τον εν λόγω υπολογισμό όπως φαίνεται και από τον τύπο)

Παρατηρούμε ότι οι 2 πίνακες είναι συμμετρικοί κάτι το οποίο ήταν αναμενόμενο όπως περιγράψαμε στις ενότητες 2,3.

Ενδεικτικές τιμές της J:

J =

(1,1)	1.9966
(2,1)	8.3857
(3,1)	8.3857
(4,1)	23.7196
(5,1)	102.9448
(6,1)	26.2753
(7,1)	6.4690
(8,1)	1.8369
(9,1)	2.1563
(10,1)	1.6771
(11,1)	2.1563
(12,1)	2.3161
(13,1)	4.5523
(14,1)	82.4996
(15,1)	51.8318
(16,1)	8.3857
(17,1)	3.2744
(18,1)	1.4376
(19,1)	1.4376

(απλώς να αναφέρουμε ότι έχουμε περάσει σε έναν πίνακα τις τιμές της πυκνότητας ρεύματος που προκύπτει για κάθε ένα από τα μπλοκ).

Σε διαγραμματική μορφή τα αποτελέσματα για την πυκνότητα ρεύματος συναρτήσει των γειτονικών μπλοκ θα είναι:

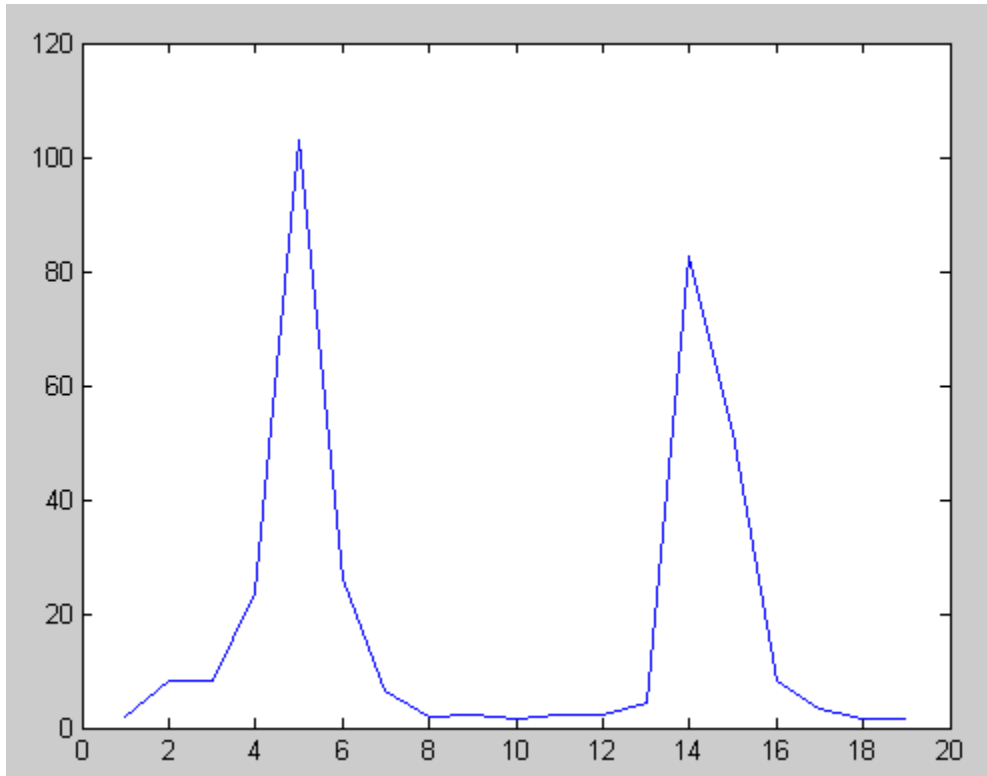


Figure 3: Πυκνότητα ρεύματος

Ενδεικτικές τιμές DOS:

H =

1.0e+018 *

(1,1) 0.0062

(2,1) 0.0000

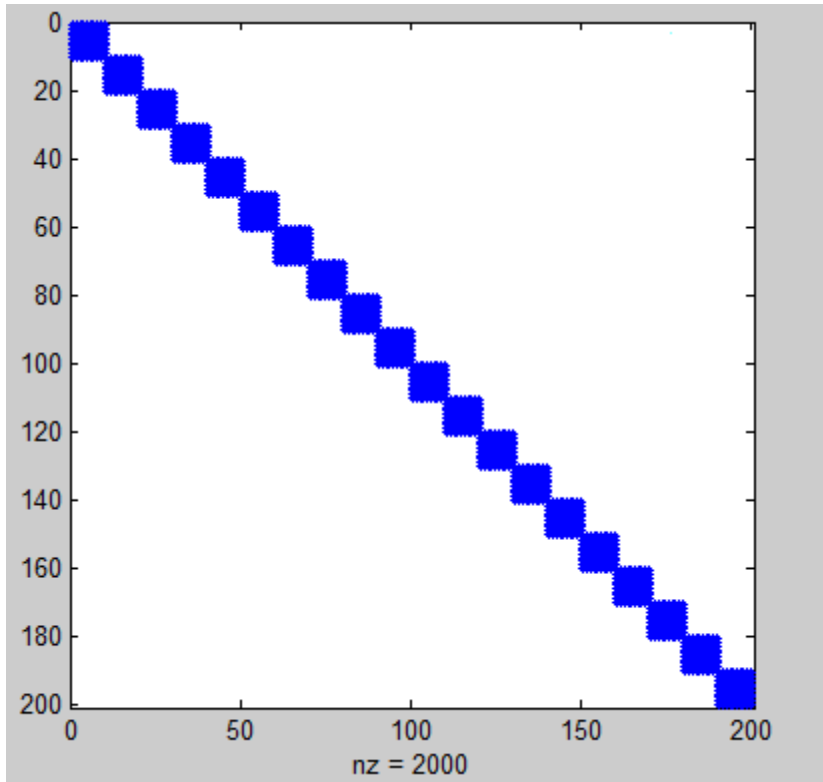
(3,1) 0.0000

(4,1)	0.0000
(5,1)	0.0000
(6,1)	0.0000
(7,1)	0.0000
(8,1)	0.0000
(9,1)	0.0000
(10,1)	0.0000
(1,2)	0.0000
(2,2)	0.0440
(3,2)	0.0000
(4,2)	0.0000
(5,2)	0.0000
(6,2)	0.0000
(7,2)	0.0000
(8,2)	0.0000
(9,2)	0.0000
(10,2)	0.0000
(1,3)	0.0000
(2,3)	0.0000
(3,3)	0.0061
(4,3)	0.0000
(5,3)	0.0000

(6,3)	0.0000
(7,3)	0.0000
(8,3)	0.0000
(9,3)	0.0000
(10,3)	0.0000
(1,4)	0.0000
(2,4)	0.0000
(3,4)	0.0000
(4,4)	0.0088
(5,4)	0.0000
(6,4)	0.0000
(7,4)	0.0000
(8,4)	0.0000
(9,4)	0.0000
(10,4)	0.0000
(1,5)	0.0000
(2,5)	0.0000
(3,5)	0.0000
(4,5)	0.0000
(5,5)	0.0573
(6,5)	0.0000
(7,5)	0.0000

(8,5)	0.0000
(9,5)	0.0000
(10,5)	0.0000
(1,6)	0.0000
(2,6)	0.0000
(3,6)	0.0000
(4,6)	0.0000
(5,6)	0.0000
(6,6)	0.0201
(7,6)	0.0000
(8,6)	0.0000
(9,6)	0.0000
(10,6)	0.0000

Ο πίνακας των πυκνοτήτων καταστάσεων έχει την εξής μορφή:



Χαρακτηριστικά του MIT25 transistor:

$$t_{ox} = 1.5e-9 \text{ m}$$

$$h = 4.13566733e-15 \text{ eV*sec}$$

$$Lp = 6.0e-8 \text{ m}$$

$$Lb = 12.0e-8 \text{ m}$$

$$Lg = 5.0e-8 \text{ m}$$

$$e_{Si} = 11.9$$

$$kT = 0.02585 \text{ eV}$$

$$length : 9.5e-8 \text{ m}$$

$$width : 1.2e-7 \text{ m}$$

Επίλογος

Στην παρούσα εργασία παρουσιάσαμε το πρόβλημα της κβαντικής μεταφοράς στο κανάλι ενός τρανζίστορ, αλλά και σε οποιαδήποτε άλλη ηλεκτρονική συσκευή σε νανοκλίμακα, όπως νανο-καλώδια.

Δώσαμε μια λεπτομερή αναφορά στην έρευνα που γίνεται και έγινε πάνω σε αυτό το αντικείμενο, καθώς και τις μελλοντικές προεκτάσεις του.

Δείξαμε πως με την χρήση των συναρτήσεων μη-ισορροπίας του Green μπορούμε να προσομοιώσουμε ικανοποιητικά υπό κάποιες συνθήκες το φαινόμενο της κβαντικής μεταφοράς τόσο σε ομογενή, όσο και σε μη-ομογενή πλέγματα.

Επίσης δείξαμε πως οι συναρτήσεις αυτές διακριτοποιούνται σε μορφή πίνακα και πως χρησιμοποιούμε αυτή την διακριτοποίηση για να εξάγουμε τιμές κατάλληλες για τις εξισώσεις G^r και $G^<$, τις οποίες χρησιμοποιούμε ύστερα για τον υπολογισμό των πυκνοτήτων φορτίου και ρεύματος.

Σε αρκετά σημεία δώσαμε μηδενικές ή τυχαίες τιμές, καθαρά και μόνο για την διευκόλυνση των πράξεων. Σε περίπτωση εμπειριστατωμένης έρευνας θα χρειαστεί να υπολογιστούν τιμές και σε αυτά τα σημεία, κάτι το οποίο δεν είναι αναγκαίο να γίνει σε επίπεδο διπλωματικής εργασίας.

Τέλος κάνουμε μια λεπτομερή παρουσίαση του κώδικα που αναπτύχθηκε για την περιγραφή του φαινομένου της κβαντικής μεταφοράς, καθώς επίσης παραθέτουμε αριθμητικά αποτελέσματα από τις προσομοιώσεις που έγιναν.

Αναφορές

- [1] **“Two-dimensional quantum mechanical modeling of nanotransistors”**, A. Svizhenko, M. P. Anantram, T. R. Govindan, B. Biegel, *Journal of Applied Physics*, Volume 91, Number 4, Feb. 2002.
- [2] **“A scalable distributed method for quantum-scale device simulation”**, Stephen Cauley, Jitesh Jain, Cheng-Kok Koh, Venkataramanan Balakrishnan, *Journal of Applied Physics* **101**, 123715, 2007.
- [3] **“Numerically stable algorithms for inversion of block tridiagonal and banded matrices”**, Jitesh Jain, Hong Li, Stephen Cauley, Cheng-Kok Koh, Venkataramanan Balakrishnan, *Purdue e-pubs*, Purdue University, March 2007.
- [4] **“Modeling of Quantum Transport in Nano-scale MOSFET Devices”**, Seonghoon Jin, *Ph. D. Dissertation*, Feb. 2006.
- [5] **“Computing entries of the inverse of a sparse matrix using the FIND algorithm”**, S. Li, S. Ahmed, G. Klimeck, E. Darve, *Journal of Applied Physics* 227, 2008.
- [6] **“Nanoscale device modeling: the Green’s function method”**, Supriyo Datta, *Superlattices and Microstructures*, Vol. 28, No. 4, 2000.
- [7] **“Modeling Quantum Transport in Nanoscale Transistors”**, Ramesh Venugopal, *Ph. D. Dissertation*, Aug. 2003.
- [8] **“Electronic transport in mesoscopic systems”**, Supriyo Datta, 1995.
- [9] **“Domain-Decomposition-Type Methods For Computing The Diagonal Of a Matrix Inverse”**, Jok M. Tang, Yousef Saa, University of Minnesota, 2010