5-1-2010

Quantum electron transport in models of nanoparticles using matrix algebra and renormalization group methods

Lazarus Solomon

QUANTUM ELECTRON TRANSPORT IN MODELS OF NANOPARTICLES USING MATRIX ALGEBRA AND RENORMALIZATION GROUP METHODS

By

Lazarus Solomon

A Thesis
Submitted to the Faculty of
Mississippi State University
in Partial Fulfillment of the Requirements
for the Degree of Doctor of Philosophy
in Engineering with concentration in Engineering Physics
in the Department of Physics and Astronomy

Mississippi State, Mississippi

May 2010

QUANTUM ELECTRON TRANSPORT IN MODELS OF

NANOPARTICLES USING MATRIX ALGEBRA AND

RENORMALIZATION GROUP METHODS

By

Lazarus Solomon

Approved:

Mark A. Novotny
Professor of Physics and Astronomy
Astronomy and Department Head
(Major Professor)

David L. Monts
Professor of Physics and Astronomy
and Graduate Coordinator

R. Torsten Clay
Associate Professor of Physics
and Astronomy
(Committee Member)

Seong-Gon Kim
Associate Professor of Physics
and Astronomy
(Committee Member)

Gautam Rupak
Assistant Professor of Physics
and Astronomy
(Committee Member)

Yaroslav Koshka
Associate Professor of Computer
and Electrical Engineering
(Committee Member)

Sarah A. Rajala
Dean of the James Worth Bagley College
of Engineering

Name: Lazarus Solomon

Date of Degree: May 2, 2010

Institution: Mississippi State University

Major Field: Engineering with concentration in Engineering Physics

Major Professor: Dr. Mark A. Novotny

Title of Study: QUANTUM ELECTRON TRANSPORT IN MODELS OF NANOPARTICLES USING MATRIX ALGEBRA AND RENORMALIZATION GROUP METHODS

Pages in Study: 123

Candidate for Degree of Doctor of Philosophy

A general expression for quantum transmission of non-interacting spinless electrons through models of a fully connected network of sites that can be regarded as a nanoparticle is obtained using matrix algebra. This matrix algebra method leads to the same results given by the Green's function method without requiring the mathematical sophistication as required by the later. The model of the nanoparticle in this study comprises a single linear array of atoms that profile the input and output leads connected to a fully connected blob of atoms. A simple tight-binding Hamiltonian motivates the quantum transmission in the discrete lattice system. If there are n atoms in the nanoparticle, the methodology requires the inverse of a $n \times n$ matrix. The solution is obtained analytically for different cases: a single atom in the nanoparticle, a single dangle atom, n fully connected atoms in a mean-field type cluster with symmetric input and output connections, and the most general case where the n fully connected atoms can be connected arbitrarily to the input and output leads. A numerical solution is also provided for the case where the intra-bonds

among the atoms in the nanoparticle are varied (a case with not-fully connected atoms). The expression for the transmission coefficient thus obtained using the matrix method is compared with the transmission coefficients derived using the real space Renormalization Group method and the Green's function method.

Key words: Nanosystems, transport, transmission coefficient

DEDICATION

This thesis is dedicated to my mother, Pramila and in honor of the memory of my late father, S. S. SunderRaj, sisters, Vinolia and Catherine, brothers, Charles and Samuel, sons, Evan and Navin, and to my dear wife Layla. Thank you for your love and support.

ACKNOWLEDGMENTS

I would like to express my very sincere gratitude to my research and thesis advisor, Dr. Mark A. Novotny whose expertise, understanding, and incredible patience has added considerably to my graduate school experience. I appreciate his vast knowledge and skills and the demeanor with which he guided me through this project. I am gratefully indebted to him for my successful completion of the doctoral program.

Thanks to Dr. Seong-Gon Kim, Dr. R. Torsten Clay, Dr. Yaroslav Koshka, and Dr. Rupak Gautam for consenting to be members of the dissertation committee and the valuable suggestions to the making of this document. Their patience and advice is appreciated.

Thanks to my wife Layla who has been a constant support without which I could not have come thus far. Thanks to my boys Evan and Navin who are always a source of joy that kept my good sense prevailing during this journey through graduate school.

I thank the Department of Physics and Astronomy for the financial support, as well as partial support from grants from the National Science Foundation.

The findings and opinions in this thesis belong solely to the author, and are not necessarily those of the sponsor.

Permission to reproduce the MSU logo was given by Mississippi State University.

Unix® is a registered trademark of The Open Group.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS, ABBREVIATIONS, AND NOMENCLATURE

E Energy of particle

E_c Conduction band edge

G Conductance

G^A Advanced Green's function

G^R Retarded Green's function

H_{op} Hamiltonian operator

$\mathcal{H}$ Hamiltonian

h Planck's constant

$\hbar = h/2\pi$

I Identity matrix

J Matrix with all elements equal to 1

k_B Boltzmann constant

k_f Fermi wavenumber

M Number of transverse modes

R Reflection coefficient

r Amplitude of reflected wave

T Transmission coefficient

$\overline{T}$ Product of number of modes

t Hopping/overlap energy parameter

$t_{a,b}$ Hopping energy parameter between sites a and b

t_T Amplitude of transmission

t_u Output connection parameter

t_w Input connection parameter

U Time-independent potential energy

$\vec{u}$ Output coupling vector

v Velocity of particle

v_f Fermi velocity

$\vec{w}$ Input coupling vector

Γ Energy broadening

ϵ Onsite potential energy

μ Electrochemical potential

Σ Self-energy

CHAPTER 1

INTRODUCTION

1.1 Nanotechnology

Nanotechnology has become popular not only by name but in extensive research and promises a lot of opportunities and benefits in every area of life. In its original sense, 'nanotechnology' refers to the projected ability to construct items from a bottom up fashion, using techniques and tools being developed to make high performance products. In the not too far future the manufacturing establishments will yield to a paradigm shift that this new technology will demand. Nanotechnology focuses on constructing structures of clusters atoms at a scale of a billionth of a meter abbreviated as $1\ nm$. The characteristic dimensions are less than about $1,000$ nanometers. To get a sense of the nanoscale, a human hair measures $50,000$ nanometers across [39].

In one of his talks given on December 29, 1959, at the annual meeting of the American Physical Society at the California Institute of Technology, the 1965 Nobel Prize winner in Physics, Richard P. Feynman has said, "the principles of physics, as far as I can see, do not speak against the possibility of maneuvering things atom by atom." The effort is on to apply at the molecular scale the concept that has demonstrated its effectiveness at the macroscopic scale: making parts go to locations where it is needed [14, 27].

Properties that are of interest with respect to the objective of this research in particular, and to the larger interests of the broad area of nanoscience in general, are the electronic transport and other transport properties at nanoscale dimensions. The focus in this treatise will primarily be that of an alternative transport coefficient derivation to those derived using standard conventional methods.

The systems of interest to nanoscience are the isolated nanostructures and their properties of conductivities. The modeling of these nanostructures and a convenient method in deriving a transmission expression is the essence of this dissertation. Transport theory lays the foundation of theoretical materials design. The questions to ask then are:

- How does one derive a transmission expression? Is there another method?
- If there is one, is it rigorous?
- Can it be implemented?
- Does the method yield the same results as other standard methods?
- Is it applicable to more difficult systems?

The answers to these questions set the tone of this dissertation.

1.2 Quantum Transport

Transport in nanosystems is currently a subject of intense experimental and theoretical investigations because of their possible applications in electronic devices. This ongoing research concerns with transport properties of various systems such as nanowires, single

atoms, molecules, quantum, dots, nanotubes. Technological advancements in recent times have permitted fabrication of devices and enabled control over matters of small dimensions. Under those circumstances, the behavior of individual atoms and electrons become important and quantum effects become very crucial. These fundamentally alter the optical, electrical and magnetic properties of materials. Many of the rules applied in the macroscopic realm do not hold in the mesoscopic realm. Certain fundamental limits have to be set on how small the electronic devices should be. Theoretical predictions along with experiments are part of that process. In the classical case, a particle or molecule is transported through the device by diffusion, a time-dependent random process, such as heat diffusion, molecular diffusion, and Brownian motion. This type of motion encounters frequent collisions and the net transport of particles takes place because of a concentration gradient, that is, from a region of higher concentration to one of a lower concentration. Diffusion is relevant in the lengths between the nanometer and millimeter scale. Unlike this classical type, when the dimensions of the device approach the mean free path, the transport mechanism changes from diffusive to ballistic. In this nanometer regime quantum effects become significant and using quantum principles becomes central. The nanometer scale regime's imperative requirement is that the transport of particles be governed by the quantum mechanical wave propagation. Transport is no longer dependent on the dimensions of the device, and the macroscopic "Ohm's law" falls short. In such a case the conductance through a narrow constriction formed between two electrodes in general is given by the Landauer formula [24, 25]

$$G = \frac{2e^2}{h} MT, \tag{1.1}$$

where e is the electronic charge, h is Planck's constant, M is the number of transverse modes, and T is the transmission probability that an electron injected at one end will emerge at the other end. $T = 1$ for ballistic transport.

Much of the discussion in this thesis will be utilizing the single-band, effective-mass Schrödinger equation. The solutions of this problem provide one of the fundamental tools available to understand and predict the behavior of quantum devices.

A large number of works in the area of quantum transport have dealt with conditions very near to thermal equilibrium and low bias. This 'very near to thermal equilibrium' and low bias condition is also referred to as the "linear-response regime," implying the current and voltage relationship is linear. Theoretical descriptions are much easier in this case than otherwise. The Kubo formula for conductivity is one approach in studying linear response theory [22, 23]. Another approach to the linear response transport is the Landauer transport formula Eq. (1.1) [24]. The conductance formula is expressed in terms of the quantum mechanical transmission coefficients for a system at absolute zero temperature, where the Fermi-Dirac distribution (Eq. 1.29) is a step function. When conditions are not too far removed from thermal equilibrium, then equilibrium statistical physics can be applied treating the departure from equilibrium as a small perturbation on the equilibrium state.

One approach to more sophisticated problems in quantum transport theory is the Green's function formalism. The basis of this approach is its concurrence with the progress of quantum electrodynamics. The non-equilibrium Green's function theory is described by Kadanoff and Baym [18] and by Keldysh [19]. The non-equilibrium functions are defined

as expectation values of single-particle creation and annihilation operators and they describe the state and time evolution of the system. The Green's functions are determined by solving a set of Dyson equations which are an integral form of Schrödinger's equation. The Dyson's equations form a convenient starting point for the development of a perturbation expansion [15].

1.3 Transport Calculations - Continuous

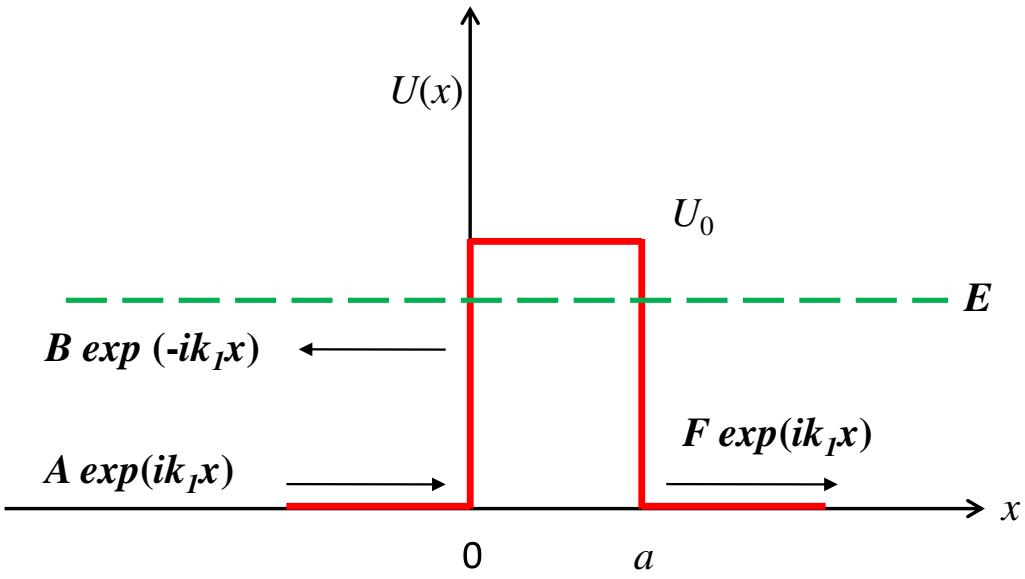

Figure 1.1

Potential barrier with propagation directions.

The simplest quantum transport problem is in terms of scattering of the electron wavefunction $\psi(x)$ in one-dimension by a spatially varying potential $U(x)$, a one-dimensional

scattering problem that will aid in the understanding of basic transport calculations. The dynamics of a quantum particle of mass m in one-dimension is given by the time-independent Schrödinger equation

$$-\frac{\hbar^2}{2m}\frac{d^2\psi(x)}{dx^2} + U(x)\psi(x) = E\psi(x), \tag{1.2}$$

where $\hbar$ is the Planck's constant, and E is the total energy of a localized wave packet moving from the left towards a one-dimensional potential barrier $U(x)$, (Fig. 1.1) satisfying the condition

$$U(x) = \begin{cases} 0 & x < 0 \\ U_0 & 0 \leq x \leq a \\ 0 & x > a. \end{cases} \tag{1.3}$$

This repulsive potential supports no bound states. We assume a particle comes in from $x = -\infty$ and is either reflected by or transmitted through the barrier. In this case where $E < U_0$, classically, every particle that arrives at the barrier at $(x = 0)$ will be reflected back. Quantum mechanically the wave function is not zero beyond the barrier, implying that there is some probability of transition. The wave equation is solved by the standard text book presentation of this scattering phenomenon of the time-independent Schrödinger equation Eq. (1.2). It requires a solution of $H\psi = E\psi$. The Schrödinger equation in the three regions yields the expressions [40, 45]

$$\psi(x) = \begin{cases} \psi_1(x) = Ae^{ik_1x} + Be^{-ik_1x} & x \leq 0 \\ \psi_2(x) = Ce^{k_2x} + De^{-k_2x} & 0 < x < a \\ \psi_3(x) = Fe^{ik_1x} & x \geq a, \end{cases} \tag{1.4}$$

where $k_1^2 = 2mE/\hbar^2$ and $k_2^2 = 2m(U_0 - E)/\hbar^2$, $A\ e^{ik_1x}$ and $F\ e^{-ik_1x}$ represent the $incident$ and $transmitted$ waves respectively moving in the positive x direction, $B\ e^{-ik_1x}$ corresponds to the $reflected$ wave moving in the negative x direction. Using the continuity of $\psi(x)$ and its first derivative at $x = 0$ and $x = a$, the constants $B, C, D,$ and F can be obtained in terms of A.

The $reflection$ and $transmission\ coefficients$, R and T can now be evaluated as [45]

$$R = \left| \frac{reflected\ current\ density}{incident\ current\ density} \right| = \left| \frac{J_{reflected}}{J_{incident}} \right|, \quad T = \left| \frac{J_{transmitted}}{J_{incident}} \right|. \tag{1.5}$$

Since the incident wave is $\psi_{\rm i} = Ae^{ik_1x}$, the incident current density is given by

$$J_{incident} = \frac{i\hbar}{2m}\left(\psi_{\rm i}(x)\frac{d\psi_{\rm i}^*(x)}{dx} - \psi_{\rm i}^*(x)\frac{d\psi_{\rm i}(x)}{dx}\right) = \frac{\hbar k_1}{m}\big|A\big|^2, \tag{1.6}$$

and similarly, since the reflected and transmitted waves are $\psi_r(x) = Be^{-ik_1x}$ and $\psi_t(x) = Ce^{ik_1x}$ respectively, then the reflected and transmitted fluxes are

$$J_{reflected} = -\frac{\hbar k_1}{m}\big|B\big|^2, \quad J_{transmitted} = \frac{\hbar k_2}{m}\big|F\big|^2. \tag{1.7}$$

Thus the transmission coefficient is

$$T = \frac{k_1|F|^2}{k_1|A|^2}. \tag{1.8}$$

Constant F can be calculated in terms of A and hence the transmission probability T.

1.4 The Propagation Matrix

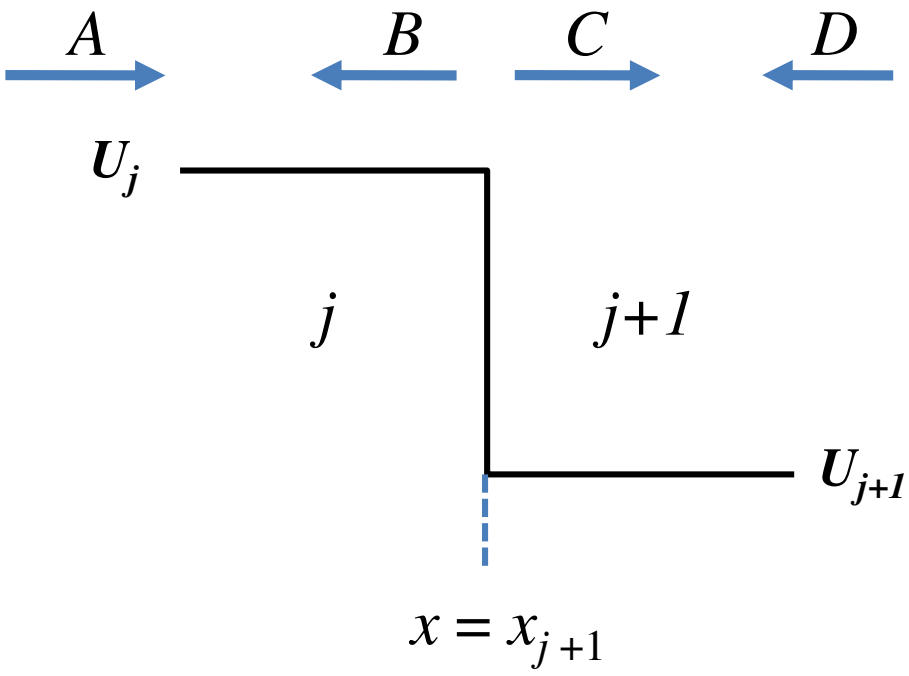

Figure 1.2

1-D step potential.

This section motivates the tight-binding approximation and the matrix method, which is the theme of this dissertation. Consider a step potential shown in Fig. (1.2). Any potential of arbitrary shape can be approximated as a series of potential steps, $i.e.$ it can be discretized, as shown in Fig. (1.3). The transmission and reflection coefficients are calculated at the first potential step for a particle energy E incident from the left as shown in Fig. (1.2). The transmitted particle propagating to the next potential step, where it again has a probability of being transmitted or reflected. For every potential step and the

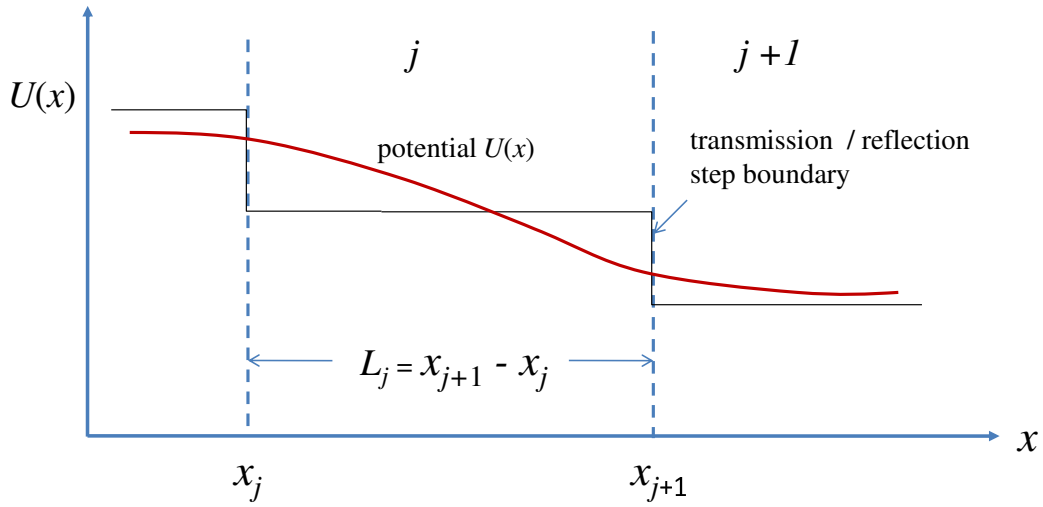

Figure 1.3

Smoothly varying 1-D potential discretized into series of potential steps.

free propagation of the wavefunction to the next potential step can be associated with a 2×2 matrix. This matrix carries all the amplitude and phase information on transmission and reflection at each potential step and the propagation to the next step. The total one-dimensional propagation probability for a potential consisting of a number of potential steps can be calculated by taking the product of each 2×2 matrix associated with each transmission and reflection at each step. This approach works well for a series of potential steps approximating a smoothly varying potential, by which it can be assumed that the errors are small if the step spacing is small compared to the wavelength of the particle.

The steps [26] implemented to calculate the transmission probability are:

1. Calculate the propagation matrix $\mathbf{p}_{\text{step}}$ for transmission and reflection of the wave function representing a particle of energy E incident on a single potential step at x_{j+1} as shown in Fig. (1.3).

2. Calculate the propagation matrix $\mathbf{p}_{\text{free}}$ for propagation of the wave function between steps. The free propagation region is between x_j and x_{j+1}, with distance L_j.

3. Calculate the propagation matrix for the j-th region. This is obtained by multiplying $\mathbf{p}_{\text{step}}$ and $\mathbf{p}_{\text{free}}$ to get matrix $\mathbf{p}_j$ for the j-th region of the discretized potential.

4. Calculate the total propagation matrix $\mathbf{P}$ for the complete potential by multiplying together the propagation matrices for each region of the discretized potential.

The particle has wave vector

$$k_j^2 = \frac{2m(E - U_j)}{\hbar^2}. \tag{1.9}$$

The propagation between steps separated by distance L_j carries a phase so that $\psi_{Aj}\, e^{ik_jL_j} = \psi_{Cj}$ and $\psi_{Bj}\, e^{-ik_jL_j} = \psi_{Dj}$, which in matrix form is

$$\begin{bmatrix} A_j \\ B_j \end{bmatrix} = \mathbf{p}_{j_{\text{free}}} \begin{bmatrix} C_{j+1} \\ D_{j+1} \end{bmatrix}, \tag{1.10}$$

where the free propagation matrix is

$$\mathbf{p}_{j_{\text{free}}} = \begin{bmatrix} e^{-ik_jL_j} & 0 \\ 0 & e^{ik_jL_j} \end{bmatrix} \tag{1.11}$$

while the step matrix is

$$\mathbf{p}_{j_{\text{step}}} = \frac{1}{2} \begin{bmatrix} \left(1 + \frac{k_{j+1}}{k_j}\right) & \left(1 - \frac{k_{j+1}}{k_j}\right) \\ \left(1 - \frac{k_{j+1}}{k_j}\right) & \left(1 + \frac{k_{j+1}}{k_j}\right) \end{bmatrix}. \tag{1.12}$$

The propagating matrix $\mathbf{p}_j$ for the j-th region is the combined effect, which is a product of $\mathbf{p}_{j_{\rm free}}$ and $\mathbf{p}_{j_{\rm step}}$ matrices, that is,

$$\mathbf{p}_j = \mathbf{p}_{j_{\rm free}}\, \mathbf{p}_{j_{\rm step}} = \begin{bmatrix} p_{11} & p_{12} \\ p_{21} & p_{22} \end{bmatrix}, \tag{1.13}$$

where

$$\begin{aligned} p_{11} &= \frac{1}{2}\left(1+\frac{k_{j+1}}{k_j}\right)\exp(-ik_jL_j) \\ p_{12} &= \frac{1}{2}\left(1-\frac{k_{j+1}}{k_j}\right)\exp(-ik_jL_j) \\ p_{21} &= \frac{1}{2}\left(1-\frac{k_{j+1}}{k_j}\right)\exp(ik_jL_j) \\ p_{22} &= \frac{1}{2}\left(1+\frac{k_{j+1}}{k_j}\right)\exp(ik_jL_j), \end{aligned} \tag{1.14}$$

note that $p_{11} = p_{22}^*$ and $p_{21} = p_{12}^*$.

For the general case of N potential steps, the total propagation matrix is the product of the propagation matrix for each region

$$\mathbf{P} = \mathbf{p}_1\mathbf{p}_2\ldots\mathbf{p}_j\ldots\mathbf{p}_N = \prod_{j=1}^{N}\mathbf{p}_j.$$

Since the particle is introduced from the left, the incident coefficient is taken to be $A = 1$. If there is no incoming particle from the right, then $D = 0$. Consequently

$$\begin{bmatrix} A \\ B \end{bmatrix} = \left(\prod_{j=1}^{N}\mathbf{p}_j\right)\begin{bmatrix} C \\ D \end{bmatrix} = \mathbf{P}\begin{bmatrix} C \\ D \end{bmatrix} \tag{1.15}$$

can be written as

$$\begin{bmatrix} 1 \\ B \end{bmatrix} = \begin{bmatrix} p_{11} & p_{12} \\ p_{21} & p_{22} \end{bmatrix}\begin{bmatrix} C \\ 0 \end{bmatrix}. \tag{1.16}$$

The transmission probability in this case is

$$|C|^2 = \left|\frac{1}{p_{11}}\right|^2. \tag{1.17}$$

The transmission probability for a rectangular potential barrier as shown in Fig. (1.1), which can be considered as composed of two parts where the wave vector changes from k_1 to k_2 due to the potential step-up at $x = 0$ and a step-down at $x = L = a$ where the vector changes from k_2 to k_1, which is symmetry. The corresponding wave function changes from ψ_1 to ψ_2 for the part step up and from ψ_2 to ψ_1 for the part step down, where

$$\begin{aligned} \psi_1 &= \frac{A}{\sqrt{k_1}}e^{ik_1x} + \frac{B}{\sqrt{k_1}}e^{-ik_1x} \\ \psi_2 &= \frac{C}{\sqrt{k_2}}e^{ik_2x} + \frac{D}{\sqrt{k_2}}e^{-ik_2x}. \end{aligned} \tag{1.18}$$

Using constraints that the wave functions and their first derivatives are continuous at the boundaries, in matrix form this becomes

$$\frac{1}{\sqrt{k_1}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}\begin{bmatrix} A \\ B \end{bmatrix} = \frac{1}{\sqrt{k_2}}\begin{bmatrix} 1 & 1 \\ \frac{k_2}{k_1} & -\frac{k_2}{k_1} \end{bmatrix}\begin{bmatrix} C \\ D \end{bmatrix} \tag{1.19}$$

Thus

$$\begin{bmatrix} A \\ B \end{bmatrix} = \frac{1}{2\sqrt{k_1k_2}}\begin{bmatrix} k_1+k_2 & k_1-k_2 \\ k_1-k_2 & k_1+k_2 \end{bmatrix}\begin{bmatrix} C \\ D \end{bmatrix}. \tag{1.20}$$

The total propagation matrix for the rectangular potential barrier of thickness L is the product of the step-up matrix and the step-down matrix, which is

$$\mathbf{P} = \frac{1}{4k_1k_2}\begin{bmatrix} (k_1+k_2)e^{-ik_2L} & (k_1-k_2)e^{ik_2L} \\ (k_1-k_2)e^{-ik_2L} & (k_1+k_2)e^{ik_2L} \end{bmatrix}\begin{bmatrix} k_2+k_1 & k_2-k_1 \\ k_2-k_1 & k_2+k_1 \end{bmatrix}. \tag{1.21}$$

Multiplying out the matrices gives the elements of the matrix $\mathbf{P}$. The transmission probability for a particle incident on the potential barrier is given by $|1/p_{11}|^2$, where here

$$p_{11} = \frac{(k_2^2 + k_1^2 + 2k_1k_2)e^{-ik_2L} - (k_1^2 + k_2^2 - 2k_1k_2)e^{ik_2L}}{4k_1k_2}. \tag{1.22}$$

With some rearrangement

$$p_{11} = -\frac{(k_1^2 + k_2^2)(e^{ik_2L} - e^{-ik_2L})}{2 \cdot 2k_1k_2} + \frac{1}{2}(e^{-ik_2L} + e^{ik_2L}). \tag{1.23}$$

For the case $E \geq U_0$, when the energy of the incident particle is greater than the potential barrier energy, using the above equation results in

$$p_{11} = -i\frac{(k_1^2 + k_2^2)}{2k_1k_2}\sin(k_2L) + \cos(k_2L).$$

Therefore the transmission probability for $E \geq U_0$ is

$$T = \frac{1}{|p_{11}|^2} = \left[\left(\frac{k_2^2 + k_1^2}{2k_1k_2}\right)^2 \sin^2(k_2L) + \cos^2(k_2L)\right]^{-1} \tag{1.24}$$

or

$$T = \left[1 + \left(\left(\frac{k_2^2 + k_1^2}{2k_1k_2}\right)^2 - 1\right)\sin^2(k_2L)\right]^{-1}. \tag{1.25}$$

For the case when $E < U_0$

$$T = \left[1 + \left(\left(\frac{k_2^2 + k_1^2}{2k_1k_2}\right)^2 - 1\right)\sinh^2(k_2L)\right]^{-1}. \tag{1.26}$$

Transmission as a function of energy can be expressed using the transmission for the case $E \geq U_0$ with the use of the relations $k_1^2 = 2mE/\hbar^2$ and $k_2^2 = 2m(E - U_0)/\hbar^2$ giving

$$T(E) = \left[1 + \frac{1}{4}\left(\frac{E - (E - U_0)}{\sqrt{E(E - U_0)}}\right)^2 \sin^2(k_2L)\right]^{-1} \tag{1.27}$$

or

$$T(E) = \left[1 + \frac{1}{4}\left(\frac{{U_0}^2}{E(U_0 - E)}\right)\sinh^2(k_2L)\right]^{-1}. \tag{1.28}$$

1.5 Calculation of Finite Temperature Conductance

The zero temperature conductance is a Fermi surface property since the current is carried by a few electrons near the Fermi energy, and can be considered to be diffusion from a region of higher electrochemical potential μ_1 to a region of lower electrochemical potential μ_2. At zero temperature transport of electrons takes place in the energy range $\mu_1 > E > \mu_2$.

When the temperature is raised, the Fermi function

$$f_0(E) = (1 + \exp\left[(E - E_f)/k_\beta T\right])^{-1} \tag{1.29}$$

changes smoothly over an energy range, instead of the abrupt discontinuous change from one to zero (Fig. 1.4). At finite temperatures, slightly above zero, the kinetic energy of the electrons are raised thereby causing some redistribution. Some levels that were vacant at absolute zero are now occupied, and some levels that were occupied at absolute zero are vacant. The electro-chemical potential μ, which is a function of temperature, causes the redistribution. The discontinuous change at absolute zero in the Fermi function is due to $\mu = \epsilon = \epsilon_F$. When the quantity μ changes due to a change in temperature, then $\epsilon - \mu > k_B T$ and is not zero.

The conductance of a one-dimensional quantum mechanical system at absolute zero is given by the well established Landauer equation Eq. (1.1) in which the current I flowing at any point in the device is given by [12]

$$I = (2e/h)MT[\mu_1 - \mu_2].$$

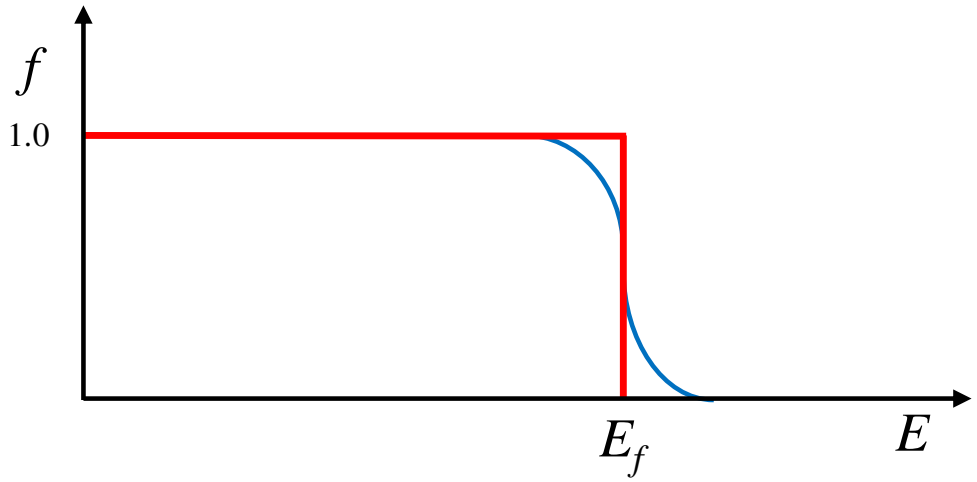

Figure 1.4

Fermi function in the degenerate limit.

This current equation shows that at zero temperature the transport is due entirely to the influx of electrons at the Fermi energy that is assumed constant over the range $\mu_1 > E > \mu_2$, where the transmission function $\overline{T}(E) = M(E)T(E)$ with $T(E)$ as the transmission probability and $M(E)$ as the number of transverse modes in the conductor.

As the temperature is increased from absolute zero, the effective transmission coefficient is drastically modified almost immediately. If inelastic scattering effects inside the device are not important, then the Landauer relation between conductivity and transmission remains valid provided T is replaced by $\int dE \left(-\frac{\partial f_0}{\partial E}\right) \overline{T}(E)$. Here, f_0 is the Fermi distribution at temperature T. The physics behind this is that the carriers are assumed to suffer no inelastic scattering inside the sample. Thus within the sample the temperature is effectively zero. However, the sources and sinks for carriers which are metallic conductors

attached to the ends of the device are in thermal equilibrium; hence the carriers available for conduction through the device, instead of being confined to E_f itself, have a thermal distribution around it given by $\partial f_0/\partial E$ [5].

The shift in the energy levels provide multiple energy paths and transport can take place through these paths. The Landauer formula for finite-temperature, also called multi-channel conduction through transverse modes, is given by

$$G = \frac{2e^2}{h} \int dE \left(-\frac{\partial f}{\partial E} \right) \overline{T}(E). \tag{1.30}$$

1.6 Green's Function Method

The Green's function method plays a vital role in condensed matter physics and particle physics. It is used in solving inhomogeneous boundary value problems and provides an effective method for analyzing the local density of states, conductance, quantum theory of scattering, and other transport-related properties of semi-conductors. In modern theoretical physics, Green's functions are used as propagators of Feynman diagrams. A brief introduction [12] to the Green's function and some of its properties will provide an understanding of the concepts of this methodology.

A response R due to an excitation S is related by means of a differential operator D_{op},

$$D_{op} R = S. \tag{1.31}$$

The response due to an excitation for instance, could be a change in the current due to a change in the potential. The Green's function can be defined in the form

$$R = D_{op}^{-1} S = GS, \tag{1.32}$$

where $G \equiv D_{op}^{-1}$. The case of a non-interacting transport problem can be expressed in the form

$$[E - H_{op}]\Psi = S, \tag{1.33}$$

where Ψ is the wave function and S is an equivalent excitation term due to a wave incident from one of the leads. The corresponding Green's function can be written as

$$G = [E - H_{op}]^{-1}, \tag{1.34}$$

where H_{op} is the Hamiltonian operator:

$$H_{op} \equiv \frac{(i\hbar\Delta)^2}{2m} + U(\mathbf{r}). \tag{1.35}$$

The inverse of a differential operator can be obtained when the boundary conditions are set, without which the system is an infinite one and the inverse of a differential operator of an open system without boundary conditions would be impossible since a real and proper solution is bounded.

As an example [12], a simple one-dimensional wire is considered with a constant potential energy U_0. From Eqs. (1.34, 1.35),

$$G = \left[E - U_0 + \frac{\hbar^2}{2m}\frac{\partial^2}{\partial x^2} \right]^{-1}, \tag{1.36}$$

which is

$$\left(E - U_0 + \frac{\hbar^2}{2m}\frac{\partial^2}{\partial x^2} \right) G(x, x') = \delta(x - x'), \tag{1.37}$$

similar to the Schrödinger equation

$$\left(E - U_0 + \frac{\hbar^2}{2m}\frac{\partial^2}{\partial x^2} \right) \Psi(x) = 0, \tag{1.38}$$

except for the source term $\delta(x - x^{'})$ on the right. The Green's function $G(x, x^{'})$ can be viewed as the wavefunction at x resulting from a unit excitation applied at $x^{'}$. Such an excitation gives rise to two waves traveling outwards from the point of excitation, with amplitudes A^{+} and A^{-} as shown in Fig. (1.6).

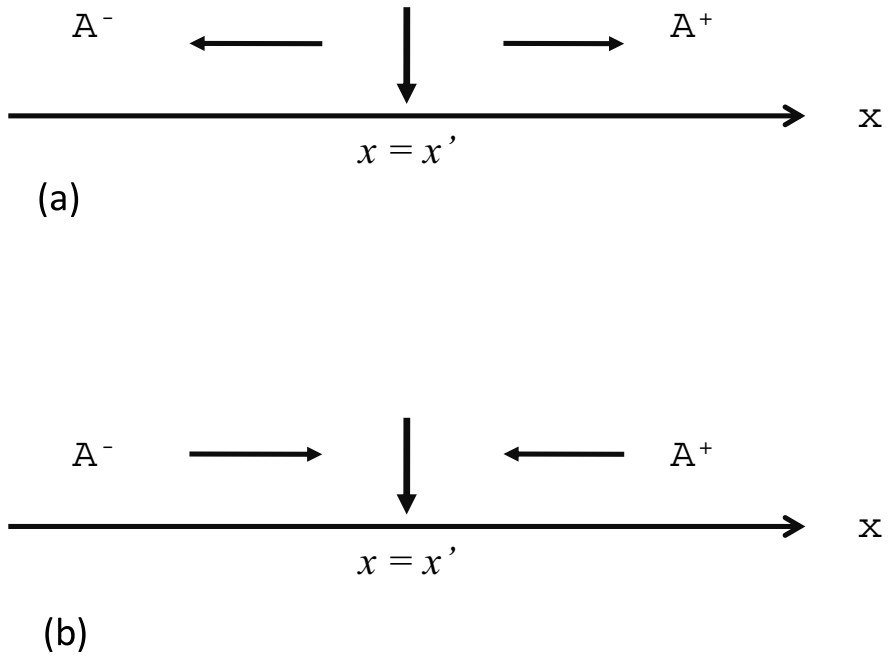

Figure 1.5

(a) Retarded and (b) advanced Green's function for infinite 1-D wire.

The solutions

$$G(x, x^{'}) = A^{+} \exp[ik(x - x^{'})], \quad x > x^{'}$$

$$G(x, x^{'}) = A^{-} \exp[-ik(x - x^{'})], \quad x < x^{'} \tag{1.39}$$

regardless of what A^+ and A^- might be, satisfy Eq. (1.37) at all points other than $x = x^{'}$. In order to satisfy Eq. (1.37) at $x = x^{'}$, the Green's function must be continuous

$$[G(x, x^{'})]_{x=x'+} = [G(x, x^{'})]_{x=x'-} \tag{1.40}$$

while the derivative must be discontinuous by $2m/\hbar^2$

$$\left[\frac{\partial G(x, x^{'})}{\partial x}\right]_{x=x'+} - \left[\frac{\partial G(x, x^{'})}{\partial x}\right]_{x=x'-} = \frac{2m}{\hbar^2}. \tag{1.41}$$

With the substitution of the solutions into the continuous and discontinuous conditions the Green's function is given by

$$\begin{aligned} G^A(x, x^{'}) &= +\frac{i}{\hbar v}\exp[-ik|x - x^{'}|] \\ G^R(x, x^{'}) &= -\frac{i}{\hbar v}\exp[ik|x - x^{'}|] \end{aligned} \tag{1.42}$$

called the advanced and retarded Green's function respectively. The former corresponds to incoming waves and the later corresponds to outgoing waves. They satisfy Eq. (1.37). To accommodate the boundary conditions, an imaginary infinitesimal term $\mathrm{i}\eta$ ($\eta > 0$) is added to the energy part in Eq. (1.37), resulting in

$$\left(E - U_0 + \frac{\hbar^2}{2m}\frac{\partial^2}{\partial x^2} + \mathrm{i}\eta\right)\mathrm{G^R(x, x^{'})} = \delta(\mathrm{x - x^{'}}), \tag{1.43}$$

then the associated wave number is

$$\begin{aligned} \kappa &= \frac{\sqrt{2m(E + \mathrm{i}\eta - \mathrm{U_0})}}{\hbar} \\ &\approx \frac{\sqrt{2m(E - U_0)}}{\hbar}\left[1 + \frac{\mathrm{i}\eta}{2(E - U_0)}\right] \\ &\equiv k(1 + \mathrm{i}\delta). \end{aligned} \tag{1.44}$$

This imaginary part makes the advanced function grow indefinitely moving away from the source of excitation which is divergent and therefore not an acceptable solution. Hence the retarded function is accepted as the solution, since it is convergent (or bounded). A similar argument for the advanced function makes it the only acceptable solution of the equation

$$\left(E - U_0 + \frac{\hbar^2}{2m}\frac{\partial^2}{\partial x^2} - \mathrm{i}\eta\right)\mathrm{G^A}(\mathrm{x},\mathrm{x}^{'}) = \delta(\mathrm{x}-\mathrm{x}^{'}). \tag{1.45}$$

Thus a general Green's function is given as

$$G^{\pm} = [E - H_{op} \pm \mathrm{i}\eta]^{-1}, \tag{1.46}$$

where $G^{+} = G^{A}$ is the advanced Green's function, and $G^{-} = G^{R}$ is the retarded Green's function. The transmission in terms of the Green's function is given as

$$\overline{T} \equiv Tr[\Gamma_1 G^{-}\Gamma_2 G^{+}] = Tr[\Gamma_2 G^{-}\Gamma_1 G^{+}], \tag{1.47}$$

where $\Gamma_1 = \mathrm{i}[\sum_1^- - \sum_1^+]$ is the coupling of the conductor to the lead 1, and $\Gamma_2 = \mathrm{i}[\sum_2^- - \sum_2^+]$ is the coupling of the conductor to lead 2. The terms $\sum^{\pm}$ are the advanced and retarded self energy terms which are Hermitian conjugates of each other.

1.7 Method of Finite Differences

The method of finite differences is another approach in obtaining a numerical solution of the Schrödinger equation, and subsequently deals with transport and transport properties in the regime of quantum effects. The steps are [13]

1. Convert an infinite dimensional open system, such as the infinite leads, to one of a finite dimensional closed system with boundaries.

2. Choose discrete lattice points at $x = na$, where n is an integer, and a, the lattice constant,

3. Convert the Hamiltonian operator H_{op} into a matrix $[H]$.

4. Convert the wave function $\Psi(x)$ into a column vector $\{\psi\}$.

The Schrödinger equation gets converted from a partial differential equation into a matrix equation:

$$i\hbar\frac{\partial}{\partial x}\Psi(x) = H_{op}\Psi(x) \quad \longrightarrow \quad i\hbar\frac{d}{dx}\{\psi(x)\} = [H]\{\psi(x)\}. \tag{1.48}$$

The matrix representation for a 1-D system is obtained by considering the Hamiltonian operator

$$H_{op} = -\frac{\hbar^2}{2m}\frac{d^2}{dx^2} + U(x) \tag{1.49}$$

and a discrete lattice (as in Fig. 2.2) whose points are located at $x = na$, n being an integer. The matrix can be written as

$$[H_{op}\psi]_{x=na} = \left[-\frac{\hbar^2}{2m}\frac{d^2\psi}{dx^2}\right]_{x=na} + U_n\psi_n, \tag{1.50}$$

where $\psi_n \rightarrow \psi(x = na)$, $U_n \rightarrow U(x = na)$. The method of finite differences can be applied to approximate the operator $d^2\psi/dx^2$ as

$$\frac{d^2\psi}{dx^2} \rightarrow \frac{1}{a}[\psi_{n+1} - 2\psi_n + \psi_{n-1}]. \tag{1.51}$$

Then

$$[H_{op}\psi]_{x=na} = (U_n + 2t)\psi_n - t\psi_{n-1} - t\psi_{n=1}, \tag{1.52}$$

where $t \equiv \hbar^2/2ma^2$. The above equation can be written as

$$[H_{op}\psi]_{x=na} = \sum[(U_n + 2t)\delta_{n,m} - t\delta_{n,m+1} - t\delta_{n,m-1}], \tag{1.53}$$

where $\delta_{n,m}$ is the Kronecker delta, which is one if n and m are equal and zero otherwise. Thus the elements of the matrix H operator for a 1-D linear chain can be written as

$$H = \begin{pmatrix} \dots & -t & 0 & 0 & 0 \\ -t & U_{-1} + 2t & -t & 0 & 0 \\ 0 & -t & U_0 + 2t & t & 0 \\ 0 & 0 & -t & U_1 + 2t & -t \\ 0 & 0 & 0 & -t & \dots \end{pmatrix}. \tag{1.54}$$

Each site is linked to its nearest neighbor by the element t, while elements along the diagonal are the potential energy U_n and $2t$. The dispersion relation for the 1-D discrete lattice with a constant potential $U(x) = U_0$ is obtained using the plane wave eigenfunctions $\psi_k(x) = \exp[ikx]$ and $E = U_0 + (\hbar^2 k^2)/2m$. The Schrödinger equation can be written using equation Eq. (1.52) as

$$E\psi_n = (U_0 + 2t)\psi_n - t\psi_{n-1} - t\psi_{n+1}. \tag{1.55}$$

This is satisfied by the solution of the form $\psi_n = \exp[ikx_n]$, where $x_n = na$, and a is the lattice constant. Substituting these wavefunctions into Eq. (1.55) results in

$$E = U_0 + 2t\,(1 - \cos(ka))\,, \tag{1.56}$$

which is the dispersion relation for the 1-D lattice. This condition imposes on E the 1-D energy band limits

$$U_0 \leq E \leq U_0 + 4t. \tag{1.57}$$

1.8 The Tight Binding Approximation

The Schrödinger equation even in one dimension provides few analytic solutions. Therefore numerical methods are sought to solve most problems. Most often the 1-D discrete Schrödinger equation is presented as the inferior numerical approximation to the 'true' continuous equation, which is exact only in the limit of infinitesimal lattice spacing. While such a position is correct mathematically, the continuous Schrödinger equation is not the most physically reasonable choice for realistic modeling of semiconductor quantum wells and other nanoelectronic devices because when applied to semiconductor quantum wells the continuous Schrödinger equation is actually an equation for the wavefunction envelope and therefore its solutions are meaningless at lengths smaller than the lattice constant [7]. The discrete modeling is more appropriate in portraying the physics of a crystal than is the continuous model. Discrete Schrödinger picture is provided by the equivalent tight-binding model for a crystalline solid [6]. Numerical results for both continuous and discrete models cannot be interpreted without the analytical results. Simple 1-D time-independent quantum mechanical models are based on Eq. (1.2), where m represents the effective mass and ψ the wave function. Equation (1.2) must be solved numerically for all but the simplest potentials U. This numerical solution is achieved using the standard finite difference method or equivalently the central difference formula for the second derivative. The difference equation is obtained by considering the spatial variation of the wave function ψ_j and using Taylor expansions.

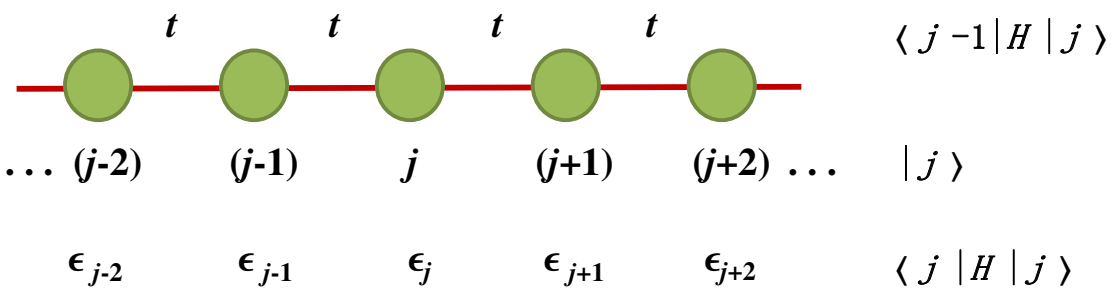

Figure 1.6

Section of infinite chain of atoms, each with an s-like orbital.

Fig. (1.6) shows a section of an infinite chain of atoms, each with an s-like orbital, of lattice spacing a. The top row shows the nearest neighbor matrix element t and the kets $|j\rangle$ represent one orbital per atom. The difference equation used is

$$\frac{d^2\psi}{dx^2} \approx \frac{1}{a^2}[\psi(x-a) - 2\psi(x) + \psi(x+a)], \tag{1.58}$$

where a is the lattice space constant. Evaluating the functions and derivatives at points $x_j = ja$ gives

$$\left(\frac{\hbar^2}{2ma^2}\right)\psi_{j-1} + \left[\frac{\hbar^2}{ma^2} + U_j - E\right]\psi_j + \left(-\frac{\hbar^2}{2ma^2}\right)\psi_{j+1} = 0, \tag{1.59}$$

where $U_n = U(x_n)$, $\psi_n = \psi(x_n)$. Eq. (1.59) is essentially the tight-binding approximation

$$H = \sum_i \epsilon_i(|i\rangle\langle i|) + \sum_{\langle ij\rangle} t_{ij}(|i\rangle\langle j| + |j\rangle\langle i|), \tag{1.60}$$

where ϵ_i and t_{ij} are the are the on-site atomic energy and the overlap integrals of the atomic orbitals at sites i and j, respectively. The tight-binding hopping matrix elements t_{ij}, also called the perturbative potential, are usually assumed to be non-zero only between

pairs of nearest-neighbor lattice sites $\langle ij \rangle$. Located on sites i and j are the tight-binding basis functions $|i\rangle$ and $|j\rangle$ respectively, applied to a chain of atoms with spacing a and one orbital per atom (site) shown in Fig.(1.6).

The tight-binding Hamiltonian is another way to calculate the electron dispersion relation in periodic potentials. It is widely used to describe the electronic band structure in condensed matter. Unlike the free electron theory, the tight-binding method describes the electronic structure in terms of localized atomic orbitals which overlap due to bonding between the neighboring valance atoms [16, 37]. The propagating Bloch states, which are responsible for electronic transport in metals, can be built up from the atomic orbitals by solving the appropriate Schrödinger equation. The tight-binding approach is very suited to numerical calculations of the conductance because it discretizes the spatial continuum in terms of the atomic sites.

Noting that solids are made up of atoms, the electron moves locally in the lowest energy s-state or the s-atomic orbital of an isolated atom. The electronic structure of a periodic array of such atoms is then developed by allowing a small overlap of electronic wavefunctions between adjacent atoms. The Hamiltonian is represented in a basis of singly occupied states, a subset of states of Fock space where all states are multiply occupied. Its matrix representation is easy to construct. The simplest is the tridiagonal matrix in the position-occupation basis. Using this tight-binding formalism for an infinite 1-D linear

chain of atoms (Fig. 1.6) with site-labeling in discrete space, the left-hand side part of the Schrödinger equation $(H - E)\psi = 0$ can be written in the site basis:

$$\begin{pmatrix} \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \\ \dots & \epsilon_{-2} - E & t & 0 & 0 & 0 & 0 & \dots \\ \dots & t & \epsilon_{-1} - E & t & 0 & 0 & 0 & \dots \\ \dots & 0 & t & \epsilon - E & t & 0 & 0 & \dots \\ \dots & 0 & 0 & t & \epsilon_1 - E & t & 0 & \dots \\ \dots & 0 & 0 & 0 & t & \epsilon_2 - E & t & \dots \\ \dots & 0 & 0 & 0 & 0 & t & \epsilon_3 - E & \dots \\ & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix}. \tag{1.61}$$

1.9 Matrix Formalism

Most descriptions of electronic transport through nanosystems use the Green's function formalism. The matrix method is another approach in obtaining numerical solution to the Schrödinger equation. Daboul $et\ al.$ [11] have described the matrix method in quantum percolation studies. The method transforms the infinitely sized Hamiltonian matrix into a reduced matrix that is finite and involves semi-infinite linear chains of atoms using an ansatz. The method uses the tight-binding Hamiltonian and Bloch type wave functions. This approach provides less mathematical sophistication than its counterpart, the Green's function, which involves many terms in the transport calculations that obscures the real physics. In this approach to studying transport problems, matrix algebra is used for the same reason it is for a transfer matrix calculation [21, 29, 30, 33, 34] in statistical physics,

that is to study a system with an infinite number of particles but where computations only using a finite matrix is sought. In statistical physics the transfer matrix calculations will ultimately lead to the calculation of the trace of an infinite matrix. And for isotropic systems the problem is one of simply finding the trace of a power of a finite-dimensional matrix [31, 32]. For electronic transport problems the inverse of an infinite matrix are required. The system in this approach will reduce to the calculation of the inverse of a finite matrix that will provide the wavefunctions for an infinite number of values of the position x. Then the numerical calculation of the probability of transmission and reflection would be possible.

The basis of the matrix formalism is in considering the discrete lattice structure, incorporating the tight-binding model, also called the finite difference method.

Consider a system where two atoms a and b located at $x = 0$, called a "blob" whose wavefuntions are ψ_a and ψ_b respectively. Attached to the blob on the left at $x = -1$ are the semi-infinite 1-D linear chain of atoms labeled $x = -\infty, \ldots, -4, -3, -2, -1$, and connected to the blob on right at $x = 1$ is the semi-infinite chain of atoms labeled $x = 1,\ 2,\ 3, \ldots \infty$ (Fig. 1.7). The Hamiltonian of the whole system includes the input and output chains of atoms, as well as the blob atoms.

The Schrödinger equation $\mathcal{H}_\infty(x)\Psi(x) = E\,\Psi(x)$ can be written in matrix form where the random hopping integral t is given a value of 1 (or -1 only in the 1-dimensional case), and the onsite energy $\epsilon = 0$ for each site in the semi-infinite chains of atoms except

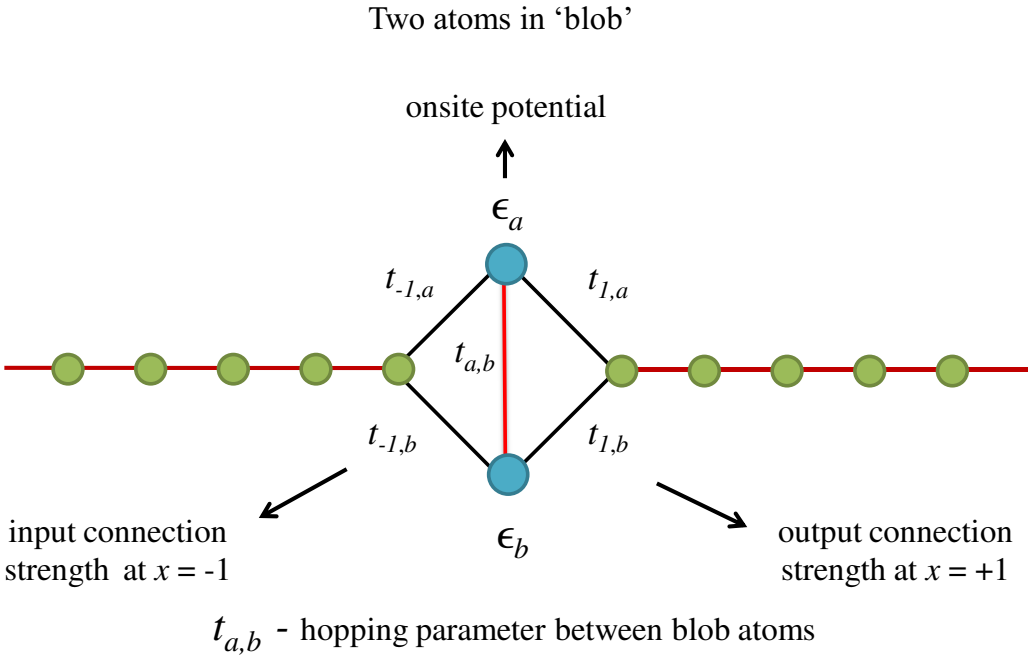

Figure 1.7

Two-site blob with input and output semi-infinite chains of atoms in 1-D.

for the sites in the blob whose onsite energies are ϵ_a and ϵ_b. The Schrödinger equation $(\mathcal{H} - E)\vec{\Psi} = 0$ in matrix form is

$$\begin{pmatrix}
\ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \\
\cdots & -E & 1 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\
\cdots & 1 & -E & 1 & 0 & 0 & 0 & 0 & 0 & \cdots \\
\cdots & 0 & 1 & -E & t_{-1,a} & t_{-1,b} & 0 & 0 & 0 & \cdots \\
\cdots & 0 & 0 & t_{-1,a} & \epsilon_a - E & t_{a,b} & t_{a,1} & 0 & 0 & \cdots \\
\cdots & 0 & 0 & t_{-1,b} & t_{a,b} & \epsilon_b - E & t_{b,1} & 0 & 0 & \cdots \\
\cdots & 0 & 0 & 0 & t_{a,1} & t_{b,1} & -E & 1 & 0 & \cdots \\
\cdots & 0 & 0 & 0 & 0 & 0 & 1 & -E & 1 & \cdots \\
\cdots & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -E & \cdots \\
 & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots
\end{pmatrix}
\begin{pmatrix} \vdots \\ \psi_{-3} \\ \psi_{-2} \\ \psi_{-1} \\ \psi_a \\ \psi_b \\ \psi_1 \\ \psi_2 \\ \psi_3 \\ \vdots \end{pmatrix}
= \begin{pmatrix} \vdots \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \vdots \end{pmatrix}, \tag{1.62}$$

where $\vec{\Psi}(x)$ is the column vector of site wavefunctions. The Hamiltonian is an infinite-dimensional matrix. In Sec. (2.2) the solution to this matrix will be reduced to the solution for an inverse of a finite-dimensional matrix.

CHAPTER 2

TRANSMISSION CALCULATIONS

2.1 Transmission Coefficient Calculations from (n + 2) Matrix

The Schrödinger equation in matrix form (Eq. 1.62) for a system that includes the input wire, the blob of two atoms, and the output wire, is an infinite-dimensional matrix equation that is to be solved for the column vector $\{\Psi\}$. The equation has to be reduced first to a finite-dimensional one so that the inverse and hence the solution could be determined numerically. Electrons are sent from $x = -\infty$ so the wavefunction for $x < 0$ are the incident wave and reflected wave with amplitude r, represented by the ansatz [11] (physical guess)

$$\psi_{-(n+1)} = e^{-i\,n\,q} + re^{i\,n\,q} \tag{2.1}$$

and the ansatz for the transmitted wave traveling to $x = \infty$ is

$$\psi_{(n+1)} = t_T e^{inq} \tag{2.2}$$

where t_T is the amplitude. The sites of the system are now labeled $n = 0, 1, 2, \ldots \infty$ implying discretization of the one-dimensional space. These physical ansatz can also be written for the negative values with $m = -n - 1$ for $x < 0$ as

$$\psi_m = e^{i(m+1)q} + re^{-i(m+1)q} \qquad m = -\infty, \cdots, -2, -1 \tag{2.3}$$

and for the positive values with $m = n + 1$ as

$$\psi_m = t_T e^{i(m-1)q} \qquad m = 1, 2, \cdots, +\infty. \tag{2.4}$$

This wavefunction with unit amplitude traveling in from $x = -\infty$, is partially reflected with an amplitude r to $x = -\infty$, and partially transmitted with amplitude t_T to $x = +\infty$. The energy E of the injected electron is the same at any location in the system, which is

$$e^{iq} + e^{-iq} = E \;\; \leftrightarrow \;\; E = 2\cos(q) \;\; \leftrightarrow \;\; q = \arccos\left(\frac{E}{2}\right). \tag{2.5}$$

This condition imposes a limit on the energy E of the incoming electron of

$$-2 \leq E \leq 2. \tag{2.6}$$

This is the dispersion relation of Eq. (1.57) with $t = 1$ and $U_0 = -2t$ so $\epsilon = U_0 + 2t = 0$. In matrix notation for the two-site blob and the two semi-infinite leads the *ansatz* is

$$\begin{pmatrix} \vdots \\ \psi_{-3} \\ \psi_{-2} \\ \psi_{-1} \\ \psi_a \\ \psi_b \\ \psi_1 \\ \psi_2 \\ \psi_3 \\ \vdots \end{pmatrix} = \begin{pmatrix} \vdots \\ e^{i(-2)q} + re^{-i(-2)q} \\ e^{i(-1)q} + re^{-i(-1)q} \\ e^{i0q} + re^{-i0q} \\ \psi_a \\ \psi_b \\ t_T e^{0iq} \\ t_T e^{1iq} \\ t_T e^{2iq} \\ \vdots \end{pmatrix} = \begin{pmatrix} \vdots \\ e^{-2iq} + re^{2iq} \\ e^{-1iq} + re^{1iq} \\ 1 + r \\ \psi_a \\ \psi_b \\ t_T \\ t_T e^{1iq} \\ t_T e^{2iq} \\ \vdots \end{pmatrix}. \tag{2.7}$$

Substituting Eq. (2.7) in Eq. (1.62) gives the matrix equation

$$\begin{pmatrix}
\ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \\
\cdots & -E & 1 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\
\cdots & 1 & -E & 1 & 0 & 0 & 0 & 0 & 0 & \cdots \\
\cdots & 0 & 1 & -E & t_{-1,a} & t_{-1,b} & 0 & 0 & 0 & \cdots \\
\cdots & 0 & 0 & t_{-1,a} & \epsilon_a - E & t_{a,b} & t_{a,1} & 0 & 0 & \cdots \\
\cdots & 0 & 0 & t_{-1,b} & t_{a,b} & \epsilon_b - E & t_{b,1} & 0 & 0 & \cdots \\
\cdots & 0 & 0 & 0 & t_{a,1} & t_{b,1} & -E & 1 & 0 & \cdots \\
\cdots & 0 & 0 & 0 & 0 & 0 & 1 & -E & 1 & \cdots \\
\cdots & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -E & \cdots \\
 & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots
\end{pmatrix}
\begin{pmatrix}
\vdots \\
e^{-2iq} + re^{2iq} \\
e^{-1iq} + re^{1iq} \\
1 + r \\
\psi_a \\
\psi_b \\
t_T \\
t_T e^{1iq} \\
t_T e^{2iq} \\
\vdots
\end{pmatrix}
=
\begin{pmatrix}
\vdots \\
0 \\
0 \\
0 \\
0 \\
0 \\
0 \\
0 \\
0 \\
\vdots
\end{pmatrix}. \tag{2.8}$$

From the product on the left side of the above matrix for sites $n > 1$ $(m < -2)$ for the incoming sites (negative) there is

$$\begin{aligned}
\psi_{m-1} - E\psi_m + \psi_{m+1} &= e^{i(m+1)q}\left[e^{-iq} - E + e^{iq}\right] + re^{-i(m+1)q}\left[e^{iq} - E + e^{-iq}\right] \\
&= e^{i(m+1)q}\left[\,0\,\right] + re^{-i(m+1)q}\left[\,0\,\right] = 0\,.
\end{aligned} \tag{2.9}$$

Similarly for the sites with $n > 1$ ($m < -2$) for the outgoing sites (positive) this gives

$$\begin{aligned}
\psi_{m-1} - E\psi_m + \psi_{m+1} &= t_T e^{i(m-2)q} - t_T E e^{i(m-1)q} + t_T e^{i(m)q} \\
&= t_T e^{i(m-1)q}\left[e^{-iq} - E + e^{iq}\right] \\
&= t_T e^{i(m-1)q}\left[\,0\,\right] = 0.
\end{aligned} \tag{2.10}$$

Hence the equation $(\mathcal{H}_\infty - E)\Psi = 0$ is satisfied everywhere except (so far) in the central part of the matrix that pertains to the blob atoms. The central part of the matrix requires that

$$\begin{pmatrix} 1-E & \vec{w}^{\mathrm{T}} & 0 & 0 \\ \vec{0} & \vec{w} & \mathcal{H} - E\mathbf{I} & \vec{u} & \vec{0} \\ 0 & 0 & \vec{u}^{\mathrm{T}} & -E1 \end{pmatrix} \begin{pmatrix} e^{-iq} + re^{iq} \\ 1 + r \\ \vec{\psi} \\ t_T \\ t_T e^{iq} \end{pmatrix} = \begin{pmatrix} e^{-iq} + re^{iq} - E - rE + \vec{w}^{\mathrm{T}}\vec{\psi} \\ \vec{w} + r\vec{w} + \mathcal{H}\vec{\psi} - E\mathbf{I}\vec{\psi} + t_T\vec{u} \\ \vec{u}^{\mathrm{T}}\vec{\psi} - Et_T + t_T e^{iq} \end{pmatrix} = \begin{pmatrix} 0 \\ \vec{0} \\ 0 \end{pmatrix}. \tag{2.11}$$

Here $\mathcal{H}$ is the portion of the Hamiltonian that governs the interactions within the blob in terms of the onsite energies ϵ and hopping parameters t, both ϵ and t are taken to be real numbers. The size of the matrix $\mathcal{H}$ is equal to the number of sites within the blob. $\mathbf{I}$ is the identity matrix of the same dimension. The vectors $\vec{w}$ and $\vec{u}$ and their transposes also have the same dimension as the dimension of $\mathcal{H}$. The vector $\vec{w}$ that couples the site at $x = -1$ to the blob is

$$\vec{w} = \begin{pmatrix} t_{-1,a} \\ t_{-1,b} \\ \vdots \end{pmatrix} \tag{2.12}$$

and the vector $\vec{u}$ (standing for the word *ut* in Norwegian, which means out) that couples the site at $x = +1$ to the blob is

$$\vec{u} = \begin{pmatrix} t_{1,a} \\ t_{1,b} \\ \vdots \end{pmatrix}. \tag{2.13}$$

The equation Eq. (2.11) can be written as

$$\begin{pmatrix} -E+e^{iq} & \vec{w}^{\mathrm{T}} & 0 \\ \vec{w} & \mathcal{H}-E\mathbf{I} & \vec{u} \\ 0 & \vec{u}^{\mathrm{T}} & -E+e^{iq} \end{pmatrix} \begin{pmatrix} 1+r \\ \vec{\psi} \\ t_T \end{pmatrix} - \begin{pmatrix} e^{iq}-e^{-iq} \\ \vec{0} \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ \vec{0} \\ 0 \end{pmatrix}, \quad (2.14)$$

which when multiplied out results in

$$\begin{pmatrix} e^{-iq}+re^{iq}-E-rE+\vec{w}^{\mathrm{T}}\vec{\psi} \\ \vec{w}+r\vec{w}+\mathcal{H}\vec{\psi}-E\mathbf{I}\vec{\psi}+t_T\vec{u} \\ \vec{u}^{\mathrm{T}}\vec{\psi}-Et_T+t_Te^{iq} \end{pmatrix} = \begin{pmatrix} 0 \\ \vec{0} \\ 0 \end{pmatrix}, \quad (2.15)$$

which is the same as Eq. (2.11).

The equation

$$\begin{pmatrix} -E+e^{iq} & \vec{w}^{\mathrm{T}} & 0 \\ \vec{w} & \mathcal{H}-E\mathbf{I} & \vec{u} \\ 0 & \vec{u}^{\mathrm{T}} & -E+e^{iq} \end{pmatrix} \begin{pmatrix} 1+r \\ \vec{\psi} \\ t_T \end{pmatrix} = \begin{pmatrix} e^{iq}-e^{-iq} \\ \vec{0} \\ 0 \end{pmatrix}. \quad (2.16)$$

is thus the key equation to solve for r and t_T. This finite and reduced matrix whose dimension is equal to the number of sites n in the blob plus 2 is the key to solve the Schrödinger equation and hence determine the coefficients of reflection $R = |r|^2$ and transmission $T = |t_T|^2$. By conservation of the number of electrons, the relationship $R + T = |r|^2 + |t_T|^2 = 1$ must always hold. The solution also yields the eigenvector $\vec{\psi}$ within the blob.

2.2 The *n*-Matrix Transmission

The equation to solve is Eq. (2.16), which can be solved formally as

$$\begin{pmatrix} 1+r \\ \vec{\psi} \\ t_T \end{pmatrix} = \left[\begin{pmatrix} -E+e^{iq} & \vec{w}^{\mathrm{T}} & 0 \\ \vec{w} & \mathcal{H}-E\mathbf{I} & \vec{u} \\ 0 & \vec{u}^{\mathbf{T}} & -E+e^{iq} \end{pmatrix} \right]^{-1} \begin{pmatrix} e^{iq}-e^{-iq} \\ \vec{0} \\ 0 \end{pmatrix}. \tag{2.17}$$

The energy is $E = e^{-iq} + e^{iq} = 2\cos(q)$ with $-2 \leq E \leq 2$. Also note that

$$-E + e^{\pm iq} = -e^{\mp iq}. \tag{2.18}$$

The wave vector q can be regarded as an angle, with

$$\cos(q) = \frac{E}{2} \tag{2.19}$$

and

$$\sin(q) = \frac{\sqrt{4-E^2}}{2}. \tag{2.20}$$

Thus one can use

$$e^{\pm iq} = \cos(q) \pm i\sin(q) = \frac{E}{2} \pm i\frac{\sqrt{4-E^2}}{2}. \tag{2.21}$$

Define

$$S = -E + e^{iq} = -E + \frac{E}{2} + i\frac{\sqrt{4-E^2}}{2} = -\frac{E}{2} + i\frac{\sqrt{4-E^2}}{2}. \tag{2.22}$$

Then Eq. (2.17) can be shown by direct matrix multiplication to be

$$\left[\begin{pmatrix} S & \vec{w}^{\mathrm{T}} & 0 \\ \vec{w} & \mathcal{H}-E\mathbf{I} & \vec{u} \\ 0 & \vec{u}^{\mathbf{T}} & S \end{pmatrix} \right]^{-1} = \begin{pmatrix} \frac{1}{S} + \frac{1}{S^2}\vec{w}^{\mathrm{T}}\mathbf{L}\vec{w} & -\frac{1}{S}\vec{w}^{\mathrm{T}}\mathbf{L} & \frac{1}{S^2}\vec{w}^{\mathrm{T}}\mathbf{L}\vec{u} \\ -\frac{1}{S}\mathbf{L}\vec{w} & \mathbf{L} & -\frac{1}{S}\mathbf{L}\vec{u} \\ \frac{1}{S^2}\vec{u}^{\mathrm{T}}\mathbf{L}\vec{w} & -\frac{1}{S}\vec{u}^{\mathrm{T}}\mathbf{L} & \frac{1}{S} + \frac{1}{S^2}\vec{u}^{\mathrm{T}}\mathbf{L}\vec{u} \end{pmatrix} \tag{2.23}$$

with the definition of the matrix

$$\mathbf{L} = \left[\mathcal{H} - E\mathbf{I} - \frac{1}{S}\vec{u}\vec{u}^{\mathrm{T}} - \frac{1}{S}\vec{w}\vec{w}^{\mathrm{T}}\right]^{-1}. \tag{2.24}$$

Multiplying through the general form of the matrix inverse in Eq. (2.17) gives the result

$$\begin{pmatrix} 1+r \\ \vec{\psi} \\ t_T \end{pmatrix} = \begin{pmatrix} \frac{1}{S} + \frac{1}{S^2}\vec{w}^{\mathrm{T}}\mathbf{L}\vec{w} & -\frac{1}{S}\vec{w}^{\mathrm{T}}\mathbf{L} & \frac{1}{S^2}\vec{w}^{\mathrm{T}}\mathbf{L}\vec{u} \\ -\frac{1}{S}\mathbf{L}\vec{w} & \mathbf{L} & -\frac{1}{S}\mathbf{L}\vec{u} \\ \frac{1}{S^2}\vec{u}^{\mathrm{T}}\mathbf{L}\vec{w} & -\frac{1}{S}\vec{u}^{\mathrm{T}}\mathbf{L} & \frac{1}{S} + \frac{1}{S^2}\vec{u}^{\mathrm{T}}\mathbf{L}\vec{u} \end{pmatrix} \begin{pmatrix} 2i\sin(q) \\ \vec{0} \\ 0 \end{pmatrix} \tag{2.25}$$

or

$$\begin{pmatrix} 1+r \\ \vec{\psi} \\ t_T \end{pmatrix} = \begin{pmatrix} \frac{2i\sin(q)}{S} + \frac{2i\sin(q)}{S^2}\vec{w}^{\mathrm{T}}\mathbf{L}\vec{w} \\ -\frac{2i\sin(q)}{S}\mathbf{L}\vec{w} \\ \frac{2i\sin(q)}{S^2}\vec{u}^{\mathrm{T}}\mathbf{L}\vec{w} \end{pmatrix}. \tag{2.26}$$

The finite matrix to solve has now been reduced in size by two. The only requirement is to solve the matrix $\mathbf{L}$ analytically or numerically finding the inverse of a matrix of size n equal to the number of sites in the blob. From Eq. (2.26), the transmission coefficient [35] is

$$T = |t_T|^2 = \left|\frac{2i\sin(q)}{S^2}\vec{u}^{\mathrm{T}}\mathbf{L}\vec{w}\right|^2. \tag{2.27}$$

2.3 Fully Connected Blob, $n > 2$

Consider a blob with n sites located at $x = 0$, in which all sites are identical and $n > 1$. Assume an on-site energy ϵ and all blob sites are equally connected to the input lead at site $x = -1$ with strength t_w and to the output lead at site $x = +1$ with strength t_u. Introduce

a vector $\vec{e}$ of length n whose elements are all 1, the identity matrix $\mathbf{I}$ of size $n \times n$, and the $n \times n$ matrix $\mathbf{J}$ whose elements are all 1. For example, for $n = 3$

$$\vec{e} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}, \quad \mathbf{I} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad \mathbf{J} = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}. \tag{2.28}$$

Relations $\mathbf{J}\tilde{\mathrm{e}} = n\tilde{\mathrm{e}}$, $\vec{e}\vec{e}^T = \mathbf{J}$ and $\mathbf{J}^2 = n\mathbf{J}$ can be proved by matrix multiplication. The input and output coupling vectors in Eq. (2.26) in terms of $\vec{e}$, and coupling strength parameter can be written as

$$\vec{w} = t_w \vec{e}, \qquad \vec{u} = t_u \vec{e}. \tag{2.29}$$

The matrix equation Eq. (2.26) in original form is $\mathcal{H} - E\mathbf{I} = (\epsilon - E - t)\mathbf{I} + t\mathbf{J}$, where each of the n blob sites is coupled with a hopping parameter t to each of the other $n - 1$ blob sites. As an example for $n = 3$ they are

$$\vec{w} = t_w \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}, \qquad \vec{u} = t_u \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}, \qquad \mathcal{H} - E\mathbf{I} = \begin{pmatrix} \epsilon - E & t & t \\ t & \epsilon - E & t \\ t & t & \epsilon - E \end{pmatrix}. \tag{2.30}$$

Eq. (2.24) is now written as

$$\mathbf{L}^{-1} = (\epsilon - E - t)\mathbf{I} + [t - S^*(t_w^2 + t_u^2)]\mathbf{J} \tag{2.31}$$

where $S = -E + \exp(iq) = -\frac{E}{2} + i\frac{\sqrt{4-E^2}}{2}$ defined in Eq. (2.22). The matrix $\mathbf{L}$ can be written as

$$\mathbf{L} = X_I \mathbf{I} + X_J \mathbf{J}, \tag{2.32}$$

where $X_I = (\epsilon - E - t)$ and $X_J = t - S^*(t_w^2 + t_u^2)$. Its inverse is

$$\mathbf{L}^{-1} = \left(\frac{1}{X_I}\mathbf{I} + Y_J\mathbf{J}\right), \tag{2.33}$$

then

$$(X_I\mathbf{I} + X_J\mathbf{J})\left(\frac{1}{X_I}\mathbf{I} + Y_J\mathbf{J}\right) = \mathbf{I} \tag{2.34}$$

provided that

$$Y_J = -\frac{X_J}{X_I(X_I + nX_J)}, \tag{2.35}$$

which when substituted in Eq. (2.34) along with the properties $\mathbf{I}^2 = \mathbf{I}$, $\mathbf{JI} = \mathbf{IJ} = \mathbf{J}$, and $\mathbf{J}^2 = n\mathbf{J}$, results in

$$\left[\frac{X_J}{X_I} + X_JY_Jn + X_IY_J\right]\mathbf{J} = \left[\frac{X_J}{X_I} + Y_J\left(X_Jn + X_I\right)\right]\mathbf{J} = \mathbf{0}. \tag{2.36}$$

The matrix cannot be singular ($X_I \neq 0$) for a plausible solution. Therefore the inverse $\mathbf{L}^{-1}$ provides a general solution for any number of sites n. The solution for the general n site case using Eq. (2.34) is

$$\mathbf{L} = \frac{1}{\epsilon - E - t}\mathbf{I} - \frac{t - S^*(t_w^2 + t_u^2)}{(\epsilon - E - t)[\epsilon - E + (n-1)t - nS^*(t_w^2 + t_u^2)}\mathbf{J}. \tag{2.37}$$

Thus the solution [35] to find for the transmission is,

$$t_T = 2i(S^*)^2\sin(q)\vec{u}^{\mathrm{T}}\mathbf{L}\vec{w} = 2i(S^*)^2t_wt_u\sin(q)\vec{e}^{\mathrm{T}}\mathbf{L}\vec{e}. \tag{2.38}$$

Using the properties $\mathbf{J}\vec{e} = n\vec{e}$, $\vec{e}^{\mathrm{T}}\vec{e} = n$, and $\vec{e}^{\mathrm{T}}\mathbf{I}\vec{e} = n$ in the transmission amplitude t_T above yields

$$t_T = 2i(S^*)^2\sin(q)t_wt_u\left[\frac{n}{\epsilon - E - t} - \frac{n^2\left[t - S^*(t_w^2 + t_u^2)\right]}{(\epsilon - E - t)[\epsilon - E + (n+1)t - nS^*(t_w^2 + t_u^2)]}\right]. \tag{2.39}$$

This solution is valid for $n > 2$, a non-singular matrix so $\epsilon - E - t \neq 0$, and then the physically measurable probability $T = |t_T|^2$.

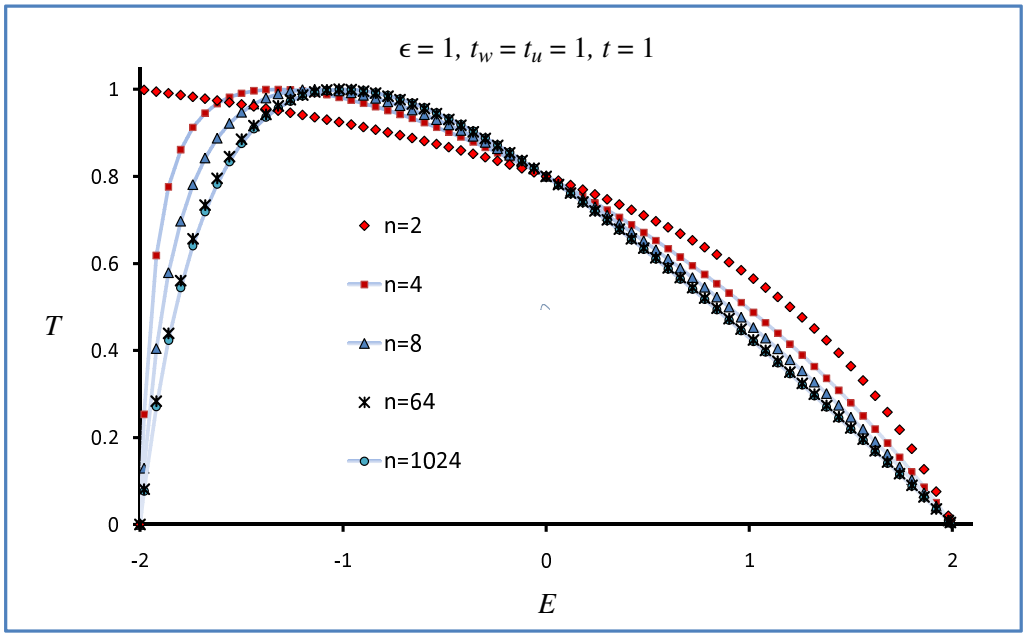

Figure 2.1

Transmission vs. Energy for n-site blob with $\epsilon = t = t_w = t_u = 1$.

Fig. (2.1) illustrates how transmission changes with the number of atoms in the blob. For more than about four atoms in the blob the transmission does not significantly change. The argument for such a behavior is that as more atoms are added to the blob the sum of input and output hopping strengths $t_w = t_u = 1$ increases accordingly and so does the sum of strengths of the intra-blob bond hopping parameters t.

2.4 Decimation Renormalization Group - Matrix

One of the most basic themes in theoretical physics is the idea that nature is described locally. The basic equations of all physics is local. In order to be able to specify local equations it is necessary to define continuum limits, including the limits that define derivatives. The idea of the derivative and a continuum is important in all of physics. A group of continuum limits is called the statistical continuum limit that has a very broad range of applications in physics. The functions of a continuous variable are themselves independent in a statistical continuum limit. If the continuum were to be replaced by discrete lattice points, the field averages would consist of integrals over the value of the field at each lattice site n. Thus for the discrete lattice case one has a multiple integration, where the variables of the integration are the fields at each site. A procedure to understand the statistical continuum limit is called the renormalization group RG [43]. Renormalization refers to a mathematical tool that allows one to change a physical system as one views it at different $distance\ scales$. It is a strategy for dealing with problems involving many scale lengths. The RG method is intimately related to "scale invariance," a symmetry by which the system appears the same at all scales. It brings to light the scale invariance in the neighborhood of a critical point. In the real space transformation, one eliminates certain degrees of freedom which are defined on a lattice, and thus carries out a partial trace operation on the partition function. The lattice constant of the resulting system is then readjusted and the interval variables are renormalized in such a manner that the new Hamiltonian corresponds to the original one in its form. The approach is to integrate out

the fluctuations in sequence, starting with fluctuations on an atomic scale and then moving on until enough scales have been averaged out [41, 43, 44].

The central equation used in the RG type of solutions here for an $n \times n$ matrix $\mathbf{A}$ is

$$\int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty} d\vec{x} \exp[-\vec{x}^{\mathrm{T}} \mathbf{A} \tilde{x} + \tilde{b}\tilde{x}] = \frac{\pi^{n/2}}{\sqrt{\det|\mathbf{A}|}} \exp\left[\frac{1}{4}\tilde{b}^{\mathrm{T}} \mathbf{A}^{-1} \tilde{b}^{-1}\right]. \tag{2.40}$$

The matrix $\mathbf{A}$ is symmetric, and Eq. (2.40) holds for a positive-definite matrix $\mathbf{A}$. It can, however, be assumed that the equation holds as long as $\mathbf{A}$ is symmetric even if it is complex. The complex entries are only along the diagonal in the cases to be considered in the forthcoming sections and chapters applying RG type of solutions. Having thus made this assumption, the determinant

$$\det|\mathbf{A}| = \frac{\pi^n \exp\left[\frac{1}{2}\vec{b}^{\mathrm{T}} \mathbf{A}^{-1} \vec{b}\right]}{\left(\int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty} d\vec{x} \exp\left[-\vec{x}^{\mathrm{T}} \mathbf{A} \vec{x} + \vec{b}\vec{x}\right]\right)^2} \tag{2.41}$$

can be utilized. The inverse of a matrix can be achieved using Cramer's rule for a square matrix such as for $\mathbf{A}$ as

$$\mathbf{A}^{-1} = \frac{\mathrm{adj}(\mathbf{A})}{\det(\mathbf{A})}, \tag{2.42}$$

where $\mathrm{adj}(\mathbf{A}) = \mathbf{C}^{\mathrm{T}}$, that is, the adjugate of $\mathbf{A}$ is the transpose of the cofactor of $\mathbf{A}$. The cofactor being $C_{ij} = (-)^{i+j} M_{ij}$, and (i, j) the minor of $\mathbf{A}$ denoted by M_{ij} the determinant of the $(n-1) \times (n-1)$ matrix that results from the deletion of row i and column j of the matrix $\mathbf{A}$: $M_{ij} = \det(ij \text{ minor of } \mathbf{A})$.

2.5 Transmission for a Single Site Blob using different methods

This section deals with the simple and special case where there is a single atom in the blob. The transmission equations are compared using different methods using solutions in

continuous space, the discrete Green's function method, the (3×3) and (1×1) matrices, and the RG method.

2.5.1 Single Site: Continuous

From Sec. (1.3), Fig. (1.1), Eq. (1.7), constants F and A are found using boundary conditions for the one-dimensional potential barrier. The constraints are that ψ and $d\psi/dx$ must be continuous at $x = 0$ and $x = a$. Implementing for the constants yields the following relationship between them [45]

$$A + B = C + D, \qquad ik_1(A - B) = ik_2(C - D), \tag{2.43}$$

and

$$Ce^{ik_2a} + De^{-ik_2a} = Fe^{ik_1a}, \qquad ik_2(Ce^{ik_2a} - De^{-ik_2a}) = ik_1Fe^{ik_1a}. \tag{2.44}$$

Solving for F results in

$$F = 4k_1k_2Ae^{-ik_1a}\left[4k_1k_2\cos(k_2a) - 2i\left(k_1^2 - k_2^2\right)\sin(k_2a)\right]. \tag{2.45}$$

The transmission coefficient Eq. (1.7) is then

$$\begin{aligned} T &= \left[1 + \frac{1}{4}\left(\frac{k_1^2 - k_2^2}{k_1k_2}\right)\sin^2(k_2a)\right]^{-1} \\ &= \left[1 + \frac{U_0^2}{4E(E - U_0)}\sin^2\left(\sqrt{2ma^2U_0/\hbar}\sqrt{E/U_0 - 1}\right)\right]^{-1} \end{aligned} \tag{2.46}$$

since

$$\left(\frac{k_1^2 - k_2^2}{k_1k_2}\right)^2 = \frac{U_0^2}{E(E - U_0)}. \tag{2.47}$$

Further simplifying gives

$$T(E) = \frac{4E(E-U_0)}{4E(E-U_0) + U_0^2 \sin^2\left(\sqrt{(E-U_0)/t_0}\right)}, \tag{2.48}$$

where $t_0 = \hbar^2/2ma^2$. This solution is given in [45].

2.5.2 Single Site: Green's Function (Discrete)

Consider the single atom device as shown in Fig. (2.2) with a (1×1) Hamiltonian. The relevant Schrödinger equation has to be discretized in order to represent the quantum system in a matrix form. As an application of the Green's function formalism [13] to a single atom with input and output leads as shown (Fig. 2.2) is considered.

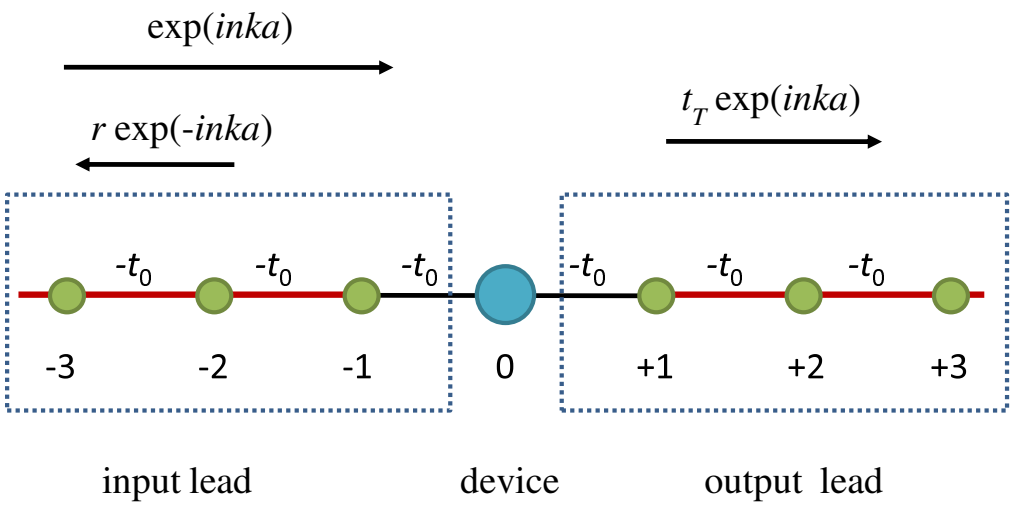

Figure 2.2

A one-atom "device" connected to input and output leads.

The device here is treated as single atom whose Hamiltonian is a one-by-one matrix,

$$[H] = E_c + 2t_0 + \frac{U_0}{a}. \tag{2.49}$$

Here E_c is the conduction band edge constant, $t_0 \equiv \hbar^2/2m_c a^2$, and U_0 is the single-electron charging energy, that is, the change in potential energy due to one extra electron. The effects of the two semi-infinite leads are represented by a 1×1 self energy matrix

$$[\Sigma_1(E)] = -t_0 e^{(ika)}, \qquad [\Sigma_2(E)] = -t_0 e^{(ika)},$$

where ka is related to the energy E by the dispersion relation

$$E = E_c + 2t_0(1 - \cos ka) \quad \longrightarrow \quad \hbar v(E) = 2at_0 \sin(ka). \tag{2.50}$$

The Green's function

$$G = [E\mathbf{I} - H - \Sigma_1 - \Sigma_2]^{-1} = [E - E_c - 2t_0 + 2t_0 e^{(ika)} - (U_0/a)]^{-1},$$

which on simplifying using the dispersion relation gives

$$G = [i2t_0 \sin ka - (U_0/a)]^{-1} = \frac{a}{(i\hbar v - U_0)}. \tag{2.51}$$

Then the transmission is given by

$$T(E) = \mathrm{Tr}[\Gamma_1 G \Gamma_2 G^+] = \frac{\hbar^2 v(E^2)}{\hbar^2 v(E)^2 + U_0^2}, \tag{2.52}$$

where $\Gamma_{1,2}(E) = i[\Sigma_{1,2} - \Sigma_{1,2}^+] = -t_0 e^{(ika)}$ is the broadening matrix, the imaginary part of the self-energy. This discrete lattice approach in real space is used in calculating the Green's function, the effects of the semi-infinite input and output leads that have been integrated through a self energy function Σ^R thus replacing them with a finite-sized non-zero isolated single atom conductor. In this formulation using the self energy function the semi-infinite leads can be eliminated, except for the points of contact between the single

atom conductor and the adjacent atoms of the leads. Such convenience requires only the Hamiltonian, which dimension is the same as the number of atoms in the conductor. In this simple case of the single atom, the dimension is 1×1.

2.5.3 Single Site: (3×3) Matrix

The equation to solve for transmission in this case is Eq. (2.17), in which the matrix $\mathcal{H} - E\mathbf{I}$ is set to, say $\mathbf{A}$, and $e^{iq} = \cos(q) + i\sin(q) = \frac{E}{2} + i\frac{\sqrt{4-E^2}}{2}$ as already defined, is

$$\begin{pmatrix} -E + e^{iq} & \vec{w}^{\mathrm{T}} & 0 \\ \vec{w} & \mathbf{A} & \vec{u} \\ 0 & \vec{u}^{\mathrm{T}} & -E + e^{iq} \end{pmatrix} \begin{pmatrix} 1 + r \\ \vec{\psi} \\ t_T \end{pmatrix} = \begin{pmatrix} 2i\sin(q) \\ \vec{0} \\ 0 \end{pmatrix}. \tag{2.53}$$

The method of reduction is employed to find the inverse, where $\mathcal{H}$ in this single site case is a 1×1 matrix. Multiply the left side for the middle equation to give

$$(1 + r)\vec{w} + \mathbf{A}\vec{\psi} + t_T\vec{u} = \vec{0}, \tag{2.54}$$

multiply by $\mathbf{A}^{-1}$:

$$(1 + r)\mathbf{A}^{-1}\vec{w} + \vec{\psi} + t_T\mathbf{A}^{-1}\vec{u} = 0. \tag{2.55}$$

Then

$$\vec{\psi} = -(1 + r)\mathbf{A}^{-1}\vec{w} - t_T\mathbf{A}^{-1}\vec{u}. \tag{2.56}$$

Substitute for ψ in the matrix equation and take the product for the top row:

$$(1 + r)(-E + e^{iq}) - (1 + r)\vec{w}^{\mathrm{T}}\mathbf{A}^{-1}\vec{u} - t_T\vec{w}^{\mathrm{T}}\mathbf{A}^{-1}\vec{u} = 2i\sin(q) \tag{2.57}$$

and similarly taking the product for the bottom row:

$$-(1 + r)\vec{u}^{\mathrm{T}}\mathbf{A}^{-1}\vec{w} - t_T\vec{u}^{\mathrm{T}}\mathbf{A}^{-1}\vec{u} + t_T(-E + e^{iq}) = 0. \tag{2.58}$$

Now there are two equations Eq. (2.57) and Eq. (2.58) with two variables $(1+r)$ and t_T to be solved in matrix form

$$\begin{pmatrix} -E+e^{iq}-\vec{w}^{\mathrm{T}}\mathbf{A}^{-1}\vec{w} & -\vec{w}^{\mathrm{T}}\mathbf{A}^{-1}\vec{u} \\ -\vec{u}^{\mathrm{T}}\mathbf{A}^{-1}\vec{w} & -E+e^{iq}-\vec{u}^{\mathrm{T}}\mathbf{A}^{-1}\vec{u} \end{pmatrix} \begin{pmatrix} 1+r \\ t_T \end{pmatrix} = \begin{pmatrix} 2i\sin(q) \\ 0 \end{pmatrix}. \quad (2.59)$$

The dimension of the matrix equation has now been reduced to the inverse of a 2×2 matrix, the inverse of which can be easily obtained.

In the case of a single atom in the blob $\mathcal{H}=\epsilon$ is a matrix of size 1×1. Now $\mathbf{A}$ is 1×1 and symmetric. Then $\mathbf{A}=\mathbf{A}^{-1}$ is symmetric, and therefore $\vec{w}\mathbf{A}^{-1}\vec{u}=\vec{u}\mathbf{A}^{-1}\vec{w}$. With these substitutions

$$\begin{aligned} \mathbf{A} &= (\epsilon - E) \\ \vec{u} &= t_u\vec{e} \\ \vec{w} &= t_w\vec{e} \\ \vec{e} &= 1 \\ \vec{w}^{\mathrm{T}}\mathbf{A}\vec{w} &= t_w^2\frac{1}{(\epsilon-E)}. \end{aligned} \quad (2.60)$$

Using Kramer's rule to find the inverse of the 2×2 matrix gives

$$t_T = \frac{\frac{-2i\sin(q)t_u t_w}{\epsilon-E}}{\left(S-\frac{t_w^2}{\epsilon-E}\right)\left(S-\frac{t_u^2}{\epsilon-E}\right)-\left(-\frac{t_w t_u}{\epsilon-E}\right)^2}. \quad (2.61)$$

Substituting $S=\frac{E}{2}+i\frac{\sqrt{4-E^2}}{2}$ and $\sin(q)=\frac{\sqrt{4-E^2}}{2}$ into Eq. (2.61) gives

$$T=|t_T|^2=\frac{(4-E^2)t_w^2t_u^2}{t_u^4+2t_u^2t_w^2+t_w^4-E^2(-1+t_u^2+t_w^2)+E(-2+t_u^2+t_w^2)\epsilon+\epsilon^2} \quad (2.62)$$

Setting $t_w=t_u=1$ in the above equation Eq. (2.62)

$$T=\frac{4-E^2}{4-E^2+\epsilon^2}. \quad (2.63)$$

A screenshot of an Easy Java Simulation showing Transmission versus Energy for the single site blob, with parameters input strength t_w, output strength t_u, and onsite energy of the single atom (impurity) ϵ is shown in Fig. (2.3). The Easy Java Simulation is available freely on the web at $\mathrm{http://quantum.ph.msstate.edu/nanotransport.html}$ and the codes for this simulation are given in Appendix A.

2.5.4 Single Site: (1×1) Matrix

Figure (2.2) shows the uniform semi-infinite input and output leads to the one-atom blob in one-dimension represented by a one-band effective mass model. The one site at $x = 0$ is different from the others. The Hamiltonian for this single atom is a one-by-one matrix. In this case, let $\vec{w} = t_w$ and $\vec{u} = t_u$ and $\mathcal{H} = \epsilon$. The required inverse matrix is given by

$$\mathbf{L} = \frac{1}{\epsilon - E - S^*(t_w^2 + t_u^2)} = \frac{1}{\epsilon - E - [\cos(\phi) - i\sin(\phi)]\,(t_w^2 + t_u^2)} \tag{2.64}$$

with $S^* = \exp(-i\phi)$, where the phase (written in computer notation) is

$$\phi = \mathrm{atan2}\left(\frac{\sqrt{4 - E^2}}{2}, -\frac{E}{2}\right). \tag{2.65}$$

Let $\tilde{t}^2 = t_w^2 + t_u^2$. Then

$$\mathbf{L} = \frac{\epsilon - E - \tilde{t}^2\cos(\phi) - i\tilde{t}^2\sin(\phi)}{\left[\epsilon - E - \tilde{t}^2\cos(\phi)\right]^2 + \left[\tilde{t}^2\sin(\phi)\right]^2}. \tag{2.66}$$

Let $\mathbf{L} = A\exp(i\omega)$ with amplitude

$$A = \frac{1}{\sqrt{\left[\epsilon - E - \tilde{t}^2\cos(\phi)\right]^2 + \left[\tilde{t}^2\sin(\phi)\right]^2}} \tag{2.67}$$

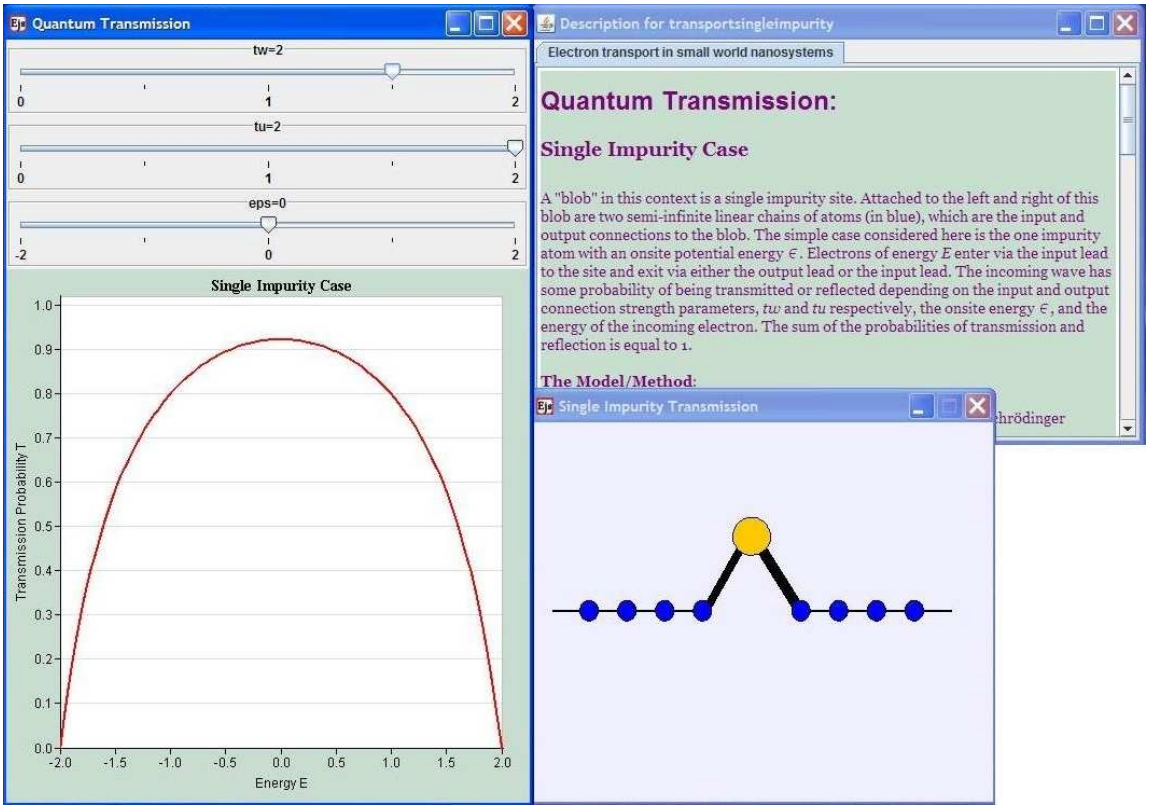

Figure 2.3

Easy Java Simulation of single atom blob T vs E.

This gives the general solution for t_T to be

$$t_T = 2i(S^*)^2 \sin(q)\vec{u}^T \mathbf{L}\vec{w} = 2A\sin(q)t_u t_w e^{\frac{i\pi}{2}} e^{-2i\phi} e^{i\omega}, \tag{2.68}$$

and the transmission is [35]

$$T = |t_T|^2 = 4A^2 \sin^2(q) t_u^2 t_w^2 = \frac{4\sin^2(q)t_u^2 t_w^2}{\left[\epsilon - E - \tilde{t}^2\cos(\phi)\right]^2 + \left[\tilde{t}^2 \sin(\phi)\right]^2}. \tag{2.69}$$

Simplifying further with the respective definitions of $\sin(q)$, and $\cos(\phi)$, transmission is given in terms of onsite energy ϵ of the single site, the energy E of the incoming electron, and the input and output parameters t_w and t_u,

$$T = |t_T|^2 = \frac{(4 - E^2)t_u^2 t_w^2}{t_u^4 + 2t_u^2 t_w^2 + t_w^4 - E^2(-1 + t_u^2 + t_w^2) + E(-2 + t_u^2 + t_w^2)\epsilon + \epsilon^2}. \tag{2.70}$$

Setting $t_w = t_u = 1$ in the above equation Eq. (2.70) gives

$$T = \frac{4 - E^2}{4 - E^2 + \epsilon^2}. \tag{2.71}$$

The plots for transmission T versus energy E are shown in Fig. (2.3) for $\epsilon = 0$ as $t_w = t_u$ is varied and how T varies as ϵ is varied. When $t_w = t_u = 1$ and $\epsilon = 0$, there is full transmission for all energy values $-2 \leq E \leq 2$. Setting input connection strength $t_w = 0$ is equivalent to disconnecting the atom from the input, showing physically zero transmission. Setting output connection strength $t_u = 0$ also gives zero transmission, when all the electrons are reflected back to the origin.

2.5.5 Single Site: RG

For an n-site blob, the matrix equation to solve is

$$\begin{pmatrix} e^{iq}-E & \vec{t}_w & t_{\rm d} \\ \vec{t}_w & \mathbf{A} & \vec{t}_u \\ t_{\rm d} & \vec{t}_u & e^{iq}-E \end{pmatrix} \begin{pmatrix} 1+r \\ \vec{\psi} \\ t_T \end{pmatrix} = \begin{pmatrix} 2i\sin(q) \\ \vec{0} \\ 0 \end{pmatrix}, \tag{2.72}$$

where

$$e^{iq} = \cos(q) + i\sin(q) = \frac{E}{2} + i\frac{\sqrt{4-E^2}}{2}.$$

The matrix for a single-site blob $\mathbf{A} = (\epsilon - E - t)\mathbf{I} + t\mathbf{J}$ is then $(\epsilon - E)$ since there is only one site in the blob, identity matrix $\mathbf{I}$ is just a 1×1 matrix, and there is no inter-blob hopping parameter ($t = 0$), $\vec{t}_w = \vec{t}_w^{\rm T}$ and $\vec{t}_w = \vec{t}_w^{\rm T}$. The matrix to solve is the inverse of the matrix in Eq. (2.72), which is

$$\mathbf{M}_3 = \begin{pmatrix} e^{iq}-E & t_w & 0 \\ t_w & \epsilon - E & t_u \\ 0 & t_u & e^{iq}-E \end{pmatrix}. \tag{2.73}$$

Then

$$1 + r = (\mathbf{M}_3^{-1})_{1,1} 2i\sin(q) \tag{2.74}$$

with transmission $T = 1 - R$ and reflection $R = |r|^2$, where $(\mathbf{M}_3^{-1})_{1,1}$ is the $(1,1)$ element of the inverse of the matrix $\mathbf{M}_3$. Number the site just before the blob as x_- and the site after the blob x_+, and the site of the blob as x_1, and form the vector

$$\vec{x} = \begin{pmatrix} x_- \\ x_1 \\ x_+ \end{pmatrix}. \tag{2.75}$$

From Sec. (2.4), Eq. (2.41), the determinant of the matrix is

$$\det|\mathbf{M}_3| = \frac{\pi^3}{\left[\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} dx_- dx_1 dx_+ \exp\left(-\vec{x}^{\mathrm{T}}\mathbf{M}_3\vec{x}\right)\right]^2}. \tag{2.76}$$

Therefore using Cramer's rule to find the $(1,1)$ element of the inverse gives

$$1 + r = 2i\sin(q)\frac{\det\left|\begin{pmatrix} \epsilon - E & t_u \\ t_u & e^{iq} - E \end{pmatrix}\right|}{\det\left|\begin{pmatrix} e^{iq} - E & t_w & 0 \\ t_w & \epsilon - E & t_u \\ 0 & t_u & e^{iq} - E \end{pmatrix}\right|} \tag{2.77}$$

which gives

$$1 + r = \frac{2i\sin(q)}{\pi}\left\{\frac{I_3}{I_2}\right\}^2, \tag{2.78}$$

where

$$I_3 = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} dx_- dx_1 dx_+ \exp\left[-\vec{x}^{\mathrm{T}}\mathbf{M}_3\vec{x}\right] \tag{2.79}$$

and

$$I_2 = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} dx_1' dx_+' \exp\left[-\begin{pmatrix} x_1' & x_+' \end{pmatrix}\begin{pmatrix} \epsilon - E & t_u \\ t_u & e^{iq} - E \end{pmatrix}\begin{pmatrix} x_1' \\ x_+' \end{pmatrix}\right]. \tag{2.80}$$

Now perform the integral I_2 over dx_1' to give

$$I_2 = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} dx_1' dx_+' \exp\left[-\left((\epsilon - E){x_1'}^2 + (e^{iq} - E){x_+'}^2 + 2t_u x_1' x_+'\right)\right] \tag{2.81}$$

and with

$$\vec{b} = 2t_u x_+' \quad \text{and} \quad \mathbf{A}^{-1} = \frac{1}{\epsilon - E}$$

$$I_2 = \int_{-\infty}^{\infty} dx'_+ \exp\left[-(e^{iq} - E){x'_+}^2\right] \frac{\sqrt{\pi}}{\sqrt{\epsilon - E}} \left[\frac{1}{4}\frac{(2t_u x'_+)^2}{\epsilon - E}\right] = \frac{\sqrt{\pi}}{\sqrt{\epsilon - E}} I_1^*. \quad (2.82)$$

The renormalized integral to do with site x_1 decimated is

$$I_1^* = \int_{-\infty}^{\infty} dx'_+ \exp\left[-\left(e^{iq} - E - \frac{t_u^2}{\epsilon - E}\right){x'_+}^2\right]. \quad (2.83)$$

Similarly performing the integral in I_3 over dx_1 with $\vec{b} = 2t_u x_+ + 2t_w x_-$ and $\mathbf{A} = \frac{1}{\epsilon - E}$ gives

$$I_3 = \frac{\sqrt{\pi}}{\sqrt{\epsilon - E}} \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} dx_+ dx_- \exp\left[-(x_-\ x_+)\begin{pmatrix} e^{iq} - E - \frac{t_w^2}{\epsilon - E} & -\frac{t_w t_u}{\epsilon - E} \\ -\frac{t_w t_u}{\epsilon - E} & e^{iq} - E - \frac{t_u^2}{\epsilon - E}\end{pmatrix}\begin{pmatrix} x_- \\ x_+\end{pmatrix}\right] = \frac{\sqrt{\pi}}{\sqrt{\epsilon - E}} I_2^*. \quad (2.84)$$

Collecting terms gives

$$1 + r = \frac{2i\sin(q)}{\pi}\left\{\frac{I_3}{I_2}\right\}^2 = \frac{2i\sin(q)}{\pi}\left\{\frac{I_2^*}{I_1^*}\right\}^2. \quad (2.85)$$

This leaves

$$1 + r = \frac{2i\sin(q)}{\pi} \frac{\det\left|e^{iq} - E - \frac{{t_u}^2}{\epsilon - E}\right|}{\det\left|\begin{pmatrix} e^{iq} - E - \frac{t_w^2}{\epsilon - E} & -\frac{t_u t_w}{\epsilon - E} \\ -\frac{t_u t_w}{\epsilon - E} & e^{iq} - E - \frac{t_u^2}{\epsilon - E}\end{pmatrix}\right|} \quad (2.86)$$

so

$$1 + r = \left(\mathbf{M}_2^{*-1}\right)_{1,1} 2i\sin(q),$$

where

$$M_2^* = \begin{pmatrix} e^{iq} - E - \frac{t_w^2}{\epsilon - E} & -\frac{t_u t_w}{\epsilon - E} \\ -\frac{t_u t_w}{\epsilon - E} & e^{iq} - E - \frac{t_u^2}{\epsilon - E}\end{pmatrix}. \quad (2.87)$$

Then

$$M_2^* \begin{pmatrix} 1+r \\ t_T \end{pmatrix} = \begin{pmatrix} e^{iq} - E - \frac{t_w^2}{\epsilon - E} & -\frac{t_u t_w}{\epsilon - E} \\ -\frac{t_u t_w}{\epsilon - E} & e^{iq} - E - \frac{t_u^2}{\epsilon - E} \end{pmatrix} \begin{pmatrix} 1+r \\ t_T \end{pmatrix} = \begin{pmatrix} 2i\sin(q) \\ 0 \end{pmatrix}. \tag{2.88}$$

This equation can be solved for t_T by taking the inverse of M_2^*, which is a simple symmetric 2×2 matrix giving

$$T = |t_T|^2 = \frac{(4 - E^2) t_u^2 t_w^2}{t_u^4 + 2t_u^2 t_w^2 + t_w^4 - E^2(-1 + t_u^2 + t_w^2) + E(-2 + t_u^2 + t_w^2)\epsilon + \epsilon^2}. \tag{2.89}$$

Setting $t_w = t_u = 1$ in the above equation Eq. (2.89) gives

$$T = \frac{4 - E^2}{4 - E^2 + \epsilon^2}. \tag{2.90}$$

The equivalence between the transmission equations Eqs. (2.63, 2.71) from the matrix solutions, Eq. (2.90) from the RG calculations, and the transmission equation (Eq. 2.52) from the Green's function calculation lies in the following relation [13]:

$$\hbar^2 v(E)^2 = 4a^2 t_0^2 \sin^2(ka). \tag{2.91}$$

Setting the hopping parameter $t_0 = 1$, lattice constant $a = 1$, and the wave vector k equal to angle q as in Eq. (2.5) with the definition $\sin(q) = \frac{\sqrt{4-E^2}}{2}$, gives $\hbar^2 v(E)^2 = 4 - E^2$, and that results in the equivalence of the transmission equations

$$T = \frac{\hbar^2 v(E)^2}{\hbar^2 v(E)^2 + U_0^2} = \frac{4 - E^2}{4 - E^2 + \epsilon^2}, \tag{2.92}$$

where U_0 and ϵ are equivalents of the onsite potential energy of the single atom. Hence the equivalence of the Green's function transmission equation with those derived from the matrix and RG solutions.

2.6 Transmission for Single Site Dangle using different methods

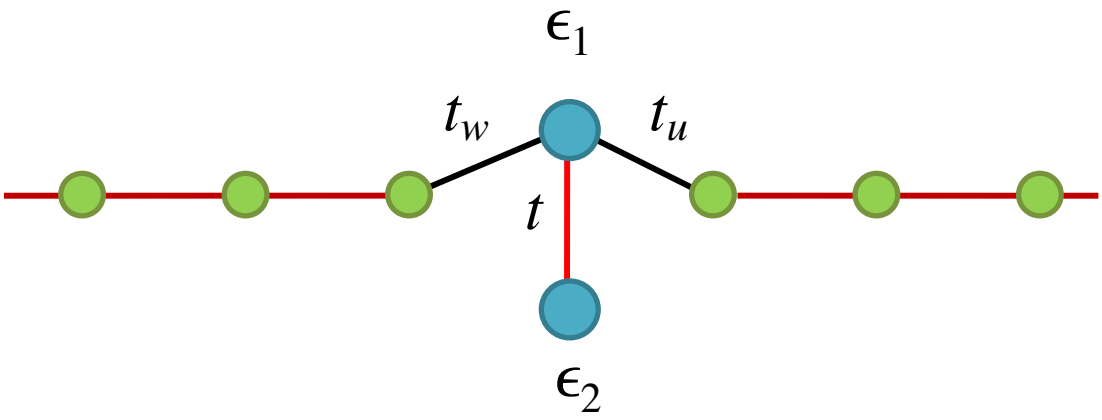

Figure 2.4

Two sites: Dangle blob.

2.6.1 Single Site Dangle: Matrix Formalism

Shown in Fig. (2.4) is a two site blob with one atom connected to the input and output. The Hamiltonian in this case is a 2×2 matrix. The input and output connecting vectors are respectively

$$\vec{w} = \begin{pmatrix} t_w \\ 0 \end{pmatrix}, \qquad \vec{u} = \begin{pmatrix} t_u \\ 0 \end{pmatrix}. \tag{2.93}$$

The two atoms in the blob with onsite energies ϵ_1 and ϵ_2 are connected by hopping parameter t. Then the Hamiltonian of this dangle blob is

$$\mathcal{H} = \begin{pmatrix} \epsilon_1 & t \\ t & \epsilon_2 \end{pmatrix}. \tag{2.94}$$

Using Eq. (2.24) with coupling matrices

$$\vec{w}\vec{w}^{\mathrm{T}} = \begin{pmatrix} t_w^2 & 0 \\ 0 & 0 \end{pmatrix}, \qquad \vec{u}\vec{u}^{\mathrm{T}} = \begin{pmatrix} t_u^2 & 0 \\ 0 & 0 \end{pmatrix} \tag{2.95}$$

gives the matrix

$$\mathbf{L} = \mathcal{H} - E\mathbf{I} - S^* \left(\vec{w}\vec{w}^{\mathrm{T}} + \vec{u}\vec{u}^{\mathrm{T}}\right) = \begin{pmatrix} \epsilon_1 - E - S^* \left(t_w^2 + t_u^2\right) & t \\ t & \epsilon_2 - E \end{pmatrix}. \tag{2.96}$$

Then the transmission equation Eq. (2.26) gives

$$T = (4 - E^2) \left| \begin{pmatrix} t_w & 0 \end{pmatrix} \mathbf{L}^{-1} \begin{pmatrix} t_u \\ 0 \end{pmatrix} \right|^2. \tag{2.97}$$

The matrix $\mathbf{L}^{-1}$ can be obtained using standard methods since it is a symmetric 2×2 matrix. The transmission therefore for a single-site dangle, for example given a set of parametric values $t_w = t_u = 1$, and $\epsilon_1 = \epsilon_2 = 0$, is

$$T(E) = \frac{4 - E^2}{4 - E^2 + \frac{t^4}{E^2}}. \tag{2.98}$$

2.6.2 Single Site Dangle: Green's Function Formalism

The above transmission equation Eq. (2.98) for a single-site dangle has been derived (shown in the Appendix of [8]) using the discrete Green's function formalism. Referring

to the same model as shown in Fig. (2.4), the Green's function is expressed in terms of the Dyson equation and the transmission obtained is for the same set of parameters discussed in the previous section (2.6.1),

$$T(E) = \frac{4 - E^2}{4 - E^2 + \frac{t^4}{E^2}}. \tag{2.99}$$

2.6.3 Single Site Dangle: RG Formalism

Sec. (2.4) gives the transmission calculation by the RG method for an n-site blob. This method can be applied for a one site in a dangle as shown in Fig. (2.5). The single dangle site with on-site energy ϵ_1 is connected via t to the site with ϵ_0. The site with energy ϵ_0 is connected on both sides to the input and output leads that have been decimated shown in the figure as crossed boxes. Applying the RG to decimate the dangling site leads to a renormalized on-site energy ϵ_0'. For an $(1+1)$-site blob of which 1 site is decimated, then the $(1+1+2) \times (1+1+2)$ matrix equation to be solved will be

$$\begin{pmatrix} e^{iq} - E & t_w & 0 & 0 \\ t_w & \epsilon_0 - E & t & t_u \\ 0 & t & \epsilon_1 - E & 0 \\ 0 & t_u & 0 & e^{iq} - E \end{pmatrix} \begin{pmatrix} 1 + r \\ \psi \\ \theta \\ t_T \end{pmatrix} = \begin{pmatrix} 2i \sin(q) \\ 0 \\ 0 \\ 0 \end{pmatrix}, \tag{2.100}$$

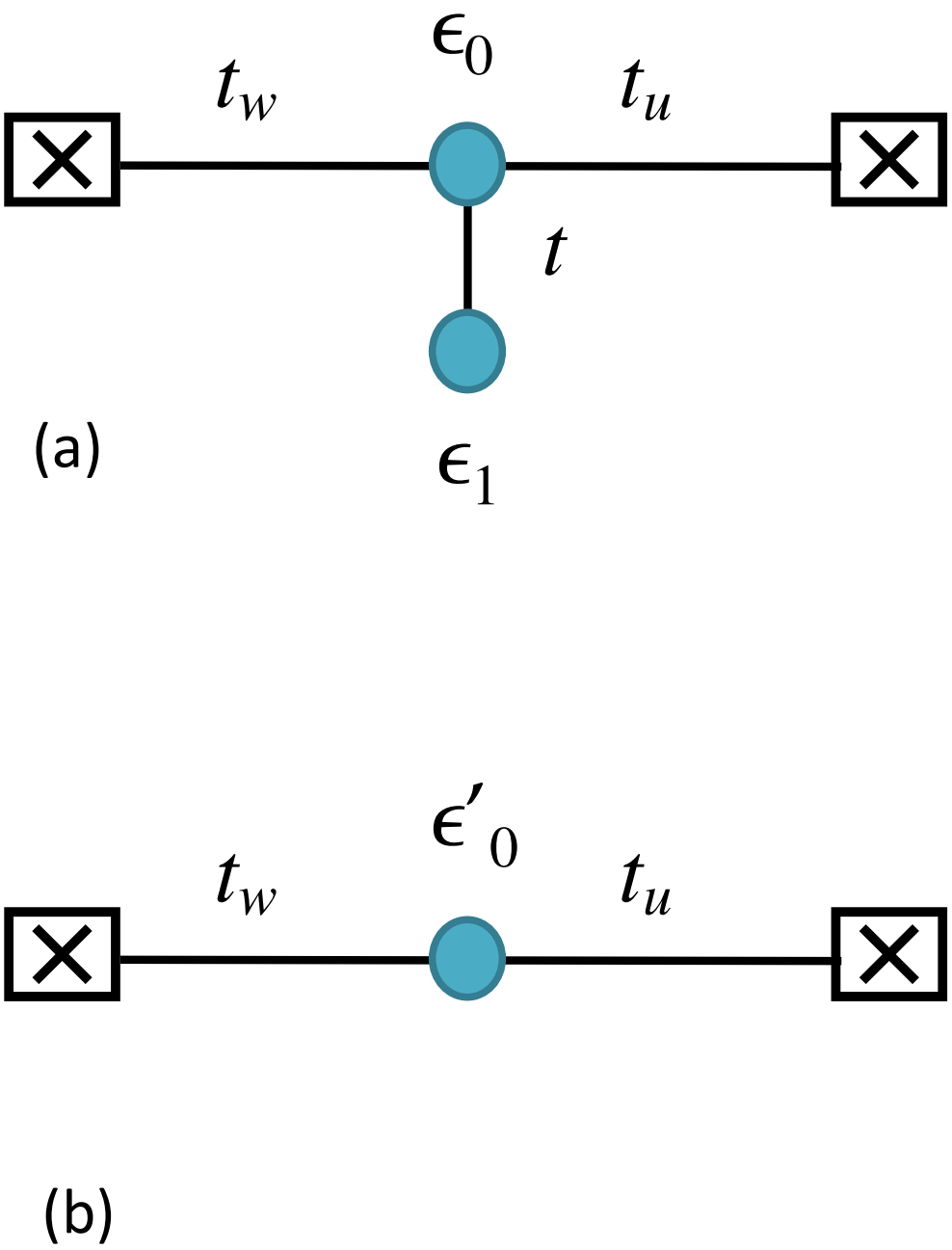

Figure 2.5

A single site dangle (a) before decimation, (b) after decimation.

where $e^{iq} = \cos(q) + i\sin(q) = \frac{E}{2} + i\frac{\sqrt{4-E^2}}{2}$. Decimating the one dangle site in the standard way, the decimated renormalized matrix is

$$\begin{pmatrix} e^{iq} - E & t_w & 0 \\ t_w & \epsilon_0' - E & t_u \\ 0 & t_u & e^{iq} - E \end{pmatrix} \begin{pmatrix} 1 + r \\ \vec{\psi} \\ t_T \end{pmatrix} = \begin{pmatrix} 2i\sin(q) \\ 0 \\ 0 \end{pmatrix} \tag{2.101}$$

with

$$\epsilon_0' = \epsilon_0 - \frac{t^2}{\epsilon_1 - E}. \tag{2.102}$$

This completes the RG decimation equations for the dangling site. Then

$$t_T = \frac{2i\sin(q)\frac{t_w\, t_u}{\epsilon' - E}}{(e^{iq} - E - \frac{t_w^2}{\epsilon' - E})(e^{iq} - E - \frac{t_u^2}{\epsilon' - E}) - (\frac{t_w t_u}{\epsilon' - E})^2}. \tag{2.103}$$

With onsite energies $\epsilon_0 = \epsilon_1$ set to zero the transmission probability is calculated using the symbolic manipulation program Mathematica [46], giving

$$T = \frac{E^2(4 - E^2)t_u^2 t_w^2}{E^2\left[(-2 + t_u^2 + t_w^2)t^2 + (t_u^2 + t_w^2)^2\right] - E^2(-1 + t_u^2 + t_w^2) + t^4}. \tag{2.104}$$

A transmission equation comparable to those (Eq. (2.98), Eq. (2.99)) derived using the other methods is obtained by setting $t_w = t_u = 1$ to give

$$T = |t_T|^2 = \frac{E^2(4 - E^2)}{E^2(4 - E^2) + t^4}. \tag{2.105}$$

Hence the transmission equation Eq. (2.98) obtained from matrix solution for a blob with one site in the dangle is equivalent to the transmission equations Eq. (2.105) and Eq. (2.99) derived using the RG method and the Green's function calculation respectively.

CHAPTER 3

APPLICATIONS

3.1 General Solution: Fully connected 5 Equations 5 Unknowns case

This section takes into account of a fully connected n-site blob with input and output leads connected to each of the sites in the blob with hopping parameter t and onsite energy ϵ. This is a general case with symmetry, (Fig. 3.1), in which $\vec{w} = \vec{u}$. The general transmission formula is obtained using the 5-Equation-5-Unknown matrix method and the RG mechanism.

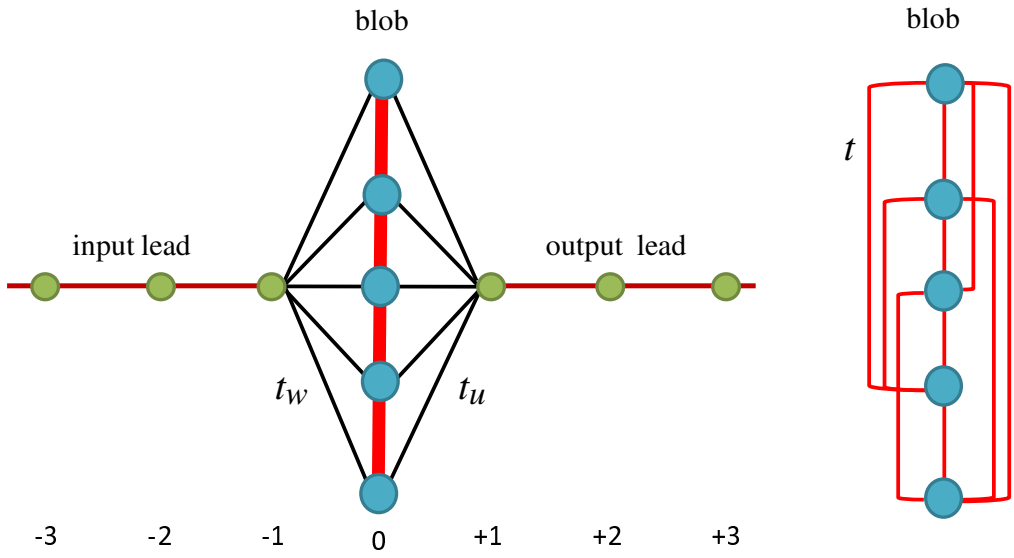

Figure 3.1

A fully-connected n=5 site blob, with all sites connected to input and output.

3.1.1 5-Equation-5-Unknown: Matrix Formalism

Equation (2.39) for the fully-connected n-blob case can be further generalized to obtain exact solutions. Each atom has a on-site energy ϵ coupled and a hopping parameter t to each of the $n-1$ other atoms in the blob. Furthermore, the blob atoms are equally coupled to the input and output by vectors $\vec{w}$ (Eq. (2.13)) and $\vec{u}$ (Eq. (2.12)) respectively. Consider the case where $(n\vec{w} = n\vec{u})$, so that the input and output leads are symmetrically coupled. This arrangement initiates a general matrix of the form $\mathbf{M}$ that can be solved and that is the intent of this section.

The 5-Equation-5-Unknown is a special case as shown in Fig. (3.1) with the n fully connected sites of potential energy ϵ coupled to input and output with coupling strengths t_w and t_u respectively. The number of input and output couplings can vary from one 1 to n, but in general the vector $\vec{w}$ $(= \vec{u})$ can be arbitrary. The Hamiltonian is

$$\mathcal{H} = (\epsilon - t)\mathbf{I} + t\mathbf{J} \tag{3.1}$$

and therefore

$$\mathbf{M} = \mathbf{L}^{-1} = \mathcal{H} - E\mathbf{I} - S^*(t_w^2 + t_u^2)\vec{w}\vec{w}^{\mathrm{T}}. \tag{3.2}$$

Substituting the Hamiltonian in Eq. (3.2) yields the general form of the matrix [35]

$$\mathbf{M} = X_I\mathbf{I} + X_J\mathbf{J} + X_{io}\vec{w}\vec{w}^{\mathrm{T}}, \tag{3.3}$$

where $X_I = \epsilon - E - t, \quad X_J = t$, and $X_{io} = -S^*(t_w^2 + t_u^2)$ corresponds to the identical coupling of a site to the input and output. The ansatz for its inverse has the form [35]

$$\mathbf{M}^{-1} = Y_I\mathbf{I} + Y_{JJ}\mathbf{J} + Y_{ioio}\vec{w}\vec{w}^{\mathrm{T}} + Y_{Jio}\vec{e}\vec{w}^{\mathrm{T}} + Y_{ioJ}\vec{w}\vec{e}^{\mathrm{T}}, \tag{3.4}$$

where

$$\begin{aligned}
\vec{e}^{\mathrm{T}}\vec{w} &= \vec{w}^{\mathrm{T}}\vec{e} = m \\
\vec{w}^{\mathrm{T}}\vec{w} &= l \\
\vec{e}^{\mathrm{T}}\vec{e} &= n \\
\mathbf{J}\vec{e} &= n\vec{e} \\
\mathbf{JJ} &= n\mathbf{J} \\
\mathbf{J}\vec{w} &= \vec{e}\vec{e}^{\mathrm{T}}\vec{w} = m\vec{e},
\end{aligned} \tag{3.5}$$

and l and m being parameters, and n the number of atoms in the cluster. Parameter l is the sum of the squares of the input connection strengths, and m is the sum of the input (or output) connection strengths. When $\frac{1}{t_w}\vec{w}$ has elements of only one or zero, parameters $l = m$ is a special case of the 5-Equation-5-Unknown. For example for the n connected blob, if only three sites are connected equally to both input and output leads, then $l = m = 3$. Thus the parameters l and m control the number and coupling strengths to the n-atom blob. Collecting like terms of the product $\mathbf{MM^{-1}}$ results in five equations with five unknowns

$$\begin{aligned}
X_I Y_I &= 1 \\
X_I Y_{JJ} + X_J Y_J + n X_J Y_{JJ} + m X_J Y_{ioJ} &= 0 \\
X_I Y_{ioio} + X_{io} Y_I + l X_{io} Y_{ioio} + m X_{io} Y_{Jio} &= 0 \\
X_I Y_{Jio} + m X_J Y_{ioio} + n X_J Y_{Jio} &= 0 \\
X_I Y_{ioJ} + m X_{io} Y_{JJ} + l X_{io} Y_{ioJ} &= 0.
\end{aligned} \tag{3.6}$$

One of the five solutions is the trivial solution

$$Y_I = \frac{1}{X_I}, \tag{3.7}$$

and the four other solutions are obtained by solving the matrix equation using Mathematica:

$$\begin{pmatrix} (X_I + nX_J) & & & mX_J \\ & (X_I + lX_{io}) & mX_{io} & \\ & X_J & (X_I + nX_J) & \\ mX_{io} & & & (X_I + lX_{io}) \end{pmatrix} \begin{pmatrix} Y_{JJ} \\ Y_{ioio} \\ Y_{Jio} \\ Y_{ioJ} \end{pmatrix} = \begin{pmatrix} \frac{-X_J}{X_I} \\ \frac{-X_{io}}{X_I} \\ 0 \\ 0 \end{pmatrix}. \tag{3.8}$$

The solutions of Eq. (3.8) are

$$\begin{aligned} Y_{JJ} &= -\frac{X_J}{X_I}\left[\frac{lX_{io} + X_I}{(X_I + nX_J)(lX_{io} + X_I) - m^2 X_J X_{io}}\right] \\ Y_{ioio} &= -\frac{X_{io}}{X_I}\left[\frac{nX_J + X_I}{(X_I + lX_{io})(nX_J + X_I) - m^2 X_J X_{io}}\right] \\ Y_{Jio} &= \frac{X_{io}}{X_I}\left[\frac{mX_J}{(X_I + lX_{io})(nX_J + X_I) - m^2 X_J X_{io}}\right] \\ Y_{ioJ} &= \frac{X_J}{X_I}\left[\frac{mX_{io}}{(X_I + nX_J)(lX_{io} + X_I) - m^2 X_J X_{io}}\right]. \end{aligned} \tag{3.9}$$

Substituting these solutions in Eq. (3.4), using Mathematica, yields the amplitude of transmission

$$t_T = 2i(S^*)^2 \sin(q)\vec{u}^{\mathrm{T}}\mathbf{M}^{-1}\vec{w}, \tag{3.10}$$

and hence the transmission coefficient $T = |t_T|^2$.

A screenshot of an Easy Java Simulation of Transmission versus Energy for the fully connected blob is shown in Fig. (3.2). The simulation is available freely on the web at $\mathrm{http://quantum.ph.msstate.edu/nanotransport.html}$. The simulation codes are given in Appendix B.

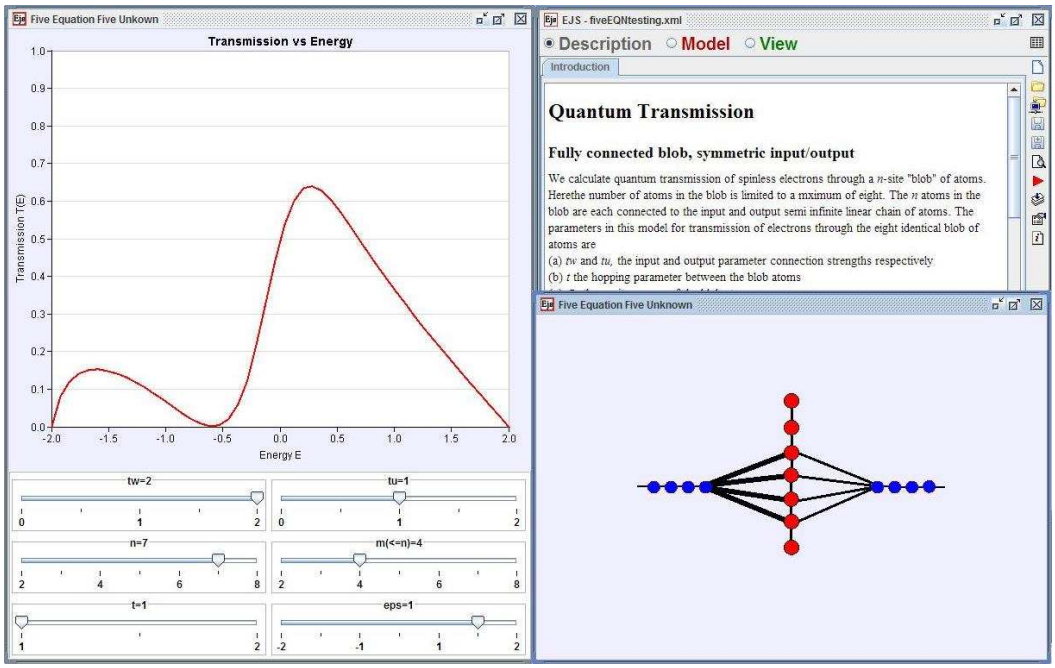

Figure 3.2

Easy Java Simulation of T vs E for the 5-Equation-5-Unknown case.

3.1.2 5-Equation-5-Unknown: RG Formalism

From Sec. (2.5.5), Eq. (2.72), the matrix equation for the fully connected n-site blob

$$\begin{pmatrix} e^{iq}-E & \vec{t}_w^{\mathrm{T}} & t_{\mathrm{d}} \\ \vec{t}_w & \mathbf{A} & \vec{t}_u \\ t_{\mathrm{d}} & \vec{t}_u^{\mathrm{T}} & e^{iq}-E \end{pmatrix} \begin{pmatrix} 1+r \\ \vec{\psi} \\ t_T \end{pmatrix} = \begin{pmatrix} 2i\sin(q) \\ \vec{0} \\ 0 \end{pmatrix}, \tag{3.11}$$

in which the matrix to find the inverse is

$$\mathbf{M}_{n+2} = \begin{pmatrix} e^{iq}-E & \vec{t}_w^{\mathrm{T}} & t_{\mathrm{d}} \\ \vec{t}_w & \mathbf{A} & \vec{t}_u \\ t_{\mathrm{d}} & \vec{t}_u^{\mathrm{T}} & e^{iq}-E \end{pmatrix}, \tag{3.12}$$

where

$$\mathbf{A} = (\epsilon - E - t)\mathbf{I} + t\mathbf{J}.$$

The identity matrix $\mathbf{I}$ is of size $n \times n$ and the matrix $\mathbf{J}$ of size $n \times n$, related as $\mathbf{J}\vec{e} = n\vec{e}$, and $\vec{e}\vec{e}^{\mathrm{T}} = \mathbf{J}$. Then

$$1 + r = \left(\mathbf{M}_{n+2}^{-1}\right)_{1,1} 2i\sin(q) \tag{3.13}$$

with transmission coefficient $T = 1 - R$ and the reflection coefficient $R = |r|^2$, where $\left(\mathbf{M}_{n+2}^{-1}\right)_{1,1}$ is the $(1,1)$ element of the inverse of the matrix $\mathbf{M}_{n+2}$. Number the site just before the blob as x_-, the site just after the blob x_+, and the sites of the blob $x_1, x_2, \ldots, x_n$ and form the vector

$$\vec{x} = \begin{pmatrix} x_- \\ x_1 \\ x_2 \\ \vdots \\ x_n \\ x_+ \end{pmatrix}. \tag{3.14}$$

The determinant is

$$\det|\mathbf{M}_{n+2}| = \frac{\pi^{n+2}}{\left[\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\cdots\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} dx_- dx_1 \ldots dx_n dx_+ \exp\left(-\vec{x}^{\mathrm{T}}\mathbf{M}_{n+2}\vec{x}\right)\right]^2}. \tag{3.15}$$

Therefore using Cramer's rule to find the $(1,1)$ element of the inverse matrix gives

$$1+r = 2i\sin(q)\frac{\det\left|\begin{pmatrix} \mathbf{A} & \vec{t}_u \\ \vec{t}_u^{\,\mathrm{T}} & e^{iq}-E \end{pmatrix}\right|}{\det\left|\begin{pmatrix} e^{iq}-E & \vec{t}_w & 0 \\ \vec{t}_w^{\,\mathrm{T}} & \mathbf{A} & \vec{t}_u \\ 0 & \vec{t}_u^{\,\mathrm{T}} & e^{iq}-E \end{pmatrix}\right|} \tag{3.16}$$

so then

$$1+r = \frac{2i\sin(q)}{\pi}\left[\frac{I_{n+2}}{I_n}\right], \tag{3.17}$$

where

$$I_{n+2} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\cdots\int_{-\infty}^{\infty} dx_- dx_1 \ldots dx_n dx_+ \exp\left[-\vec{x}^{\mathrm{T}}\mathbf{M}_{n+2}\vec{x}\right] \tag{3.18}$$

and

$$I_n = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\cdots\int_{-\infty}^{\infty} dx'_1 \ldots dx'_n dx'_+ \exp\left[-(x'_1\ldots x'_n x'_+)\begin{pmatrix} \mathbf{A} & t_u \\ t_u^{\mathrm{T}}\, e^{iq}-E \end{pmatrix}\begin{pmatrix} x'_1 \\ \vdots \\ x'_n \\ x'_+ \end{pmatrix}\right]. \tag{3.19}$$

Perform the integration over all n sites in the blob to give I_n, where the integrals use the matrix $\mathbf{A} = (\epsilon - E - t)\mathbf{I} + t\mathbf{J}$ of size n, and the vector

$$\vec{b} = \begin{pmatrix} 2t_{w1}x_- + 2t_{u1}x_+ \\ 2t_{w2}x_- + 2t_{u2}x_+ \\ \vdots \\ 2t_{wn}x_- + 2t_{un}x_+ \end{pmatrix} \tag{3.20}$$

and since $\mathbf{J}^2 = n\mathbf{J}$, then $\mathbf{A}^{-1} = \frac{1}{\epsilon - E - t}\mathbf{I} - \frac{t}{(\epsilon - E - t)(\epsilon - E - t + nt)}\mathbf{J}$. The eigenvalues of $\mathbf{A}$ are $\epsilon - E - t + nt$ with eigenvector $\vec{e}$ and $\epsilon - E - t$ which is $n - 1$ fold degenerate, so

$$\det|\mathbf{A}| = (\epsilon - E - t + nt)(\epsilon - E - t)^{n-1}. \tag{3.21}$$

Also

$$\vec{b}^{\mathrm{T}}\mathbf{J}\vec{b} = \vec{b}^{\mathrm{T}}\vec{e}\vec{e}^{\mathrm{T}}\vec{b} = \left[\vec{e}^{\mathrm{T}}\vec{b}\right]^2 = 4\left[\left(\sum_{i=1}^{n} t_{wi}\right)x_+ + \left(\sum_{j=1}^{n} t_{uj}\right)x_-\right]^2, \tag{3.22}$$

so

$$\vec{b}^{\mathrm{T}}\mathbf{J}\vec{b} = 4\left[\left(\sum_{i=1}^{n} t_{wi}\right)^2 x_+^2 + 2\left(\sum_{i=1}^{n} t_{wi}\right)\left(\sum_{j=1}^{n} t_{uj}\right)x_+x_- + \left(\sum_{i=1}^{n} t_{ui}\right)^2 x_-^2\right] \tag{3.23}$$

and

$$\vec{b}^{\mathrm{T}}\mathbf{I}\vec{b} = 4\sum_{i=1}^{n}(t_{wi}x_+ + t_{ui}x_-)^2 = 4\left[x_+^2\sum_{i=1}^{n} t_{wi}^2 + 2x_-x_+\sum_{i=1}^{n} t_{wi}t_{ui} + x_-^2\sum_{i=1}^{n} t_{ui}^2\right]. \tag{3.24}$$

The constant terms cancel out, leaving the final RG result that can be hypothesized to be given by

$$M_n^* \begin{pmatrix} 1+r \\ \\ t_T \end{pmatrix} = \begin{pmatrix} 2i\sin(q) \\ \\ 0 \end{pmatrix} \tag{3.25}$$

with

$$M_n^* = \begin{pmatrix} e^{iq} - E - \frac{\sum t_{wi}^2}{\epsilon - E - t} + \frac{t(\sum t_{wi})^2}{(\epsilon - E - t)(\epsilon - E - t + nt)} & \frac{t\sum t_{wi}\sum t_{ui}}{(\epsilon - E - t)(\epsilon - E - t + nt)} - \frac{\sum t_{wi}t_{ui}}{\epsilon - E - t} \\ \\ \frac{t\sum t_{wi}\sum t_{ui}}{(\epsilon - E - t)(\epsilon - E - t + nt)} - \frac{\sum t_{wi}t_{ui}}{\epsilon - E - t} & e^{iq} - E - \frac{\sum t_{ui}^2}{\epsilon - E - t} + \frac{t(\sum t_{ui})^2}{(\epsilon - E - t)(\epsilon - E - t + nt)} \end{pmatrix}. \tag{3.26}$$

Therefore the general solution has only the five parameters which are

$$\begin{aligned} &\textstyle\sum_{i=1}^{n} t_{wi}, \\ &\textstyle\sum_{i=1}^{n} t_{ui}, \\ &\textstyle\sum_{i=1}^{n} t_{wi}^2, \\ &\textstyle\sum_{i=1}^{n} t_{ui}^2, \\ &\textstyle\sum_{i=1}^{n} t_{wi} t_{ui}. \end{aligned} \tag{3.27}$$

These can now be specialized to the case where $t_{wi} = t_{ui}$, for comparison with the 5-Equation-5-Unknown case. Then the parameters are

$$\begin{aligned} \sum_{i=1}^{n} t_{wi} &= m\, t_w \\ \sum_{i=1}^{n} t_{ui} &= m\, t_u \\ \sum_{i=1}^{n} t_{wi}^2 &= l\, t_w^2 \\ \sum_{i=1}^{n} t_{ui}^2 &= l\, t_u^2 \\ \sum_{i=1}^{n} t_{wi} t_{ui} &= l\, t_w t_u. \end{aligned} \tag{3.28}$$

A simpler case to be solved via the RG case is now considered. For an $n + m$-site fully-connected blob of which m sites will be decimated, the $(n + m + 2) \times (n + m + 2)$

matrix equation to be solved with no direct connections between the input and output, no ϵ_c, no connection of the decimated sites to input or output leads, is

$$\begin{pmatrix} e^{iq}-E & \vec{t}_w^{\,\mathrm{T}} & \vec{0}_m^{\,\mathrm{T}} & \vec{0}_{m'}^{\,\mathrm{T}} & 0 \\ \vec{t}_w & \mathbf{A} & t\mathbf{J}_{n\times m} & t\mathbf{J}_{n\times m'} & \vec{t}_u \\ \vec{0}_m & t\mathbf{J}_{n\times m} & \mathbf{C} & t\mathbf{J}_{n\times m'} & \vec{0}_m \\ \vec{0}_{m'} & t\mathbf{J}_{m'\times n} & t\mathbf{J}_{m'\times m} & \mathbf{C}_d & \vec{0}_{m'} \\ 0 & \vec{t}_u^{\,\mathrm{T}} & \vec{0}_m^{\,\mathrm{T}} & \vec{0}_{m'}^{\,\mathrm{T}} & e^{iq}-E \end{pmatrix} \begin{pmatrix} 1+r \\ \vec{\psi} \\ \vec{\theta} \\ \vec{\theta}_d \\ t_T \end{pmatrix} = \begin{pmatrix} 2i\sin(q) \\ \vec{0}_n \\ \vec{0}_m \\ \vec{0}_{m'} \\ 0 \end{pmatrix}, \tag{3.29}$$

where $e^{iq} = \cos(q) + i\sin(q) = \frac{E}{2} + i\frac{\sqrt{4-E^2}}{2}$, the matrix $\mathbf{A} = (\epsilon - E - t)\mathbf{I} + t\mathbf{J}$ is $n \times n$ and the matrix $\mathbf{C} = (\epsilon - E - t)\mathbf{I} + t\mathbf{J}$ is $m \times m$ so $\mathbf{C}_d = (\epsilon - E - t)\mathbf{I} + t\mathbf{J}$ is $m^{'} \times m^{'}$ and the dimensions for matrices $\mathbf{J}$ and vectors are shown.

The decimated renormalized matrix has the form

$$\begin{pmatrix} e^{iq}-E & \vec{t}_w^{\,\mathrm{T}} & \vec{0}_m & 0 \\ \vec{t}_w & \mathbf{A} - t^2\mathbf{J}_{n\times m'}\mathbf{C}_d^{-1}\mathbf{J}_{m'\times n} & t^{'}\mathbf{J}_{n\times m} & \vec{t}_u \\ \vec{0}_m & t^{'}\mathbf{J}_{m\times n} & \mathbf{C} - t^2\mathbf{J}_{m\times m'}\mathbf{C}_d^{-1}\mathbf{J}_{m'\times m} & \vec{0}_m \\ 0 & vect_u^{\mathrm{T}} & \vec{0}_m & e^{iq}-E \end{pmatrix} \begin{pmatrix} 1+r \\ \vec{\psi} \\ \vec{\theta} \\ t_T \end{pmatrix} = \begin{pmatrix} 2i\sin(q) \\ \vec{0}_n \\ \vec{0}_m \\ 0 \end{pmatrix}. \tag{3.30}$$

This completes the final RG decimation equations in matrix form for the fully connected (easy-case) blob. The required inverse is

$$\mathbf{C}_d^{-1} = \frac{1}{(\epsilon - E - t)}\mathbf{I}_{m'\times m'} - \frac{t}{(\epsilon - E - t)(\epsilon - E - t + m't)}\mathbf{J}_{m'\times m'}. \tag{3.31}$$

Using $\mathbf{J}_{m\times l}\mathbf{J}_{l\times n} = l\mathbf{J}m \times n$ gives

$$\mathbf{J}_{l\times m'}\mathbf{C}_d^{-1}\mathbf{J}_{m'\times k} = \frac{m^{'}}{(\epsilon - E - t)}\mathbf{J}_{l\times k} - \frac{tm^{'}m^{'}}{(\epsilon - E - t)(\epsilon_e - t + m't)}\mathbf{J}_{l\times k}. \tag{3.32}$$

This gives the complete RG transformation. Finally, one has the RG transformations [3, 4], [36]

$$t^{'} = t - \frac{t^2 m^{'}}{\epsilon - E - t}\left[1 - \frac{t m^{'}}{\epsilon - E - t + m^{'} t}\right] \tag{3.33}$$

and

$$\epsilon^{'} = \epsilon - \frac{t^2 m^{'}}{\epsilon - E - t}\left[1 - \frac{t m^{'}}{\epsilon - E - t + m^{'} t}\right]. \tag{3.34}$$

A screenshot of the Mathematica Manipulate Plot showing transmission coefficient T employing the matrix method and the RG method for the 5-Equation-5-Unknown case is shown in Fig. (3.3). The results are identical for the two methods.

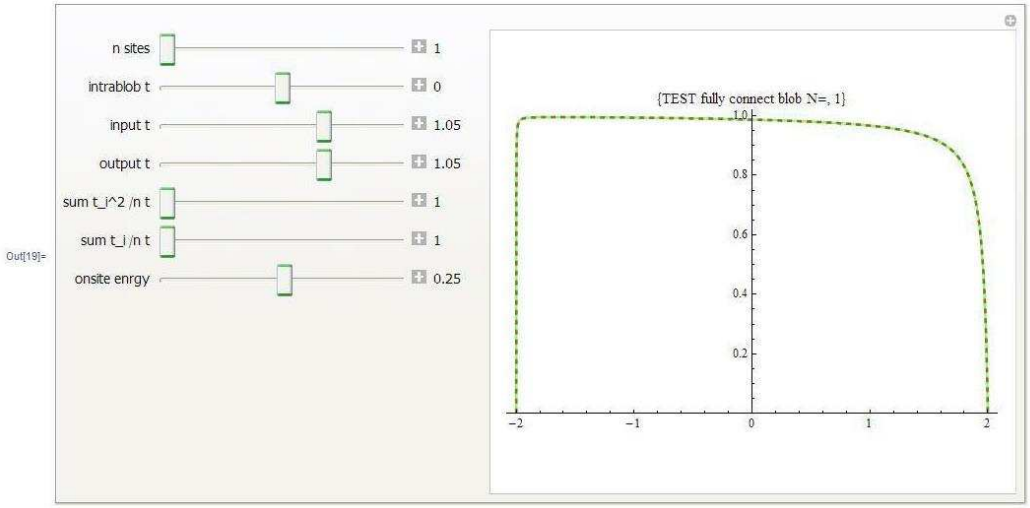

Figure 3.3

Plots for the matrix and RG methods, 5-Equation-5-Unknown case .

3.2 General Solution: Fully connected 10 Equations 10 Unknowns case

This case is the most general case with all sites connected to all other sites of potential energy ϵ and coupling strength t, and where at least one or more sites are connected to the input lead with strength t_w and to the output lead with strength t_u. This general case takes account of both the non-symmetrical case (Fig. 3.4) where $\vec{w}$ and $\vec{u}$ are not proportional and the symmetrical 5-Equation-5-Unknown case (Fig. 3.1).

3.2.1 10-Equation-10-Unknown: Matrix Formalism

The same fully connected n-site blob considered in Sec. (3.1) is connected at random such that all blob atoms are not necessarily equally coupled to the input and output leads. Such an assortment calls for a more general matrix of the form $\mathbf{M}$ written as [42]

$$\mathbf{M} = X_I\mathbf{I} + X_J\mathbf{J} + X_w\vec{w}\vec{w}^{\mathrm{T}} + X_u\vec{u}\vec{u}^{\mathrm{T}}, \tag{3.35}$$

and the ansatz for its inverse has the form [42]

$$\begin{aligned}\mathbf{M}^{-1} = {} & Y_I\mathbf{I} + Y_{JJ}\mathbf{J} + Y_{ww}\vec{w}\vec{w}^{\mathrm{T}} + Y_{uu}\vec{u}\vec{u}^{\mathrm{T}} + Y_{Ju}\vec{e}\vec{u}^{\mathrm{T}} \\ & + Y_{uJ}\vec{u}\vec{e}^{\mathrm{T}} + Y_{Jw}\vec{e}\vec{w}^{\mathrm{T}} + Y_{wJ}\vec{w}\vec{e}^{\mathrm{T}} + Y_{uw}\vec{u}\vec{w}^{\mathrm{T}} + Y_{wu}\vec{w}\vec{u}^{\mathrm{T}}.\end{aligned} \tag{3.36}$$

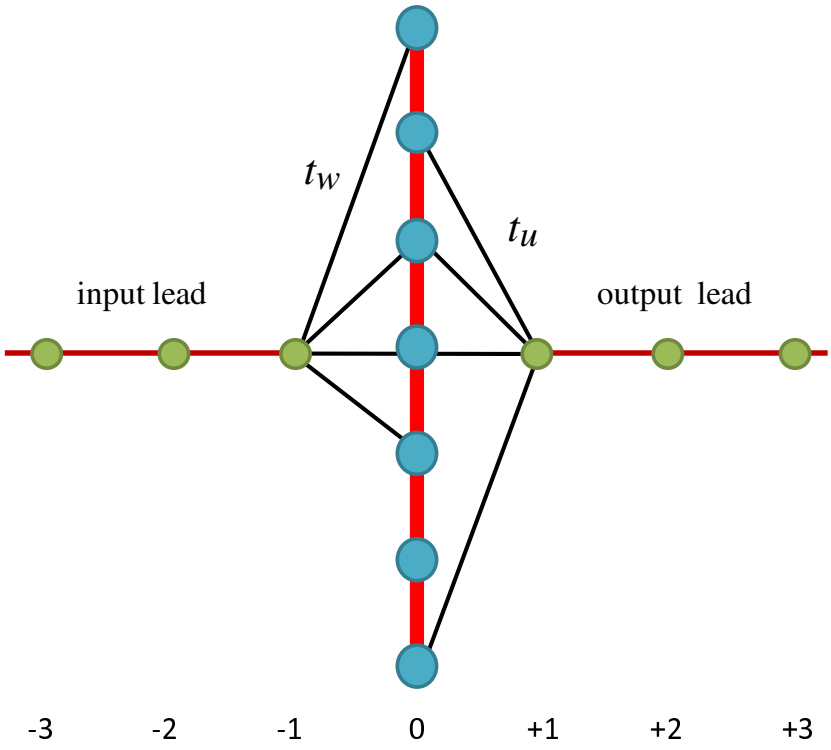

Figure 3.4

Fully-connected n=7 sites, not all symmetrically connected to input, output.

Then $\mathbf{M}\mathbf{M}^{-1} = I$. The forty terms of the product $\mathbf{M}\mathbf{M}^{-1}$ are shown in Table (3.1). The product of vectors and the transposes of vectors are defined as:

$$
\begin{aligned}
&\mathbf{J} = \vec{e}\vec{e}^{\mathrm{T}}, \\
&\vec{e}^{\mathrm{T}}\mathbf{J} = n\vec{e}^{\mathrm{T}}, \\
&\mathbf{J}\mathbf{I} = \mathbf{J} \\
&\mathbf{J}\mathbf{J} = n\mathbf{J}, \\
&\mathbf{J}\vec{e} = n\vec{e}, \\
&\mathbf{J}\vec{w} = \vec{e}\vec{e}^{\mathrm{T}}\vec{w} = m\vec{e}, \\
&\vec{w}^{\mathrm{T}}\vec{w} = l, \\
&\vec{u}^{\mathrm{T}}\vec{u} = k, \\
&\vec{e}^{\mathrm{T}}\vec{e} = n, \\
&\vec{e}^{\mathrm{T}}\vec{w} = \vec{w}^{\mathrm{T}}\vec{e} = m, \\
&\vec{e}^{\mathrm{T}}\vec{u} = \vec{u}^{\mathrm{T}}\vec{e} = p, \\
&\vec{w}^{\mathrm{T}}\vec{u} = \vec{u}^{\mathrm{T}}\vec{w} = h,
\end{aligned}
\tag{3.37}
$$

where l, m, n, p, h, k are constants or parameters of the 10-Equation-10-Unknown general case. Parameter l is the sum of the square of the input strengths, k is the sum of squares of the output strengths, m is the sum of the input strengths, h is the sum of the product of input and output strengths, and n is the number of fully connected atoms in the blob. Using these constants in the combined product $\mathbf{M}\mathbf{M}^{-1}$ in Table (3.1) gives a modified set of terms (see Table (3.2)). From Table (3.2), collecting like terms yields the ten equations with ten

unknowns. The solutions of these equations are solved with the help of Mathematica. In particular, the ten equations are

$$\begin{aligned}
[X_I Y_I] &= 1 \\
[(X_I + nX_J)Y_{JJ} + mX_J Y_{wJ} + pX_J Y_{uJ}] &= 0 \\
[X_w Y_I + (X_I + lX_w)Y_{ww} + mX_w Y_{Jw} + hX_w Y_{uw}] &= 0 \\
[X_u Y_I + (X_I + kX_u)Y_{uu} + hX_u Y_{wu} + pX_u Y_{Ju}] &= 0 \\
[mX_J Y_{wu} + pX_J Y_{uu} + (X_I + nX_J)Y_{Ju}] &= 0 \\
[(X_I + kX_u)Y_{uJ} + hX_u Y_{wJ} + pX_u Y_{JJ}] &= 0 \\
[pX_J Y_{uw} + mX_J Y_{ww} + (X_I + nX_J)Y_{Jw}] &= 0 \\
[hX_w Y_{uJ} + (X_I + lX_w)Y_{wJ} + mX_w Y_{JJ}] &= 0 \\
[hX_w Y_{ww} + pX_u Y_{Ju} + (X_I + kX_u)Y_{uw}] &= 0 \\
[hX_w Y_{uu} + mX_w Y_{Ju} + (X_I + lX_w)Y_{wu}] &= 0. \qquad (3.38)
\end{aligned}$$

The solution of the first equation is the same trivial form as Eq. (3.7). The other nine equations can be solved in matrix form Eq. (3.39),

$$\left[X_I\mathbf{I}+\left(\begin{array}{ccccccccc} nX_J & 0 & 0 & 0 & pX_J & 0 & mX_J & 0 & 0 \\ 0 & lX_w & 0 & 0 & 0 & mX_w & 0 & hX_w & 0 \\ 0 & 0 & kX_u & pX_u & 0 & 0 & 0 & 0 & hX_u \\ 0 & 0 & pX_J & nX_J & 0 & 0 & 0 & 0 & mX_J \\ pX_u & 0 & 0 & 0 & kX_u & 0 & hX_u & 0 & 0 \\ 0 & mX_J & 0 & 0 & 0 & nX_J & 0 & pX_J & 0 \\ mX_w & 0 & 0 & 0 & hX_w & 0 & lX_w & 0 & 0 \\ 0 & hX_w & 0 & 0 & 0 & pX_u & 0 & kX_u & 0 \\ 0 & 0 & hX_w & mX_w & 0 & 0 & 0 & 0 & lX_w \end{array} \right) \right] \left[\begin{array}{c} Y_{JJ} \\ Y_{ww} \\ Y_{uu} \\ Y_{Ju} \\ Y_{uJ} \\ Y_{Jw} \\ Y_{wJ} \\ Y_{uw} \\ Y_{wu} \end{array} \right] = \left[\begin{array}{c} \frac{-X_J}{X_I} \\ \frac{-X_w}{X_I} \\ \frac{-X_u}{X_I} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{array} \right] \tag{3.39}$$

using Mathematica, and the nine solutions are [42]

$$\begin{aligned} Y_{JJ} &= -\frac{X_J(X_I^2+kX_IX_u+lX_IX_w-h^2X_uX_W+klX_uX_w)}{\Omega} \\ Y_{uJ} &= \frac{X_JX_u(-hmX_w+p(X_I+lX_w))}{\Omega} \\ Y_{uu} &= -\frac{X_u(X_I^2+nX_IX_J+lX_IX_w-m^2X_JX_w+lnX_JX_w)}{\Omega} \\ Y_{Ju} &= \frac{X_JX_u(-hmX_w+p(X_I+lX_w))}{\Omega} \\ Y_{Jw} &= \frac{X_J(-hpX_u+m(X_I+kX_u))X_w}{\Omega} \\ Y_{uw} &= \frac{(-mpX_J+h(X_I+nX_J))X_uX_w}{\Omega} \\ Y_{wu} &= \frac{(-mpX_J+h(X_I+nX_J))X_uX_w}{\Omega} \\ Y_{wJ} &= \frac{X_J(-hpX_u+m(X_I+kX_u))X_w}{\Omega} \\ Y_{ww} &= -\frac{(X_I^2+nX_IX_J+kX_IX_u+knX_JX_u-p^2X_JX_u)X_w}{\Omega}, \end{aligned} \tag{3.40}$$

where

$$\begin{aligned}\Omega &= X_I(X_I^3 - (h^2n + k(m^2 - ln) - 2hmp + lp^2) \\ & X_J X_u X_w + X_I^2(nX_J + kX_u + lX_w + \\ & X_I(-p^2 X_J X_u - m^2 X_J X_w + lnX_J X_w - \\ & h^2 X_u X_w + kX_u(nX_J + lX_w))).\end{aligned} \tag{3.41}$$

Plugging these solutions of Eq. (3.40) into Eq. (3.36) with corresponding parameters for $\mathbf{M^{-1}}$ gives the transmission coefficient $T = |t_T|^2$, where $t_T = 2i(S^*)^2 \sin(q)\vec{u}^{\mathrm{T}}\mathbf{M}^{-1}\vec{w}$.

Table 3.1

The forty terms of the product $\mathbf{M}\mathbf{M}^{-1}$.

$\mathbf{M}\mathbf{M}^{-1}$	$X_I\mathbf{I}$	$X_J\mathbf{J}$	$X_w\vec{w}\vec{w}^{\mathrm{T}}$	$X_u\vec{u}\vec{u}^{\mathrm{T}}$
$Y_I\mathbf{I}$	$X_IY_I\mathbf{I}^2$	$X_JY_J\mathbf{J}$	$X_wY_I\vec{w}\vec{w}^{\mathrm{T}}\mathbf{I}$	$X_uY_I\vec{u}\vec{u}^{\mathrm{T}}\mathbf{I}$
$Y_{JJ}\mathbf{J}$	$X_IY_{JJ}\mathbf{I}\mathbf{J}$	$X_IY_{JJ}\mathbf{J}^2$	$X_wY_{JJ}\mathbf{J}\vec{w}\vec{w}^{\mathrm{T}}$	$X_uY_{JJ}\mathbf{J}\vec{u}\vec{u}^{\mathrm{T}}$
$Y_{ww}\vec{w}\vec{w}^{\mathrm{T}}$	$X_IY_{ww}\vec{w}\vec{w}^{\mathrm{T}}\mathbf{I}$	$X_JY_{ww}\mathbf{J}\vec{w}\vec{w}^{\mathrm{T}}$	$X_wY_{ww}\vec{w}\vec{w}^{\mathrm{T}}\vec{w}\vec{w}^{\mathrm{T}}$	$X_uY_{ww}\vec{u}\vec{u}^{\mathrm{T}}\vec{w}\vec{w}^{\mathrm{T}}$
$Y_{uu}\vec{u}\vec{u}^{\mathrm{T}}$	$X_IY_{uu}\vec{u}\vec{u}^{\mathrm{T}}\mathbf{I}$	$X_JY_{uu}\mathbf{J}\vec{u}\vec{u}^{\mathrm{T}}$	$X_wY_{uu}\vec{w}\vec{w}^{\mathrm{T}}\vec{u}\vec{u}^{\mathrm{T}}$	$X_uY_{uu}\vec{u}\vec{u}^{\mathrm{T}}\vec{u}\vec{u}^{\mathrm{T}}$
$Y_{Ju}\vec{e}\vec{u}^{\mathrm{T}}$	$X_IY_{Ju}\vec{e}\vec{u}^{\mathrm{T}}\mathbf{I}$	$X_JY_{Ju}\mathbf{J}\vec{e}\vec{u}^{\mathrm{T}}$	$X_wY_{Ju}\vec{w}\vec{w}^{\mathrm{T}}\vec{e}\vec{u}^{\mathrm{T}}$	$X_uY_{Ju}\vec{u}\vec{u}^{\mathrm{T}}\vec{e}\vec{u}^{\mathrm{T}}$
$Y_{uJ}\vec{u}\vec{e}^{\mathrm{T}}$	$X_IY_{uJ}\vec{u}\vec{e}^{\mathrm{T}}\mathbf{I}$	$X_JY_{uJ}\mathbf{J}\vec{u}\vec{e}^{\mathrm{T}}$	$X_wY_{uJ}\vec{w}\vec{w}^{\mathrm{T}}\vec{u}\vec{e}^{\mathrm{T}}$	$X_uY_{uJ}\vec{u}\vec{u}^{\mathrm{T}}\vec{u}\vec{e}^{\mathrm{T}}$
$Y_{Jw}\vec{e}\vec{w}^{\mathrm{T}}$	$X_IY_{Jw}\vec{e}\vec{w}^{\mathrm{T}}\mathbf{I}$	$X_JY_{Jw}\mathbf{J}\vec{e}\vec{w}^{\mathrm{T}}$	$X_wY_{Jw}\vec{w}\vec{w}^{\mathrm{T}}\vec{e}\vec{w}^{\mathrm{T}}$	$X_uY_{Jw}\vec{u}\vec{u}^{\mathrm{T}}\vec{e}\vec{w}^{\mathrm{T}}$
$Y_{wJ}\vec{w}\vec{e}^{\mathrm{T}}$	$X_IY_{wJ}\vec{w}\vec{e}^{\mathrm{T}}\mathbf{I}$	$X_JY_{wJ}\mathbf{J}\vec{w}\vec{e}^{\mathrm{T}}$	$X_wY_{wJ}\vec{w}\vec{w}^{\mathrm{T}}\vec{w}\vec{e}^{\mathrm{T}}$	$X_uY_{wJ}\vec{u}\vec{u}^{\mathrm{T}}\vec{w}\vec{e}^{\mathrm{T}}$
$Y_{uw}\vec{u}\vec{w}^{\mathrm{T}}$	$X_IY_{uw}\vec{u}\vec{w}^{\mathrm{T}}\mathbf{I}$	$X_JY_{uw}\mathbf{J}\vec{u}\vec{w}^{\mathrm{T}}$	$X_wY_{uw}\vec{w}\vec{w}^{\mathrm{T}}\vec{u}\vec{w}^{\mathrm{T}}$	$X_uY_{uw}\vec{u}\vec{u}^{\mathrm{T}}\vec{u}\vec{w}^{\mathrm{T}}$
$Y_{wu}\vec{w}\vec{u}^{\mathrm{T}}$	$X_IY_{wu}\vec{w}\vec{u}^{\mathrm{T}}\mathbf{I}$	$X_JY_{wu}\mathbf{J}\vec{w}\vec{u}^{\mathrm{T}}$	$X_wY_{wu}\vec{w}\vec{w}^{\mathrm{T}}\vec{w}\vec{u}^{\mathrm{T}}$	$X_uY_{wu}\vec{u}\vec{u}^{\mathrm{T}}\vec{w}\vec{u}^{\mathrm{T}}$

Table 3.2

The forty terms of product $\mathbf{M}\mathbf{M}^{-1}$ modified with constants.

$\mathbf{M}\mathbf{M}^{-1}$	$X_I\mathbf{I}$	$X_J\mathbf{J}$	$X_w\vec{w}\vec{w}^{\mathrm{T}}$	$X_u\vec{u}\vec{u}^{\mathrm{T}}$
$Y_I\mathbf{I}$	X_IY_I	$X_JY_J\mathbf{J}$	$X_wY_I\vec{w}\vec{w}^{\mathrm{T}}$	$X_uY_I\vec{u}\vec{u}^{\mathrm{T}}$
$Y_{JJ}\mathbf{J}$	$X_IY_{JJ}\mathbf{J}$	$X_IY_{JJ}\ n\ \mathbf{J}$	$X_wY_{JJ}\ m\ \vec{w}\vec{e}^{\mathrm{T}}$	$X_uY_{JJ}\ p\ \vec{u}\vec{e}^{\mathrm{T}}$
$Y_{ww}\vec{w}\vec{w}^{\mathrm{T}}$	$X_IY_{ww}\vec{w}\vec{w}^{\mathrm{T}}$	$X_JY_{ww}\ m\ \vec{e}\vec{w}^{\mathrm{T}}$	$X_wY_{ww}\ l\ \vec{w}\vec{w}^{\mathrm{T}}$	$X_uY_{ww}\ h\ \vec{u}\vec{w}^{\mathrm{T}}$
$Y_{uu}\vec{u}\vec{u}^{\mathrm{T}}$	$X_IY_{uu}\vec{u}\vec{u}^{\mathrm{T}}$	$X_JY_{uu}\ p\ \vec{e}\vec{u}^{\mathrm{T}}$	$X_wY_{uu}\ h\ \vec{w}\vec{u}^{\mathrm{T}}$	$X_uY_{uu}\ k\ \vec{u}\vec{u}^{\mathrm{T}}$
$Y_{Ju}\vec{e}\vec{u}^{\mathrm{T}}$	$X_IY_{Ju}\vec{e}\vec{u}^{\mathrm{T}}$	$X_JY_{Ju}\ n\ \vec{e}\vec{u}^{\mathrm{T}}$	$X_wY_{Ju}\ m\ \vec{w}\vec{u}^{\mathrm{T}}$	$X_uY_{Ju}\ p\ \vec{u}\vec{u}^{\mathrm{T}}$
$Y_{uJ}\vec{u}\vec{e}^{\mathrm{T}}$	$X_IY_{uJ}\vec{u}\vec{e}^{\mathrm{T}}$	$X_JY_{uJ}\ p\ \mathbf{J}$	$X_wY_{uJ}\ h\ \vec{w}\vec{e}^{\mathrm{T}}$	$X_uY_{uJ}\ k\ \vec{u}\vec{e}^{\mathrm{T}}$
$Y_{Jw}\vec{e}\vec{w}^{\mathrm{T}}$	$X_IY_{Jw}\vec{e}\vec{w}^{\mathrm{T}}$	$X_JY_{Jw}\ n\ \vec{e}\vec{w}^{\mathrm{T}}$	$X_wY_{Jw}\ m\ \vec{w}\vec{w}^{\mathrm{T}}$	$X_uY_{Jw}\ p\ \vec{u}\vec{w}^{\mathrm{T}}$
$Y_{wJ}\vec{w}\vec{e}^{\mathrm{T}}$	$X_IY_{wJ}\vec{w}\vec{e}^{\mathrm{T}}$	$X_JY_{wJ}\ m\ \mathbf{J}$	$X_wY_{wJ}\ l\ \vec{w}\vec{e}^{\mathrm{T}}$	$X_uY_{wJ}\ h\ \vec{u}\vec{e}^{\mathrm{T}}$
$Y_{uw}\vec{u}\vec{w}^{\mathrm{T}}$	$X_IY_{uw}\vec{u}\vec{w}^{\mathrm{T}}$	$X_JY_{uw}\ p\ \vec{e}\vec{w}^{\mathrm{T}}$	$X_wY_{uw}\ h\ \vec{w}\vec{w}^{\mathrm{T}}$	$X_uY_{uw}\ k\ \vec{u}\vec{w}^{\mathrm{T}}$
$Y_{wu}\vec{w}\vec{u}^{\mathrm{T}}$	$X_IY_{wu}\vec{w}\vec{u}^{\mathrm{T}}$	$X_JY_{wu}\ m\ \vec{e}\vec{u}^{\mathrm{T}}$	$X_wY_{wu}\ l\ \vec{w}\vec{u}^{\mathrm{T}}$	$X_uY_{wu}\ h\ \vec{u}\vec{u}^{\mathrm{T}}$

3.2.2 10-Equation-10-Unknown: RG formalism

In the most general case of the fully connected n-site blob (10-Equation-10-Unknown) where t_w is not necessarily proportional to t_u, in which case the five corresponding parameters $l, m, n, p, h,$ and k are comparable to those in Eq. (3.27), and they are

$$\begin{aligned}
\sum_{i=1}^{n} t_{wi} &= m \\
\sum_{i=1}^{n} t_{ui} &= p \\
\sum_{i=1}^{n} t_{wi}^2 &= l \\
\sum_{i=1}^{n} t_{ui}^2 &= k \\
\sum_{i=1}^{n} t_{wi} t_{ui} &= h.
\end{aligned} \tag{3.42}$$

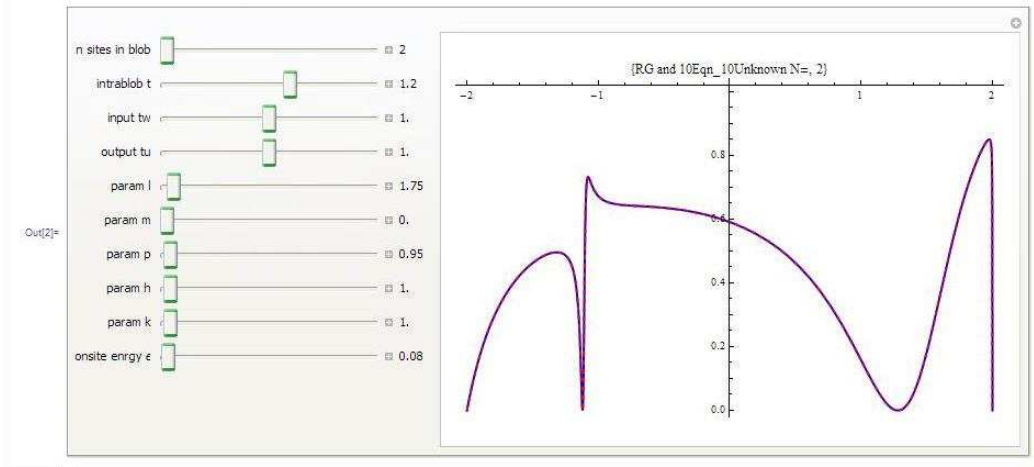

Figure 3.5

Comparison of the Matrix and RG methods, 10-Equation-10-Unknown case.

For an $n+m$-site fully-connected blob of which m sites will be decimated, then the matrix equation to be solved is

$$\begin{pmatrix} e^{iq}-E & \vec{t}_w^{\mathrm{T}} & \vec{t}_{wd}^{\mathrm{T}} & t_s \\ \vec{t}_w & \mathbf{A} & \mathbf{J} & \vec{t}_u \\ \vec{t}_{wd} & \mathbf{J}^{\mathrm{T}} & \mathbf{C} & \vec{t}_{ud} \\ t_s & \vec{u}^{\mathrm{T}} & \vec{t}_{ud}^{\mathrm{T}} & e^{iq}-E \end{pmatrix} \begin{pmatrix} 1+r \\ \vec{\psi} \\ \vec{\theta} \\ t_T \end{pmatrix} = \begin{pmatrix} 2i\sin(q) \\ \vec{0}_n \\ \vec{0}_m \\ 0 \end{pmatrix}, \tag{3.43}$$

where $e^{iq} = \cos(q) + i\sin(q) = \frac{E}{2} + i\frac{\sqrt{4-E^2}}{2}$. The matrix $\mathbf{A} = (\epsilon - E - t)\mathbf{I} + t\mathbf{J}$ is a $n \times n$ and the matrix $\mathbf{C} = (\epsilon - E - t)\mathbf{I} + t\mathbf{J}$ is $m \times m$ so $\mathbf{J}$ is of size $n \times m$. Similarly vectors $\vec{t}_w, \vec{t}_u, \vec{\psi}, \vec{0}_n$ are all of length n while the vectors $\vec{t}_{wd}, \vec{t}_{ud}, \vec{\theta}, \vec{0}_m$ are all of length m.

The decimated renormalized matrix has the form

$$\begin{pmatrix} e^{iq}+\epsilon_c-E-\vec{t}_{wd}^{\mathrm{T}}\mathbf{C}^{-1}\vec{t}_{wd} & \vec{t}_w^{\mathrm{T}}-\vec{t}_{wd}^{\mathrm{T}}\mathbf{C}^{-1}\mathbf{B}^{\mathrm{T}} & t_s-\vec{t}_{wd}^{\mathrm{T}}\mathbf{C}^{-1}\vec{t}_{ud} \\ \vec{t}_w-\mathbf{B}\mathbf{C}^{-1}\vec{t}_{wd} & \mathbf{A}-\mathbf{B}\mathbf{C}^{-1}\mathbf{B}^{\mathrm{T}} & \vec{t}_u-\mathbf{B}\mathbf{C}^{-1}\vec{t}_{ud} \\ t_s-\vec{t}_{ud}^{\mathrm{T}}\mathbf{C}^{-1}\vec{t}_{wd} & \vec{t}_u^{\mathrm{T}}-\vec{t}_{ud}^{\mathrm{T}}\mathbf{C}^{-1}\mathbf{B}^{\mathrm{T}} & e^{iq}+\epsilon_c-E-\vec{t}_{ud}^{\mathrm{T}}\mathbf{C}^{-1}\vec{t}_{ud} \end{pmatrix} \begin{pmatrix} 1+r \\ \vec{\psi} \\ t_T \end{pmatrix} = \begin{pmatrix} 2i\sin(q) \\ \vec{0}_n \\ 0 \end{pmatrix}, \tag{3.44}$$

where $\mathbf{C} = \mathbf{C}^{\mathrm{T}}$ implying that $\mathbf{C}^{-1}$ is also symmetric. Again $\vec{t}_{ud}^{\mathrm{T}}\mathbf{C}^{-1}\vec{t}_{wd} = \vec{t}_{wd}^{\mathrm{T}}\mathbf{C}^{-1}\vec{t}_{ud}$, so the final matrix is symmetric. This completes the final RG decimation equations in matrix form for the fully connected blob.

Now assuming that all input and output vectors are proportional to $\vec{e}$, for example $\vec{t}_u = t_{io}\vec{e}$ and $\vec{t}_{ud} = t_{io}\vec{e}$. Then, since

$$\mathbf{C}^{-1} = \frac{1}{\epsilon - E - t}\mathbf{I} - \frac{t}{(\epsilon - E - t)(\epsilon - E - t + mt)}\mathbf{J} \tag{3.45}$$

one has the RG transformations [3, 4, 28]

$$
\begin{aligned}
t_s' &= t_s - \frac{mt_{io}^2}{\epsilon - E - t} + \frac{tt_{io}^2 m^2}{(\epsilon - E - t)(\epsilon - E - mt)} = t_s + R_m \\
\epsilon' &= \epsilon_c - \frac{mt_{io}^2}{\epsilon - E - t} + \frac{tt_{io}^2 m^2}{(\epsilon - E - t)(\epsilon - E - mt)} = \epsilon_c + R_m \\
t_{io}' &= t_{io} - \frac{mt_{io}^2}{\epsilon - E - t} + \frac{tt_{io}^2 m^2}{(\epsilon - E - t)(\epsilon - E - mt)} = t_{io} + R_m \\
t' &= t - \frac{mt_{io}^2}{\epsilon - E - t} + \frac{tt_{io}^2 m^2}{(\epsilon - E - t)(\epsilon - E - mt)} = t + R_m \\
\epsilon' &= \epsilon - \frac{mt_{io}^2}{\epsilon - E - t} + \frac{tt_{io}^2 m^2}{(\epsilon - E - t)(\epsilon - E - mt)} = \epsilon + R_m
\end{aligned}
\tag{3.46}
$$

with

$$
R_m = -\frac{mt_{io}^2}{\epsilon - E - t} + \frac{tt_{io}^2 m^2}{(\epsilon - E - t)(\epsilon - E - mt)} \tag{3.47}
$$

3.3 Missing Bonds in Fully Connected Blob

Another of the many applications is to consider transmission when one or more of the intra-blob bonds are removed. This course of action is to shift from an almost perfect blob toward that of a real nanoparticle model. In essence this section examines how the "mean-field" like solutions of Sec. (2.3) are affected when a few values for the hopping parameters within the blob are removed.

Consider the case of a general blob, with the Hamiltonian dependent on the general values for the hopping parameter between the n sites within the blob, $t_{l,j}$ for $1 \leq l, j \leq n$, and the onsite energy of each of the n blob sites, ϵ_l for $1 \leq l \leq n$. In addition, the solution will depend on the connections between the left site at $x = -1$, elements of $\vec{t}_w$ of $t_{L,l}$ for $1 \leq l \leq n$, and between the right site at $x = +1$, elements of $\vec{t}_u$ of $t_{l,R}$ for $1 \leq l \leq n$. Thus the symmetric matrix to solve is given by Eq. (2.24) and Eq. (2.26), has no symmetry in $\mathcal{H}$, or $\vec{u}$ or $\vec{w}$ can only be solved numerically, at least for large n. The matrix $\mathbf{L}$ of Eq. (2.24)

is an $n \times n$ matrix with complex elements. Physically the matrix $\mathbf{L}$ must be non-singular, except perhaps at particular special values of the energy E such as $|E| = 2$. The matrix is not Hermitian but is symmetric.

This matrix is solved numerically [35] using the program for linear algebraic solutions in $Numerical\ Recipes$ [38], particularly the LUDCMP program to perform the LU decomposition of the matrix $\mathbf{L}$ and the subroutine LUBKSB to solve the set of linear equations after the LU decomposition has been carried out. The routines are first converted from real arithmetic to complex arithmetic for the input and output matrices and vectors. Besides, checks are placed to determine singularities for particular values of the energy.

A numerical solution in this fashion has been performed by Cuansing and Nakanishi to study quantum interference effects in particle transport through square lattices. [9, 10] The focus here is on analyzing the effect on the "mean-field" like solutions when some of the intra-blob bonds are removed.

Figure (3.6) shows results for $n = 32$ blob atoms. The number of bonds that connect all atoms to all other atoms is given by $n(n-1)/2$. Therefore for $n = 32$, there will be 496 bonds. The transmission curve for this fully connected case is similar to those shown in Fig. (2.1). When two bonds chosen at random are removed the transmission remains unchanged over much of the energy spectrum, but shows a very distinct drop in transmission near $E = -0.8$ for this particular two-bond removal. Further changes in the transmission are shown in Fig. (3.6) when 16 randomly chosen bonds have been removed. As the number of bonds continue to decrease the transmission curve takes additional dips and resonances. As the number of bonds approach one-half the number for a fully con-

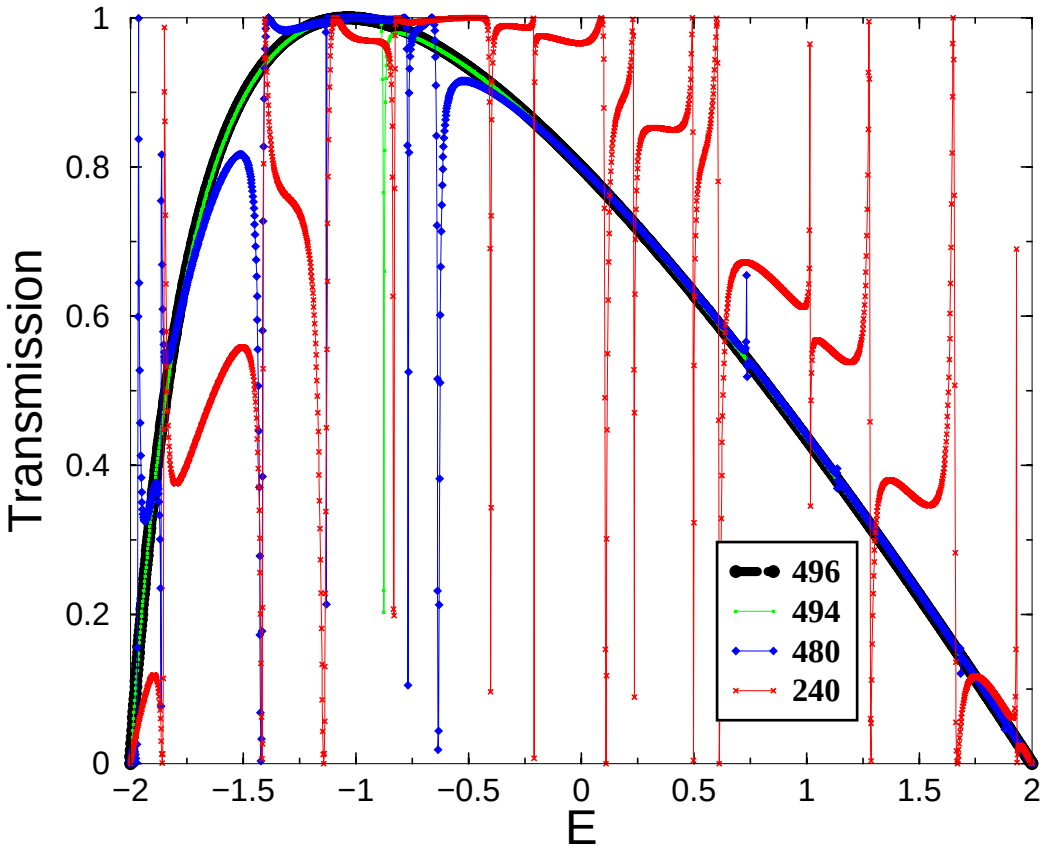

Figure 3.6

Transmission versus energy for decreasing number of inter-blob connections.

nected configuration, the spectrum looks very much like that of any disordered system with random impurities in one or two dimensions [1, 2, 8, 17, 20]. The combination of a one-dimensional incoming and outgoing lead wires attached to an “almost perfect” blob that is multiply connected producing such a disordered transmission is remarkable.

CHAPTER 4

CONCLUSIONS AND DISCUSSION

An general expression for quantum transmission of spinless electrons through a model of a nanoparticle has been derived using finite-dimensional matrix algebra. The general expressions required the calculation of the inverse of an $n \times n$ matrix. Particular models solved were a single site in the blob, a two-site one-dangle blob, a fully connected symmetric case that resulted from 5-Equations and 5-Unknowns, and the most general case of the 10-Equations and 10-Unknowns solutions method where the fully connected atoms in the blob can have both symmetrical and asymmetrically connections to the input and output leads. Similar expressions were obtained using the RG method and compared to the conventional Green's function method. The comparisons of different solutions methods for the assortments of connections show no difference in the transmission. The case where some of the inter-blob hopping bonds were randomly removed required numerical solutions using matrix algebra. The numerical solutions are found to be consistent with those analytical solutions.

From all the above considerations it can be prudently concluded that the matrix algebra method is more feasible since it is not inundated with mathematical sophistication, and thereby does not obscure the physics. It is more economic in the sense it requires half the number of the matrix calculations that is required by the Green's function method. For

example, the transmission coefficient Eq. (1.47) along with Eq. (1.46) for the general case of n atoms in the blob requires the trace of the products of two coupling terms Γ_1 and Γ_2 which are matrices with their respective self energy terms $\Sigma^{\pm}$, and two more matrices $G^{\pm}$ that has the Hamiltonian of n blob atoms. This requirement plus the need to find the inverses of $G^{\pm}$ makes this method less economic when compared with the matrix method that requires only the inverse of the matrix of the blob of n atoms. The transmission calculations using the matrix method are in par with those derived by the standard methods. And furthermore the matrix algebra method provides parameters that facilitates one to modify the nanoparticle model in terms of the hopping energies, the onsite energies, the connection strengths, and the number of sites in the model. The method is applicable to other complex systems, and also has numerical solutions for the model discussed.

This dissertation has developed a general formalism using the matrix algebra method and the RG method for calculating the transmission coefficient of quantum electrons through a model of a nanoparticle. This formalism can now be applied to more realistic models, either analytically or numerically.

REFERENCES

[1] E. Abrahams, P. W. Anderson, D. C. Licciardello, and T. V. Ramakrishnan, "Scaling Theory of Localization: Absence of Quantum Diffusion in Two Dimensions," *Phys. Rev. Lett.*, vol. 42, 1979, pp. 673–676.

[2] P. W. Anderson, "Absence of Diffusion in Certain Random Lattices," *Phys. Rev*, vol. 109, 1958, pp. 1492–1505.

[3] H. Aoki, "Real-space Renormalization-group Theory for Anderson Localization: Decimation Method for Electron Systems," *J. Phys. C*, vol. 13, 1980, pp. 3369–86.

[4] H. Aoki, "Decimation Method of Real-space Renormalization for Electron Systems with Application to Random Systems," *Physica A*, vol. 114, 1982, pp. 538–542.

[5] M. Y. Azbel and D. P. DiVincenzo, "Finite-Temperature Conductance in One Dimension," *Physical Review B*, vol. 30, 1984, pp. 6877–6888.

[6] T. B. Boykin, "Tight-binding-like Expressions for the Continuous-space Electromagnetic Coupling Hamiltonian," *Am. J. Phys.*, vol. 69, 2001, pp. 793 – 798.

[7] T. B. Boykin, "The Discretized Schrödinger Equation and Simple Models for Semiconductor Quantum Wells," *Eur. J. Phys.*, vol. 25, 2004, pp. 503 – 514.

[8] S. Çalişkan, M. A. Novotny, and J. I. Cerdá, "Transport through Small World Networks," *J. Appl. Phys.*, vol. 102, 2007, article 013707 [1-7 pages].

[9] E. Cuansing and H. Nakanishi, "Quantum Transport in Randomly Diluted Quantum Percolation Cluster in Two Dimensions," *Physica A: Statistical Mechanics and its Applications*, vol. 387, 2008, pp. 806–816.

[10] E. Cuansing and H. Naknishi, "Quantum Interference Effect, in Particle Transport through Square Lattices," *Phys. Rev. E*, vol. 70, 2004, article 0661421 [1-4 pages].

[11] D. D. Daboul, I. Chang, and A. Aharony, "Series Expansion Study of Quantum Percolation on the Square Lattice," *Eur. Phys. J. B*, vol. 16, 2000, pp. 303–316.

[12] S. Datta, *Electronic Transport in Mesoscopic Systems*, Cambridge University Press, Cambridge, 1995.

[13] S. Datta, *Quantum Transport: Atom to Transistor*, Cambridge University Press, Cambridge, 2005.

[14] K. E. Drexler, *Nanosystems: Molecular Machinary, Manufacturing, and Computation*, John Wiley & Sons, New York, 1992.

[15] W. R. Frensley, *VLSI Electronic: Miscrostructure Science*, Academic Press, San Diego, 1995.

[16] W. A. Harrison, *Electronic Structure and Properties of Solids*, New York, Dover, 1989.

[17] Y. Imry and R. Landauer, "Conductance Viewed as Transmission," *Rev. Mod. Phys.*, vol. 71, 1999, pp. S306–S312.

[18] L. P. Kadanoff and G. Baym, *Quantum Statistical Mechanics*, Benjamin-Cummings, Reading, MA, 1962.

[19] L. V. Keldysh, "Diagram Technique for Non-equilibrium Processes," *Sov. Phys. JETP, 1018*, vol. 20, 1965, p. 1515.

[20] B. Kramer and A. MacKinnon, "Localization: Theory and Experiment," *Rep. Prog. Phys.*, vol. 56, 1993, pp. 1469–1564.

[21] H. A. Kramers and G. H. Wannier, "Statistics of Two-dimensional Ferromagnet. Part I," *Phys. Rev.*, vol. 60, 1941, pp. 252–262.

[22] R. Kubo, "Statistical-Mechanical Theory of Irreversible Process. 1.General Theory and Simple Applications to Magnetic and Conduction Problems," *J. Phys. Soc.*, vol. 12, 1957, pp. 570 – 586.

[23] R. Kubo, M. Toda, and N. Hashitsume, *Statistical Physics II. Nonequilibrium Statistical Mechanics*, Springer-Verlag, Berlin, 1985.

[24] R. Landauer, "Spatial Variation of Currents and Fields due to Localized Scatterers in Metallic Conduction," *IBM J. Res. Dev. 1*, 1957, pp. 223–231.

[25] R. Landauer, "Electrical Resistance of One-dimensional Lattices," *Philosoph. Mag.*, vol. 21, 1970, pp. 863–867.

[26] A. F. J. Levi, *Applied Quantum Mechanics: for Engineers and Physicists*, 1st edition, Cambridge University Press, Cambridge, 2003.

[27] R. C. Merkle, "Nanotechnology: What will it mean?," *IEEE Spectrum, Inside Technology*, 2001.

[28] C. Monthus and T. Garel, "Statistics of Renormalized On-site Energies and Renormalized Hoppings for Anderson Localization in Two and Three Dimensions," *Phys. Rev. B*, vol. 80, no. 2, 2009, article 024203 [1-11 pages].

[29] M. P. Nightingale, *Finite-size Scaling and Simulation of Statistical Mechanical Systems*, World Scientific, Singapore, 1990.

[30] M. P. Nightingale and H. W. J. Blöte, "Transfer-matrix Monte Carlo Estimates of Critical Points in the Simple-cubic Ising, Planar, and Heisenberg Models," *Phys. Rev. B*, vol. 54, 1996, pp. 1001–1008.

[31] M. A. Novotny, "Matrix Products with Applications to Classical Statistical Mechanics," *J. Math. Phys.*, vol. 20, 1979, pp. 1146–1150.

[32] M. A. Novotny, "Equivalence of Transfer Matrices," *J. Math. Phys.*, vol. 29, 1988, pp. 2280–2287.

[33] M. A. Novotny, "Critical Exponents for the Ising Model between One and Two Dimensions," *Phys. Rev. E*, vol. 46, 1992, pp. 2939–2950.

[34] M. A. Novotny and P. A. Rikvold, "Conventional and Unconventional Numerical Transfer Methods," Mat. Res. Soc. Symp. Proc. 278, 1992, pp. 15–20.

[35] M. A. Novotny and L. Solomon, "Electronic Transport through Nanoparticles using Finite-Dimensional Matrix Algebra," submitted to *Am. J. Phys*, (detailed steps in article at undergraduate level).

[36] M. A. Novotny, L. Solomon, C. Varghese, and S. Boettcher, "Renormalization Group Calculation of Electron Transport through a Fully Connected Blob," *Physica Procedia*, 2010, in press.

[37] D. G. Pettifer, *Bonding and Structure of Molecules and Solids*, Oxford, Clarendon Press, 1995.

[38] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, *Numerical Recipes: The Art of Scientific Computing*, Cambridge University Press, Cambridge, 1986.

[39] M. A. Ratner and D. Ratner, *Nanotechnology: A Gentle Introduction to the Next Big Idea*, Prentice Hall Professional Technical Reference, New Jersey, 2002.

[40] L. I. Schiff, *Quantum Mechanics*, McGraw Hill Co., New York, 1949.

[41] F. Schwabl, *Statistical Mechanics*, 3rd edition, Springer - Verlag Berlin Hiedelberg, 2000.

[42] L. Solomon and M. A. Novotny, "Matrix and Renormalization Group Calculations of Electronic Transport through Nanoparticles: Fully Connected General Case - 10 Equations 10 Unknowns," to be submitted to *Phys. Rev. E*.

[43] K. G. Wilson, "The Renormalization Group: Critical Phenomena and the Kondo Problem," *Rev. Mod. Phys.*, vol. 47, 1975, pp. 773 – 840.

[44] K. G. Wilson, "The Renormalization Group and Critical Phenomena," *Rev. Mod. Phys.*, vol. 55, 1983, pp. 583 – 600.

[45] N. Zettili, *Quantum Mechanics Concepts and Applications*, John Wiley &,Sons, Ltd. Chichester, 2003.

[46] R. L. Zimmerman and F. I. Olness, *Mathematica for Physics*, 2nd edition, Addison-Wesley, New York, 2002.

APPENDIX A

EASY JAVA SIMULATION CODE: SINGLE IMPURITY TRANSPORT

```
/*
 * Class : transportsingleimpurity.java
 *  Generated using  *  Easy Java Simulations Version 4.1
 */

package transportsingleimpurity_pkg;

import org.colos.ejs.library._EjsConstants;

import javax.swing.event.*;
import javax.swing.*;
import java.awt.event.*;
import java.awt.*;
import java.net.*;
import java.util.*;
import java.io.*;
import java.lang.*;

public class transportsingleimpurity extends org.colos.ejs.
                                      library.AbstractModel {

 public transportsingleimpuritySimulation _simulation=null;
  public transportsingleimpurityView _view=null;
  public transportsingleimpurity _model=this;

  public int _getStepsPerDisplay() { return 1; }

  static public String _getEjsModel() {
                 return "./transportsingleimpurity.xml"; }

  static public String _getModelDirectory() { return ""; }

  static public java.util.Set<String> _getEjsResources() {
    java.util.Set<String> list = new java.util.
                           HashSet<String>(); return list;
  };

  static public void main (String[] args) {
    String lookAndFeel = null;
    boolean decorated = true;
    if (args!=null) for (int i=0; i<args.length; i++) {
      if      (args[i].equals("-_lookAndFeel"))
                                      lookAndFeel = args[++i];
```

```
    else if (args[i].equals("-_decorateWindows"))
                                          decorated = true;
    else if (args[i].equals("-_doNotDecorateWindows"))
                                         decorated = false;
  }
  if (lookAndFeel!=null) org.opensourcephysics.display.
        OSPRuntime.setLookAndFeel(decorated,lookAndFeel);
  org.opensourcephysics.tools.ResourceLoader.
                                       addSearchPath(".");
        // This is for relative resources
  boolean pathsSet = false;
  try { // in case of security problems
    if (System.getProperty("osp_ejs")!=null) {
        // Running under EJS
     org.colos.ejs.library.Simulation.setPathToLibrary
  ("C:/Documents and Settings/LAZARUS/Desktop/
                                  EJS_4.1/bin/config/");
        // This is for classes (such as EjsMatlab)
           which needs to know where the library is
      pathsSet = true;
    }
  }
  catch (Exception _exception) { pathsSet = false; }
        // maybe an unsigned Web start?
  try { org.colos.ejs.library.control.EjsControl.
        setDefaultScreen(Integer.parseInt
        (System.getProperty("screen"))); }
        // set default screen
  catch (Exception _exception) { }
        // Ignore any error here
  if (!pathsSet) {
    org.colos.ejs.library.Simulation.setPathToLibrary
  ("C:/Documents and Settings/LAZARUS/Desktop/
                                   EJS_4.1/bin/config/");
        // This is for classes (such as EjsMatlab)
           which needs to know where the library is
  }
  transportsingleimpurity __theModel =
                    new transportsingleimpurity (args);
}

public transportsingleimpurity () {
             this (null, null, null,null,null,false); }
```

```
        // slave application

  public transportsingleimpurity (String[] _args) {
              this (null, null, null,null,_args,true); }

  public transportsingleimpurity (String _replaceName,
   java.awt.Frame _replaceOwnerFrame, java.net.
    URL _codebase, org.colos.ejs.library.
     LauncherApplet _anApplet, String[] _args,
       boolean _allowAutoplay) {
    __theArguments = _args;
    __theApplet = _anApplet;
    java.text.NumberFormat _Ejs_format =
                    java.text.NumberFormat.getInstance();
    if (_Ejs_format instanceof java.text.DecimalFormat) {
      ((java.text.DecimalFormat) _Ejs_format).
           getDecimalFormatSymbols()
                             .setDecimalSeparator('.');
    }
    _simulation = new transportsingleimpuritySimulation
(this,_replaceName,_replaceOwnerFrame,_
                                codebase,_allowAutoplay);
    _view = (transportsingleimpurityView)
                                    _simulation.getView();
    _simulation.processArguments(_args);
  }

  public org.colos.ejs.library.View getView()
                                          { return _view; }

  public org.colos.ejs.library.Simulation getSimulation() {
                                       return _simulation; }

  public void _resetSolvers() {
    _external.resetIC();
  }
  // --- Implementation of ExternalClient ---

  public String _externalInitCommand
                             (String _applicationFile) {
    StringBuffer _external_initCommand=new StringBuffer();
    return _external_initCommand.toString();
  }
```

```
 public synchronized void _externalSetValues(boolean _any,
 org.colos.ejs.library.external.ExternalApp _application){
 }

 public synchronized void _externalGetValues(boolean _any,
 org.colos.ejs.library.external.ExternalApp _application){
 }

 // --- End of implementation of ExternalClient ---

// ------------------------------------------
// Variables defined by the user
// ------------------------------------------

 public double E  = -2; // Variables.Var Table:1
 public double eps  = 0.0; // Variables.Var Table:2
 public double tw  = 1.0; // Variables.Var Table:3
 public double tu  = 1.0; // Variables.Var Table:4
 public Object myColor  = null; // Variables.Var Table:5

// ------------------------------------------
// Methods defined by the user
// ------------------------------------------

// --- Initialization

 public void _initialization1 () {
          // > Initialization.Init Page
 }  // > Initialization.Init Page

// --- Evolution

// --- Constraints

 public void _constraints1 () {
      // > Fixed relations.FixRel Page
  int e = (int) eps; // > Fixed relations.FixRel Page:1
   {   // > Fixed relations.FixRel Page:2
     if (e == -2.0) myColor = Color.black;
       // > Fixed relations.FixRel Page:3
     if (e == -1.0) myColor = Color.red;
       // > Fixed relations.FixRel Page:4
```

```
    if (e == 0.0) myColor = Color.orange;
      // > Fixed relations.FixRel Page:5
    if (e == 1.0) myColor = Color.yellow;
      // > Fixed relations.FixRel Page:6
    if (e == 2.0) myColor = Color.white;
      // > Fixed relations.FixRel Page:7
      // > Fixed relations.FixRel Page:8
   }  // > Fixed relations.FixRel Page:9
 }  // > Fixed relations.FixRel Page

// --- Custom

 //public void lib_Page () {  // > Custom.Lib Page:1
 //}  // > Custom.Lib Page:2

// --- Methods for view elements

 public double _method_for_tw_inputStrength_lineWidth ()
                                        { return 5*tw; }
 public double _method_for_tu_outputStrength_lineWidth ()
                                        { return 5*tu; }
// ----------------------------------------
// Implementation of interface Model
// ----------------------------------------

 public synchronized void reset () {
   E = -2; // Variables.Var Table:1
   eps = 0.0; // Variables.Var Table:2
   tw = 1.0; // Variables.Var Table:3
   tu = 1.0; // Variables.Var Table:4
 }

 public synchronized void initialize () {
   _initialization1 ();
   _resetSolvers();
 }

 public synchronized void step () {
 }

 public synchronized void update () {
   _constraints1 ();
 }
```

```
  public void _freeMemory () {
    System.gc(); // Free memory from unused old arrays
  }

} // End of class transportsingleimpurityModel


/*
 * Class : transportsingleimpurity.java
 *  Generated using  *  Easy Java Simulations Version 4.1
 */

package transportsingleimpurity_pkg;

import org.colos.ejs.library._EjsConstants;

import javax.swing.event.*;
import javax.swing.*;
import java.awt.event.*;
import java.awt.*;
import java.net.*;
import java.util.*;
import java.io.*;
import java.lang.*;

public class transportsingleimpurity extends org.colos.ejs.
                                        library.AbstractModel {

 public transportsingleimpuritySimulation _simulation=null;
  public transportsingleimpurityView _view=null;
  public transportsingleimpurity _model=this;

  public int _getStepsPerDisplay() { return 1; }

  static public String _getEjsModel() {
                 return "./transportsingleimpurity.xml"; }

  static public String _getModelDirectory() { return ""; }

  static public java.util.Set<String> _getEjsResources() {
    java.util.Set<String> list = new java.util.
                          HashSet<String>(); return list;
  };
```

```
static public void main (String[] args) {
  String lookAndFeel = null;
  boolean decorated = true;
  if (args!=null) for (int i=0; i<args.length; i++) {
    if      (args[i].equals("-_lookAndFeel"))
                                    lookAndFeel = args[++i];
    else if (args[i].equals("-_decorateWindows"))
                                          decorated = true;
    else if (args[i].equals("-_doNotDecorateWindows"))
                                         decorated = false;
  }
  if (lookAndFeel!=null) org.opensourcephysics.display.
        OSPRuntime.setLookAndFeel(decorated,lookAndFeel);
  org.opensourcephysics.tools.ResourceLoader.
                                        addSearchPath(".");
        // This is for relative resources
  boolean pathsSet = false;
  try { // in case of security problems
    if (System.getProperty("osp_ejs")!=null) {
        // Running under EJS
     org.colos.ejs.library.Simulation.setPathToLibrary
  ("C:/Documents and Settings/LAZARUS/Desktop/
                                  EJS_4.1/bin/config/");
        // This is for classes (such as EjsMatlab)
           which needs to know where the library is
      pathsSet = true;
    }
  }
  catch (Exception _exception) { pathsSet = false; }
        // maybe an unsigned Web start?
  try { org.colos.ejs.library.control.EjsControl.
        setDefaultScreen(Integer.parseInt
        (System.getProperty("screen"))); }
        // set default screen
  catch (Exception _exception) { }
        // Ignore any error here
  if (!pathsSet) {
    org.colos.ejs.library.Simulation.setPathToLibrary
  ("C:/Documents and Settings/LAZARUS/Desktop/
                                   EJS_4.1/bin/config/");
        // This is for classes (such as EjsMatlab)
           which needs to know where the library is
```

```
    }
    transportsingleimpurity __theModel =
                        new transportsingleimpurity (args);
  }

  public transportsingleimpurity () {
               this (null, null, null,null,null,false); }
          // slave application

  public transportsingleimpurity (String[] _args) {
               this (null, null, null,null,_args,true); }

  public transportsingleimpurity (String _replaceName,
   java.awt.Frame _replaceOwnerFrame, java.net.
    URL _codebase, org.colos.ejs.library.
     LauncherApplet _anApplet, String[] _args,
       boolean _allowAutoplay) {
    __theArguments = _args;
    __theApplet = _anApplet;
    java.text.NumberFormat _Ejs_format =
                      java.text.NumberFormat.getInstance();
    if (_Ejs_format instanceof java.text.DecimalFormat) {
      ((java.text.DecimalFormat) _Ejs_format).
           getDecimalFormatSymbols()
                              .setDecimalSeparator('.');
    }
    _simulation = new transportsingleimpuritySimulation
(this,_replaceName,_replaceOwnerFrame,_
                                  codebase,_allowAutoplay);
    _view = (transportsingleimpurityView)
                                     _simulation.getView();
    _simulation.processArguments(_args);
  }

  public org.colos.ejs.library.View getView()
                                              { return _view; }

  public org.colos.ejs.library.Simulation getSimulation() {
                                         return _simulation; }

  public void _resetSolvers() {
    _external.resetIC();
  }
```

```
 // --- Implementation of ExternalClient ---

 public String _externalInitCommand
                              (String _applicationFile) {
   StringBuffer _external_initCommand=new StringBuffer();
   return _external_initCommand.toString();
 }

 public synchronized void _externalSetValues(boolean _any,
 org.colos.ejs.library.external.ExternalApp _application){
 }

 public synchronized void _externalGetValues(boolean _any,
 org.colos.ejs.library.external.ExternalApp _application){
 }

 // --- End of implementation of ExternalClient ---

// ------------------------------------------
// Variables defined by the user
// ------------------------------------------

 public double E  = -2; // Variables.Var Table:1
 public double eps  = 0.0; // Variables.Var Table:2
 public double tw  = 1.0; // Variables.Var Table:3
 public double tu  = 1.0; // Variables.Var Table:4
 public Object myColor  = null; // Variables.Var Table:5

// ------------------------------------------
// Methods defined by the user
// ------------------------------------------

// --- Initialization

 public void _initialization1 () {
          // > Initialization.Init Page
 }  // > Initialization.Init Page

// --- Evolution

// --- Constraints

 public void _constraints1 () {
```

```
      // > Fixed relations.FixRel Page
  int e = (int) eps; // > Fixed relations.FixRel Page:1
   {   // > Fixed relations.FixRel Page:2
     if (e == -2.0) myColor = Color.black;
       // > Fixed relations.FixRel Page:3
     if (e == -1.0) myColor = Color.red;
       // > Fixed relations.FixRel Page:4
     if (e == 0.0) myColor = Color.orange;
       // > Fixed relations.FixRel Page:5
     if (e == 1.0) myColor = Color.yellow;
       // > Fixed relations.FixRel Page:6
     if (e == 2.0) myColor = Color.white;
       // > Fixed relations.FixRel Page:7
       // > Fixed relations.FixRel Page:8
    }  // > Fixed relations.FixRel Page:9
 }  // > Fixed relations.FixRel Page

// --- Custom

 //public void lib_Page () {  // > Custom.Lib Page:1
 //}  // > Custom.Lib Page:2

// --- Methods for view elements

 public double _method_for_tw_inputStrength_lineWidth ()
                                        { return 5*tw; }
 public double _method_for_tu_outputStrength_lineWidth ()
                                        { return 5*tu; }
// ------------------------------------------
// Implementation of interface Model
// ------------------------------------------

 public synchronized void reset () {
   E = -2; // Variables.Var Table:1
   eps = 0.0; // Variables.Var Table:2
   tw = 1.0; // Variables.Var Table:3
   tu = 1.0; // Variables.Var Table:4
 }

 public synchronized void initialize () {
   _initialization1 ();
   _resetSolvers();
 }
```

```
  public synchronized void step () {
  }

  public synchronized void update () {
    _constraints1 ();
  }

  public void _freeMemory () {
    System.gc(); // Free memory from unused old arrays
  }

} // End of class transportsingleimpurityModel


/*
 * Class : transportsingleimpurity.java
 *  Generated using  *  Easy Java Simulations Version 4.1
 */

package transportsingleimpurity_pkg;

import org.colos.ejs.library._EjsConstants;

import javax.swing.event.*;
import javax.swing.*;
import java.awt.event.*;
import java.awt.*;
import java.net.*;
import java.util.*;
import java.io.*;
import java.lang.*;

public class transportsingleimpurity extends org.colos.ejs.
                                     library.AbstractModel {

 public transportsingleimpuritySimulation _simulation=null;
  public transportsingleimpurityView _view=null;
  public transportsingleimpurity _model=this;

  public int _getStepsPerDisplay() { return 1; }

  static public String _getEjsModel() {
                return "./transportsingleimpurity.xml"; }
```

```
static public String _getModelDirectory() { return ""; }

static public java.util.Set<String> _getEjsResources() {
  java.util.Set<String> list = new java.util.
                        HashSet<String>(); return list;
};

static public void main (String[] args) {
  String lookAndFeel = null;
  boolean decorated = true;
  if (args!=null) for (int i=0; i<args.length; i++) {
    if      (args[i].equals("-_lookAndFeel"))
                                    lookAndFeel = args[++i];
    else if (args[i].equals("-_decorateWindows"))
                                          decorated = true;
    else if (args[i].equals("-_doNotDecorateWindows"))
                                         decorated = false;
  }
  if (lookAndFeel!=null) org.opensourcephysics.display.
        OSPRuntime.setLookAndFeel(decorated,lookAndFeel);
  org.opensourcephysics.tools.ResourceLoader.
                                       addSearchPath(".");
        // This is for relative resources
  boolean pathsSet = false;
  try { // in case of security problems
    if (System.getProperty("osp_ejs")!=null) {
        // Running under EJS
     org.colos.ejs.library.Simulation.setPathToLibrary
  ("C:/Documents and Settings/LAZARUS/Desktop/
                                  EJS_4.1/bin/config/");
        // This is for classes (such as EjsMatlab)
           which needs to know where the library is
      pathsSet = true;
    }
  }
  catch (Exception _exception) { pathsSet = false; }
        // maybe an unsigned Web start?
  try { org.colos.ejs.library.control.EjsControl.
        setDefaultScreen(Integer.parseInt
        (System.getProperty("screen"))); }
        // set default screen
  catch (Exception _exception) { }
```

```
        // Ignore any error here
    if (!pathsSet) {
      org.colos.ejs.library.Simulation.setPathToLibrary
    ("C:/Documents and Settings/LAZARUS/Desktop/
                                    EJS_4.1/bin/config/");
        // This is for classes (such as EjsMatlab)
           which needs to know where the library is
    }
    transportsingleimpurity __theModel =
                      new transportsingleimpurity (args);
  }

  public transportsingleimpurity () {
              this (null, null, null,null,null,false); }
        // slave application

  public transportsingleimpurity (String[] _args) {
              this (null, null, null,null,_args,true); }

  public transportsingleimpurity (String _replaceName,
   java.awt.Frame _replaceOwnerFrame, java.net.
    URL _codebase, org.colos.ejs.library.
     LauncherApplet _anApplet, String[] _args,
       boolean _allowAutoplay) {
    __theArguments = _args;
    __theApplet = _anApplet;
    java.text.NumberFormat _Ejs_format =
                    java.text.NumberFormat.getInstance();
    if (_Ejs_format instanceof java.text.DecimalFormat) {
      ((java.text.DecimalFormat) _Ejs_format).
           getDecimalFormatSymbols()
                             .setDecimalSeparator('.');
    }
    _simulation = new transportsingleimpuritySimulation
(this,_replaceName,_replaceOwnerFrame,_
                                codebase,_allowAutoplay);
    _view = (transportsingleimpurityView)
                                    _simulation.getView();
    _simulation.processArguments(_args);
  }

  public org.colos.ejs.library.View getView()
                                           { return _view; }
```

```
 public org.colos.ejs.library.Simulation getSimulation() {
                                      return _simulation; }

 public void _resetSolvers() {
   _external.resetIC();
 }
 // --- Implementation of ExternalClient ---

 public String _externalInitCommand
                             (String _applicationFile) {
   StringBuffer _external_initCommand=new StringBuffer();
   return _external_initCommand.toString();
 }

 public synchronized void _externalSetValues(boolean _any,
 org.colos.ejs.library.external.ExternalApp _application){
 }

 public synchronized void _externalGetValues(boolean _any,
 org.colos.ejs.library.external.ExternalApp _application){
 }

 // --- End of implementation of ExternalClient ---

// ----------------------------------------
// Variables defined by the user
// ----------------------------------------

 public double E  = -2; // Variables.Var Table:1
 public double eps  = 0.0; // Variables.Var Table:2
 public double tw  = 1.0; // Variables.Var Table:3
 public double tu  = 1.0; // Variables.Var Table:4
 public Object myColor  = null; // Variables.Var Table:5

// ----------------------------------------
// Methods defined by the user
// ----------------------------------------

// --- Initialization

 public void _initialization1 () {
          // > Initialization.Init Page
```

```
 }  // > Initialization.Init Page

// --- Evolution

// --- Constraints

 public void _constraints1 () {
      // > Fixed relations.FixRel Page
  int e = (int) eps; // > Fixed relations.FixRel Page:1
   {   // > Fixed relations.FixRel Page:2
     if (e == -2.0) myColor = Color.black;
       // > Fixed relations.FixRel Page:3
     if (e == -1.0) myColor = Color.red;
       // > Fixed relations.FixRel Page:4
     if (e == 0.0) myColor = Color.orange;
       // > Fixed relations.FixRel Page:5
     if (e == 1.0) myColor = Color.yellow;
       // > Fixed relations.FixRel Page:6
     if (e == 2.0) myColor = Color.white;
       // > Fixed relations.FixRel Page:7
       // > Fixed relations.FixRel Page:8
    }  // > Fixed relations.FixRel Page:9
 }  // > Fixed relations.FixRel Page

// --- Custom

 //public void lib_Page () {  // > Custom.Lib Page:1
 //}  // > Custom.Lib Page:2

// --- Methods for view elements

 public double _method_for_tw_inputStrength_lineWidth ()
                                        { return 5*tw; }
 public double _method_for_tu_outputStrength_lineWidth ()
                                        { return 5*tu; }
// ----------------------------------------
// Implementation of interface Model
// ----------------------------------------

 public synchronized void reset () {
   E = -2; // Variables.Var Table:1
   eps = 0.0; // Variables.Var Table:2
   tw = 1.0; // Variables.Var Table:3
```

```
    tu = 1.0; // Variables.Var Table:4
  }

  public synchronized void initialize () {
    _initialization1 ();
    _resetSolvers();
  }

  public synchronized void step () {
  }

  public synchronized void update () {
    _constraints1 ();
  }

  public void _freeMemory () {
    System.gc(); // Free memory from unused old arrays
  }

} // End of class transportsingleimpurityModel
```

APPENDIX B

EASY JAVA SIMULATION CODE: 5-EQUATIONS-5-UNKNOWNS

```
/*
 * Class : FullyConnectedBlob.java
 *  Generated using  *  Easy Java Simulations Version 4.1
 */

package FullyConnectedBlob_pkg;

import org.colos.ejs.library._EjsConstants;

import javax.swing.event.*;
import javax.swing.*;
import java.awt.event.*;
import java.awt.*;
import java.net.*;
import java.util.*;
import java.io.*;
import java.lang.*;

public class FullyConnectedBlob extends org.colos.ejs.
                               library.AbstractModel {

  public FullyConnectedBlobSimulation _simulation=null;
  public FullyConnectedBlobView _view=null;
  public FullyConnectedBlob _model=this;

  public int _getStepsPerDisplay() { return 1; }

  static public void main (String[] args) {
    String lookAndFeel = null;
    boolean decorated = true;
    if (args!=null) for (int i=0; i<args.length; i++) {
      if      (args[i].equals("-_lookAndFeel"))
                          lookAndFeel = args[++i];
      else if (args[i].equals("-_decorateWindows"))
                               decorated = true;
      else if (args[i].equals("-_doNotDecorateWindows"))
                              decorated = false;
    }
    if (lookAndFeel!=null) org.opensourcephysics.display.
        OSPRuntime.setLookAndFeel(decorated,lookAndFeel);
    org.opensourcephysics.tools.ResourceLoader.
     addSearchPath("."); // This is for relative resources
    boolean pathsSet = false;
```

```
  try { // in case of security problems
    if (System.getProperty("osp_ejs")!=null)
      { // Running under EJS
      org.colos.ejs.library.Simulation.setPathToLibrary
  ("C:/Documents and Settings/LAZARUS/Desktop/EJS_4.1
                                          /bin/config/");
       // This is for classes (such as EjsMatlab) which
          needs to know where the library is
      pathsSet = true;
    }
  }
  catch (Exception _exception) { pathsSet = false; }
      // maybe an unsigned Web start?
  try { org.colos.ejs.library.control.EjsControl
                                      .setDefaultScreen
  (Integer.parseInt(System.getProperty("screen"))); }
      // set default screen
  catch (Exception _exception) { }
      // Ignore any error here
  if (!pathsSet) {
    org.colos.ejs.library.Simulation.setPathToLibrary
  ("C:/Documents and Settings/LAZARUS/Desktop/EJS_4.1
                                        /bin/config/");
      // This is for classes (such as EjsMatlab)
         which needs to know where the library is
  }
  FullyConnectedBlob __theModel = new
                            FullyConnectedBlob (args);
}

public FullyConnectedBlob ()
 { this (null, null, null,null,null,false); }
      // slave application

public FullyConnectedBlob (String[] _args)
 { this (null, null, null,null,_args,true); }

public FullyConnectedBlob
(String _replaceName, java.awt.Frame _replaceOwnerFrame,
  java.net.URL _codebase, org.colos.ejs.library.
    LauncherApplet _anApplet, String[] _args,
     boolean _allowAutoplay) {
  __theArguments = _args;
```

```
  __theApplet = _anApplet;
  java.text.NumberFormat _Ejs_format =
                    java.text.NumberFormat.getInstance();
  if (_Ejs_format instanceof java.text.DecimalFormat) {
    ((java.text.DecimalFormat) _Ejs_format).
      getDecimalFormatSymbols().setDecimalSeparator('.');
  }
  _simulation = new FullyConnectedBlobSimulation
(this,_replaceName,_replaceOwnerFrame,_codebase,
                                          _allowAutoplay);
  _view = (FullyConnectedBlobView) _simulation.getView();
  _simulation.processArguments(_args);
}

public org.colos.ejs.library.View getView ()
                                       { return _view; }

public org.colos.ejs.library.Simulation getSimulation()
 { return _simulation; }

public void _resetSolvers() {
  _external.resetIC();
}
// --- Implementation of ExternalClient ---

public String _externalInitCommand
                              (String _applicationFile) {
  StringBuffer _external_initCommand=new StringBuffer();
  return _external_initCommand.toString();
}

public synchronized void _externalSetValues
(boolean _any, org.colos.ejs.library.external.
   ExternalApp _application) {
}

public synchronized void _externalGetValues
(boolean _any, org.colos.ejs.library.external.
   ExternalApp _application) {
}

// --- End of implementation of ExternalClient ---
```

```
// -----------------------------------------
// Variables defined by the user
// -----------------------------------------

 public double L  = 2.0;  // Variables.variables5Eq5Unkn:1
 public double tu  = 1.0; // Variables.variables5Eq5Unkn:2
 public double tw  = 1.0; // Variables.variables5Eq5Unkn:3
 public double n = 2.99995;//Variables.variables5Eq5Unkn:4
 public double E ;       // Variables.variables5Eq5Unkn:5
 public double m  = 6.299823999999999;
                         // Variables.variables5Eq5Unkn:6
 public double t  = 1.5; // Variables.variables5Eq5Unkn:7
 public double eps  = 0.7332239999999999;
                         // Variables.variables5Eq5Unkn:8
 public Object myColor  = null;
                         // Variables.variables5Eq5Unkn:9
 public boolean show2 = true;// Variables.showVariables:1
 public boolean show3a ; // Variables.showVariables:2
 public boolean show3b ; // Variables.showVariables:3
 public boolean show4a ; // Variables.showVariables:4
 public boolean show4b ; // Variables.showVariables:5
 public boolean show5a ; // Variables.showVariables:6
 public boolean show5b ; // Variables.showVariables:7
 public boolean bnd2a3a ; // Variables.showVariables:8
 public boolean bnd3a4a ; // Variables.showVariables:9
 public boolean bnd4a5a ; // Variables.showVariables:10
 public boolean bnd2b3b ; // Variables.showVariables:11
 public boolean bnd3b4b ; // Variables.showVariables:12
 public boolean bnd4b5b ; // Variables.showVariables:13
 public boolean in3a ; // Variables.Var Table:1
 public boolean out3a ; // Variables.Var Table:2
 public boolean in3b ; // Variables.Var Table:3
 public boolean out3b ; // Variables.Var Table:4
 public boolean in4a ; // Variables.Var Table:5
 public boolean out4a ; // Variables.Var Table:6
 public boolean in4b ; // Variables.Var Table:7
 public boolean out4b ; // Variables.Var Table:8
 public boolean in5a ; // Variables.Var Table:9
 public boolean out5a ; // Variables.Var Table:10
 public boolean in5b ; // Variables.Var Table:11
 public boolean out5b ; // Variables.Var Table:12

// -----------------------------------------
```

```
// Methods defined by the user
// ----------------------------------------

// --- Initialization

 public void _initialization1 () {
    // > Initialization.IntializationForT
 }  // > Initialization.IntializationForT

// --- Evolution

// --- Constraints

 public void _constraints1 () {
   // > Fixed relations.number of atoms n
  int i = (int)n;// > Fixed relations.number of atoms n:1
   switch (i)  // > Fixed relations.number of atoms n:2
   {  // > Fixed relations.number of atoms n:3
     case 2:  // > Fixed relations.number of atoms n:4
    {show3a = false; show3b = false;
              // > Fixed relations.number of atoms n:5
     show4a = false; show4b=false;
              // > Fixed relations.number of atoms n:6
     show5a = false; show5b = false;
              // > Fixed relations.number of atoms n:7
     bnd2a3a = false; bnd3a4a = false; bnd4a5a = false;
              // > Fixed relations.number of atoms n:8
     bnd2b3b = false; bnd3b4b = false; bnd4b5b= false;};
              // > Fixed relations.number of atoms n:9
     break;   // > Fixed relations.number of atoms n:10
     case 3:  // > Fixed relations.number of atoms n:11
     { show3a = true; bnd2a3a = true;
              // > Fixed relations.number of atoms n:12
      show3b = false;
              // > Fixed relations.number of atoms n:13
      show5a = false; show5b = false;
              // > Fixed relations.number of atoms n:14
      show4a = false; show4b = false;
              // > Fixed relations.number of atoms n:15
              // > Fixed relations.number of atoms n:16
      bnd3a4a = false; bnd4a5a = false;
              // > Fixed relations.number of atoms n:17
      bnd2b3b = false; bnd3b4b = false; bnd4b5b= false;};
```

```
          // > Fixed relations.number of atoms n:18
 break;  // > Fixed relations.number of atoms n:19
          // > Fixed relations.number of atoms n:20
case 4:  // > Fixed relations.number of atoms n:21
{show3b = true; bnd2b3b = true;
          // > Fixed relations.number of atoms n:22
 show3a = true; bnd2a3a = true;
          // > Fixed relations.number of atoms n:23
 show4a = false; show5a = false;
          // > Fixed relations.number of atoms n:24
 show4b = false; show5b = false;
          // > Fixed relations.number of atoms n:25
 bnd3a4a = false; bnd4a5a = false; bnd3b4b = false;
 bnd4b5b= false;};
          // > Fixed relations.number of atoms n:26
 break;  // > Fixed relations.number of atoms n:27
          // > Fixed relations.number of atoms n:28
case 5:  // > Fixed relations.number of atoms n:29
{ show4a = true; bnd3a4a = true;
          // > Fixed relations.number of atoms n:30
 show3b = true; bnd2b3b = true;
          // > Fixed relations.number of atoms n:31
 show3a = true; bnd2a3a = true;
          // > Fixed relations.number of atoms n:32
 show4b = false; show5b = false;
          // > Fixed relations.number of atoms n:33
 bnd4a5a = false; bnd3b4b = false; bnd4b5b= false;};
          // > Fixed relations.number of atoms n:34
 break;  // > Fixed relations.number of atoms n:35
          // > Fixed relations.number of atoms n:36
case 6:  // > Fixed relations.number of atoms n:37
{ show4b = true; bnd3b4b = true;
          // > Fixed relations.number of atoms n:38
 show4a = true; bnd3a4a = true;
          // > Fixed relations.number of atoms n:39
 show3b = true; bnd2b3b = true;
          // > Fixed relations.number of atoms n:40
 show3a = true; bnd2a3a = true;
          // > Fixed relations.number of atoms n:41
 show5a = false; show5b = false;
          // > Fixed relations.number of atoms n:42
 bnd4a5a = false; bnd4b5b = false;};
          // > Fixed relations.number of atoms n:43
```

```
    break;   // > Fixed relations.number of atoms n:44
      // > Fixed relations.number of atoms n:45
    case 7:  // > Fixed relations.number of atoms n:46
    { show5a = true; bnd4a5a = true;
             // > Fixed relations.number of atoms n:47
     show4b = true; bnd3b4b = true;
             // > Fixed relations.number of atoms n:48
     show4a = true; bnd3a4a = true;
             // > Fixed relations.number of atoms n:49
     show3b = true; bnd2b3b = true;
             // > Fixed relations.number of atoms n:50
     show3a = true; bnd2a3a = true;
             // > Fixed relations.number of atoms n:51
     show5b = false; bnd4b5b = false;
             // > Fixed relations.number of atoms n:52
     in5b = false; out5b = false; };
             // > Fixed relations.number of atoms n:53
     break;  // > Fixed relations.number of atoms n:54
             // > Fixed relations.number of atoms n:55
    case 8:  // > Fixed relations.number of atoms n:56
    { show3a = true; show4a = true; show5a = true;
             // > Fixed relations.number of atoms n:57
     show3b = true; show4b = true; show5b = true;
             // > Fixed relations.number of atoms n:58
     bnd4b5b = true; bnd2a3a = true; bnd3a4a = true;
             // > Fixed relations.number of atoms n:59
     bnd4a5a = true; bnd2b3b = true; bnd3b4b = true;
     bnd4b5b = true;};
             // > Fixed relations.number of atoms n:60
     break;  // > Fixed relations.number of atoms n:61
             // > Fixed relations.number of atoms n:62
    }  // > Fixed relations.number of atoms n:63
}  // > Fixed relations.number of atoms n

public void _constraints2 () { // > Fixed relations.
                                          L parameter
  int j = (int)L;  // > Fixed relations.L parameter:1
  {  // > Fixed relations.L parameter:2
   if (L>n) L = n;  // > Fixed relations.L parameter:3
  switch (j)  // > Fixed relations.L parameter:4
  {  // > Fixed relations.L parameter:5
  case 2:  // > Fixed relations.L parameter:6
     {  // > Fixed relations.L parameter:7
```

```
   in3a = false; in4a = false; in5a = false;
   in3b = false; in4b = false; in5b = false;
      // > Fixed relations.L parameter:8
   }; break;  // > Fixed relations.L parameter:9
     // > Fixed relations.L parameter:10
case 3:  // > Fixed relations.L parameter:11
  {  // > Fixed relations.L parameter:12
   in3a = true; in4a = false; in5a = false;
   in3b = false; in4b = false; in5b = false;
     // > Fixed relations.L parameter:13
  }; break;  // > Fixed relations.L parameter:14
case 4:  // > Fixed relations.L parameter:15
   {  // > Fixed relations.L parameter:16
   in3a = true; in4a = false; in5a = false;
   in3b = true; in4b = false; in5b = false;
      // > Fixed relations.L parameter:17
   }; break;  // > Fixed relations.L parameter:18
case 5:   // > Fixed relations.L parameter:19
   {  // > Fixed relations.L parameter:20
   in3a = true; in3b = true; in4a = true;
   in4b = false; in5a = false; in5b = false;
      // > Fixed relations.L parameter:21
   }; break;  // > Fixed relations.L parameter:22
case 6:  // > Fixed relations.L parameter:23
   {  // > Fixed relations.L parameter:24
   in3a = true; in3b = true; in4a = true;
   in4b = true; in5a = false; in5b = false;
      // > Fixed relations.L parameter:25
   }; break;  // > Fixed relations.L parameter:26
case 7:  // > Fixed relations.L parameter:27
   {  // > Fixed relations.L parameter:28
   in3a = true; in3b = true; in4a = true;
   in4b = true; in5a = true; in5b = false;
      // > Fixed relations.L parameter:29
   }; break;  // > Fixed relations.L parameter:30
     // > Fixed relations.L parameter:31
    // > Fixed relations.L parameter:32
case 8:  // > Fixed relations.L parameter:33
   {  // > Fixed relations.L parameter:34
   in3a = true; in3b = true; in4a = true;
   in4b = true; in5a = true; in5b = true;
      // > Fixed relations.L parameter:35
   }; break;  // > Fixed relations.L parameter:36
```

```
  }  // > Fixed relations.L parameter:37
  }  // > Fixed relations.L parameter:38
}  // > Fixed relations.L parameter

public void _constraints3 () {
   // > Fixed relations.m parameter
  int M = (int)m;  // > Fixed relations.m parameter:1
  {  // > Fixed relations.m parameter:2
   if (m>n) m = n; // > Fixed relations.m parameter:3
  switch (M)  // > Fixed relations.m parameter:4
  {  // > Fixed relations.m parameter:5
  case 2:  // > Fixed relations.m parameter:6
    {  // > Fixed relations.m parameter:7
     in3a = false; in4a = false; in5a = false;
     in3b = false; in4b = false; in5b = false;
       // > Fixed relations.m parameter:8
     out3a = false; out4a = false; out5a = false;
     out3b = false; out4b = false; out5b = false;
       // > Fixed relations.m parameter:9
     }; break;  // > Fixed relations.m parameter:10
      // > Fixed relations.m parameter:11
  case 3:  // > Fixed relations.m parameter:12
    {   // > Fixed relations.m parameter:13
     in3a = true; in4a = false; in5a = false;
     in3b = false; in4b = false; in5b = false;
        // > Fixed relations.m parameter:14
     out3a = true; out3b = false; out4a = false;
     out4b = false; out5a = false; out5b = false;
        // > Fixed relations.m parameter:15
    }; break;  // > Fixed relations.m parameter:16
  case 4:  // > Fixed relations.m parameter:17
     {  // > Fixed relations.m parameter:18
     in3a = true; in4a = false; in5a = false;
     in3b = true; in4b = false; in5b = false;
        // > Fixed relations.m parameter:19
     out3a = true; out3b = true; out4a = false;
     out4b = false; out5a = false; out5b = false;
        // > Fixed relations.m parameter:20
     }; break;  // > Fixed relations.m parameter:21
  case 5:   // > Fixed relations.m parameter:22
     {  // > Fixed relations.m parameter:23
     in3a = true; in4a = true; in5a = false;
     in3b = true; in4b = false; in5b = false;
```

```
        // > Fixed relations.m parameter:24
      out3a = true; out3b = true; out4a = true;
      out4b = false; out5a = false; out5b = false;
        // > Fixed relations.m parameter:25
      }; break;  // > Fixed relations.m parameter:26
    case 6:  // > Fixed relations.m parameter:27
      {   // > Fixed relations.m parameter:28
      in3a = true; in4a = true; in5a = false;
      in3b = true; in4b = true; in5b = false;
          // > Fixed relations.m parameter:29
      out3a = true; out3b = true; out4a = true;
      out4b = true; out5a = false; out5b = false;
          // > Fixed relations.m parameter:30
      }; break;  // > Fixed relations.m parameter:31
    case 7:  // > Fixed relations.m parameter:32
      {  // > Fixed relations.m parameter:33
      in3a = true; in4a = true; in5a = true;
      in3b = true; in4b = true; in5b = false;
        // > Fixed relations.m parameter:34
      out3a = true; out4a = true; out5a = true;
      out3b = true; out4b = true; out5b = false;
        // > Fixed relations.m parameter:35
      }; break;  // > Fixed relations.m parameter:36
        // > Fixed relations.m parameter:37
    case 8:  // > Fixed relations.m parameter:38
      {   // > Fixed relations.m parameter:39
      in3a = true; in4a = true; in5a = true;
      in3b = true; in4b = true; in5b = true;
          // > Fixed relations.m parameter:40
      out3a = true; out3b = true; out4a = true;
      out4b = true; out5a = true; out5b = true;
          // > Fixed relations.m parameter:41
      }; break;  // > Fixed relations.m parameter:42
    }  // > Fixed relations.m parameter:43
    }  // > Fixed relations.m parameter:44
  }  // > Fixed relations.m parameter

  public void _constraints4 () { // > Fixed relations.eps
                                                 parameter
    int e = (int)eps;  // > Fixed relations.eps parameter:1
    {  // > Fixed relations.eps parameter:2
    if (e == -2.0) myColor = Color.black;
      // > Fixed relations.eps parameter:3
```

```
    if (e == -1.0) myColor = Color.orange;
       // > Fixed relations.eps parameter:4
    if (e == 0.0) myColor = Color.yellow;
       // > Fixed relations.eps parameter:5
    if (e == 1.0) myColor = Color.red;
       // > Fixed relations.eps parameter:6
    if (e == 2.0) myColor = Color.white;
       // > Fixed relations.eps parameter:7
    }  // > Fixed relations.eps parameter:8
  }  // > Fixed relations.eps parameter

// --- Custom

// --- Methods for view elements

  public void _method_for_twSlider_dragaction () {
    _simulation.disableLoop();
       // Make the simulation thread not to step the model
    _getArguments();
    _simulation.enableLoop();
       // Make the simulation thread not to step the model
  }
  public void _method_for_tuSlider_dragaction () {
    _simulation.disableLoop();
       // Make the simulation thread not to step the model
    _getArguments();
    _simulation.enableLoop();
       // Make the simulation thread not to step the model
  }
  public double _method_for_in3a_lineWidth ()
                                              { return 3*tw; }
  public double _method_for_out3a_lineWidth ()
                                              { return 3*tu; }
  public double _method_for_in3b_lineWidth ()
                                              { return 3*tw; }
  public double _method_for_out3b_lineWidth ()
                                              { return 3*tu; }
  public double _method_for_in2a2b_lineWidth ()
                                              { return 3*tw; }
  public double _method_for_out2a2b_lineWidth ()
                                              { return 3*tu; }
  public double _method_for_in4a_lineWidth ()
                                              { return 3*tw; }
```

```
 public double _method_for_in4b_lineWidth ()
                                              { return 3*tw; }
 public double _method_for_out4a_lineWidth ()
                                              { return 3*tu; }
 public double _method_for_out4b_lineWidth ()
                                              { return 3*tu; }
 public double _method_for_in5a_lineWidth ()
                                              { return 3*tw; }
 public double _method_for_out5a_lineWidth ()
                                              { return 3*tu; }
 public double _method_for_in5b_lineWidth ()
                                              { return 3*tw; }
 public double _method_for_out5b_lineWidth ()
                                              { return 3*tu; }
 public double _method_for_bond2aTo2b_lineWidth ()
                                               { return 3*t; }
 public double _method_for_bond2bTo3b_lineWidth ()
                                               { return 3*t; }
 public double _method_for_bond2aTo3a_lineWidth ()
                                               { return 3*t; }
 public double _method_for_bond3aTo4a_lineWidth ()
                                               { return 3*t; }
 public double _method_for_bond3bTo4b_lineWidth ()
                                               { return 3*t; }
 public double _method_for_bond4aTo5a_lineWidth ()
                                               { return 3*t; }
 public double _method_for_bond4bTo5b_lineWidth ()
                                               { return 3*t; }
// -----------------------------------------
// Implementation of interface Model
// -----------------------------------------

 public synchronized void reset () {
   L = 2.0; // Variables.variables5Eq5Unkn:1
   tu = 1.0; // Variables.variables5Eq5Unkn:2
   tw = 1.0; // Variables.variables5Eq5Unkn:3
   n = 2.999956; // Variables.variables5Eq5Unkn:4
   m = 6.299823999999999; // Variables.variables5Eq5Unkn:6
   t = 1.5; // Variables.variables5Eq5Unkn:7
   eps = 0.7332239999999999;
                            // Variables.variables5Eq5Unkn:8
   show2 = true; // Variables.showVariables:1
 }
```

```
  public synchronized void initialize () {
    _initialization1 ();
    _resetSolvers();
  }

  public synchronized void step () {
  }

  public synchronized void update () {
    _constraints1 ();
    _constraints2 ();
    _constraints3 ();
    _constraints4 ();
  }

  public void _freeMemory () {
    System.gc(); // Free memory from unused old arrays
  }

} // End of class FullyConnectedBlobModel
```

APPENDIX C

MATHEMATICA CODE: 10-EQUATION-10-UNKNOWNS CASE

Mathematica Code for Transmission Coefficent for the General Case 10 Equations 10 Unknowns through a blob of any number of sites n>0:

\\Define Identity Matrix Z of size n:

$Z = \text{IdentityMatrix}[n];$

\\Define or generate vector e of length n with all \\elements equal to 1:

$e = [1, \{i, n\}];$

\\Matrix J of size nxn, all elements equal to 1:

$J = \text{Outer}[\text{Times}, e, e];$

\\Define in terms of hopping energy t, site energy epsilon, \\energy of incoming electron E (ee):

$\text{Xi} = \epsilon - ee - t;$
$\text{Xj} = t;$
$\text{Xu} = -S\text{tu}^2;$
$\text{Xw} = -S\text{tw}^2;$

\\output coupling vector u

$u = \text{tu}e;$

\\input coupling vector w

$w = \text{tw}e;$

\\Define S in terms of egienvalues E (ee):

$S = \frac{-\text{ee}}{2} - I\frac{\sqrt{4-\text{ee}^2}}{2};$

\\Ansatz for Matrix M for blob:

$M = \text{Xi}Z + \text{Xj}J + \text{Xw}(\text{Outer}[\text{Times}, w, w]) + \text{Xu}(\text{Outer}[\text{Times}, u, u]);$

\\Ansatz for inverse of matrix M:

$L = \text{Yi}Z + \text{Yjj}J + \text{Yww}(\text{Outer}[\text{Times}, w, w]) + \text{Yuu}(\text{Outer}[\text{Times}, u, u]) +$
$\text{Yju}(\text{Outer}[\text{Times}, e, u]) + \text{Yuj}(\text{Outer}[\text{Times}, u, e]) + \text{Yjw}(\text{Outer}[\text{Times}, e, w]) +$
$\text{Ywj}(\text{Outer}[\text{Times}, w, e]) + \text{Ywu}(\text{Outer}[\text{Times}, w, u]) + \text{Yuw}(\text{Outer}[\text{Times}, u, w]);$

\\Products of transposes of vectors and vectors:

$e.w = w.e == m; w.w == l; u.u == k; e.u = u.e == p; e.e == n;$
$e.J = J.e == ne; J.J == nJ; u.w = w.u == h; J.Z = J;$

\\Form matrix R in terms of parameters;
\\solve matrix equation for Y's the 10 unknowns :

$\text{Solve}[R.B == A, \{\text{Yjj}, \text{Yww}, \text{Yuu}, \text{Yju}, \text{Yuj}, \text{Yjw}, \text{Ywj}, \text{Ywu}, \text{Yuw}\}];$

$A = \{-\text{Xj}/\text{Xi}, -\text{Xw}/\text{Xi}, -\text{Xu}/\text{Xi}, 0, 0, 0, 0, 0, 0\};$

$B = \{\text{Yjj}, \text{Yww}, \text{Yuu}, \text{Yju}, \text{Yuj}, \text{Yjw}, \text{Ywj}, \text{Ywu}, \text{Yuw}\};$

\\Matrix in terms parameters:

$R = \{\{(\text{Xi} + n\text{Xj}), 0, 0, 0, p\text{Xj}, 0, m\text{Xj}, 0, 0\},$
$\{0, (\text{Xi} + l\text{Xw}), 0, 0, 0, m\text{Xw}, 0, h\text{Xw}, 0\},$
$\{0, 0, (\text{Xi} + k\text{Xu}), p\text{Xu}, 0, 0, 0, 0, h\text{Xu}\},$
$\{0, 0, p\text{Xj}, (\text{Xi} + n\text{Xj}), 0, 0, 0, 0, m\text{Xj}\},$
$\{p\text{Xu}, 0, 0, 0, (\text{Xi} + k\text{Xu}), 0, h\text{Xu}, 0, 0\},$
$\{0, m\text{Xj}, 0, 0, 0, (\text{Xi} + n\text{Xj}), 0, p\text{Xj}, 0\},$
$\{0, h\text{Xu}, 0, 0, 0, p\text{Xu}, 0, (\text{Xi} + k\text{Xu}), 0\},$
$\{0, 0, h\text{Xw}, m\text{Xw}, 0, 0, 0, 0, (\text{Xi} + l\text{Xw})\}\};$

\\Transmission amplitude;
\\substitute solutions B in matrix L

$\text{tT} = 2IS^2 \frac{\sqrt{4-\text{ee}^2}}{2} \text{tu}(\text{Transpose}[w, \{1\}].L.u)\text{tw};$

\\Transmission Coefficient T:

$T = \text{Simplify}[\text{ComplexExpand}\left[\text{Abs}\left[\text{tT}^2\right]\right], \{\text{ee}, -2, 2\}];$