Two Statistical Physics Problems: Phase Diagram Calculation of Spatially Anisotropic, Surfaced d=3 Layered Systems by Renormalization-Group Theory and Vehicle-Route Optimization with Traffic Factors for Migros Home Delivery System by Simulated Annealing

by

Aykut Erbaş

A Thesis Submitted to the
Graduate School of Sciences and Engineering
in Partial Fulfillment of the Requirements for
the Degree of

Master of Science

in

Physics

Koç University

June, 2007

Koç University

Graduate School of Sciences and Engineering

This is to certify that I have examined this copy of a master's thesis by

Aykut Erbaş

and have found that it is complete and satisfactory in all respects,
and that any and all revisions required by the final
examining committee have been made.

Committee Members:

_____

Prof. A. Nihat Berker

_____

Assoc. Prof. Özgür Müstecaplıoğlu

_____

Assist. Prof. Alkan Kabakçıoğlu

Date: _____

To World and the Peace

ABSTRACT

In this thesis, two effective methods of statistical physics are applied to new frameworks. Hierarchical lattices which are solved exactly by renormalization-group theory and simulating annealing processes which minimize a complex function of any system are used in conceptually new and practical situations. Independent isotropic hierarchical lattices with d = 2 which are stacked along the z direction in order to obtain layered anisotropic hierarchical lattices with surfaces were constructed. These systems admit exact solutions as hierarchical models and constitute approximate solutions for uniaxially anisotropic d = 3 physical systems with bulk, surface, and possible anisotropy. The global phase diagrams, with layer-by-layer ordering, and layer energy densities of these magnetic models were obtained for different anisotropy parameters. A two-level simulated annealing process was developed to find the most cost-effective delivery scheme on a two-dimensional costumer array with multiple vehicles used in parallel. The minimization process is jointly achieved for dividing customers between vehicles and for optimizing the route of each vehicle, in a two-level hierarchy. Different traffic factors for city neighborhoods are included. Finally, as a fundamental current problem of statistical mechanics, quenched random fields which are equivalent to the traffic factors are added to this traveling salesman problem.

ÖZETÇE

Bu tez çalışmasında, istatistik fiziğin iki etkili yöntemi iki farklı problem üzerinde başarılı bir şekilde uygulanmıştır. Bu yöntemler sırasıyla, renormalizasyon-grup teorisi ile kesin çözülebilen hiyerarşik örgüler ve herhangi bir sisteme ait durum fonksiyonun minimum değerini bulunmasında kullanılan benzetilmiş tavlamadır. Birbirinden bağımsız 2 boyutlu sistemler z ekseni doğrultusunda istiflenerek katmerli yüzeyli anisotropik hiyerarşik örgüler oluşturulmuştur. Oluşturulan yeni örgü hiyerarşik bir model olarak kesin çözüme sahip olup, ayrıca boyutlu yüzey, hacim vs. gibi farklı anisotropilere sahip fiziksel sistemler için yaklaşık çözümler vermektedir. Bu manyetik modellerin değişik etkileşim parametreleri için, yüzeylerin teker teker faz geçişi verdiği faz diyagramları ve enerji yoğunlukları elde edilmiştir. İki aşamalı benzetilmiş tavlama yöntemi, müşteri verilerinin iki boyutlu serilerde tutulduğu ve birden fazla dağıtıcı araçın kullanıldığı bir sistem için en etkili ve verimli dağıtım mekanizmasının bulunması için kurulmuştur. Bu iyileştirme işlemi, önce müşterilerin araçlara en iyi şekilde dağılımı ardından her araç için en uygun dağıtım rotasının bulunması suretiyle yapılmıştır. Son olarak, bu satıcı adam problemine önemli bir istatistik fizik kavramı olan ve trafik etkilerine karşılık gelen donmuş rastgele alanlar da eklenmiştir.

ACKNOWLEDGMENTS

First, I would like to thank my supervisor Prof. A. Nihat Berker who has been a great source of inspiration for me with his patience and pioneering. The day I met Mr.Berker was a first order phase transition in my life. He is the one who made me give the decision of being a physicist when I first came to İ.T.Ü.

I also want to thank my friends in our research group and those of in room Sci-131.

I would like to thank those people who work in Migros Türk A. Ş. and Kangurum for their supports to our co-project with KÜMPEM.

I am grateful to members of my thesis committee for critical reading of this thesis and for their valuable comments.

Finally, I thank Mrs. Muzaffer Erim for her patience and support of accommodation in Kalamış

TABLE OF CONTENTS

LIST OF FIGURES

NOMENCLATURE

RG	Renormalization Group
MC	Monte Carlo
SM	Simulating Annealing
GPS	Global Positioning System
GA	Greedy Algorithm
FM	Ferromagnetic
PM	Paramagnetic
TSP	Traveling Salesman Problem

Chapter 1

INTRODUCTION

1.1 Phase Transition

There are many physical systems which undergo a phase transition from one state to another by a change in an intensive variable such as temperature, pressure, or magnetic field. The liquid-solid transition (freezing), the normal-to-superconducting transition in electrical conductors, the paramagnet-ferromagnet transition in magnetic materials, and the superfluid transition in liquid helium can be noted as some well-known examples of phase transitions. However, the mechanism of the transition can vary from one system to another, and to distinguish the characteristic of a phase transition, one should check that whether changes in the thermodynamic properties are sudden or not.

Phase transitions generally occur when the free energy of the system is non-analytic. Thereby, the classification of the transition can be determined by the degree of this analyticity. If the first derivative of the free energy is discontinuous then the transition is given the name *First-Order Phase Transition*. If the transition is *Second Order*, the discontinuity is exhibited in the second or higher derivatives. By employing the latent heat, distinction between two mechanism can be expressed more accurately. During the transition, if any amount of (latent) heat is absorbed or emitted by the system at the transition temperature (T_c), then the transition is discontinuous (first order). Two phases can coexist as in vapor-water transition because some parts of the system complete their transition while other parts do not yet. On the contrary, in continuous (second order) phase transition there is no latent heat. Two-phase coexistence cannot be observed. The paramagnet-ferromagnet transition of magnetic

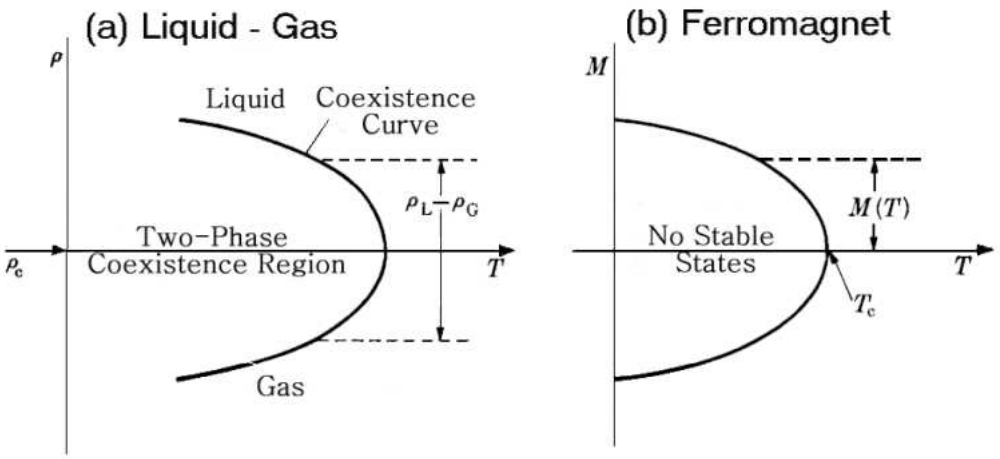

Figure 1.1: (a) Phase diagram for the liquid-gas transition where ρ is the order parameter. On right side, magnetization as the order parameter for a paramagnet-ferromagnet transition [3].

materials can be counted as an example for second order phase transition.

As mentioned above, for each system, there are unique critical values of its thermodynamic variables such as temperature, pressure or magnetic field, etc. at which system undergoes a phase transition. Combination of these critical values is labeled as the *Critical Point* in parameter space. There also needs be a quantity that is able to distinguish the system below and above the critical point. In many systems, this quantity, which is termed as *Order Parameter*, is the first derivative of the free energy. A jump in the order parameter is both a signature for a phase transition and a tool to determine the degree of transition. For instance, for a ferromagnetic system, order parameter is the magnetization which is the number of magnetic dipole moments per unit volume. While the magnetization is zero above the critical temperature (T_c), namely, in the paramagnetic phase, it begins to continuously deviate from zero and transforms to the ferromagnetic phase below T_c.

The studies on the second-order phase transitions have always been much more attractive to scientists than those on first order because of their complicated nature. The phenomenology of continuous phase transitions is also known as *critical phenomena*. In a continuous transition, the system becomes self similar as the tran-

sition point is approached. This means that no matter at which scale the system is looked into, one always observes the same structure like looking into a fractal. More precisely, the system has no characteristic length scale and is totally scale-invariant near the critical point. This characteristic implies that all thermodynamic functions are homogeneous and can be mathematically defined by power laws. Near the transition point, all the important information is carried by a set of critical exponents which hold all the singularities. For example, the specific heat can be indicated as $C \propto |T - T_c|^{-\alpha}$, or the magnetization diverges as $M \propto |T - T_c|^{-\beta}$ in the vicinity of T_c, etc.

Another interesting point about critical exponents is that they are universal. This means that different systems share the same critical exponents like the relationship between liquid-gas and paramagnet-ferromagnet transitions or that of planar magnets and two-dimensional Coumlomb plasmas. Although there are almost infinite number of systems obeying the critical phenomena, there are only a limited number of universality classes. Therefore, each system with totally different microscopic structures has to be in one of the universality classes. In addition, when critical points for different systems are calculated and drawn on dimensionless parameter space, they form a critical surface or curve which is also a phase boundary.

Also, as a short note on history, it can be said that researches on critical phenomena have started in 1940's. Although it has been focused for the last fifty years, many of the important concepts have been released in recent years. It had taken a long time that critical phenomena was accepted as a separate discipline which is worth the effort in physics.

1.1.1 A Statistical Physics Approach to Phase Transition

Statistical physics was born with the invention of the relation between probability and entropy by Ludwig Boltzmann in 1896. His famous formula,

$$S = k_B log(\Omega), \tag{1.1}$$

where Ω is the number of possible microscopic states corresponding to the same

macro state and k is the well-known Boltzmann constant($1.38036 \times 10^{-23} JK^{-1}$), gave a significant explanation to the distribution of energy in an N-particle system. We can illustrate this formula for N particles in M boxes. n_i is the number of particles at i th box. Then Ω can be obtained as

$$\Omega = \frac{N!}{n_1!n_2!..n_m!}. \tag{1.2}$$

Thus, once the entropy is obtained, one can easily find the free energy which plays an important role to determine the mechanism of phase transitions and other thermodynamic properties by subtracting the entropic enegy (TS) from the total energy (U).

$$F = U - TS \tag{1.3}$$

As we mentioned above, the thermodynamic variables or the degree of a phase transition can also be represented by derivatives of free energy. However, for many systems, (especially, systems of interacting particles) the definition of the number of possible microstates Ω is not as easy as in Eq. (1.2). So, there needs be a definition of the "Partition Function, Z", which comprises all important quantities of a system in global thermodynamic equilibrium in which macroscopic variables of the system do not change anymore. There are different types of partition functions for different systems. Such as, for a system at fixed temperature the "canonical partition function", or the "grand canonical partition function" for the systems which can exchange particles at fixed temperature and chemical potential, etc.

$$Z = \sum_i \Omega_i e^{-\beta E_i}, \tag{1.4}$$

in Eq. (1.4), E_i is the macroscopic energy of the ith state, Ω_i is the degeneracy of the E_i, β is the inverse temperature multiplied by Boltzmann constant, and the partition function is the sum over all possible states. We can define the free energy in terms of Z as

$$F = -k_B T log(Z) = -log(Z)/\beta. \tag{1.5}$$

The partition function is not a quantity different than that of given in Eq. (1.1). Both of them have similar meanings statistically. Eq. (1.2) is the number of possible configurations for N particles divided between M boxes while Z refers to all possible states in which any system can be. Therefore, the probability of a microstate with energy E_j can also be calculated as

$$P_i = \frac{e^{-\beta E_i}}{Z} \tag{1.6}$$

and the average energy,

$$\langle E \rangle = \frac{\sum_i E_i e^{-\beta E_i}}{Z} = -\frac{\partial ln Z}{\partial \beta}. \tag{1.7}$$

Many other thermodynamic quantities can be obtained from Z such as heat capacity C, susceptibility χ, etc.

One can automatically ask how the partition function exhibits singularities although it is a fully analytic function, which can be expanded in polynomial series. In 1952, Yang and Lee showed that in thermodynamic limit, Z begins to represent singularities for escaping from non-analyticity. As shown in Figure (1.2), the evolution of the phase transition can also be observed from the free energy - magnetization curve of a ferromagnetic system in which spins have two degree of freedom - up or down - for different temperature and magnetic field values. In Figure (1.2), above the critical temperature T_c, the magnetization M is zero, namely the system is in the paramagnetic phase. If $T = T_c$, free energy curve flattens. First and second derivatives with respect to magnetization on that point are zero. When T is greater than T_c, there are two roots of F. Physically speaking, spins can choose either up or down direction. This continuous transition from the paramagnetic phase to the ferromagnetic phase gives rise to a second-order phase transition. A first-order phase transitions can be observed when an external magnetic field is introduced. Thereby, spins tend to align with the direction of the magnetic field no matter what their previous configuration is. This change from up to down phase (or down to up) is a first-order phase transition as represented in Figure (1.2).

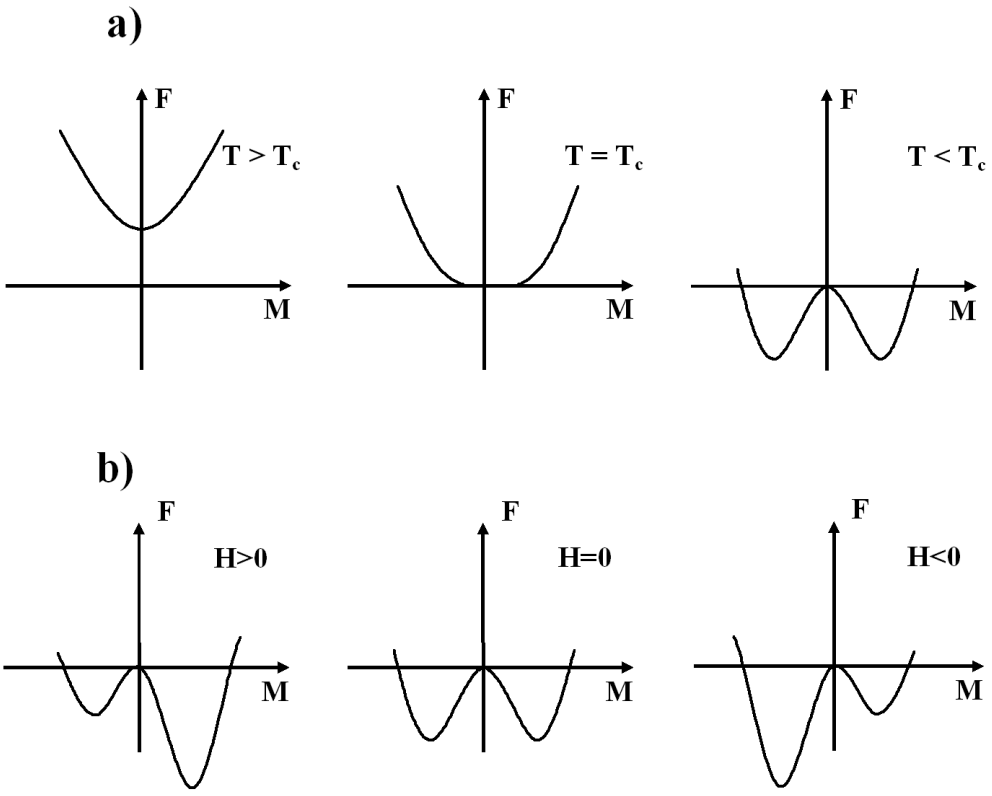

Figure 1.2: Evolution of the free energy of a system undergoing a second order phase transition, (a). First order phase transition is observed below T_c as external magnetic field deviates around zero, (b).

1.1.2 Simulated Annealing

Statistical mechanics deals with a great number of particles, and for a thermodynamic system of N particles, the total number of configurations is incredible. As discussed earlier, there may be an extremely large number of microscopic states corresponding to the same macroscopic state which is an observable reflection of measurable properties in equilibrium such as pressure, magnetization, etc. For instance, suppose that a room is divided into two equal parts. While some of the molecules can go to the one half, others can go to the other half. As a result, the number of possible configurations which N gas molecules can choose is 2^N. However, all these 2^N contributions represent the same macroscopic properties in the room. Note that this 2^N is nothing but Ω we have defined earlier.

In thermodynamic systems, when there is no particle-particle interaction, we manage to calculate averages and configurations analytically. However, when particle interactions are introduced, systems get more complicated. Thus, we need approximations both analytically and numerically to find an acceptable equilibrium state. One of the examples for the numerical case is the Metropolis algorithm, which checks in a thermal way the possible best (sometimes worst) configurations having a probability of e^{-E/k_BT} [4]. As represented in Eq. (1.6), the probability of finding a system in a particular energy is proportional to the Boltzmann factor e^{-E/k_BT}. This means that the higher the temperature the greater the energy of system. On the contrary, the lower the temperature the smaller the energy. Thus, if the temperature of the system is decreased in a right cooling schedule, the state with the lowest energy can be obtained. Suppose that we have two different energy states with probabilities P_i and P_j,

$$P_i = \frac{e^{-E_i/k_BT}}{Z},$$
$$P_j = \frac{e^{-E_j/k_BT}}{Z}.$$

The problem about these probabilities is that we do not know the normalization factor Z, which is almost impossible to calculate. However, the relative probability between these two energy states can be introduced as

$$\frac{P_i}{P_j} = \frac{e^{-E_i/k_BT}}{e^{-E_j/k_BT}} = e^{-(E_i-E_j)/k_BT} = e^{-\Delta E/k_BT}. \tag{1.8}$$

Then we do not need Z anymore. We can compute the most probable energy obeying equilibrium statistical mechanics for any given temperature by searching the space of possible configurations of system. To flow in this space, small changes are introduced one after another, and each point on this flow is checked according to the the rules given below.

Suppose our state has an energy of E_i initially, and after making a small change on state, its energy is E_j,

- If $E_i > E_j$, then replace the new system with the old one, and set E_i=E_j
- If $E_i < E_j$, do not throw up this configuration quickly, hold it. If $e^{-(E_i-E_j)/k_BT} > p$ where p is the any normalized random number, then accept this new system and set E_i=E_j.

The second rule is the most important step of the algorithm. Although accepting the higher energies may seem like an error, this helps the system climbing up local minimums and brings to the global minimum energy.

In 1983, Kirkpatrick, *et al.* applied this process to optimize functions with a new numerical approach named *Simulating Annealing* (*SA*) [5]. Term of "annealing" was inspired from the real annealing processes of semiconductors, metals, and glasses to form perfect crystals. These substances carry many structural defects, and one way of removing these defects is raising the temperature first and then cooling it so slowly that at each temperature step, particles are allowed to sit on a stable state. As the temperature decreases with small steps to zero, particles find the best configuration for themselves at each temperature. Finally, when the system is completely cooled, imperfections in structure are removed. Note that, when the system is hot, it is not in a real stable state because the higher kinetic energy of the system, the higher probability of the system is to escape from stable state.

Optimization of the functions follows the same way we mentioned above. However, this time our system is our function to optimize, and the temperature has no

physical interpretation other than being a parameter. We have to create different configurations by perturbations which, in a real physical system, correspond to the change of configuration by mobile particles. To find an optimized function or a value, at each step we have to give small perturbations to the system and then we have to check the new energy according to rules of the Metropolis algorithm until the temperature decreases to zero. Here, when we say "step", it does not mean the temperature step, it means "proper number" of small changes given to system at any temperature step. In doing so, chance of finding the best configuration is increased . Finally, one can ask what the starting temperature and proper number are. These parameters vary from problem to problem. In this study, there is going to be given a particular value for the route optimization problem in *Chapter* 3.

1.2 Renormalization-Group Theory

Renormalization group (RG) analysis is an efficient mathematical tool that allows one to find the macroscopic behavior of a system and also the mechanism of the phase transition [14]. It was first realized by Kenneth Wilson, and his success was awarded by Nobel Prize in 1982. The birth of the idea of recursive interaction coefficients goes back to 1960's. Kadanoff's "block spin" approach can be counted as a pioneer for RG type of calculations [9].

As we mentioned earlier, at the phase transition point, the system exhibits singularities and has no characteristic length scale. In saying so, degrees of freedom are correlated with any degrees of freedom no matter what the distances between them are because correlation length (ξ) goes to infinity. If we focus on a particular example such as a system composed of Ising-1/2 spins, we can say that at the critical point, each spin is correlated with all other spins in the system. Near the critical point, long-range correlations are as important as the short-range correlations. So, if one wants to investigate the mechanism of phase transition and critical phenomena, long-range correlations are also considered. However, for a system of N spins, calculation of the long-range interactions is very difficult.

An infinite ξ at critical point is a necessity for presence of the singularities observed

in the thermodynamic functions. Suppose that the magnetic susceptibility χ for an Ising system is

$$\chi = \sum_j (\langle s_0 s_j \rangle - \langle s_0 \rangle \langle s_j \rangle) = \sum_j \Gamma(r_0 - r_j),$$

where Γ is the correlation function of the system. As seen explicitly from the above equation, at the critical point, it is impossible that χ diverges unless the j sum is non-negligible over all the system instead of the nearest neighbors under the limit of infinite N.

RG method is based on the idea that long-range effective interactions can replace short range interactions [2]. If one can obtain the long-range interaction coefficients, then the critical properties of the system can be determined. The RG method achieves this by summing out the short-range couplings to obtain the same system but with new interactions. This coarse graining process continues until one known characteristic interaction value (as at the critical point) is reached. As shown in Eq. (1.9), we can define the same system by dividing the summation into two sub summations. After performing only one of the sums, what leaves behind is a "primed" system with new coupling coefficients (β' and H' of the new system).

$$\sum_s e^{-\beta H(\{s\})} = \sum_\sigma (\sum_s e^{-\beta H(\{\sigma\},\{s\})}) = \sum_\sigma e^{-\beta' H'(\{\sigma\})} \tag{1.9}$$

We can illustrate the RG method with the one-dimensional Ising model, which undergoes a zero-temperature phase transition. Its well-known Hamiltonian is

$$-\beta H = \sum_{\langle ij \rangle} e^{J s_i s_j}, \tag{1.10}$$

where J is the interaction coefficient between two nearest-neighbor spins and each spin can take value of $+1$ or -1. We start to apply RG by eliminating s spins with a length rescaling factor of b=2. As a result, we have a new σ spin system with new renormalized J' interaction.

When we start from an arbitrary value of J, namely an arbitrary point on our phase space (for 1-d Ising model it is only the J axis), the J value flows on phase space by RG recursion relations. These iteration flows are indicated by arrows in Figure

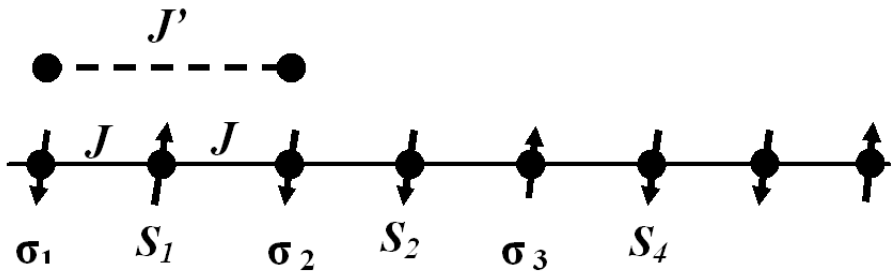

Figure 1.3: One-dimensional Ising spin chain. The coupling between two nearest neighbor spins is J. After RG procedure by aggregation of s spins, new interaction coefficient of the decimated system is J'.

(1.4). After many iterations, flows stay unaltered on special characteristic points which are called *Fixed Points*. Generally, parameter spaces are divided into different phases, and by following flows, all phases and phase boundaries can be identified easily. Therefore, RG flows play an essential role to determine the thermodynamic phases. There are some fixed points which does not characterize any phase, however, which are needed to complete the RG mechanism:

1. Stable fixed point: When we start at any nearby point on parameter space, flows are toward this point and eventually stays there. We can say that they are attractive fixed points. Physically speaking, approaching these points is like looking at the system from a larger and larger scale. On fixed points, the system is ordered or disordered (in a phase).

2. Unstable fixed point: It is impossible for a flow to reach these points no matter the number of iterations is. Even if for a flow manages to approach to these points, eventually turns away.

3. Generic class of fixed point: These are neither stable nor unstable fixed points and are directly related to a phase transition. These points are located on the critical boundary (or surface), and any flow starting on the critical boundary stays on it until it reaches the fixed point of the boundary. This boundary (or surface) separates two or more phases as shown in Figure (1.4).

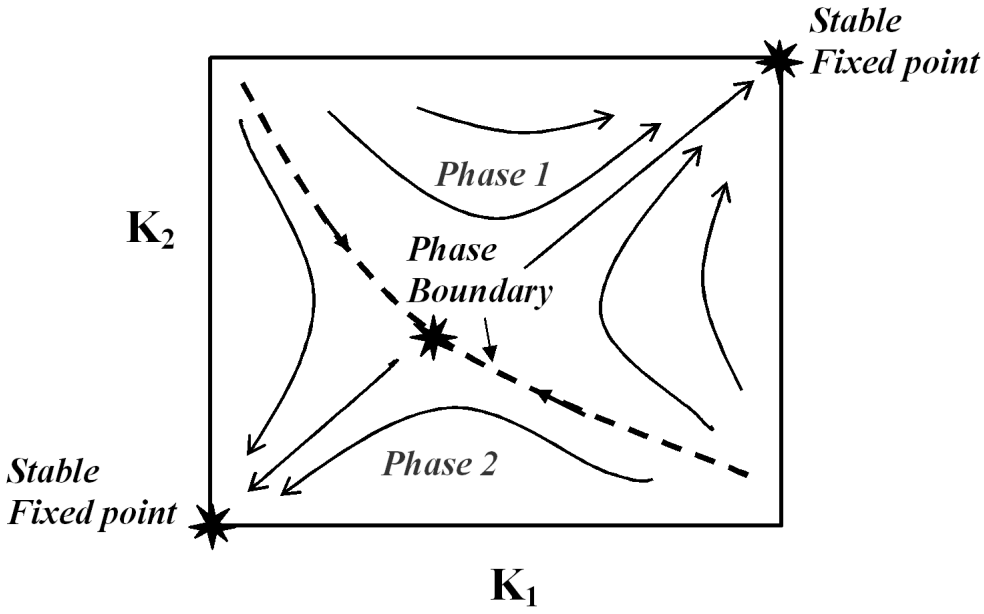

Figure 1.4: RG flows over an arbitrary parameter space. K_1 and K_2 are the parameters of the microscopic Hamiltonian. The critical curve is the border between two different phases, and the stars indicate the fixed points.

1.3 Hierarchical Lattices

In theoretical physics, exactly soluble models are very important tools to check the validity of more complex approximations and theories even if they do not refer to real physical models. For example, although 2-dimensional Ising model has no physical counterparts, it has opened the door for a wide variety of new theories [10].

In this study, we are going to introduce *layered hierarchical lattices* which can be solved exactly by RG procedure. Hierarchical lattices were first introduced by Berker and Ostlund in 1979 [6, 7]. Because the construction of hierarchical lattices is the opposite of direction of RG analysis, each hierarchical lattice has its own unique recursion relations yielding exact solutions for thermodynamic properties. Hierarchical lattices also exhibit phase transitions, and thus they are efficient tools to obtain the critical behaviors, fixed points and all critical exponents of a system.

Basically, hierarchical lattices can be considered as the extensions of long range interactions which are actions of short range interactions as mentioned in the previous

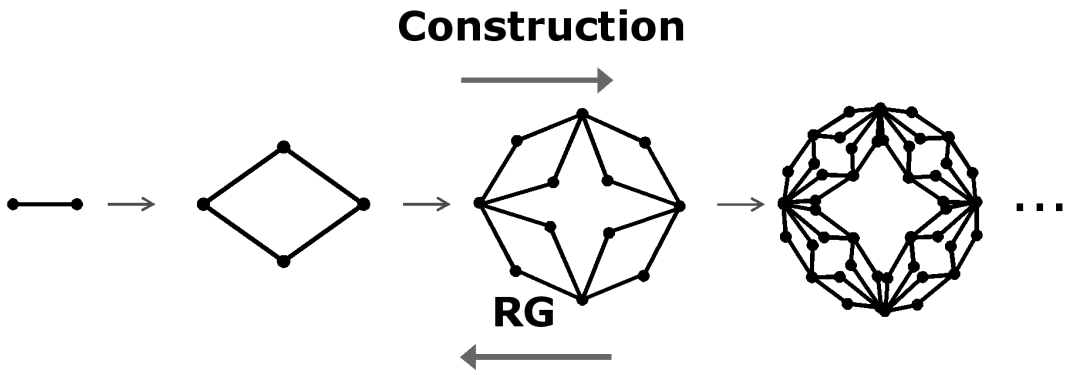

Figure 1.5: Diamond hierarchical lattice. At each step one diamond graph is replaced with a single bond many times to obtain an infinite lattice. Also, the construction direction and the RG direction are opposite of each other.

section. (This is also why this name was chosen for these lattices.) Suppose that we have a bond representing the interaction between two particles at two separate edges of the system. This long range interaction includes many shorter range interactions. To start constituting the hierarchical lattice, we replace this bond with a diamond (or baklava) shaped lattice shown in Figure (1.5) which is our unit structure for this model. Now, the long-range couplings at the beginning are defined by four bonds, namely by less longer couplings. If each bond is replaced with these diamonds one more time, then a more expanded scheme of the initial bond is obtained (lattice on the right side of Figure (1.5)). If this embedding process is repeated infinitely, consequently, an infinite lattice corresponding to a two-dimensional Ising lattice can be obtained. The essential point is that this infinite lattice is exactly soluble.

When a hierarchical lattice is introduced, the value which determines number of iterated bonds is the volume rescaling factor b^d. Here, b is the length rescaling factor, namely the shortest distance between two spins on the edges of the unit structure in terms of number of bonds, and d is dimension (b^d also indicates the number of decimated bonds at each RG step). For our example, b is 2, and the total number of bonds in the unit structure is 4. Therefore, automatically $d = 2$. Of course, for different lattices representing different models, these values change according to the structure of the lattice.

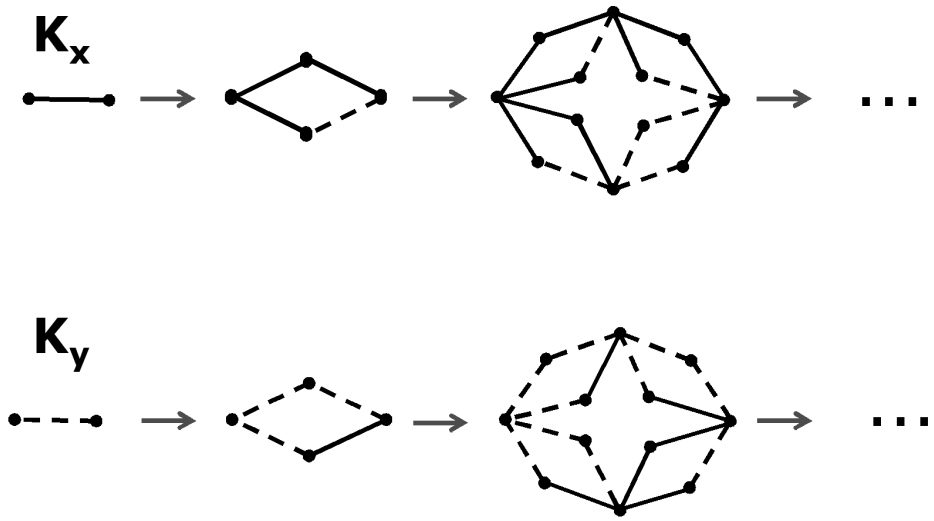

Figure 1.6: An example to the anisotropic hierarchical lattices. K_x and K_y are interactions along x and y directions respectively. The hierarchical lattice is constituted by parallel and mutual embeddings of these graphs repeatedly.

1.3.1 Anisotropic Hierarchical Lattices

All bonds composing the hierarchical lattices are connected to each other due to their iterative natures. This feature gives them a powerful ability to define even more complicated systems in which anisotropy is established along different directions. To construct anisotropic hierarchical lattices, different types of unit lattices must be developed and embedded mutually in a parallel manner[11]. Thus, one kind of interaction include another kind of interaction, and they iteratively form an infinite lattice. When calculating the volume rescaling factor b^d of the lattice, the ratio of the total number of bonds in between two hierarchic steps is taken into account. Note that, when counting the bonds, non-iterative bonds should not be counted.

There are two important requirements to be followed for the process of constitution of any anisotropic hierarchical lattice.

1. When all the interactions are equivalent to each other, the hierarchical lattice should become an isotropic model whose interactions do not depend on the direction. In Figure (1.6), when K_x is equal to K_y , the lattice turns into a 2-dimensional diamond lattice represented in Figure (1.5).

2. When one or more interactions are set to zero, the hierarchical lattice should obey the proper reductions to the lower dimensions. As illustrated in Figure (1.6), when either K_y or K_x is made zero, the hierarchical lattice reduces to a one-dimensional model.

Chapter 2

SPATIALLY ANISOTROPIC D=3 ISING LAYERED SYSTEMS

Hierarchical lattices which give exact solutions with renormalization-group theory can be applied to low symmetry problems such as magnets, surface systems, superconductivity, networks and the like [12, 13]. In this study, anisotropic hierarchical lattices which were successfully applied to the anisotropic hierarchical models are employed to construct finite-thickness surface systems. Anisotropic hierarchical lattices are constructed by the parallel, mutual embedding of each graph like isotropic hierarchical lattices but interactions along each direction are different in anisotropic hierarchical lattices. At each step one bond is replaced by b^d bonds where b is the length rescaling factor. The final system obtained from this process corresponds to an infinite lattice. For our model, anisotropy is introduced among the layers, the interaction strength on each layer is different but can be related by a parameter or a function. Hereby, more realistic models in which the anisotropy may depend on the distance from bulk, the features of the substance on each layer, etc. can be modeled.

2.1 Fully Anisotropic Model and Related Phase Diagrams

In our hierarchical models, each spin interacts with its nearest- neighbor spins on the same surface and either upper or down, or both surfaces in $d = 3$ as indicated in Figure (2.1). Our interaction coefficients for each layer are chosen as J_1, J_2 .. J_n , where n is the number of layers and there is a proper relation between these interactions. For our model, the number of layers is selected as three and the coupling are defined as $J_2 = \alpha J_1$ and $J_3 = \alpha^2 J_1$ where $0 < \alpha < 1$. When α is equal to one, it turns into an isotropic model. The layer-layer interaction between spins was defined as K (coupling of two spin on neighboring layers). When the number of layers goes

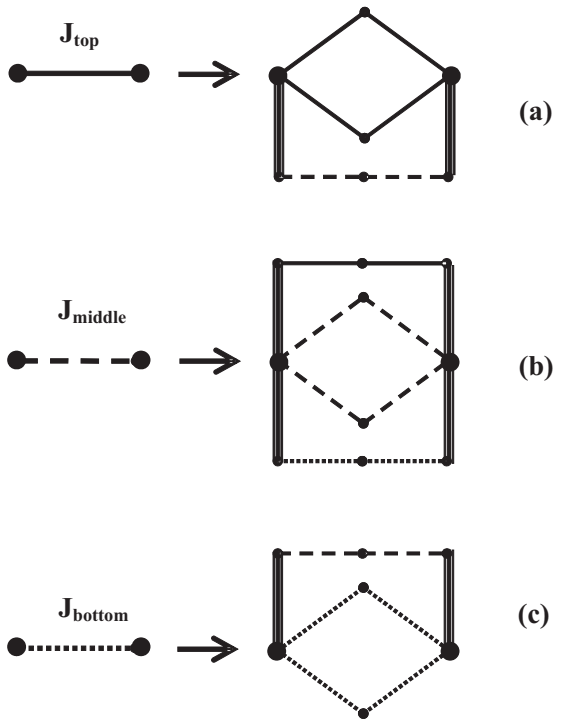

Figure 2.1: Hierarchical models: (a) upper layer, (c) bottom layer (inverse one is upper layer), (b) bulk layers. Doubled lines refer to K interactions while thin ones are J interactions. Solid and dashed lines refer to different layers.

to infinity and all J_i and K_i coefficients are set equal to each other, the system can turn into a $d = 3$ Ising model. If all the K_i couplings go to zero, the system behaves like a sum of many independent two-dimensional systems. Because of the hierarchy, construction of the models and renormalization rescaling are in the opposite directions as mentioned above.

Our Hamiltonian for all models is

$$-\beta H = \sum_k J_k \sum_{\langle ij \rangle} s_{ik} s_{jk}, + \sum_k K_k \sum_i s_{ik} s_{ik+1}, \tag{2.1}$$

where k index defines the layer number.

The lattices shown in Figure (2.1) retain their form under the renormalization group transformations. The term s_{ik} defined in Eq. (2.1) is an Ising spin at the site i of layer k with all possible nearest-neighbor interactions and $s_{ik} = \pm 1$. The most general form of the recursion relations used in our calculations is

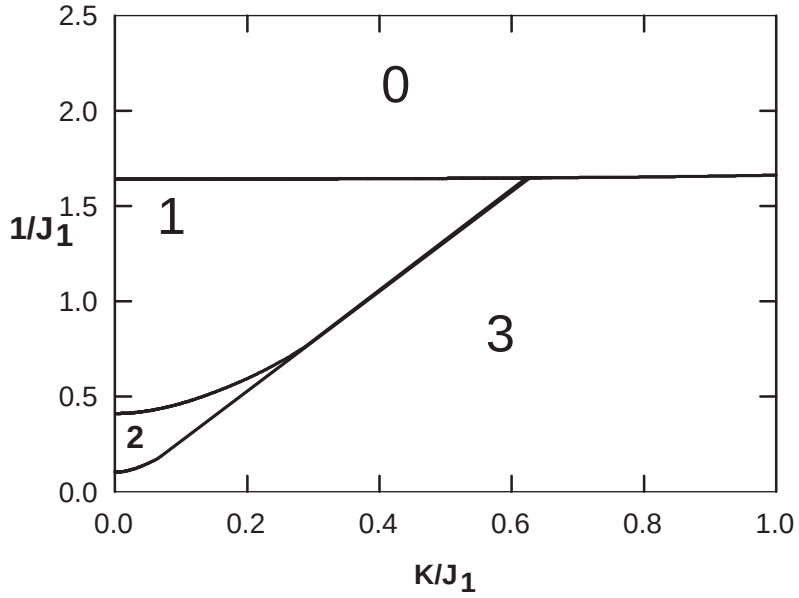

Figure 2.2: Phase diagram of 3-layer system in temperature versus layer-layer interaction. $J_2 = \alpha J_1$ and $J_3 = \alpha^2 J_1$ for $\alpha = 0.25$. The numbers indicate how many of the layers are ordered; 0 and 3 refer to ferromagnetic and paramagnetic phases respectively.

$$J_i' = log(\frac{R_i(+,+)}{R_i(+,-)}), \quad G_i' = b^d G_i + log(R_i(+,+)R_i(-,-)), \tag{2.2}$$

where R_i functions can be obtained from the renormalization-group transformations for any combination of two external spins between s_1 and s_3 in Eq. (2.3). $(+)$ is defined as spin-up and $(-)$ is defined as spin-down.

$$R_i(s_1, s_3) = e^{-\beta' H'(s_1,s_3)} = \sum_{s_3,\ldots} e^{-\beta H(s_1,s_2,\ldots s_3)}. \tag{2.3}$$

Figure (2.2) shows that, for a 3-layered system, above a finite temperature ($1/J_c = 1.66$) all the layers are disordered. When the temperature is decreased (where K/J_1 is smaller than 0.6) the magnetization of layers begins to alter from zero; firstly, the top layer then also the center layer and, finally, all of them are in the ordered phase (the numbers inside the bounded area indicate how many of the layers are ordered). Critical points for 2^{nd} and 3^{rd} layers also increase with K up to a finite value of K.

After this point ($K/J_1 = 0.6$), T_c remains nearly steady for all layers.

2.1.1 Calculation of Density with the Recursion-Matrix Method

The recursion matrix, which is constructed by the first derivatives of the interaction coefficients, can link the densities of two successive points along the renormalization trajectory on the parameter space. In our recursion space, these matrix elements remain unchanged when the transformation approaches a fixed point or a sink. The interaction coefficients also remain unchanged on these specific points. The layer energy densities can be obtained from the left eigenvector of this recursion matrix with an eigenvalue equal to the rescaling factor b^d.

$$b^d \vec{M}^* = \vec{M}^* \bar{T}^*, \tag{2.4}$$

where $\vec{M}^*$ is the density vector and $\bar{T}^*$ is our recursion matrix on a fixed point or on a sink. The recursion matrix on fixed points can be calculated easily because all the interactions are of their limit values. Left eigenvectors corresponding to the b^d eigenvalue of the T^* matrix are the densities at the sinks. For instance, one of the left eigenvectors for our model satisfying Eq. (2.4) is $[1, 1, ..1, 1]$ which means that all spins choose up (or down) direction. As mentioned earlier, any point in the parameter space flows to a fixed point by RG recursion relations. Hereby, if at each following point on the flow, recursion matrices are calculated, then after sufficient RG steps, say n, the final matrix $T^{(n)}$ reaches T^* as represented in Eq. (2.5). The only unknown of this equation, the density vector (M_0) at any temperature, can be calculated by simple linear algebra.

$$b^d \vec{M}_0 = \vec{M}^{(nd)} \bar{T}^{(n)} \bar{T}^{(n-1)} ..... \bar{T}^{(2)} \bar{T}^{(1)} \tag{2.5}$$

As shown in Figure (2.3), the energy density value of the system goes to its saturation value, 1, and all spins in the system are completely aligned at $T = 0$. How quick all the spins on the each layer join the ordered phase is determined by the α parameter, which

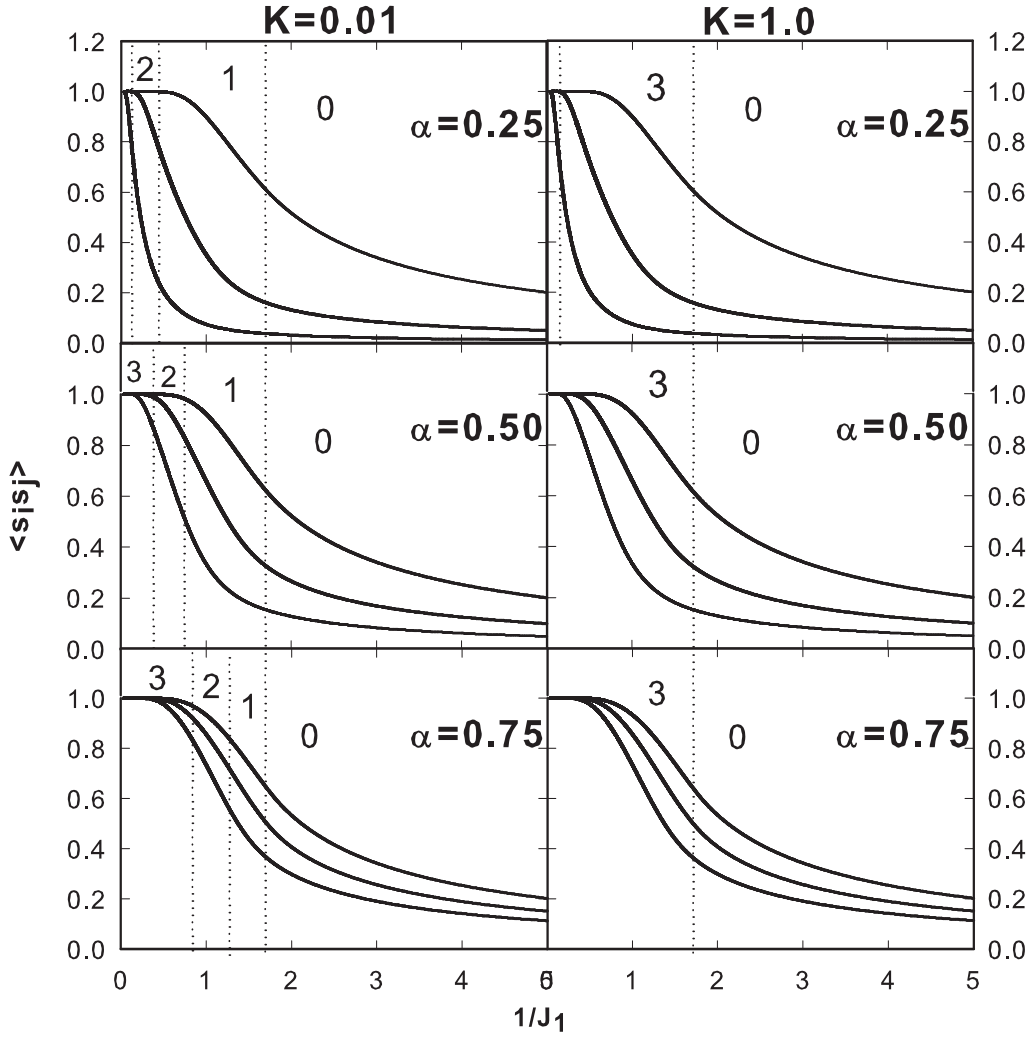

Figure 2.3: Calculated densities ($<s_i s_j>$) for different α and K values. As α goes to 1, all the densities become equivalent. The numbers indicate how many of the layers are ordered.

defines the relationship between the coupling strengths of the layers. For example, when α is equal to 1, all layers saturate at same temperature as expected. As α decreases, the gap between layer densities gets larger as shown in Figure (2.3).

The general construction of our density vectors for the 3-layer system can be written as $[1, 1, 1, < s_i s_j >_1, < s_i s_j >_2, < s_i s_j >_3]$. These vectors become $[1, 1, 1, 1, 1, 1]$, known as the 1 sink where the system is in the FM phase at low temperatures, or $[1, 1, 1, 0, 0, 0]$, known as the 0 sink of the PM phase at high temperatures. Below $K = 0.6$, these vectors become $[1, 1, 1, 1, 0, 0]$ and $[1, 1, 1, 1, 1, 0]$ with decreasing temperature. These sink vectors indicate one-layer and two-layer ordering phases, respectively. The $M^{(n)}$ in Eq. (2.5) has to be one of these vectors because each flow definitely is supposed to choose one of the given sinks. The components of T matrix are in the general form of

$$T_{\alpha\beta} = \frac{N_\alpha}{N_\beta} \frac{\partial F_\alpha}{\partial F_\beta}, \tag{2.6}$$

where F_α and F_β are the general representations of the various interactions, K, J and the additive constant, G. N_α and N_β are the total number of α and β type interactions, and $F'_\alpha = F'_\alpha(F_1, F_2, F_3, G_1, G_2, G_3)$ in our system. Terms with prime refer to the renormalized interactions. In addition, N_α and N_β are calculated numerically by considering that the system is an infinite lattice.

2.2 Bulk Anisotropic Model

By setting different anisotropy parameters, α, phase diagrams can be obtained in J space. For instance, a combination like $J_1 = J_3$ and $J_2 = \alpha J_1$ can be used to observe the phases in 2-dimensional J space. Figure (2.4) shows how our phase diagrams evolve with change in K: when K is equal to zero, we have three independent systems with $d = 2$. As represented in Figure (2.4), the four different regions indicate (I) only the center layer is ordered, (II) all layers are ordered, (III) all layers are disordered, (IV) the upper and the bottom layers are ordered. With the increase in K, two fixed points at $(J_2 \to \infty, J_1 = 0)$ and $(J_2 \to \infty, J_1 = 0.6)$ begin to approach and then annihilate each other. The same behavior is observed on the $J_1 \to \infty$ axis.

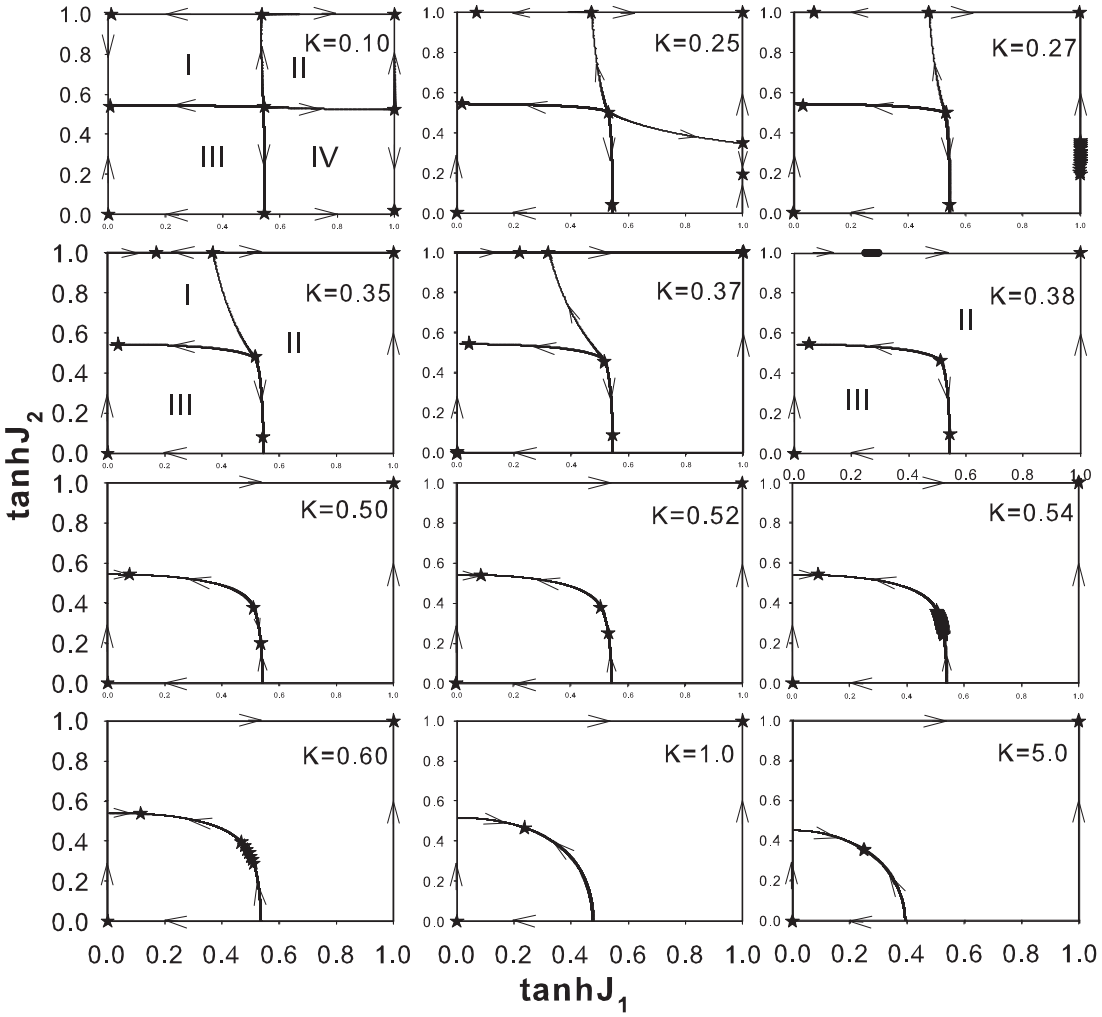

Figure 2.4: Evolution of the phase diagrams with K. I, II, III and IV represent the phases in which only the center layer is ordered, both are ordered, all layers are disordered, and only the upper and the bottom layers are ordered, respectively. As K become large, only II and III phases occur in the system as expected

The reason of the cancelation first occurs at J_2 is that, the center layer enters into the ordered phase at a lower temperature compared to the top and the bottom layers. Also two fixed points at $(J_1 = 0.5, J_2 = 35)$ and $(J_1 = 0, J_2 = 0.6)$ get closer and cancel each other, too. On the final phase diagram, for finite values of K, there is only one fixed point left on the critical curve separating the paramagnetic and ferromagnetic phases. Above a K value, the system is not allowed to undergo phase transition layer-by-layer anymore. The asymmetry of the fixed point on the curve is because spins on the center layer have more neighboring spins than those on the top and the bottom layers.

Chapter 3

ROUTE OPTIMIZATION FOR MIGROS HOME DELIVERY SYSTEM

Migros Sanal Market is an internet company which is founded in 1990 to serve its customers as a cyber supermarket in Turkiye. The working mechanism of *Sanal Market* is based on its internet site "*www.kangurum.com.tr*" via which costumers can shop and choose specific delivery times for their orders. Delivery times are not flexible, costumers have to choose one of 4 time periods within a day. Orders collected in the main server which is located in headquarter are sent to the nearest (to costumer) and proper store via internet connection. Lists of received orders for a specific service time are prepared, by a staff, by collecting items from shelves manually. Finally, packages are distributed to vehicles to complete the delivery process according to personal experiences of drivers. The number of the orders per vehicle is determined by the drivers as well as the delivery sequence of items. These two process, delivery routes and sharing of items, are the backbone of all the process as they directly effect the fuel expenses and efficient use of delivery times.

As mentioned above, each vehicle is supposed to deliver a given number of items. The order of the delivery is drawn by drivers, and they generally follow the nearest costumer's address one after another, namely they apply *greedy algorithm* unconsciously. They always have a tendency to go to the closest costumer, however, there are cases in which greedy algorithm does not work well. To obtain a more cost-efficient global path, sometimes a farther costumer should be preferred instead of a closer one. However, such a path cannot be found by the drivers individually even if they know all distances between all costumers. Therefore, usage of an optimization process is necessary and inevitable for a certain success. If the delivery paths can be optimized, this shortens the total delivery time and total distances driven per vehicle, meaning

a decrease in costs.

Determining the number of orders per vehicle and their distribution are another factor which plays an important role in raising the efficiency. In a normal process, how costumers are separated into groups are determined by drivers. However, even for a small number of costumers, there are thousands of different configurations of grouping them. For example, there are 3^12 different ways of dividing 12 costumers into 3 vehicles, and finding the best grouping among these configurations is another hard problem to get over. Thus, there is a too low probability that the drivers always find the most optimized result.

In this thesis study, both route and sharing optimization were introduced by the simulated annealing process mentioned in *Chapter I*. In the beginning, real geographical distances obtained from GPS(Global Positioning System) maps and direct mathematical distances between costumer addresses were used for energy functions needed in the metropolis algorithm, then traffic factors was also added into the energy functions as an effective parameter.

3.1 Optimization of Delivery Routes of Migros Sanal Market

The simulating annealing algorithm, which is mainly an energy minimization process, is introduced to optimize the delivery route of each sanal market vehicle. For the beginning of this project, one of the sanal market stores and its delivery range is chosen for a pilot application area in the Asia side of Istanbul. At the service of this store there are 3 vehicles whose carrying capacities are enough for the maximum 8 costumer packages. For this problem, maximum costumer number for each service period is taken as $3 \times 8 = 24$, 3 vehicles multiplied by 8 costumer addresses. The purpose of this problem is to find the minimum route corresponding to the minimum energy with an analogy to the simulating annealing process mentioned in previous chapters. In saying so, vehicles starting from the store stop at each costumer's house to deliver packages and then come back to the store by following the shortest path among the given addresses. That is to say we apply Traveling salesman problem (TSP) for Migros sanal market vehicles: In TSP problem, a salesman needs to visit

a number of cities without visiting the same city twice by using the shortest path. SM algorithm optimizes the route of the salesman by creating new routes and then checking them according to the Metropolis rules. Finally when the temperature of the SM algorithm decreases to zero, one of the shortest path is obtained.

In the optimization process, the total distance of journey is taken as the energy value needed for the SM algorithm. These distances can be obtained in two different ways:

1. From the distance equation between two points in 2-dimensional space as given in Eq. (3.1). The sum starts from the 0^{th} element of the coordinate array to the $N+1^{st}$ element. Here, the $N+1^{th}$ stop is equal to the 0^{th} stop because the driver should complete a circle path starting from the store and finishing at the store. The coordinates in our problem are latitude and longitude values of approximately 3000 costumers served from this store which are recorded as a 2-dimensional data set. Latitudes should be multiplied by $111km.$ which is the distance between two adjacent lines parallel to equator to obtain the real distances, and coefficient of the longitudes is approximately $88km.$ which is the distance between two meridians pinching this area of the city.

$$E = \sum_i \sqrt{111^2(x_i - x_{i+1})^2 + 88^2(y_i - y_{i+1})^2}. \tag{3.1}$$

2. From the real walking distances obtained from GPS data which is stored as a NxN $\overline{E}$ matrix where N is the number of costumers. Any element E_{ij} of this matrix is the distance between ith and jth coordinates. Therefore, all diagonal elements are automatically zero.

To be able to start applying the SM algorithm, new configurations should be obtained by changing the initial generation. In this problem, for consistency and for a fair comparison of results, the greedy algorithm (GA) route is considered as the initial path. Note that for SM, the initial state is not important. New generations are produced by small changes in the current route such as changing the order of

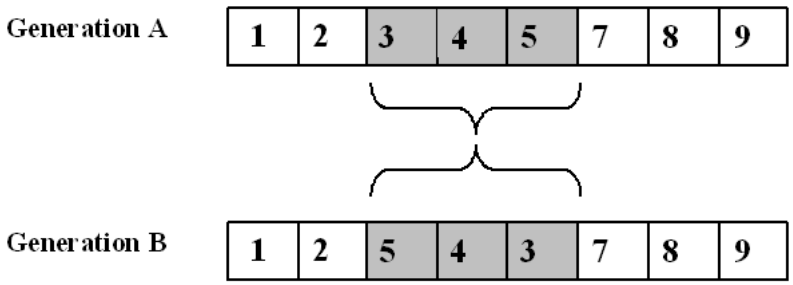

Figure 3.1: A randomly chosen fragment in the generation A is reversed to create the new generation B

two (or more) costumer coordinates. In this thesis, fragments whose lengths are randomly determined are reversed to obtain new generations as shown in Figure (3.1). While perturbing the system in this way to create new configurations, there are a few strategies which can be followed to earn computational time. Suppose that the number of costumer addresses is N:

- Always hold $i < j$ where i and j are randomly chosen two indices between 0 and N-1
- If the length of the fragment between i and j is zero, namely $i = j$, then increase j by 1.
- If $i = 0$ and $j = N - 1$ then decrease j by 1. This hinders a meaningless step to create a generation.

After creating a new generation (path), this new configuration should be compared to previous generation to decide whether new one is replaced with the old one or not. To achieve this, the first thing to do is checking the energy differences as done in the Metropolis algorithm mentioned in *Chapter* 1. If the new energy is smaller than the old one, accepting the new generation brings the system to a lower energy namely, a shorter path, and this is exactly what we want. What about if the situation is opposite of this: the energy of the new configuration generated by small changes being larger than the old one. Classically, this state is automatically refused, whereas,

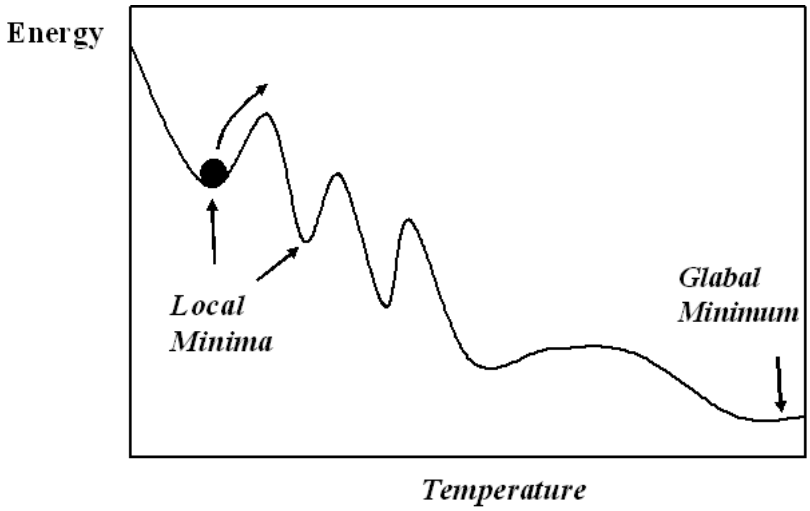

Figure 3.2: Illustration of the Simulating annealing algorithm; a ball being stuck in a local minimum. To make the ball climb up and reach the global minimum, a temporary increment of energy should be accepted.

in SM, higher energies still have a chance to be accepted. This difference refers to the main idea of the Metropolis algorithm and SM. Although, at first glance, acceptance of high energy seems as a loss, this sacrifice rescues the system from being stuck in local minima. The probability of the new state to be accepted with respect to the old state is proportional to $e^{-\Delta E/k_B T}$, where ΔE is the energy of the new state minus that of older one. When this finite normalized probability is greater than a specific normalized number, the new state can take the place of the old state. This action may get the system out of a local minimum by a jump-up. This move can be depicted as a ball rolling down on a cliff as represented in Figure (3.2). At that point, one can ask how the algorithm knows where it should make the height of the jumps smaller. The answer is hidden under the fact that the SM algorithm presumes that if each minimum is represented as a crater, global minimum is the largest of them. Therefore, when the ball falls into the largest crater, it is harder to climb up and escape from this crater. The magnitude of these jumps gets shorter as the temperature approaches to its zero value.

Producing new generations and controlling them are repeated "many" times at each temperature value as the program runs to reach zero temperature. In commonly

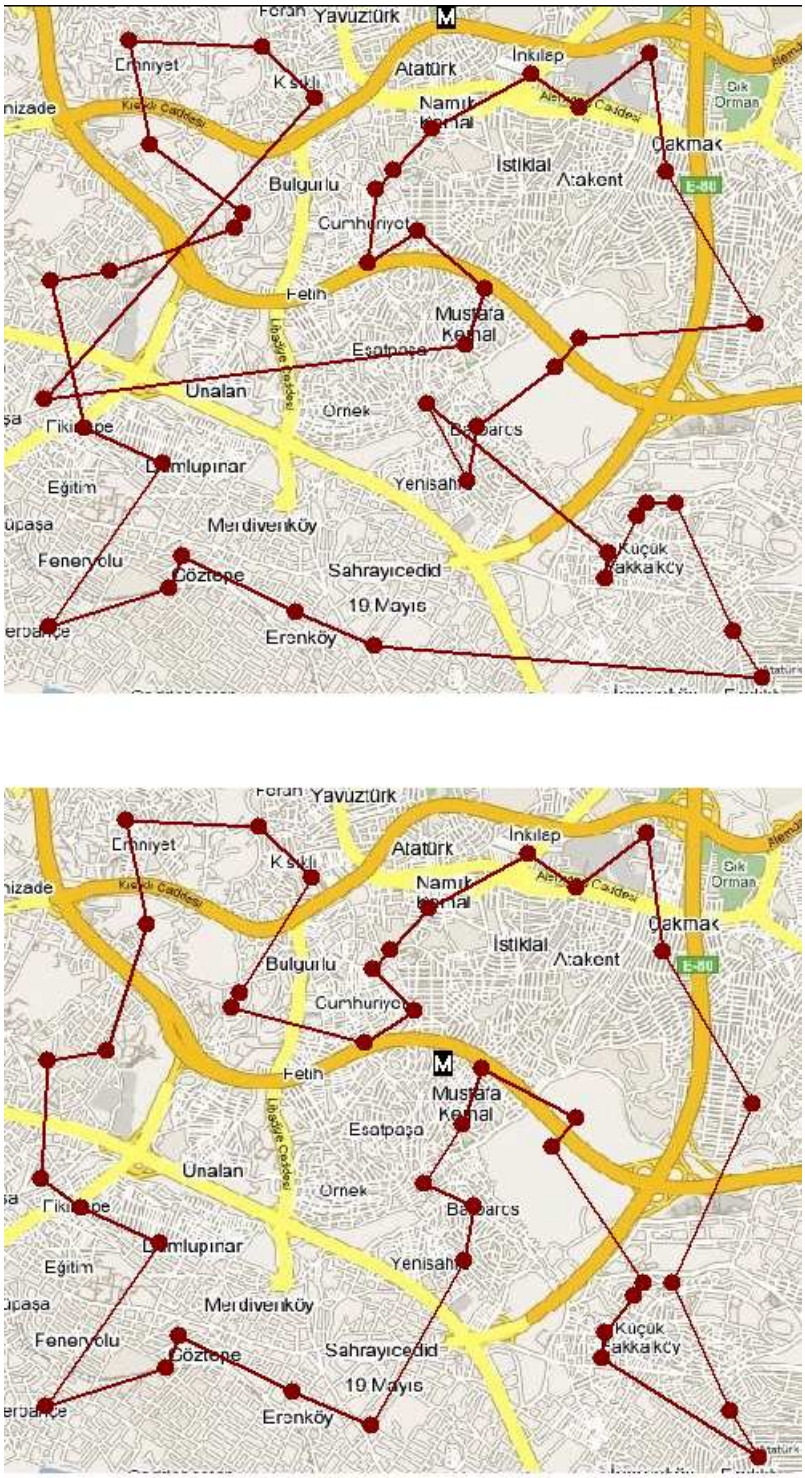

Figure 3.3: Optimization of a delivery route starting from a particular store. The greedy algorithm, in which the nearest points are preferred (upper), is compared to optimized route (bottom). %17 improvement is obtained with respect to the GA for a specific group of costumers. Distances are obtained from Eq. (3.1).

used terminology, "many" is a number times *Monte Carlo* (MC) steps where each MC step is the number of costumer addresses N. In our work, MC steps were changed with respect to temperature to speed up the algorithm. The less the temperature is, the more the MC steps are, hence at high temperatures the system does not spend too much time.

The cooling schedule and initial value of temperature are other important concepts in SM. As we mentioned before, the SM algorithm obeys the rule that all states have a probability of the Boltzmann factor e^{-E/k_BT}. At high temperatures, system can accept almost all configurations. However, the lower the temperature, the less configurations are accepted and the system needs to stay more at low temperatures to test many more generations. Therefore, the cooling schedule is chosen in such a way that the system spends more time at lower temperatures. Figure (3.3) represents the result of an optimization process for a given number of costumers. In our work, temperature is decreased by $\%5 - 10$ until the absolute zero. On the other hand, the starting temperature can be chosen close to the largest energy difference as represented in Eq. (1.8).

3.2 Optimization of Sharing the Items between the Trucks

As summarized above, generally there are more than one vehicles to deliver all orders, and there must be a mechanism determining to which vehicles deliver which addresses. This necessity of planning automatically gives birth to a parameter effecting the optimization which is to divide the costumer addresses between the vehicles in a cost-efficient way. This is the counterpart of the problem of how m balls can be put into n different boxes. However, this time there are some limiting conditions on the number of balls in each box, and as an extra requirement, the balls in each box must localize in a "special manner". In our problem, boxes are vehicles, and the balls correspond to the costumer coordinates. In our case, "special manner" used for boxes above is nothing but the minimum route for each vehicle. Two optimization processes are run hierarchically to guarantee both the minimum path and the most efficient grouping; first most-ideal groups are established, then addresses in each group

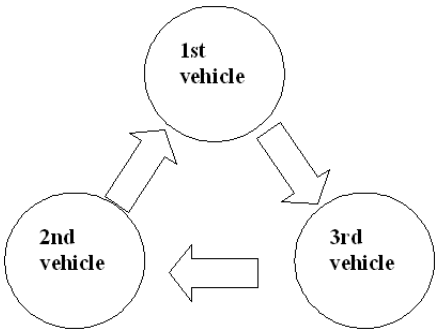

Figure 3.4: To obtain new groups for the process of delivering the costumers between the vehicles, a randomly-chosen number of costumers is exchanged among two vehicles. Periodic boundary condition is employed for final vehicle

are route-optimized.

The SM algorithm is employed one more time for this minimization step too. Same arguments described above for the SM algorithm keep their validity such as an imaginary temperature, energy, and proper number of MC steps. However this time, two optimization processes should run together to find the solution. This is to say that two different temperatures, MC steps, and processes of generating new states have to be run together in a proper hierarchy. The outer loop is for optimization of grouping while the inner loop optimizes route for each group separately and in parallel. If the process is explained explicitly, initially, the total number of addresses is divided between vehicles equally, then the sum of lengths of their paths are calculated as the initial energy. Following, new generations need to be created, namely, new configurations of addresses need to be assigned to each vehicle. Here, the same method described in *Section* 3.1 can be used again. This time fragments, whose lengths are determined by a set of two random indices, are exchanged among the different vehicles as shown in Figure (3.4). For instance, addresses of the first vehicle between two indices i and j are exchanged with addresses of the second vehicle between i and j, and a fragment between k and l of the second vehicle performs the same with the third vehicle and so on. Periodic boundary condition can be introduced for the last group so that it can interact with the first vehicle. The new groups which are candidates for the most efficient sharing are sent to the inner loop where the addresses

of each group are arrayed by route optimization process independently in a parallel manner. This means that for each group, the process defined in *Section* 3.1 is applied like an individual optimization process. Finally, the sum of their optimized routes is compared to initial energy so that if smaller, it is accepted. Otherwise, a check is made that $e^{-\Delta E/k_B T}$ is larger than a probability defined randomly. If the answer is still negative, then we start from the old generation again until the temperature of the outer loop is down to zero. We emphasize again that there are two temperatures, outer and inner, and two MC step loops related to these temperatures. However, while the inner temperature decreases to zero "the number of vehicles" times at each outer MC step, the outer temperature approaches zero only once where the whole optimization process is completed.

The result of the above process gives one of the most cost-efficient combinations for vehicle routes which include equal number of costumer addresses as illustrated in Figure (3.5) for a given set of costumers. However, in real life there is no restriction that each vehicle should carry an equal number of items. Also, sometimes in given optimization results, although two points are too close to each other, they are members of different groups, namely, carried by different vehicles. So, one can naturally ask what about adding these two addresses into the same group. This is completely possible and can increase efficiency. During the process of generating new configurations among the groups, another way of perturbation is to allow groups to give an address to the other group but not to take another back instead of the given one. Thus, costumer addresses assigned to each vehicle do not have to be equal anymore, and another degree of freedom is introduced to effect optimization positively. Of course, there need be upper and lower limits for the number of items carried by each vehicle to consider carriage limits and other capacity problems. In this work, the number of addresses assigned to each vehicle can deviate ± 2 with respect to the other vehicles' load under the condition that the total number of costumers is constant. Note that this deviation can be controlled by keeping distances driven by each vehicle close to each other.

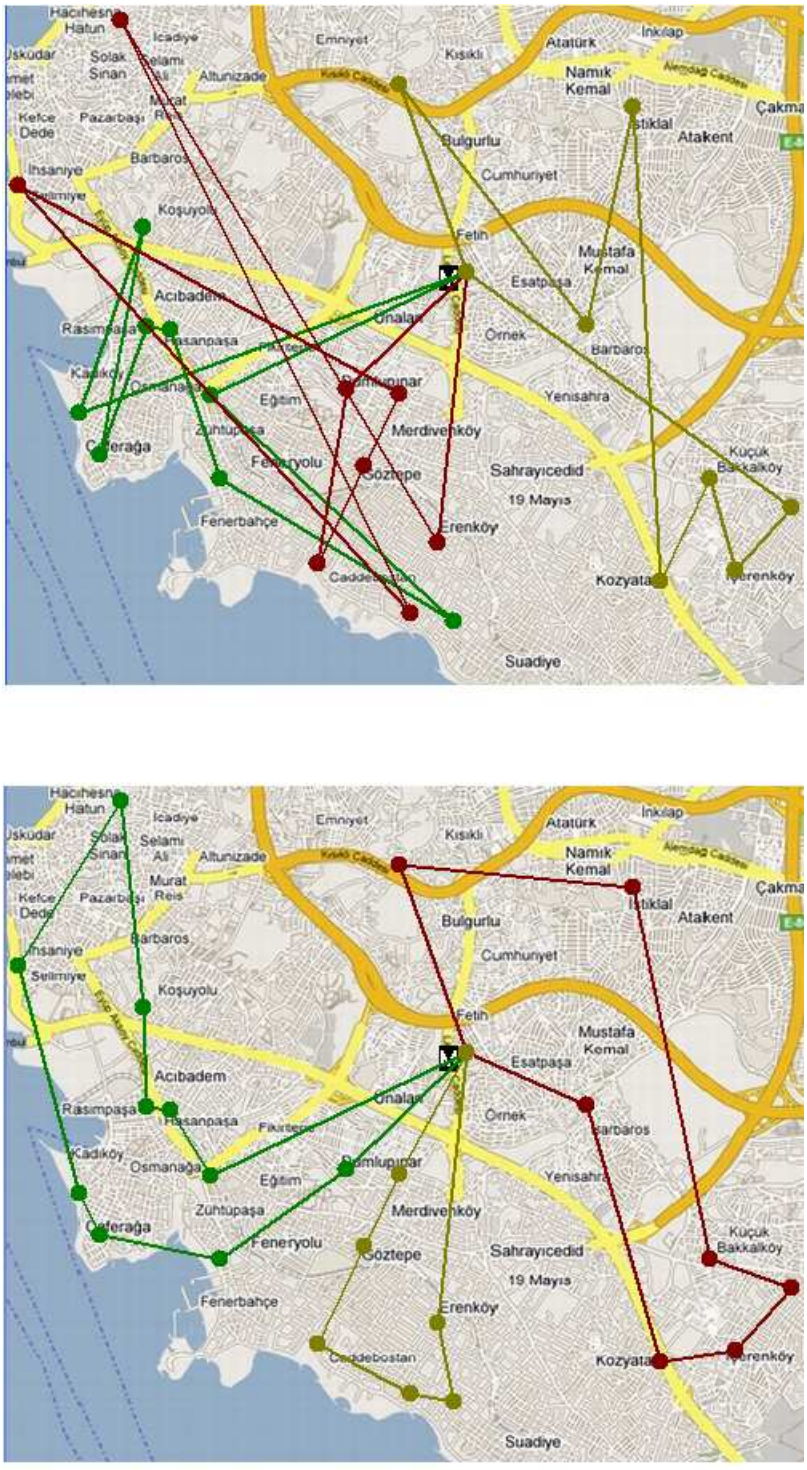

Figure 3.5: Optimization of grouping and delivery routes for 3 vehicles starting from a particular store. The greedy algorithm in which the nearest points are always preferred (upper). Optimized route (bottom) gives %40 efficiency compared to greedy algorithm , and distances are calculated as given in Eq. (3.1).

3.3 Numerical Details

In this project, the C++ programming language is used for writing the code of optimization algorithms. The object-oriented features of C++, classes, are employed, hence we are able to create special variables such as three-column vectors. Explanation of the vector standard template library used in our coding process and the logic of the algorithm is briefly given below.

3.3.1 Vector Standard Template Library

The costumer data is stored in the one-dimensional but three-column special vectors in which the first column is assigned to latitudes, the second to longitudes and the last one to the costumer identity numbers. Vectors are preferred instead of ordinary c++ arrays due to their flexibility and compatible structures with classes.

Vector: C++ vector is a container template packed in the standard template library (STD) and a kind of sequence container whose elements are ordered by following a strict linear sequence. But unlike regular arrays, storage in vectors is handled automatically. They are also allowed to be expanded and contracted as needed, so this saves us from defining too large arrays in the memory at the beginning of the code. A few useful features of vectors are

* Accessing individual elements by their position index.

* Iterating over the elements in any order.

* Add and remove elements from its end.

* They have the ability to be easily resized.

Note that the $\langle vector \rangle$ header file should be included in all the header files of the code.

In addition, two-dimensional vectors also can be constructed by embedding one another, namely they are vector of vector. vector¡int¿

vector $\langle int \rangle$ *v(M);*

vector $\langle$*vector* $\langle int \rangle \rangle$ *v2(N,v);*

Vectors have their own special functions to implement the stored data and these functions are used in our code rarely to speed up the algorithm. These functions can

be listed as

The push_ back() function appends value to the end of the vector.

Its syntax: void push_ back(const TYPE & value);

The function resize() changes the size of the vector . If val is specified then any newly-created elements will be initialized to have a value.

Its syntax: void resize(size_type num, const TYPE & value = TYPE());

The swap_ ranges() function exchanges the elements in the range [start1,end1) with the range of the same size starting at start2.

Its syntax: iterator swap_ ranges(iterator start1, iterator end1, iterator start2);

The function begin() returns an iterator to the first element of the vector.

Its syntax: iterator begin();

The size() function returns the number of elements in the current vector.

Its syntax: size_ type size() const;

3.3.2 Illustration of the Algorithm by C++

As we mentioned earlier, optimization process were both used for finding the most cost-efficient route and the best configuration of costumer-vehicle grouping. Our code has a capability to find both of the results in the short-time scales by using vector STD and class features of C++. The code illustration given below explains the algorithm and programming-language syntax step by step

int main()

Number of vehicles and costumers

int as=3, N=24;

Latitudes and longitudes are stored in a 2-dimensional array after multiplying by real distances

ana[i][j]=pow((pow(111(citx[i]-citx[j]), 2)+pow(83*(city[i]-city[j]), 2)), 0.5);*

Embed the latitudes, longitudes, and costumer identification numbers into a three-column vector which is created by classes

Vec[i]=vectors(citx[i], city[i],sanalid[i]);

Coordinate of the store is added to the end of the vector

store=vectors(citx[0], city[0], 0);

Apply GA and assign the vector itself

Vec=greedy(Vec);

24 costumers are divided between 3 vehicles and stored in the new $VecAS$ vectors, then the store coordinate is added to the end of each group.

VecAS[number of vehicles][capacity of each vehicle]=Vec[total number];

VecAS[number of vehicles].push_back(store);

Calculate the lengths of the routes of each vehicle, then sum to obtain total route and store it in "Total Initial Energy"

Initial Energy[vehicle i]=LENGTH(VecAS[vehicle i]]);

Total Initial Energy+=Initial Energy[vehicle i];

Start the outer loop

while(decrease the temperature of the outer loop to zero)

for(Monte Carlo Steps of the outer loop)

Define a set of loops to exchange costumers between vehicles. First and second, second and third,... last one and first one.

Store the initial configuration of costumers before starting to generate new ones

Storage[vehicles]=VecAS[vehicles];

There are three ways of creating new configurations.

(I)Replace a randomly chosen fragment from vehicle i with vehicle j by swap function

swap_ ranges(start of i, end of i, start of j);

(II)Exchange only one costumer between the vehicles

(III)Vehicle i can give one costumer to vehicle j, but cannot take one back

During these process, each costumer group assigned to vehicles needs contain the coordinate of the store due to the condition that vehicles start from the store and return to the store again.

Start the inner loop to optimize the route of each vehicle.

while(decrease the temperature of the inner loop to zero)

for(Monte carlo steps of the inner loop)

for(Number of vehicles)

Store the initial route of each vehicle

Storage Inner[vehicle i]=VecAS[vehicle i];

Generate new paths by reversing a randomly chosen fragment of each vehicle,

for(length of fragment)

VecAS[vehicle i][fragment]=Storage Inner[vehicle i][fragment];

Calculate the energy and energy difference

energy[vehicle i]=LENGTH(VecAS[vehicle i]);

deltaE[vehicle i]=energy[vehicle i]-Initial Energy[vehicle i];

Control the new path according to the Metropolis rules

if(metropolis(deltaE[vehicle i]==TRUE)

Initial Energy[vehicle i]=energy[vehicle i];

if(metropolis(deltaE[vehicle i]==FALSE)

VecAS[vehicle i]==Storage Inner[vehicle i];

end of the inner MC steps loop

end of the inner temperature loop

There are three optimized independent routes whose sum is equal to total energy

Total Energy+=LENGTH(VecAS[vehicle i]);

deltaE Total=(Total Energy-Total Initial Energy);

Then Metropolis for the outer loop

if(metropolis(deltaE Total==TRUE)

Total Initial Energy=Total Energy;

if(metropolis(deltaE Tota==FALSE)

VecAS[vehicles]==Storage[vehicles];

end of the outer MC steps loop

end of the outer temperature loop

system("PAUSE");

return 0;

Chapter 4

CONCLUSION

In this study, two subjects of statistical mechanics were worked successfully by introducing numerical and mathematical approaches. In the first part, hierarchical lattices were used to implement anisotropic hierarchical lattices corresponding to a 3-dimensional spin-1/2 system of interacting layers. Second part of the thesis includes a minimization application. The simulated annealing was introduced to optimize the delivery mechanism of *Migros Sanal Market* which is a business-to-costumer (B2C) company. B2C is a way of electronic commerce in which costumers can order products via internet to their homes directly. Optimization was performed on the delivery system of these orders which can be analyzed under two title, the delivery sequence of costumer addresses for each vehicle and the determination of which vehicles accomplish the orders of which costumers.

In our anisotropic hierarchical lattice, while spins which dwell on the same layers interact with J_i where i is the indices of the layers, the coupling between spins on different layers is K. An α parameter is employed to determine the relation among the couplings of the layers such that $J_i = \alpha^{i-1} J_1$ where J_1 is the interaction of the upper surface. Hierarchical lattices are constituted by parallel and mutual embeddings of unit graphs repeatedly. In our anisotropic model, only J type interactions are allowed to iterate, and K interaction is kept constant at each step. In construction of these lattices, two important rules, proper reductions to the lower dimensions when one of the interactions set to zero, and recovery of isotropy as all coefficients set to zero, are followed. Our model turns into a group of independent 2-dimensional models if K is set to zero and becomes an isotropic lattice, where all J_i and K couplings are equal, namely $\alpha = 1$.

As mentioned earlier, hierarchical lattices exhibit phase transitions, and in our

case, this situation was exemplified with Figures (2.2) and (2.4). The phase boundaries of a 3-layer system in Figure (2.2) were depicted for different K values of which each corresponds to a different physical model. At high temperatures, all layers are disordered, hence in paramagnetic phase, as represented by "Phase 0" in Figure (2.2). As the temperature decreases to zero, layers begin to get into the ordered (ferromagnetic) phase one by one. However, these are valid only when inter layer coupling K/J_1 is smaller than 0.6. Above this value, all layers undergo a phase transition at the same temperature. The results are logical because of the presence of the K interaction. When K/J_1 is smaller than 0.6, each layer undergoes a phase transition from PM to FM at its specific temperature, and its spins begin to align in parallel. Spins on the firstly aligned layer, say the upper layer, try to flip the spins on the bottom layer with an effect proportional to K. However, for small K values, this effect cannot induce enough force to achieve this. When the temperature decreases to a sufficient value, or in renormalization language, coupling of upper layer strengthens enough to change the orientation of spins on the bottom layer, both the upper and the bottom surfaces cross to the ordered phase from the disordered phase as represented with "Phase 2" in Figure (2.2) and so on. On the contrary, when K/J_1 is greater than 0.6, then all the layers act together. The system exhibits a phase transition from "Phase 0" to "Phase 3" at one particular temperature no matter what the J_i values of the surfaces are.

Phase diagrams of the 3-layer system in Figure (2.4), which were obtained by setting interaction of the upper and the bottom layers equal to each other, show explicitly how the system evolves with different K values. When K is zero, the system is a bunch of 2-dimensional lattices. If K is altered from 0 slowly, then the connections between these layers are automatically established, and the system turns into a 3-dimensional lattice. 4 phase regions were observed for $K = 0.1$, all layers are ordered (II), all layers are disordered (III), only surface layers are ordered (IV), and only the center layer is ordered (I). Increment in K directly effects the number of phases observed, for instance, phase (IV) and phase (I) are getting shrunk as K increases. Above the $K = 0.5$, there are only two phases: Phase (III) in which all spins

in the system are aligned in the same direction, and Phase (II) in which the system is in the paramagnetic phase. While K approaches to infinity, layers are getting closer and closer, and finally the system begins to exhibit the behaviors of a 2-dimensional system.

Evolution of the fixed points can also be observed from Figure (2.4). As shown in graphs, while K is small, there are 9 fixed points. However, as K increases, fixed points on the $J_1 \to \infty$ and $J_2 \to \infty$ lines come closer and then cancel each other. Phase (II) and Phase (IV) merge after the related fixed points disappear. The same happens for Phase (II) and Phase (I), respectively. As K reaches a relatively higher value, only one fixed point - the fixed point of isotropy - can survive on the critical curve. The reason why this fixed point is asymmetric is that the unit graph of center layer shown in Figure (2.1) includes more bonds than those of the upper and the bottom layers, so it reaches the fixed point of isotropy faster.

Also for our anisotropic hierarchical lattice, internal energies were calculated for different isotropy parameters α. When $\alpha = 1$, the system is fully isotropic. The saturation value of $< s_i s_j >$ at T=0 is 1, meaning that all spins in the system align along the same direction. As indicated in Figure (2.3), above a K value, although all layers exhibit a phase transition at the same temperature, there is no significant change in their internal energies.

In future studies, the number of layers will be increased to observe the *Roughening Transition* which is the transition of interface under opposing external surface magnetic fields. When the magnetic fields with opposite signs are applied to the top and the bottom layers separately, a finite number of layers at the interface, which are effected by these two magnetic fields at the same time, exhibit zero magnetization. However, below the roughening transition T_R, no paramagnetic layers are observed. Namely, magnetization of the upper half of the system is 1, while the bottom is -1 as indicated in Figure (4.1).

The other project worked in this thesis study was the optimization of *Migros Sanal Market* routes. Both the shortest path for the delivery vehicles and the most ideal separation of addresses into the vehicles were successfully optimized. Thus,

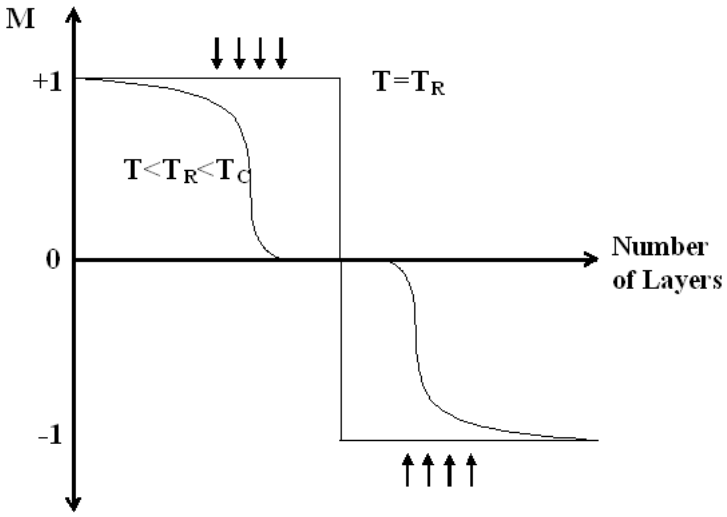

Figure 4.1: Illustration of the roughening transition; below the roughening temperature there are layers having zero magnetization at the intersection of two opposite signed external magnetic fields

each vehicle can complete its travel by visiting all costumers belonging to its group which is chosen among the many possibilities. The simulated annealing algorithm was used for both minimization processes, and remarkable results are obtained. To check and compare our results, real costumer data of a store in Istanbul were used in collaboration with *Migros Sanal Market* whose service capacity is 8 costumers per each of 3 vehicles, 24 in total. The greedy algorithm, which runs by visiting the nearest neighbors one after another, was taken as the initial path in accordance with the drivers' general attitude to draw their own paths. The efficiency obtained by the SM algorithm ranges between % 10-20 compared to this initial path for any route completed by single vehicle as shown in (3.3). When the algorithm is used to distribute the costumer addresses to the vehicles, the total length of the paths assigned to each vehicle is shorter than those of drivers, and an efficiency between % 20-30 is obtained for various combinations of the problem like the different number of vehicles, costumers, etc. When dividing the costumer packages into the trucks, the number of packages per vehicle may be flexible. Vehicles are sometimes allowed to visit different number of costumers to combine close addresses.

The gain of our algorithm was also directly tested with the original data recorded

on vehicles of the Migros Store at different periods. Times spent and distances traveled spent by each vehicle were recorded, and the routes given by SM and drivers' attitudes were compared in real time. As a result, efficiencies were found similar to above results. Note that our algorithm only gives the order of delivery and related groups. Therefore, how drivers travel between given two addresses completely depends on themselves.

In this algorithm, routes are coded in a such way that each vehicle starts from the Migros store and after completing its route comes back to the Migros store. Other compulsory check points such as petrol stations, bridges, highways can be adapted into the algorithm. For instance, only after the last service period of day, drivers are allowed to go to their home by vehicles, so for this time period, final stop can be drivers' home, etc. Energy values needed for SM are calculated linearly or obtained by GPS data. Moreover, some special values can also be added by modifying the energy function such as amount of packages, emergencies, or traffic jams. In the map shown in Figure (4.2), a quenched random field on a two-dimensional frame corresponds to the traffic jam in different neighborhoods of Istanbul. The magnetite of the field increase with effect of the traffic congestion. For example, while 10-point refers to a heavy-traffic flows, 3-point is used for empty traffic. To obtain more realistic optimization results, these random fields are going to be adopted into the energy function. These traffic factors can vary within a day or according to weather conditions. However, fields need to remain quenched during the delivery time of one vehicle.

Currently, our algorithm is being encoded by the computer engineers of Migros to actively use it for delivery system of the sanal market. At the first stage, the aim of this project is going to be increasing the number of costumers per vehicle from 8 to 9 and total 27 per period. However, in some cases, this procedure may cause delays on delivery times because some addresses could be hard to reach compared to the others, and this automatically takes more time than predictions. Therefore, to avoid such an action, the computers connected to the web server which records costumer data make the SM run after the total number of costumers exceeds a limit value. As new costumers get added to the order list, the SM runs and calculates

Figure 4.2: Quenched random field corresponds to the rate of traffic congestions. Numbers increase as the traffic flows get slower in that area

the total distances. If the total allowed length of route per vehicle is exceeded, the system stops taking new costumers even if the upper limit of number of costumers is not reached yet. The second aim is going to be that after shortening the average circling times, the number of daily services is going to be 5 instead of 4. In addition, some technological tools can be easily combined with our software, such as mobil GPS receivers assembled on vehicles can direct the driver to the costumer addresses by giving detailed road descriptions according to SM outputs. As a consequence, once this algorithm is applied, the same algorithm can be used for delivery of items from main depots to stores by trucks, collecting of items from store shelves, etc.

APPENDIX

Recursion relations for $J_{surface}$ and J_{bulk} type interactions can be obtained from RG procedure. As seen in Figure (2.1), while surface interaction diagrams consist of one side arm and one diamond lattice which is illustrated with Figure (1.5), bulk graphs have two side arm and one diamond lattice. Explicit form of the recursion relations corresponding to these graphs are given below. Their proper combinations shown in Eq. (2.2) can be used to obtain the full recursion relations. For instance, The recursion relation for the upper layer indicated with J_1 can be calculated as a function of J_1 and J_2, here J_2 is the coupling of the center layer.

$$\mathrm{R}(\mathrm{s}_1, s_5) = e^{J_1' s_1 s_5 + G} = \sum_{s_2,s_3,s_4,s_6,s_7} e^{K s_1 s_2 + J_2 s_2 s_3 + J_2 s_3 s_4 + K s_4 s_5 + J_1 s_1 s_6 + J_1 s_6 s_5 + J_1 s_5 s_7 + J_1 s_7 s_1},$$

$$\mathrm{R}(\mathrm{s}_1 = +1, s_5 = +1) = e^{J_1' + G} = 8 + \frac{4}{e^{4J_1}} + 4\,e^{4J_1} + 2\,e^{-2K-2J_2} + 2\,e^{2K-2J_2} + e^{-2K-4J_1-2J_2} + e^{2K-4J_1-2J_2} + e^{-2K+4J_1-2J_2} + e^{2K+4J_1-2J_2} + 2\,e^{-2K+2J_2} + 2\,e^{2K+2J_2} + e^{-2K-4J_1+2J_2} + e^{2K-4J_1+2J_2} + e^{-2K+4J_1+2J_2} + e^{2K+4J_1+2J_2},$$

$$\mathrm{R}(\mathrm{s}_1 = +1, s_5 = -1) = e^{-J_1' + G} = \frac{8}{e^{2K}} + 8\,e^{2K} + \frac{8}{e^{2J_{1+i}}} + 8\,e^{2J_{1+i}}.$$

Other combinations of $R(s_1, s_5)$'s are

$$R(s_1 = +1, s_5 = -1) = R(s_1 = -1, s_5 = +1),$$

$$R(s_1 = +1, s_5 = +1 = R(s_1 = -1, s_5 = -1).$$

Eliminating the G from above equations, final recursion relation for upper layer is

$$J_1' = 0.25 ln(\frac{R(s_1 = +1, s_5 = +1)}{R(s_1 = +1, s_5 = -1)}).$$

BIBLIOGRAPHY

[1] K. Huang, "Statistical Mechanics, Second Edition" Wiley, 1987.

[2] A. Altland and. Simons, "Condensed Matter Field Theory," Cambridge, 2006.

[3] H.E. Stanley, "Introduction to Phase Transition and Critical Phenomena," Oxford, 1971.

[4] N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller, and E. Teller, J. Chem. Phys. **21**, 1087 (1953).

[5] S. Kirkpatrick, D.C. Gerlatt, Jr. and M.P. Vecchi, "Optimization by simulated annealing," Science **220**, 671 (1983).

[6] A.N. Berker and S. Ostlund, J. Phys. C **12**, 4961 (1979).

[7] M. Kaufman and R.B. Griffiths, Phys. Rev. B **24**, 496 (1981).

[8] A.A. Migdal, Zh. Eksp. Teor. Fiz. **69**, 1457 (1975) [Sov. Phys. JETP **42**, 743 (1976)].

[9] L.P. Kadanoff, Ann. Phys. (N.Y.) **100**, 359 (1976).

[10] L. Onsager, Phys. Rev. **65**, 117 (1944).

[11] A. Erbaş, A. Tuncer, B. Yücesoy, and A. N. Berker, "Phase diagrams and crossover in spatially anisotropic d=3 Ising, XY magnetic,and percolation systems: Exact renormalization-group solutions of hierarchical models," Phys. Rev. E **72**, 026129 (2005).

[12] M. Hinczewski and A.N. Berker, "d=3 anisotropic and d=2 tJ models: Phase diagrams, thermodynamic croperties, and chemical Potential Shift," Eur. Phys. J. B **51**, 461 (2006).

[13] M. Hinczewski, "Griffiths singularities and algebraic order in the exact solution of an Ising model on a fractal modular network," M.Hinczewski, Phys. Rev. E **75**, 061104 (2007).

[14] K. Wilson, "Renormalization group and critical phenomena I: Renormalization group and the Kadanoff scaling picture," Phys. Rev. B **4**, 3174 (1971) .

VITA

AYKUT ERBAŞ was born in Istanbul, Turkey on September 20, 1980. He received his B.Sc. degree in Physics from Istanbul Technical University, Istanbul, in 2005. In August 2005, he was accepted to Master in Physics with full scholarship in Koç University and worked as a teaching assistant until June 2007. He has studied on different statistical mechanics problems as a member of Prof. A. Nihat Berker's research group since 2003. He gave an invited talk at Max-Planck Inst. Dresden, Germany in July 2006 and a contributed talk in American Physical Society March Meeting, Baltimore, U.S.A. in March 2006.