NOVEL ARCHITECTURES AND DEVICES FOR COMPUTING

A thesis presented

by

Frederick Rogers Waugh

to

the Department of Physics

in partial fulfillment of the requirements

for the degree of

Doctor of Philosophy

in the subject of

Physics

Harvard University

Cambridge, Massachusetts

October, 1994

ABSTRACT

This thesis explores some of the more unusual architectures and devices being considered today as the basis for information processing, emphasizing architectures that are highly parallel and devices that are extremely small compared to current standards.

The first part of this thesis theoretically and numerically analyzes analog electronic neural networks in which competition within neuron clusters leads to pattern classification and feature extraction abilities. Global stability theorems, derived using a Liapunov approach, provide general guidelines for network design and operation. The theorems state that with continuous-time updating, competitive networks converge only to fixed points, while with discrete-time, parallel updating, they converge to either fixed points or period-two limit cycles. A stability criterion guarantees that discrete-time networks converge only to fixed points when a quantity related to the neuron gain, or transfer function slope, is sufficiently small.

A set of analytical phase diagrams for competitive associative memories is derived using a combination of statistical mechanics and nonlinear dynamics. The diagrams classify attractor types as a function of pattern storage fraction and neuron gain. Numerical tests agree well with the diagrams.

Analog annealing, a technique for improving network performance by reducing neuron gain, is shown to improve performance in an analog associative memory by dramatically reducing the number of fixed points. The number of fixed points decreases exponentially with network size with a scaling exponent that decreases with neuron gain. Numerical data based on fixed-point counts in small networks support the results.

The second part of this thesis discusses low-temperature tunneling measurements at zero magnetic field through double and triple quantum dots with adjustable inter-dot

coupling, fabricated in a GaAs/AlGaAs heterostructure. The devices have capacitances so small that the charging energy of adding an electron is much greater than the thermal energy at dilution refrigerator temperatures. The measurements, which explore how changing inter-dot coupling affects device conductance, are important for quantum dots used as "artificial atoms" or as "single-electron transistors" in larger arrays.

For single quantum dots, single-electron charging leads to dramatic conductance peaks. For arrays of two and three quantum dots, the conductance peaks each split into two (double dot) or three (triple dot) peaks as the inter-dot coupling increases. The splitting closely tracks the measured tunnel conductance and experimentally determines the interaction energy. Coupled double and triple dots with different gate capacitance show quasiperiodic beating. Monte Carlo simulations of a classical capacitive charging model qualitatively reproduce the observed structure, even though the underlying splitting mechanism is most likely quantum mechanical.

ACKNOWLEDGMENTS

I could not have completed this thesis without the advice and support of family, friends, and peers.

It's hard to imagine a better environment for doing physics than the one Bob Westervelt has created in Gordon McKay Laboratory. Bob's guidance is gentle but sure, his grasp of physics deep, and his research ideas always on the mark. He has put together a lab filled with great equipment and, much more importantly, great people. In particular, Charlie Marcus inspired virtually the entire first half of this thesis, while Michael Berry and Doug Mar patiently taught me the skills and concepts I needed for the second half. John Baskey, Scott Yang, Aram Adourian, and Jordan Katine all contributed their experimental expertise; Pete Hopkins, Andy Kahn, Alex Rimberg, Raj Seshadri, and Junmin Hu provided friendship; and Catherine Crouch, Mark Eriksson, Rex Beck, and Carol Livermore are ably carrying on. As my career moves forward, I can only hope to work with people as knowledgeable, creative, congenial, and generous as my colleagues in the Westervelt group have been.

A bonus of being in the Westervelt group is being down the hall from the Tinkham group. Jack Hergenrother and Dan Ralph gave me crucial experimental pointers; Lydia Sohn and Mark Itzler have been and will always be great friends; and the rest of the group graciously put up with my frequent theft of their CPU cycles and their Ithaco 1211.

Other Gordon McKay people who have contributed vitally to this thesis include Dave Carter, Steve Shepard, Clive Hayzeldon, and Yuan Lu. I would like also to thank Bill Chase and Jack Smith of Tower Hill School for first sparking my interest in physics, and Steve Boughn and Stuart Trugman for inspiring me when it flagged at Princeton.

By far the most important contributions to this thesis have come from my wife Jessica, my brother Ted and his wife Karen, and my parents Rod and Fran. Jessica shared with me the occasional triumphs and many setbacks of my research, always spurring me on with her advice and her love. Ted and Karen have been tremendously generous and supportive, particularly in this last very busy year. My gratitude to my parents is hard to put into words: everything that I have achieved was made possible by their lifelong love, inspiration, and sacrifice.

TABLE OF CONTENTS

INTRODUCTION

"Novel" is a dangerous word to put on paper: with time, the novel becomes commonplace. It is all the more dangerous when applied to computers. For decades, computing power as measured by almost any criterion has doubled every eighteen months [Keyes, 1993]; the supercomputer of the 1970s sits atop millions of home and office desks today. To a large extent, the gains have arisen from device miniaturization and concomitant improvements in circuit speed and complexity. More recently, additional performance advances have been shown to arise when large numbers of interconnected processors compute in parallel, even if each processor itself is not particularly fast.

However, the topics of this thesis—neural networks and nanoelectronics—represent not incremental improvements to existing technologies, but rather dramatically new approaches to computing. Neural networks are more than highly parallel computers, and nanoelectronic devices are more than extremely small transistors. Together, they challenge the very foundations of today's computer industry, the von Neumann approach to computing and its realization in CMOS VLSI (complementary metal-oxide semiconductor very large-scale integration).

One may understandably object that von Neumann architectures and CMOS integrated circuits do the job quite nicely and that exotic architectures and devices are not needed. Table 1, however, hints that trouble may already be appearing. The table summarizes a dilemma faced by Intel Corporation, today's dominant microprocessor manufacturer. In retooling from the 80486 to the Pentium processor generations, Intel found in late 1993 that market valuation of the Pentium's extra computing power did not nearly compensate for the extra technological difficulties in building it. Specifically, while income per device rose by

device	trans-sistors	fab steps	income/ device	total devices/ wafer	working devices/ wafer	income/ wafer
80486	< 1M	14	$385	187	84	$32,340
Pentium	> 3M	17	$985	47	5.5	$5,418

Table 1 Typical yield per 6" silicon wafer for Intel 80486 and Pentium microprocessors as of late 1993 [*The Economist*, 1993]. Market valuation of Pentium failed to compensate for low manufacturing yield.

less than a factor of three (from $385 to $985), typical working device yield per 6" wafer fell by more than a factor of 15 (from 84 to 5.5), leading to a sharp decline in income per wafer [*The Economist*, 1993]. Nanoelectronics and neural networks may dramatically increase the total and working devices per wafer, respectively—the first by increasing the device density, and the second by making devices more robust to fabrication errors.

Again one may object that, while trouble may be appearing, there are still a myriad ways to squeeze additional performance gains from existing technologies before scrapping them altogether. But one of the main motivations of this thesis is that, in the longer term, the ability to do this may not keep up with the desire to perform ever more complex computations. The von Neumann and silicon VLSI standards may need to be complemented with new architectures realized in new physical systems. Whether the new paradigms will involve neural networks or nanoelectronics is at this point uncertain. But it is worthwhile now, before the current technologies have lost their steam, to consider the advantages and limitations of truly novel approaches to computing.

NEURAL NETWORKS

In 1977, Marvin Minsky summarized the plight of artificial intelligence research as follows:

> Our first foray into artificial intelligence was a program that did a credible job of solving problems in college calculus. Armed with that success, we tackled high school algebra; we found, to our surprise, that it was much harder. Attempts at grade school arithmetic, involving the concept of number, etc., provide problems of current research interest. An exploration of the child's world of blocks proved insurmountable, except under the most rigidly constrained circumstances. It finally dawned on us that the overwhelming majority of what we call intelligence is developed by the end of the first year of life [Mead, 1989].

Nearly twenty years later, Minsky's observations remain relevant. While conventional computers can process numerical data at blinding speeds, tasks that even infants perform without effort—recognizing a parent's face, for example, or manipulating objects with their hands—prove much more difficult. The premise of artificial intelligence, that human reasoning can be reduced to a set of instructions to be executed in sequence by a computer, has not yet been borne out after decades of research.

In contrast, the premise of neural network research is that biological computation is a hardware problem, not a software problem. Neural networks are an attempt to capture the remarkable abilities of biological computational systems by mimicking their architectures. The human brain differs from a computer in several salient ways: it has an enormous number of processors (10^{11} neurons) that are highly connected (10^4 interconnections per neuron); it is fault tolerant; and perhaps most importantly, it learns and adapts based on experience without needing to be reprogrammed. Of course, there are countless other

differences: for example, how gates regulate currents in CMOS transistors as compared to how ion channels regulate currents across neuron membranes. A crucial assumption of neural network research is that these details do not matter. In this view, biological computation is a collective effect; and therefore, like other collective effects such as magnetism or superconductivity, it is insensitive to the details of how its elements operate and interact.

The first part of this thesis, then, is motivated by the observation that performance improvement is by itself unlikely to enable conventional computers to duplicate computations that biological brains routinely perform. The goal is to "reverse-engineer" biological computation, keeping only those architectural features deemed essential—namely, distributed processing by a large number of simple, highly interconnected elements. Biology thus serves as an inspiration but need not be copied in detail. The methods are those of nonlinear dynamics and the statistical mechanics of disordered systems. The results describe the theory of neural networks that perform a variety of pattern recognition and feature detection tasks and that are readily implementable in analog circuitry [Marcus, Waugh, and Westervelt, 1990, 1991; Waugh, Marcus, and Westervelt, 1990, 1991; Waugh and Westervelt, 1993a, 1993b].

NANOELECTRONICS

To the VLSI design engineer charged with shrinking circuits to ever smaller dimensions, the quantum-mechanical world that looms ahead looks daunting. The CMOS transistor, workhorse of today's digital circuits, operates in the more comfortable realm of classical physics: its characteristics can be understood by treating electrons as particles and using ensemble averaging. Shrink this transistor down enough, and concepts like tunneling, interference, energy levels, and discreteness of electron charge can adversely

affect its performance. Ensemble averaging no longer works, leading to large performance variations from one transistor to the next. Perhaps even worse, communication between transistors becomes ever slower, since the resistance of narrow wires can increase exponentially with length [Buot, 1993].

Nanoelectronics research aims to turn these quantum-mechanical liabilities into assets—that is, to make devices whose operation relies on tunneling or interference or discreteness of electron charge. For such devices, continued shrinking typically improves performance.

Computers built with nanoelectronic devices could operate in a variety of ways, any one of which would be nothing short of revolutionary given the decades-long dominance of silicon VLSI. The simplest and least radical proposal is to replace each CMOS transistor with an equivalent quantum device [Tucker, 1992]. The resulting machine would operate very much like existing computers, with bits represented by high or low voltages. More radical is the proposal to represent bits by the presence of a single electron at some location in a device [Averin and Likharev, 1991, 1992]. Such a computer would physically look quite different from today's computers, but its logical structure would still be the same. Perhaps the most radical proposals are those for fully quantum computers in which bits can be placed in superpositions of 0 and 1 and logical operations take place coherently [Lloyd, 1993]. Such a machine would be quite different from today's computers.

The second part of this thesis is motivated by the need for physical systems in which such computers might be realized. The goal is to understand how small arrays of GaAs quantum dots interact as a step toward building nanoelectronic circuitry. In fact, the devices studied have been proposed as single-electron memories [Averin and Likharev, 1991, 1992; Bock and Hartnagel, 1993; Yano *et al.*, 1993; Dresselhaus *et al.*, 1994]. The methods are those of experimental low-temperature condensed matter physics. The results show how single-electron charging affects the conductance of quantum dot arrays

and how the conductance changes as the coupling between dots is varied [Waugh *et al.*, 1994].

QUANTUM NEUROCOMPUTERS?

It is a good idea not to get too carried away in assessing ideas as speculative as neural networks or nanoelectronics. Enormous obstacles have already arisen and will continue to arise in translating these technologies into useful applications. The point of this thesis is not to tout either one as the wave of the future, but rather to use physics to understand their limitations as well as their advantages.

That being said, it is interesting to speculate on how a hypothetical computer incorporating neural networks *and* single-electron nanoelectronics might operate. Table 2 compares operating characteristics if such a machine were built using (i) today's technology, (ii) what currently appears to be the limit of nanolithography, and (iii) speculative self-assembling macromolecular structures [Averin and Likharev, 1991, 1992]. Cases (i) and (ii) assume fabrication using $Al/AlO_x/Al$ tunnel junctions, which are probably more easily integrated into large-scale circuitry than split-gate GaAs/AlGaAs heterostructure devices (though, as noted in Chapter 6, the latter provide more flexibility in experiments). While case (iii) is considerably more speculative, a number of recent proposals and even experiments have raised the possibility of using macromolecules for information processing [Oesterhelt, Brauchle, and Hampp, 1991; Drexler, 1992; Bradley, 1993; Birge, 1994].

For the three fabrication technologies, the table shows typical scales for junction size, capacitance, voltage, power dissipation, and switching time, as well as operating temperature and active device density per unit area. Even using today's fabrication technology, the device density is astoundingly large and is predicted to increase to well over a billion gates/cm^2. Other performance characteristics, however, are less

fab tech-nology	junction area (nm^2)	capa-citance (aF)	voltage (mV)	power (nW)	time (ps)	temp-erature (K)	device density ($gates/cm^2$)
(i)	30×30	30	2.5	0.01	30	0.3	3×10^8
(ii)	10×10	3	25	1	3	3	3×10^9
(iii)	3×3	0.3	250	100	0.3	30	$\sim 10^{10}$

Table 2 Estimated operating characteristics of hypothetical computer built using (i) today's technology, (ii) what currently appears to be the limit of nanolithography, and (iii) speculative self-assembling macromolecular structures [Averin and Likharev, 1991, 1992]. Power and time scales are calculated assuming junction resistance of 600 kΩ.

impressive. The operating temperature for today's technology, 0.3 K, requires He dilution refrigeration, placing it well beyond practical limits. The predicted increase to 30 K, within reach of simpler closed-cycle refrigerators, is more promising. Furthermore, large junction resistances lead to typical gate *RC* time constants of 30 ps, predicted to slip under 1 ps. Actual switching times would probably need to be longer to reduce errors, making the devices comparable to or slower than some currently available room-temperature devices.

For a number of reasons, the architecture of these hypothetical machines may incorporate at least some aspects of neural networks. Without the fault tolerance inherent in neural networks, fabricating billion-gate devices without defects may be virtually impossible. Fabrication defects aside, it may be extremely difficult even to design and program a billion-gate device without errors; a better route may be to let the device configure itself through neural network learning. Furthermore, impedance mismatch between junctions and transmission lines may introduce long communication delays,

favoring neural network-like cellular automaton architectures with mostly local interconnections. Finally, as noted in Table 2, power dissipation becomes an increasing concern as device dimensions shrink, requiring that only a fraction of gates be switched at a given time, as is typical in biological neural systems and easily implementable in neural networks.

THESIS OVERVIEW

This thesis is divided into two parts. Chapters 1 through 3 theoretically and numerically analyze the dynamics of analog neural networks; Chapters 4 through 6 present low-temperature experiments on small arrays of coupled quantum dots. The chapters are described in more detail below.

Chapter 1 introduces competitive neural networks, in which analog-valued neurons interact both through synaptic connections and through a local competitive mechanism. Global stability theorems derived from Liapunov functions provide general guidelines for network design and operation. The theorems state that with continuous-time updating, competitive networks converge only to fixed points, while with discrete-time, parallel updating, they converge to either fixed points or period-two limit cycles. A stability criterion guarantees that discrete-time networks converge only to fixed points when their cluster gains, which are related to the slopes of the neuron transfer functions, are sufficiently small. Finally, applications of competitive networks to graph partitioning and image processing are described. These applications illustrate the types of problems neural networks can solve and also show how competition can often significantly improve computational abilities.

Chapter 2 contains a statistical analysis of the attractors of competitive associative memories. The analysis leads to bifurcation diagrams for finite memory loading (meaning

that the ratio α of stored patterns to neurons approaches zero) and to analytical phase diagrams for extensive memory loading (meaning that α approaches a constant). The bifurcation diagrams show how memories evolve with neuron transfer function steepness; the phase diagrams show attractor types as a function of pattern storage fraction and neuron transfer function steepness. In addition, storage capacities are derived for extensive loading in the limit of infinite transfer function steepness. Numerical tests are presented that support the phase diagrams.

Chapter 3 presents analytical results for analog annealing, a technique for improving network performance. The results show that the number of fixed points in an analog associative memory neural network scales exponentially with network size. The scaling exponent decreases with neuron gain for a typical neuron transfer function, implying that reducing gain improves performance by eliminating unwanted fixed points. The calculation uses techniques previously developed to count solutions of the Thouless-Anderson-Palmer (TAP) equations for infinite-range Ising spin glasses and to count metastable states of neural networks with two-state neurons, and analogies between spin systems and associative memories are discussed. Numerical counts of fixed points in small networks agree well with the analytical calculations.

Chapters 4 and 5 provide background material for the experimental results of Chapter 6. Chapter 4 reviews basic properties of two-dimensional electron gases formed in GaAs heterostructures, discusses conductance quantization in quantum point contacts, and discusses single-electron charging in single and coupled quantum dots using semiclassical and quantum-mechanical models. Chapter 5 is a practical guide to nanoelectronic device fabrication and low-temperature measurements.

Chapter 6 discusses low-temperature tunneling measurements at zero magnetic field through double and triple quantum dots with adjustable inter-dot coupling, fabricated in a GaAs/AlGaAs heterostructure. The measurements explore how changing inter-dot

coupling affects device conductance, important for quantum dots used in large arrays as "artificial atoms" or as "single-electron transistors." For single quantum dots, single-electron charging leads to dramatic conductance peaks. For arrays of two and three quantum dots, the conductance peaks each split into two (double dot) or three (triple dot) peaks as the inter-dot coupling increases. The splitting closely tracks the measured tunnel conductance and experimentally determines the interaction energy. Quasiperiodic beating occurs in the conductance of coupled double and triple dots with mismatched gate capacitance. Monte Carlo simulations of a classical capacitive charging model for the devices qualitatively reproduce the observed structure, even though the underlying splitting mechanism is most likely quantum mechanical.

CHAPTER 1*

NEURAL NETWORK ARCHITECTURES AND DYNAMICS

1.1 INTRODUCTION: NEURAL NETWORKS

The last few decades have seen a growing trend toward parallel computer architectures, especially for high-performance machines. The main advantage of parallelism is the speed that arises from division of labor. By chopping repetitious calculations into smaller pieces and working on each with a separate processor, parallel computers often realize substantial performance gains over computers doing the same calculation serially. Another advantage is cost: as processor prices decline, stringing together existing processors becomes a more economically viable alternative to designing new ones.

At first glance, neural networks represent the extreme limit of parallelism. Neural networks typically consist of many highly interconnected processing elements, each of which has a simple assignment: to turn its output (an electrical current, for example) on or off depending on whether its input (a voltage) is positive or negative. Computation occurs when an initial system state (the question) evolves in time to some later state (the answer), which may be a dynamical attractor. Often each neuron represents a single bit of information.

Yet neural networks are more than highly parallel conventional computers. Speed is only one advantage that neural networks gain from parallelism. Others are the abilities to

---

*A version of this chapter has appeared as *Phys. Rev. E* **47**, 4524 (1993).

learn and compute simultaneously, adapting to changes in data in real time; to capture subtle high-order correlations in data; and to generalize, applying previously learned rules to new data [Forrest, 1990; Hammerstrom, 1993]. These abilities originate in neuron interactions. High connectivity and extensive feedback enable neural networks to compute collectively, so that neurons strongly influence each other in the process of making decisions. In conventional parallel computers, on the other hand, interactions between processors slow computation and are a nuisance to be avoided. The aim of neural network research is to understand how collective behavior can lead to useful computation.

The inspiration for neural networks, of course, comes from biology. Figure 1.1(a) schematically depicts a human cortical neuron (henceforth a "biological neuron"), which consists of a *cell body*, a number of treelike *dendrites*, and an *axon* with branches that terminate in *synapses* or connections to the dendrites of other neurons. A neuron *fires*, or sends a voltage pulse down its axon, when the electrical potential across its cell body membrane increases above a threshold. The pulse changes the electrical potentials of neighboring neurons through chemical processes occuring at each synapse. The potential of a neighboring neuron increases if the synapse is *excitatory* and decreases if it is *inhibitory*. Often it is assumed that stray capacitances lead to time integration of voltage pulses, making only the average firing rate important, not the details of pulse shape or timing. The extent to which biological neurons employ pulse coding is an open question [Bressloff and Taylor, 1992; Kruglyak and Bialek, 1993; Rieke, Warland, and Bialek, 1993].

An artificial neuron (henceforth simply a "neuron") appears in Fig. 1.1(b). Its behavior can be described by equations like

$$\frac{dx_i(t)}{dt} = -x_i(t) + F_i\big(h_i(t)\big) \quad \text{(continuous time)}, \tag{1.1}$$

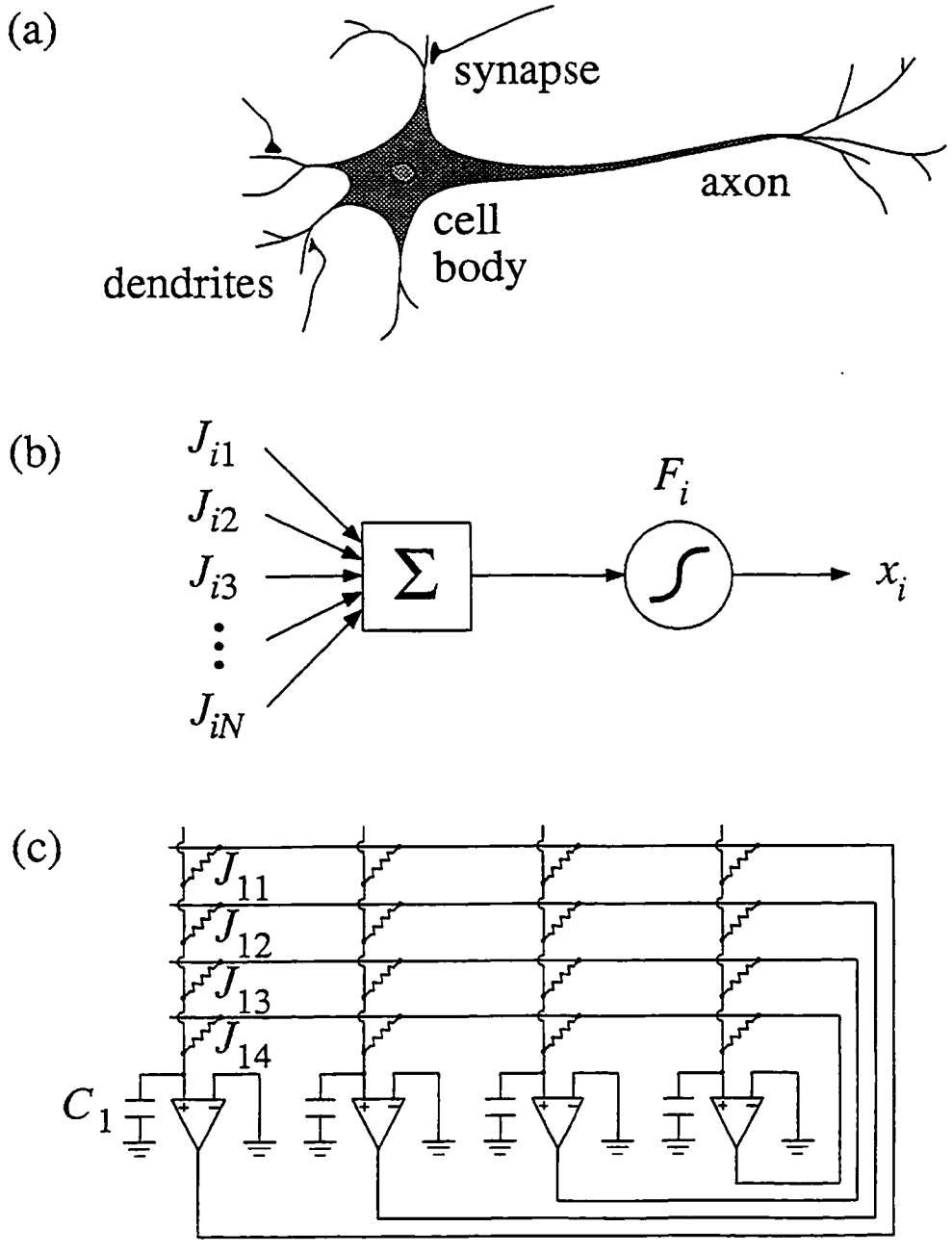


Fig. 1.1 (a) Human cortical neuron. Signals received through synapses alter cell body potential; voltage pulses traverse axon when potential exceeds threshold. (b) Model neuron. Output $x_i(t)$ is nonlinear function $F_i(z)$ of weighted sum of signals received from other neurons. (c) Four-amplifier electronic network with transfer characteristics $f_i(z)$ and input capacitances C_i. Conductance J_{ij} connects output of amplifer j to input of amplifier i.

$$x_i(t+1) = F_i\big(h_i(t)\big) \quad \text{(discrete time)}, \tag{1.2}$$

where

$$h_i(t) = \sum_{j=1}^{N} J_{ij}\, x_j(t). \tag{1.3}$$

The quantities $x_i(t)$ and $h_i(t)$ are the *output* and *input*, respectively, of neuron i. The matrix J_{ij} is called the *interconnection matrix*, and $F_i(z)$ is the *transfer function* of neuron i. Equations (1.1) through (1.3) capture many salient features of biological neurons: $x_i(t)$ corresponds to the average firing rate; $h_i(t)$ corresponds to the electrical potential across the cell body membrane; J_{ij} represents the synaptic interconnections between neurons; and $F_i(z)$ describes how the average firing rate depends on the membrane potential.

Figure 1.1(c) shows a realization of Eq. (1.1) in an analog electronic circuit consisting of nonlinear amplifiers connected by resistors. Applying Kirchoff's current law to the input of amplifier i yields

$$R_i C_i \frac{dV_i(t')}{dt} = -V_i(t') + R_i \sum_{j=1}^{N} J_{ij}\, f_j\big(V_j(t')\big), \tag{1.4}$$

where $V_i(t)$ is the input voltage for each nonlinear amplifier, C_i is the input capacitance, J_{ij} is the conductance between amplifiers i and j, and

$$R_i = \left(\sum_{j=1}^{N} J_{ij} \right)^{-1}. \tag{1.5}$$

Equations (1.1) and (1.4) are the same when the time constants R_iC_i are all equal, as can be seen by applying the transformation

$$V_i(t)/R_i = \sum_{j=1}^{N} J_{ij}\, x_j(t) \ , \qquad f_i(z) = F_i(z/R_i) \ , \qquad t' = R_iC_i\, t \tag{1.6}$$

to Eq. (1.1). It is easy to see how a clocked version of the circuit in Fig. 1.1(c) is described similarly by Eq. (1.2).

Much of neural network research boils down to the study of dynamical systems much like Eqs. (1.1) and (1.2). As the equations are quite general, there are still numerous choices in designing a network for a particular task. The design process consists of picking parameters like the interconnection matrix and transfer functions so that a chosen set of initial conditions evolves to a chosen set of dynamical attractors. The rest of this chapter is devoted to deriving design criteria for a specific class of networks having symmetric interconnection matrices, analog transfer functions, and parallel, deterministic updating rules. There are, of course, many other types of networks: interconnection matrices can be asymmetric, transfer functions can be digital, updating can be serial or sequential or stochastic. As will become apparent, the particular choices made here are meant to facilitate implementation of fast and robust networks in analog circuitry.

Specifically, this chapter focuses on the dynamics of two types of neural networks: *standard* networks like those of Eqs. (1.1) and (1.2), in which analog-valued neurons interact through synaptic connections only, and *competitive* networks, in which analog-valued neurons interact both through synaptic connections and through a local competitive mechanism. The networks are described in more detail in Sec. 1.2. Liapunov functions are used in Sec. 1.3 to derive global stability theorems that provide important guidelines for

network design and operation. (The same Liapunov functions are used in Chapters 2 and 3 to study the statistics of attractors in associative memories.) Finally, applications of both network types to graph partitioning and image processing are described in Sec. 1.4. These applications illustrate the types of problems neural networks can solve and also show how competition can often significantly improve computational abilities.

An implicit theme of the chapter is the close relationship between neural network structure and function: how well a network performs can depend on whether its architecture is well-suited to a particular task. Exactly what is meant by well-suited is one of the outstanding issues of neural network research. A common approach is first to write down an energy function for a given task and then to derive the network dynamics for which the energy function is a Liapunov function [Simic, 1990, 1991; Geiger and Yuille, 1991]. The drawback is that a new architecture must be derived for each application. This chapter takes the opposite approach, starting with a general network architecture and dynamics for competitive networks and deriving a Liapunov function. As is shown below, the competitive network architecture is well-suited to a wide variety of feature extraction and pattern classification tasks because it directly incorporates constraints involved in these tasks.

1.2 DYNAMICS OF ANALOG COMPETITIVE NETWORKS

This section introduces a network architecture in which analog-valued neurons interact not only through synaptic interconnections but also through a novel competitive mechanism. These *competitive analog networks* are related to the networks of Eqs. (1.1) and (1.2), which are referred to as *standard analog networks.* A comparison of standard and competitive analog networks is made in Fig. 1.2. As indicated in the figure, neurons

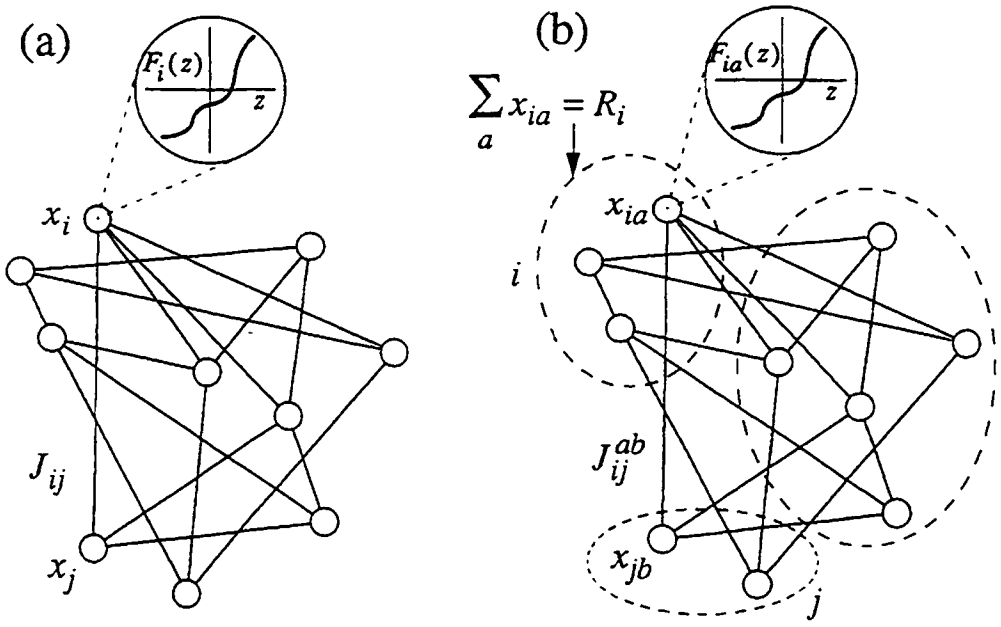


Fig. 1.2 Comparison of standard and competitive analog neural networks. Circles denote neurons, lines denote symmetric interconnections. (a) Schematic diagram of standard analog network with 10 neurons. Nonlinear input-output transfer function $F_i(z)$ of neuron i and interconnection J_{ij} between neurons i and j are shown. (b) Same network with competitive interactions, denoted by dashed ellipses. Nonlinear input-output transfer function $F_{ia}(z)$ of neuron a in cluster i and interconnection J_{ij}^{ab} between neuron a in cluster i and neuron b in cluster j are shown. Competition is enforced by requiring neuron outputs in cluster i to sum to constant R_i.

in both architectures have analog input-output transfer functions and communicate through synaptic interconnections. The important difference is that neurons in competitive networks are grouped into localized clusters, within which they compete through the constraint that their outputs sum to a constant at all times. Competition makes the output of a neuron depend on the inputs of *all* the neurons in its cluster, rather than just on its own input. As a result, clusters of neurons in competitive networks are capable of performing more complicated calculations—including the winner-take-all function, or more generally, the k-winner function—than are possible in standard analog networks.

The particular competitive mechanism studied here, which constrains the sum of neuron outputs in a cluster, has several attractive features. It requires only one connection for each neuron in a cluster; it is free of the stability problems often associated with competition [Feldman and Ballard, 1982]; and it can be implemented simply in a variety of physical systems by exploiting conservation laws [Pankove *et al.*, 1990; Perfetti, 1990; Anderson, 1990; Johnson and Jalaleddine, 1991]. Other competitive mechanisms involving all-to-all inhibitory interconnections, auxiliary neurons, or multi-neuron interactions can be substantially more cumbersome to implement [Feldman and Ballard, 1982; Lippmann, 1987; Kanter, 1988].

The rest of this section shows how to implement competition in clusters of analog neurons, describes a simple competitive electronic circuit, discusses how competitive clusters are connected to form networks, and draws an analogy between competing neurons and Potts spins.

1.2.1 Competitive clusters

Neurons in competitive networks are grouped into clusters of two or more (see Fig. 1.2(b)). A cluster i contains Q_i neurons labelled by a, $a = 1, \ldots, Q_i$. The neurons are

characterized by input-output transfer functions $F_{ia}(z)$ that may be different for each. All Q_i neurons are updated simultaneously according to a set of Q_i equations that map real-valued neuron inputs h_{ia} onto real-valued neuron outputs x_{ia}. These equations may be either the continuous-time differential equations

$$\frac{dx_{ia}(t)}{dt} = -x_{ia}(t) + F_{ia}\left(h_{ia}(t) + B_i(t) \right), \qquad a = 1, \ldots, Q_i, \tag{1.7}$$

or the discrete-time, parallel-update equations

$$x_{ia}(t+1) = F_{ia}\left(h_{ia}(t) + B_i(t) \right), \qquad a = 1, \ldots, Q_i. \tag{1.8}$$

These equations are similar to the update equations of standard analog networks, Eqs. (1.1) and (1.2). The important difference is that, in addition to inputs h_{ia}, neurons also experience a time-dependent bias $B_i(t)$ that is the same for each neuron in a cluster. The bias $B_i(t)$ is determined implicitly at time t by the requirement that the outputs of all neurons in cluster i sum to a constant R_i at the *same* time t in the continuous-time case,

$$\sum_{a=1}^{Q_i} x_{ia}(t) = R_i \,, \tag{1.9}$$

and at the *next* time step $t+1$ in the discrete-time case,

$$\sum_{a=1}^{Q_i} x_{ia}(t+1) = R_i \,. \tag{1.10}$$

Groups of neurons obeying Eqs. (1.7) through (1.10) are *competitive clusters*, and Eqs. (1.9) and (1.10) are *competitive constraints*.

Competition makes the output x_{ia} of neuron a depend not only on its own input $h_{ia}(t)$ but also on the inputs $h_{ib}(t)$, $b \neq a$, of the other neurons in cluster i. In doing so, it constrains the Q_i neuron outputs in cluster i to a $(Q_i - 1)$-dimensional space. (Thus a cluster of $Q_i = 2$ neurons is equivalent to a standard analog neuron, because its two neuron outputs, which are related by $x_{i1} = R_i - x_{i2}$, lie in a one-dimensional space.) In hardware applications the quantity R_i often is a limited resource for which the neurons in cluster i compete (see Sec. 1.2.2). While it is possible to set R_i to zero through the transformation

$$F_{ia}(z) \rightarrow F_{ia}(z) - R_{ia}, \qquad \sum_{a=1}^{Q_i} R_{ia} = R_i, \tag{1.11}$$

it is useful to allow it to be nonzero, since in hardware implementations its value may not be adjustable.

The transfer functions $F_{ia}(z)$ determine how the neurons in cluster i compete for the quantity R_i. Three restrictions are sufficient to ensure that the competitive constraints (1.9) or (1.10) have a single, unique solution for the bias $B_i(t)$: (i) all $F_{ia}(z)$ must be continuous; (ii) all $F_{ia}(z)$ must either increase or decrease monotonically (without loss of generality, all $F_{ia}(z)$ can be assumed to increase monotonically); and (iii) for all possible values of the neuron inputs h_{ia}, R_i must lie within the range of the function

$$S_i(z) \equiv \sum_{a=1}^{Q_i} F_{ia}(h_{ia} + z), \tag{1.12}$$

which, if (i) and (ii) are satisfied, is a continuous, monotonically increasing function over

all real numbers z. In addition, to ensure boundedness of the solutions to Eqs. (1.7) and (1.8), all $F_{ia}(z)$ must asymptotically increase less than linearly either for large negative values of their arguments, for large positive values of their arguments, or for both (see Sec. 1.3). These requirements are very general and are easily met by transfer functions that lead to useful competitive behavior.

As an example of competition, consider a cluster i that uses the discrete-time update equation (1.8) to compute an analog winner-take-all function of its inputs, meaning that the neuron with the largest input has the largest output while the other neuron outputs are suppressed. Suppose that $R_i = 1$, and suppose that each neuron a in the cluster has the same transfer function

$$F_{ia}(z) = \exp(\gamma z), \qquad a = 1, \ldots, Q_i, \tag{1.13}$$

where the parameter γ, the *neuron gain*, controls the transfer function slope. This transfer function is a natural choice in that it arises in the statistical mechanical treatment of the winner-take-all problem [Geiger and Yuille, 1991]. It is also similar to the threshold-linear functions used in the neocognitron [Fukushima, 1980, 1988] and in other networks with competitive or inhibitory behavior [Treves and Amit, 1988; Treves, 1990]. For this transfer function, the bias $B_i(t)$ can be expressed explicitly in terms of the neuron inputs $h_{ia}(t)$, so that the update equation (1.8) reads

$$x_{ia}(t+1) = \frac{\exp[\,\gamma h_{ia}(t)\,]}{\sum_{b=1}^{Q_i} \exp[\,\gamma h_{ib}(t)\,]}, \qquad a = 1, \ldots, Q_i \tag{1.14}$$

[for similar results, see Geiger and Yuille, 1991; Peterson and Söderberg, 1989; van den

Bout, 1990; Bridle and Cox, 1991]. Now consider how the neuron outputs $x_{ia}(t+1)$ vary with γ in the case that neuron 1 has the largest input $h_{i1}(t)$ at time t. In the limit $\gamma \to \infty$ of very steep transfer functions, $x_{i1}(t+1) \to 1$ for neuron 1, while $x_{ia}(t+1) \to 0$ for the other neurons $a > 1$ in the cluster. As γ decreases, $x_{i1}(t+1)$ decreases from 1 but remains the largest output in the cluster, while the other neuron outputs increase from 0. Finally, in the limit $\gamma \to 0$ of nearly flat transfer functions, all neuron outputs approach the same value $x_{ia}(t+1) \to 1/Q_i$. Thus, in this example, the competitive constraint (1.10) rations the quantity R_i among the Q_i neuron outputs in the cluster according the size of their inputs at each time step, with the neuron gain γ controlling how much is awarded to each neuron.

Competitive clusters are not limited to calculating the analog winner-take-all function; the broad class of possible transfer functions allows a great variety in cluster functionality. A cluster can be configured to calculate an analog k-winner function—meaning that the neurons with the k largest inputs have large outputs while the other neurons are suppressed—by setting the constant $R_i = k$ and giving each neuron the same sigmoidal or s-shaped transfer function $F(z) = \left[1 + \exp(-\gamma z)\right]^{-1}$. More complicated calculations are possible as well: for example, the number of winners can be made to depend on which neurons have the largest inputs by allowing some neurons to have exponential transfer functions and others to have sigmoids.

In standard analog networks, the neuron gain, defined as the steepest slope of a neuron transfer function, plays an important role. In competitive analog networks, the analogous quantity is the *cluster gain*, defined precisely in Sec. 1.3. The gain β_i of a competitive cluster i is a measure of the change in its neuron outputs in response to a change in their inputs. The cluster gain is thus related to the slopes of the neuron transfer functions; for

example, rescaling all transfer functions by $F_{ia}(z) \rightarrow F_{ia}(cz)$ changes the cluster gain by $\beta_i \rightarrow c\beta_i$.

1.2.2 Competition in an analog electronic circuit

An attractive feature of the competitive mechanism described above is that it can be easily implemented in a variety of physical systems by using conservation laws to enforce the competitive constraints (1.9) or (1.10). Competition can be implemented using current conservation in an electronic circuit and gain conservation in a laser resonator [Pankove *et al.*, 1990; Perfetti, 1990; Anderson, 1990; Johnson and Jalaleddine, 1991]. This subsection describes a simple analog electronic circuit that uses Kirchoff's current law to implement a Q-neuron analog winner-take-all cluster.

A schematic diagram of the circuit appears in Fig. 1.3. Each neuron a, $a = 1,\ldots,Q$, consists of an n-channel enhancement metal-oxide-semiconductor field-effect transistor (MOSFET) T_a. The input of neuron a is the gate voltage V_a of transistor T_a, and the output of neuron a is the current I_a flowing through transistor T_a. The sources of the transistors are connected to a current source I_0 and their drains to a voltage source V_0. If the transistors are operated in the sub-threshold regime, then

$$\begin{aligned} I_a &= I \exp(\gamma V_a)\left[\exp(-\gamma V) - \exp(-\gamma V_0) \right] \\ &\cong I \exp\left[\gamma(V_a - V)\right] \end{aligned} \tag{1.15}$$

[Sze, 1981; Mead, 1989]. In Eq. (1.15), I is a device-dependent parameter (typically 1 nA to 1 μA), V is the voltage of the common source line, V_0 is the power supply voltage, and $\gamma = e/k_B T \cong 40\,\mathrm{V}^{-1}$, where e is the electron charge, k_B is Boltzmann's constant, and

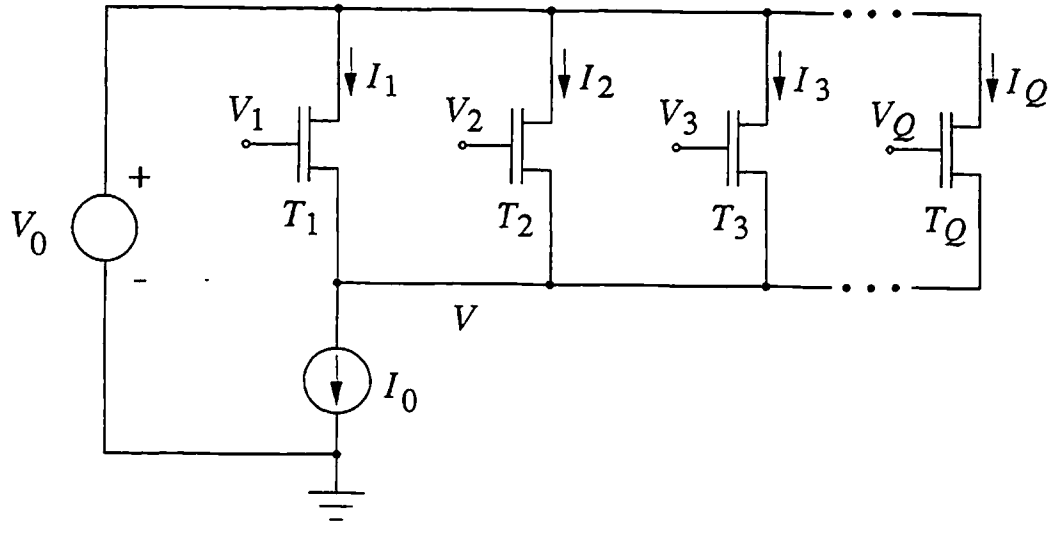


Fig. 1.3 Schematic of Q-neuron winner-take-all analog circuit. Each neuron a, $a = 1,\ldots,Q$, is an n-channel enhancement MOSFET T_a. Input of neuron a is gate voltage V_a of T_a; output of neuron a is current I_a flowing through T_a. Transistor sources are connected in parallel to current source I_0; transistor drains are connected in parallel to voltage source V_0. Kirchoff's current law constrains outputs I_a to sum to I_0.

T is the temperature. The approximation holds in Eq. (1.15) when V_0 is much larger than V. The voltage V of the common source line adjusts to make the sum of the transistor currents equal the current through the current source, in accordance with Kirchoff's current law:

$$\sum_{a=1}^{Q} I_a = I_0. \tag{1.16}$$

Comparing Eqs. (1.15) and (1.16) to Eqs. (1.7) through (1.10) shows that $-V$ plays the role of the bias term B_i, I_0 plays the role of the constant R_i, and Kirchoff's current law enforces the competitive constraint. Using Eq. (1.16) to eliminate V leads to

$$I_a \cong I_0 \frac{\exp(\gamma V_a)}{\sum_{b=1}^{Q} \exp(\gamma V_b)}, \tag{1.17}$$

which is identical to the analog winner-take-all transfer function (1.13).

To demonstrate that competition can be implemented simply and robustly in standard analog electronics, the circuit of Fig. 1.3 was constructed using Intersil VN86HF MOSFET transistors, a 15V voltage source, and a standard op amp-based 75mA current source. The transfer function for a typical transistor operating in the subthreshold regime is shown in Fig. 1.4(a). In Fig. 1.4(b), competitive behavior is shown in a cluster of $Q = 3$ transistors. The gate voltages and currents (as measured by the voltage drop across a series 100 Ω resistor) of each transistor are shown in the case that the gates of transistors 1 and 2 are held at $V_1 = V_2 = 6$ V while the gate of transistor 3 is ramped between $V_3 = 5$ V and 7 V. When $V_3 > 6$ V, transistor 3 has the maximum gate voltage and wins the

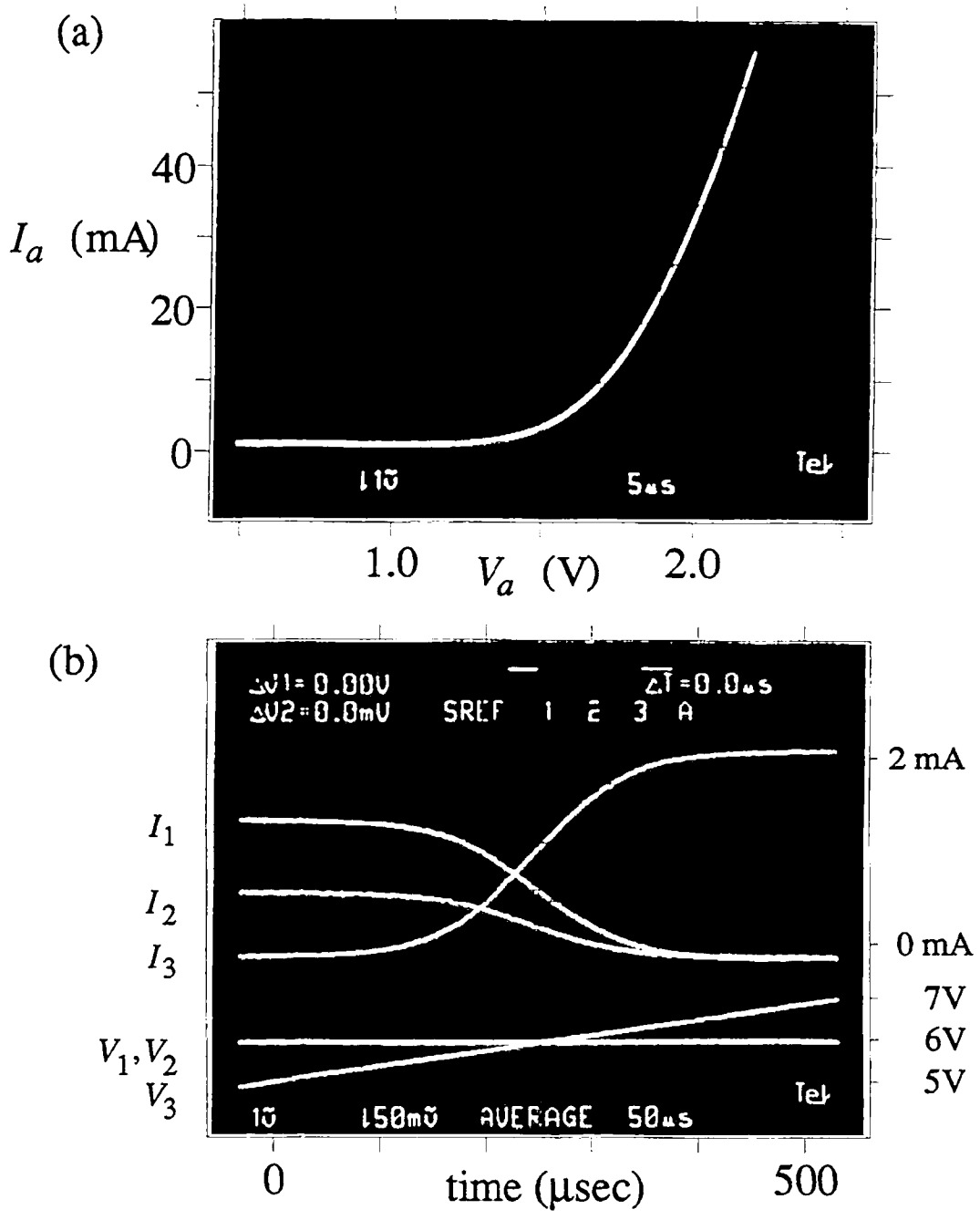


Fig. 1.4 Oscilloscope traces showing (a) transfer function of single analog electronic neuron; (b) competition in electronic circuit with $Q = 3$ neurons. Competition leads to sigmoidal response even though transfer function is exponential. In (b), input voltages (bottom) and output currents (top) of all three neurons are shown. Inputs 1 and 2 are held constant at 6 V while input 3 is ramped between 5 V and 7 V.

competition. When $V_3 < 6$ V, transistors 1 and 2 both have the maximum gate voltage; a mismatch in transistor characteristics causes transistor 2 to win the competition.

1.2.3 Competitive networks

Now consider networks of N interacting competitive clusters labelled by i, $i = 1, \ldots, N$. The clusters may contain different numbers Q_i of neurons, so that the total number N_{tot} of neurons is

$$N_{tot} = \sum_{i=1}^{N} Q_i . \tag{1.18}$$

The neurons have arbitrary transfer functions $F_{ia}(z)$ subject to the conditions stated above. An example of such a network is shown in Fig. 1.2(b).

The neurons are updated in parallel according to either the N_{tot} coupled nonlinear differential equations

$$\frac{dx_{ia}(t)}{dt} = -x_{ia}(t) + F_{ia}\left[\sum_{j=1}^{N}\sum_{b=1}^{Q_j} J_{ij}^{ab} x_{jb}(t) + B_i(t) + I_{ia}\right],$$

$$i = 1, \ldots, N, \quad a = 1, \ldots, Q_i, \tag{1.19}$$

or the N_{tot} discrete-time equations

$$x_{ia}(t+1) \;=\; F_{ia}\left[\sum_{j=1}^{N}\sum_{b=1}^{Q_j} J_{ij}^{ab}x_{jb}(t) \;+\; B_i(t) + \; I_{ia}\;\right],$$

$$i = 1, \ldots, N, \quad a = 1, \ldots, Q_i, \tag{1.20}$$

with parallel update. The interconnection matrix J_{ij}^{ab}, which couples the output of neuron b in cluster j to the input of neuron a in cluster i, is assumed to be real-valued and to satisfy the symmetry condition

$$J_{ij}^{ab} = J_{ji}^{ba}. \tag{1.21}$$

The N biases $B_i(t)$ enforce the competitive constraints (1.9) or (1.10) for each cluster. The N_{tot} external biases I_{ia} are time-independent and may differ for each cluster i and neuron a. Networks that obey Eqs. (1.20) or (1.21) are referred to as *competitive networks*.

Because competition restricts the Q_i neuron outputs in cluster i to a space of dimension $Q_i - 1$, the dynamical systems (1.20) and (1.21) lie in a space of dimension $N_{tot} - N$. Thus, of the $N_1 = N_{tot}(N_{tot}+1)/2$ independent elements of the symmetric matrix J_{ij}^{ab}, only $N_2 = (N_{tot} - N)(N_{tot} - N + 1)/2$ are necessary, and it is possible to impose up to $N_1 - N_2 = N(2N_{tot} - N + 1)/2$ constraints on the interconnections without altering a network's trajectories in state space [Nadal and Rau, 1991]. One useful choice of constraints is to require that, for $N(N+1)/2$ constants $J_{ij} = J_{ji}$,

$$\frac{1}{Q_i}\sum_{a=1}^{Q_i} J_{ij}^{ab} \;=\; \frac{1}{Q_j}\sum_{b=1}^{Q_j} J_{ij}^{ab} \;=\; J_{ij}. \tag{1.22}$$

Equation (1.22) constrains the eigenvectors of the interconnection matrix either to lie in the same $(N_{tot} - N)$-dimensional space as the dynamical systems (1.20) and (1.21) or else to be orthogonal to this space. (The eigenvalues and eigenvectors of the interconnection matrix, which has four indices, are found by flattening it into an $N_{tot} \times N_{tot}$ matrix with two indices.)

A simple counting argument shows that Eq. (1.22) imposes exactly $N_1 - N_2$ constraints on the interconnections. For the $Q_i Q_j$ interconnections J_{ij}^{ab} between two clusters $i \neq j$, Eq. (1.22) imposes $Q_i + Q_j - 1$ constraints (the number of independent column and row sums of a $Q_i \times Q_j$ matrix). For the $Q_i(Q_i+1)/2$ independent interconnections J_{ii}^{ab} within a cluster i, Eq. (1.22) imposes Q_i constraints (the number of independent column and row sums of a $Q_i \times Q_i$ symmetric matrix). Thus the total number of constraints is

$$\sum_{i=1}^{N} \sum_{j>i}^{N} \left(Q_i + Q_j - 1\right) + \sum_{i=1}^{N} Q_i = N_1 - N_2 . \tag{1.23}$$

An arbitrary, symmetric interconnection matrix J_{ij}^{ab} can be made to satisfy Eq. (1.22) by applying the symmetry-preserving transformation

$$J_{ij}^{ab} \rightarrow J_{ij}^{ab} + J_{ij} - \frac{1}{Q_i} \sum_{r=1}^{Q_i} J_{ij}^{rb} - \frac{1}{Q_j} \sum_{s=1}^{Q_j} J_{ij}^{as} + \frac{1}{Q_i Q_j} \sum_{r=1}^{Q_i} \sum_{s=1}^{Q_j} J_{ij}^{rs} . \tag{1.24}$$

This transformation leaves the network dynamics unchanged if in addition each neuron transfer function $F_{ia}(z)$ is shifted horizontally by an amount that depends on i and a as

derived below. Applying (1.24) transforms the neuron inputs $h_{ia}(t)$ by

$$h_{ia}(t) = \sum_{j=1}^{N} \sum_{b=1}^{Q_j} J_{ij}^{ab} x_{jb}(t) + I_{ia}$$

$$\rightarrow \sum_{j=1}^{N} \sum_{b=1}^{Q_j} \left[J_{ij}^{ab} - \frac{1}{Q_i} \sum_{r=1}^{Q_i} J_{ij}^{rb} \right] \left[x_{jb}(t) - \frac{R_j}{Q_j} \right] + \sum_{j=1}^{N} J_{ij} R_j + I_{ia}. \qquad (1.25)$$

Of the four new terms appearing on the right-hand side of Eq. (1.25) as a result of the transformation, three are independent of a and so contribute the same bias to the input of each neuron in cluster i, which is cancelled by the bias $B_i(t)$. The other term does depend on a, and to cancel it the transfer function $F_{ia}(z)$ must be shifted:

$$F_{ia}(z) \rightarrow F_{ia}\left(z + \sum_{j=1}^{N} \sum_{b=1}^{Q_j} J_{ij}^{ab} \frac{R_j}{Q_j} \right). \qquad (1.26)$$

The freedom to choose the constants J_{ij} can be useful in hardware applications, since a well-chosen set of J_{ij} can, for example, reduce the number of interconnections or make the interconnections all positive. Henceforth it is assumed that the transformation (1.24) has been carried out with arbitrary values of J_{ij}.

1.2.4 Competition and Potts networks

When competitive clusters are configured as analog winner-take-all units, they can be viewed as a generalization of Potts spins, just as standard analog neurons are a generalization of Ising spins. A Potts spin [for a review, see Wu, 1982] has Q real-valued inputs and an output that takes on one of Q possible discrete values; at each time step, the

output state is chosen to correspond with the largest input. The analog winner-take-all competitive cluster, Eq. (1.14), reduces to a Potts spin in the limit $\gamma \to \infty$. Moreover, for finite γ, the function on the right-hand side of Eq. (1.14) describes the thermally averaged state of a Potts spin at temperature $T = 1/\gamma$, just as the hyperbolic tangent function describes the thermally averaged state of an Ising spin.

However, the analogy between gain in deterministic, analog systems and temperature in stochastic, discrete-state systems is not exact. As for Ising systems [Thouless *et al.*, 1977], the mean-field treatment of finite-temperature Potts systems yields a term known as the *reaction field* acting on each spin [Lage and Nunes da Silva, 1984; Gross *et al.*, 1984]. By design, no reaction field appears in the update equations for competitive networks [for further discussion, see Marcus *et al.*, 1990; Fukai and Shiino, 1990; Shiino and Fukai, 1990; Takayama and Nemoto, 1990; Nishimura *et al.*, 1990].

The computational abilities of Potts networks have been studied by a number of authors [Kanter, 1988; Bollé *et al.*, 1989, 1991, 1992a, 1992b, 1992c; Nadal and Rau, 1991; Ferrari *et al.*, 1992; Vogt and Zippelius, 1992; Shim *et al.*, 1992]. Competitive clusters are considerably more general than Potts spins because they are analog rather than digital and because they can calculate other functions of their inputs besides the winner-take-all function.

1.3 GLOBAL STABILITY ANALYSIS

This section uses a Liapunov function approach to analyze the dynamics of competitive networks. It is shown that, for symmetric interconnection matrices and the broad class of transfer functions described in Sec. 1.2, (i) the attractors of competitive networks with continuous-time updating are fixed points; (ii) the attractors of competitive networks with

discrete-time, parallel updating are either fixed points or period-two limit cycles; and (iii) period-two limit cycles are eliminated from discrete-time networks, leaving only fixed-point attractors, when the cluster gains are sufficiently reduced as described below. The analysis of discrete-time, parallel-update networks follows Marcus and Westervelt, 1989b.

1.3.1 Continuous-time updating

Continuous-time competitive networks, whose time evolution is described by Eq. (1.19), are shown to have only fixed-point attractors by constructing the function

$$L(t) = -\frac{1}{2}\sum_{i,j=1}^{N}\sum_{a=1}^{Q_i}\sum_{b=1}^{Q_j} J_{ij}^{ab} x_{ia}(t) x_{jb}(t) \; - \sum_{i=1}^{N}\sum_{a=1}^{Q_i} I_{ia} x_{ia}(t) \; + \; \sum_{i=1}^{N}\sum_{a=1}^{Q_i} G_{ia}(x_{ia}(t)), \tag{1.26}$$

where $G_{ia}(x)$ is the integral of the transfer function inverse:

$$G_{ia}(x) \; \equiv \; \int_{x_0}^{x} F_{ia}^{-1}(z)\, dz. \tag{1.27}$$

The quantity x_0 is an arbitrary constant. The function $L(t)$ and similar functions have been used by a number of authors to study the dynamics and stability of analog networks [Cohen and Grossberg, 1983; Hopfield, 1984; Golden, 1986; Marcus and Westervelt, 1989; Fogelman Soulie *et al.*, 1989]. Here $L(t)$ is proven to be a Liapunov function for continuous-time competitive networks by showing that (i) the derivative of $L(t)$ with respect to time is always less than or equal to zero, and (ii) $L(t)$ is bounded below. Using the symmetry of the interconnection matrix, the time derivative is

$$\frac{dL(t)}{dt} = \sum_{i=1}^{N} \sum_{a=1}^{Q_i} \frac{dx_{ia}(t)}{dt} \left[-\sum_{j=1}^{N} \sum_{b=1}^{Q_j} J_{ij}^{ab} x_{jb}(t) - I_{ia} + G'_{ia}(x_{ia}(t)) \right], \tag{1.28}$$

where $G'_{ia}(x) = F_{ia}^{-1}(x)$ is the derivative of $G_{ia}(x)$. The first two terms in square brackets can be rewritten using Eq. (1.19), with the result

$$\frac{dL(t)}{dt} = \sum_{i=1}^{N} \sum_{a=1}^{Q_i} \frac{dx_{ia}(t)}{dt} \left[-F_{ia}^{-1}\left(x_{ia}(t) + \frac{dx_{ia}(t)}{dt} \right) + F_{ia}^{-1}(x_{ia}(t)) + B_i(t) \right]. \tag{1.29}$$

The sum over the third term in square brackets in Eq. (1.29) vanishes, since

$$\sum_{a=1}^{Q_i} \frac{dx_{ia}(t)}{dt} = \frac{d}{dt} \sum_{a=1}^{Q_i} x_{ia}(t) = \frac{dR_i}{dt} = 0. \tag{1.30}$$

Equation (1.29) may therefore be written

$$\frac{dL(t)}{dt} = \sum_{i=1}^{N} \sum_{a=1}^{Q_i} \frac{dx_{ia}(t)}{dt} \left[-F_{ia}^{-1}\left(x_{ia}(t) + \frac{dx_{ia}(t)}{dt} \right) + F_{ia}^{-1}(x_{ia}(t)) \right]. \tag{1.31}$$

As long as each transfer function increases monotonically, the quantity in square brackets in Eq. (1.31) always has the opposite sign of $dx_{ia}(t)/dt$. Thus

$$\frac{dL(t)}{dt} \leq 0, \tag{1.32}$$

with equality holding only when $dx_{ia}(t)/dt = 0$ for all clusters i and neurons a, implying

that the network has reached a fixed-point attractor. Furthermore, because all transfer functions in a cluster asymptotically increase less than linearly either for large negative values of their arguments or for large positive values of their arguments, or for both, the third sum in Eq. (1.26) increases more than quadratically when the neuron outputs are large in magnitude. The third sum therefore dominates the other two sums in $L(t)$, which are quadratic and linear in the neuron outputs, causing $L(t)$ to be bounded below. The result (1.32) and the boundedness of $L(t)$ imply that $L(t)$ is a Liapunov function of continuous-time competitive networks: the networks seek out the local minima of $L(t)$ as they evolve in time. Continuous-time competitive networks therefore have only fixed-point attractors.

1.3.2 Discrete-time, parallel updating

The proof that competitive networks with discrete-time, parallel updating—whose time evolution is given by Eq. (1.20)—have both fixed points and period-two limit cycles uses the function

$$E(t) = -\sum_{i,j=1}^{N} \sum_{a=1}^{Q_i} \sum_{b=1}^{Q_j} J_{ij}^{ab} x_{ia}(t) x_{jb}(t-1) - \sum_{i=1}^{N} \sum_{a=1}^{Q_i} I_{ia}\left[x_{ia}(t) + x_{ia}(t-1)\right]$$

$$+ \sum_{i=1}^{N} \sum_{a=1}^{Q_i} \left[G_{ia}(x_{ia}(t)) + G_{ia}(x_{ia}(t-1))\right], \tag{1.33}$$

where $G_{ia}(x)$ is defined as in Eq. (1.27) and the time t is now discrete. $E(t)$ is proven to be a Liapunov function for discrete-time, parallel-update competitive networks by showing (i) that the change in $E(t)$ between successive time steps, $\Delta E(t) \equiv E(t+1) - E(t)$, is a nonincreasing function of time, and (ii) that $E(t)$ is bounded below. Using the update equations (1.20) and the symmetry condition (1.21), $\Delta E(t)$ can be written as

$$\Delta E(t) = -\sum_{i=1}^{N}\sum_{a=1}^{Q_i} F_{ia}^{-1}(x_{ia}(t+1))\Delta_2 x_{ia}(t) - \sum_{i=1}^{N} B_i(t)\sum_{a=1}^{Q_i}\Delta_2 x_{ia}(t)$$

$$+\sum_{i=1}^{N}\sum_{a=1}^{Q_i}\left[G_{ia}(x_{ia}(t+1)) - G_{ia}(x_{ia}(t-1))\right] \qquad (1.34)$$

where $\Delta_2 x_{ia}(t) \equiv x_{ia}(t+1) - x_{ia}(t-1)$ is the change in $x_{ia}(t)$ between two time steps. The second term in Eq. (1.34) is identically zero, since

$$\sum_{a=1}^{Q_i}\Delta_2 x_{ia}(t) = \sum_{a=1}^{Q_i} x_{ia}(t+1) - \sum_{a=1}^{Q_i} x_{ia}(t-1) = R_i - R_i = 0. \qquad (1.35)$$

Thus Eq. (1.34) becomes

$$\Delta E(t) = -\sum_{i=1}^{N}\sum_{a=1}^{Q_i} F_{ia}^{-1}(x_{ia}(t+1))\Delta_2 x_{ia}(t) + \sum_{i=1}^{N}\sum_{a=1}^{Q_i}\left[G_{ia}(x_{ia}(t+1)) - G_{ia}(x_{ia}(t-1))\right]. \qquad (1.36)$$

An upper bound for the contribution from each cluster to the last term in Eq. (1.36) is

$$\sum_{a=1}^{Q_i}\left[G_{ia}(x_{ia}(t+1)) - G_{ia}(x_{ia}(t-1))\right] \le \sum_{a=1}^{Q_i} G'_{ia}(x_{ia}(t+1))\Delta_2 x_{ia}(t), \qquad (1.37)$$

where $G'_{ia}(x)$ is the derivative of $G_{ia}(x)$. Equation (1.37), which states that the Q_i-dimensional surface $\sum_a G_{ia}(x_{ia})$ lies everywhere on or above its tangent planes, holds because the neuron transfer functions increase monotonically. Equation (1.37) is illustrated

in Fig. 1.5 for a cluster of $Q_i = 2$ neurons with transfer functions $F_{i1}(z) = F_{i2}(z) = \exp(\gamma z)$ and with $R_i = 1$. Figure 1.5(a) depicts the surface $G_{i1} + G_{i2}$. The competitive constraint $x_{i1} + x_{i2} = 1$ restricts the possible values of $G_{i1} + G_{i2}$ to the dashed curve lying on the surface. Also shown is the tangent plane and a tangent paraboloid (to be used below) at a particular point (x_{i1}, x_{i2}) such that $x_{i1} + x_{i2} = 1$. Figure 1.5(b) is a detailed view of the tangent plane and the dashed curve; the left-hand side of Eq. (29) appears as the quantity A and the right-hand side as the quantity B.

Combining Eqs. (1.36) and (1.37) and using the result $G'_{ia}(x) = F_{ia}^{-1}(x)$ leads to

$$\Delta E(t) \leq 0, \tag{1.38}$$

where equality holds when $\Delta_2 x_{ia}(t) = 0$ for all i and a, which implies that the network has reached either a fixed-point or a period-two attractor. The result (1.38) and the fact that $E(t)$, like $L(t)$, is bounded below, together prove that $E(t)$ is a Liapunov function of competitive networks with discrete-time, parallel updating, and that all attractors of these networks are either fixed points or period-two limit cycles.

Period-two limit cycle attractors can be eliminated from discrete-time, parallel-update competitive networks by reducing the cluster gains sufficiently. The proof shows that the function $L(t)$ of Eq. (1.26), now applied to discrete-time, parallel-update competitive networks, is a Liapunov function when all cluster gains are sufficiently small. Using the update equation and the symmetry of the interconnection matrix, the change in $L(t)$ between successive time steps, $\Delta L(t) \equiv L(t+1) - L(t)$, can be written

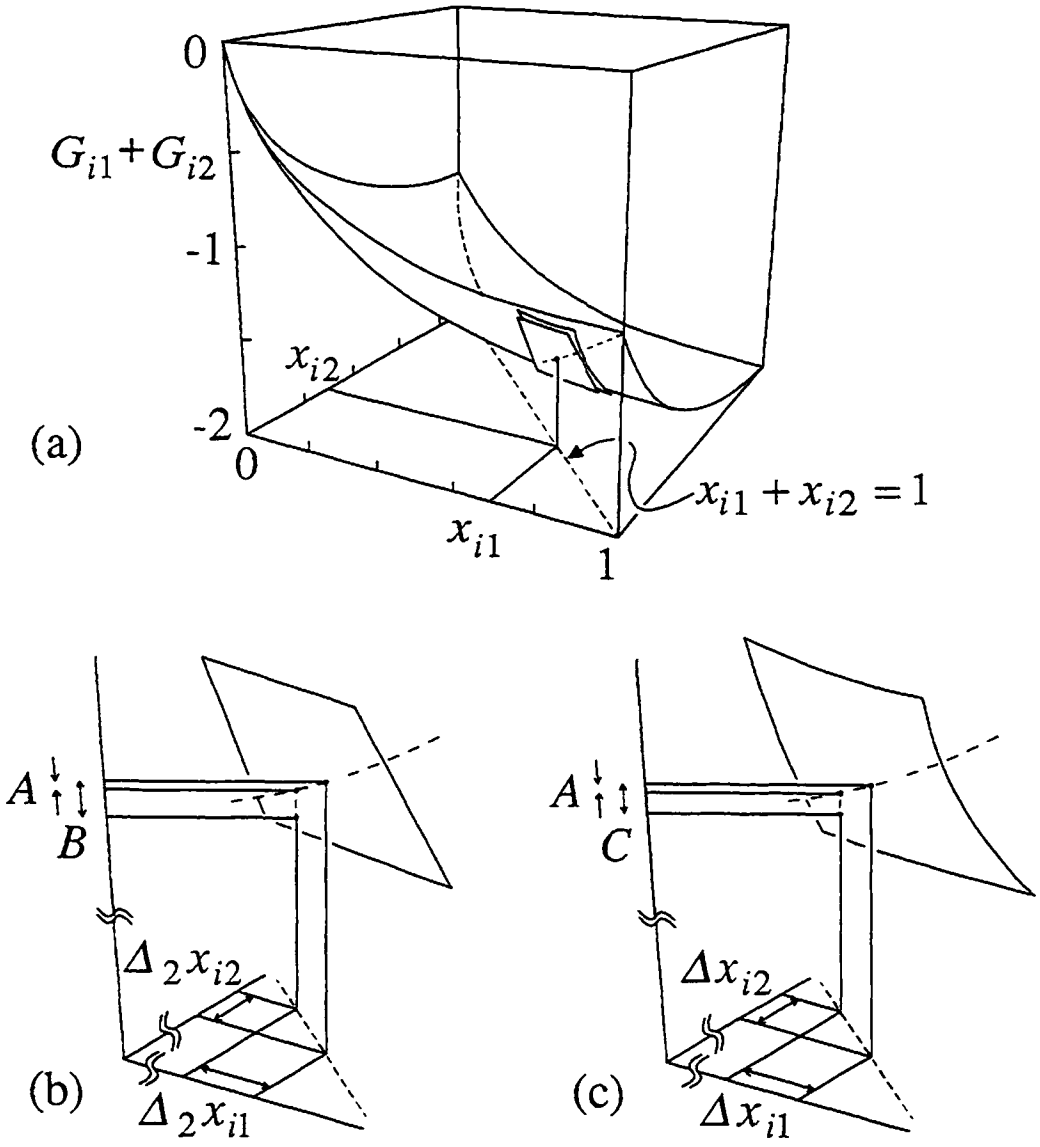


Fig. 1.5 Illustration of inequalities (1.37) and (1.41) for analog winner-take-all cluster i of $Q_i = 2$ neurons with transfer functions $F_{i1}(z) = F_{i2}(z) = \exp(z)$ and $R_i = 1$. (a) Surface $G_{i1}(x_{i1}) + G_{i2}(x_{i2})$ vs. neuron outputs x_{i1} and x_{i2}. Competition constrains system to dashed curve lying on surface. Also shown are tangent plane and paraboloid at (x_{i1}, x_{i2}). Parabola curvature $1/\beta_i$ equals greatest curvature of dashed curve; β_i is cluster gain. (b) Detail of tangent plane; Eq. (1.37) is represented by $A \le B$. (c) Detail of tangent parabola; Eq. (1.41) is represented by $A \le C$.

$$\Delta L(t) = -\frac{1}{2}\sum_{i,j=1}^{N}\sum_{a=1}^{Q_i}\sum_{b=1}^{Q_j} J_{ij}^{ab}\,\Delta x_{ia}(t)\,\Delta x_{jb}(t) - \sum_{i=1}^{N}\sum_{a=1}^{Q_i} F_{ia}^{-1}\big(x_{ia}(t+1)\big)\,\Delta x_{ia}(t)$$

$$- \sum_{i=1}^{N} B_i(t)\sum_{a=1}^{Q_i}\Delta x_{ia}(t) + \sum_{i=1}^{N}\sum_{a=1}^{Q_i}\big[G_{ia}\big(x_{ia}(t+1)\big) - G_{ia}\big(x_{ia}(t)\big)\big], \qquad (1.39)$$

where $\Delta x_{ia}(t) = x_{ia}(t+1) - x_{ia}(t)$ is the change in x_{ia} between consecutive time steps. As in Eq. (1.35), the third term on the right-hand side of (1.39) is zero, so that

$$\Delta L(t) = -\frac{1}{2}\sum_{i,j=1}^{N}\sum_{a=1}^{Q_i}\sum_{b=1}^{Q_j} J_{ij}^{ab}\,\Delta x_{ia}(t)\,\Delta x_{jb}(t) - \sum_{i=1}^{N}\sum_{a=1}^{Q_i} F_{ia}^{-1}\big(x_{ia}(t+1)\big)\,\Delta x_{ia}(t)$$

$$+ \sum_{i=1}^{N}\sum_{a=1}^{Q_i}\big[G_{ia}\big(x_{ia}(t+1)\big) - G_{ia}\big(x_{ia}(t)\big)\big]. \qquad (1.40)$$

An inequality relating the last term in Eq. (1.40) to $\Delta x_{ia}(t)$—similar to (1.37) but including a term quadratic in $\Delta x_{ia}(t)$—can be constructed:

$$\sum_{a=1}^{Q_i}\big[G_{ia}\big(x_{ia}(t+1)\big) - G_{ia}\big(x_{ia}(t)\big)\big] \le \sum_{a=1}^{Q_i} G'_{ia}\big(x_{ia}(t+1)\big)\Delta x_{ia}(t) - \frac{1}{2\beta_i}\sum_{a=1}^{Q_i}\big(\Delta x_{ia}(t)\big)^2. \qquad (1.41)$$

The quantity β_i is the gain of cluster i, defined as the smallest number such that the inequality

$$\frac{1}{\beta_i}\sum_{a=1}^{Q_i}(\Delta x_{ia}(t))^2 \le \sum_{a=1}^{Q_i} G_{ia}''(x_{ia})(\Delta x_{ia}(t))^2, \tag{1.42}$$

where $G_{ia}''(x)$ is the second derivative of $G_{ia}(x)$, is satisfied for any possible values of $x_{ia}(t)$ and $\Delta x_{ia}(t)$ subject to the competitive constraint (1.10). Note that, if there were no constraints on $x_{ia}(t)$ and $\Delta x_{ia}(t)$, β_i would equal the steepest slope among all the transfer functions in cluster i. However, because competition constrains the sums of $x_{ia}(t)$ and $\Delta x_{ia}(t)$ in each cluster to equal R_i and zero respectively, β_i takes on a value less than or equal to the steepest slope. This value is given by

$$\frac{1}{\beta_i} = \min_{\{z_a,\varepsilon_a\}} \sum_{a=1}^{Q_i} \varepsilon_a^2 \left(\frac{dF_{ia}^{-1}(z)}{dz}\right)_{z=z_a}. \tag{1.43}$$

The minimization in (1.43) is over the dummy variables z_a and ε_a, $a = 1,\ldots,Q_i$, subject to the constraints

$$\sum_{a=1}^{Q_i} z_a = R_i, \qquad \sum_{a=1}^{Q_i} \varepsilon_a = 0, \qquad \sum_{a=1}^{Q_i} \varepsilon_a^2 = 1, \tag{1.44}$$

and to the restriction that each z_a lie in the range of the transfer function $F_{ia}(z)$. In cases for which the minimization (1.43) is difficult to carry out, a useful upper bound for β_i is the steepest slope of any of the transfer functions $F_{ia}(z)$, $a = 1,\ldots,Q_i$, over the range to which the competitive constraint (1.10) restricts them. For clusters in which $R_i = 1$ and all neurons have the exponential transfer function of Eq. (1.13), Eqs. (1.43) and (1.44) give the result $\beta = \gamma/2$ independent of Q_i.

The inequality (1.41) and the geometrical meaning of the cluster gain can be understood by referring again to Fig. 1.5. The tangent paraboloid in Fig. 1.5(a) has curvature $1/\beta_i$. The minimization procedure of Eqs. (1.43) and (1.44) sets β_i equal to the greatest curvature of the dashed curve lying on the surface $G_{i1} + G_{i2}$ (recall that competition constrains the system to the dashed curve). Thus the dashed curve lies everywhere on or above the paraboloid when the paraboloid is tangent to it. Figure 1.5(c) is a detailed view of the tangent paraboloid and the dashed curve; the left-hand side of Eq. (1.41) appears as the quantity A and the right-hand side as the quantity C. For any cluster i, β_i is the greatest curvature of the $(Q_i - 1)$-dimensional region of the surface $\sum_a G_{ia}(x_{ia})$ to which the neurons of the cluster are restricted. The vector z_a and the unit vector ε_a produced by the minimization (1.43) indicate, respectively, the location and the direction of the curvature β_i on the surface.

Equations (1.40) and (1.41) and the result $G_i'(x) = F_i^{-1}(x)$ lead to

$$\Delta L(t) \;\le\; -\frac{1}{2}\sum_{i,j=1}^{N}\sum_{a=1}^{Q_i}\sum_{b=1}^{Q_j}\left(J_{ij}^{ab} + \delta_{ij}\delta_{ab}\frac{1}{\beta_i}\right)\Delta x_{ia}(t)\Delta x_{jb}(t), \tag{1.45}$$

where $\delta_{ij} = 1$ for $i = j$ and 0 otherwise. If the matrix $M_{ij}^{ab} = \left(J_{ij}^{ab} + \delta_{ij}\delta_{ab}\beta_i^{-1}\right)$ is positive definite, then the inequality

$$\Delta L(t) \;\le\; 0 \tag{1.46}$$

is satisfied. Equality holds in (1.46) only when $\Delta x_{ia}(t) = 0$, implying that the network has reached a fixed-point attractor. The requirement that M_{ij}^{ab} be positive definite leads to the stability criterion

$$\frac{1}{\beta_i} > -\lambda_{min}, \quad i = 1,\ldots,N, \tag{1.47}$$

where λ_{min} is the smallest or most negative eigenvalue of the interconnection matrix J_{ij}^{ab}, and β_i is the gain of cluster i. Equation (1.46) and the boundedness of $L(t)$ prove that, when the cluster gains satisfy (1.47), $L(t)$ is a Liapunov function of discrete-time, parallel-update competitive networks and the only attractors are fixed points. Thus networks with interconnection matrices for which $\lambda_{min} \geq 0$ do not have period-two limit cycles, while those with interconnection matrices for which $\lambda_{min} < 0$ can have period-two limit cycles if any one of the cluster gains exceeds $1/|\lambda_{min}|$.

The stability results of this section provide guidelines for designing competitive networks that compute by relaxing to fixed-point attractors. Specifically, continuous-time networks are guaranteed to converge only to fixed points for any cluster gain values, while discrete-time, parallel-update networks converge only to fixed points when the cluster gains are chosen to satisfy the stability criterion (1.47).

1.4 OPTIMIZATION AND IMAGE PROCESSING APPLICATIONS

So far this chapter has presented general results for competitive networks (valid also for standard networks when $Q = 2$), subject only to the restrictions of interconnection symmetry and transfer function monotonicity of Sec. 1.2. This section describes how competition can be used to extract features and classify patterns. In these applications, each neuron in a cluster represents one feature or pattern, and competition among them determines which features or patterns are present. Localized competitive interactions have been used to detect elementary image features in image-processing networks such as the

neocognitron [Fukushima, 1980, 1988], and competition appears in other networks that classify patterns, learn regularities in inputs, and perform vector quantization and optimization [Grossberg, 1987; Peterson and Söderberg, 1989; van den Bout, 1990; Rose, Gurewitz, and Fox, 1990]. The two particular applications considered in this section are graph partitioning and image processing. The main points are that competitive networks perform these tasks well in comparison to standard networks, and that the stability criterion (1.47) is useful in designing networks to have only fixed-point attractors.

1.4.1 Graph partitioning

Graph partitioning [Fu and Anderson, 1986] is a classic optimization problem, arising for example in placing logic circuits on different chips and chips on different boards when a computer is designed [Kirkpatrick, Gelatt, and Vecchi, 1983]. Ideally, logic circuits (chips) should be distributed more or less uniformly among the chips (boards) to minimize crowding; but care should also be taken to minimize the circuit paths that travel between packages, since these can reduce speed and increase power consumption. The application of graph partitioning to computer design is illustrated in Fig. 1.6(a).

In abstract terms, graph partitioning involves dividing a set of N arbitrarily interconnected nodes (a *graph*) into Q subsets (or *subgraphs*) such that (i) each subset contains N/Q nodes and (ii) the number of links between subsets (the *cutset*) is minimized. What makes the problem difficult is that these two goals are at odds. Conflicting goals are the essence of optimization problems; usually they lead to *frustration*, meaning that none of the goals can be satisfactorily attained, and to a large number of solutions that are nearly equally good [Hopfield and Tank, 1985, 1986; Basharan, Fu, and Anderson, 1986; Fu and Anderson, 1986; Burgess and Moore, 1989]. These issues are the subject of Chapter 3.

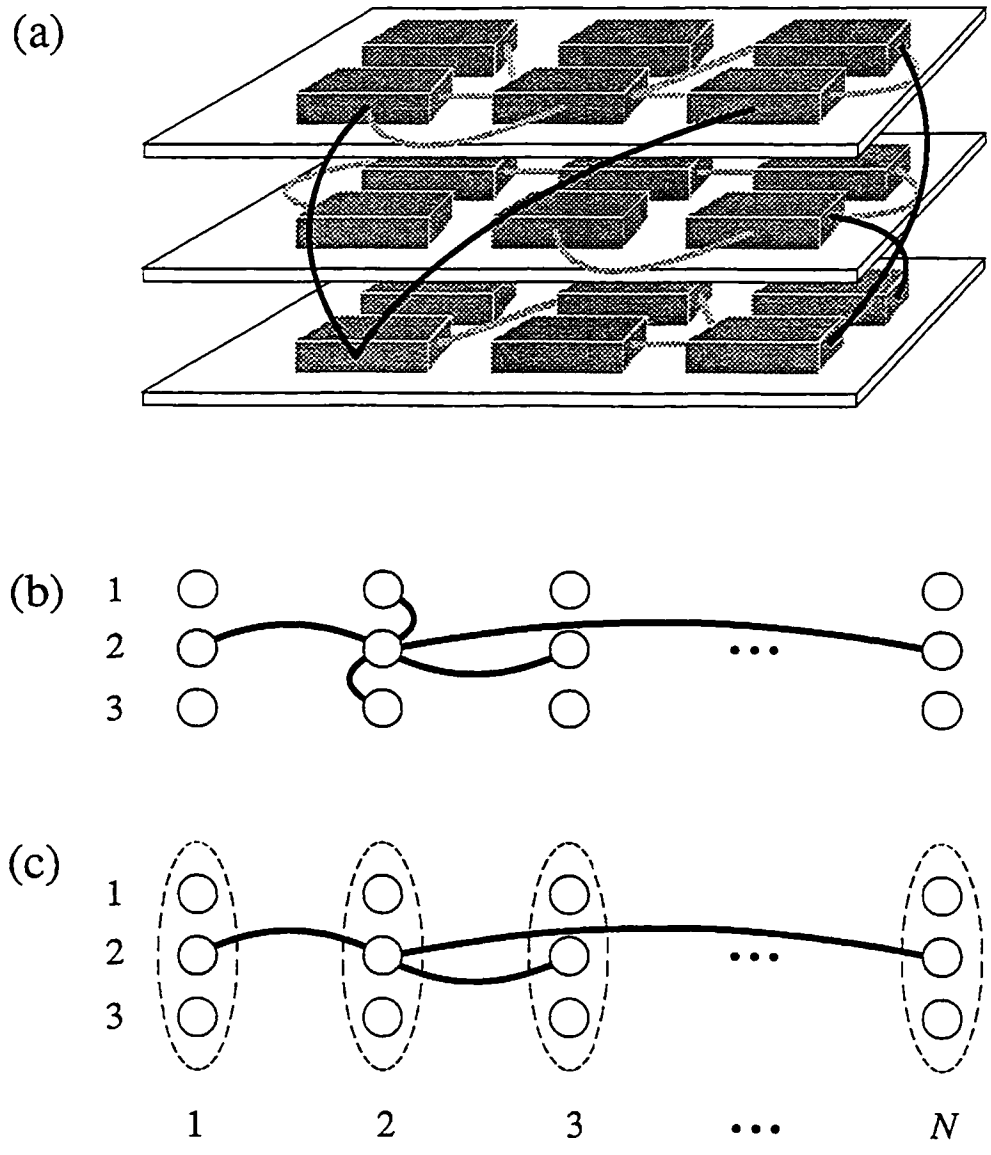


Fig. 1.6 (a) Example of graph partitioning problem in which $N = 18$ logic chips with connectivity matrix T_{ij} must be assigned to $Q = 3$ circuit boards. Chip interconnections forming cutset are black; other interconnections are gray. (b) Standard and (c) competitive analog networks that perform graph partitioning for $Q = 3$. Both networks contain NQ neurons; positive output of neuron in column i and row a indicates that node i is assigned to subgraph a.

The rest of this subsection compares the performance of standard and competitive analog neural networks that perform graph partitioning. As depicted schematically in Fig. 1.6(b) and (c), both networks have a total of NQ neurons. A positive output x_{ia} for neuron in column i and row a means a network has assigned node i to subgraph a. In a valid solution, then, only one neuron should have positive output in each column. In standard networks, this restriction is implemented using *lateral inhibition* [Dowling, 1987; Mead, 1989; Marcus, Waugh, and Westervelt, 1991], or strong negative interconnections among neurons in each column; competition implements it in competitive networks. The difference is that lateral inhibition imposes a *soft* constraint that can be violated, whereas competition imposes a *hard* constraint that cannot [Simic, 1990, 1991].

Numerical results, shown in Fig. 1.7, indicate that rigid constraint enforcement improves performance significantly. The figure compares two performance measures, *cutset size* and *error in subgraph size,* as a function of the number N of nodes for standard and competitive networks performing graph partitioning with $Q = 3$, 4, and 5. Both performance measures are given as fractions. Cutset size compares the total number of links between nodes in different subgraphs to the number when nodes are randomly assigned to subgraphs. Error in subgraph size compares the deviation in the number of nodes in each subgraph from N/Q to the deviation in the worst possible case, in which all nodes are assigned to the same subgraph. The competitive network nearly always beats the standard network in both performance measures for a given N and Q and always scales better than the standard network as N increases.

The standard networks for graph partitioning obey the update rule

$$x_{ia}(t+1) = F\left[\sum_{j=1}^{N}\sum_{b=1}^{Q} J_{ij}^{ab} x_{jb}(t)\right] \tag{1.48}$$

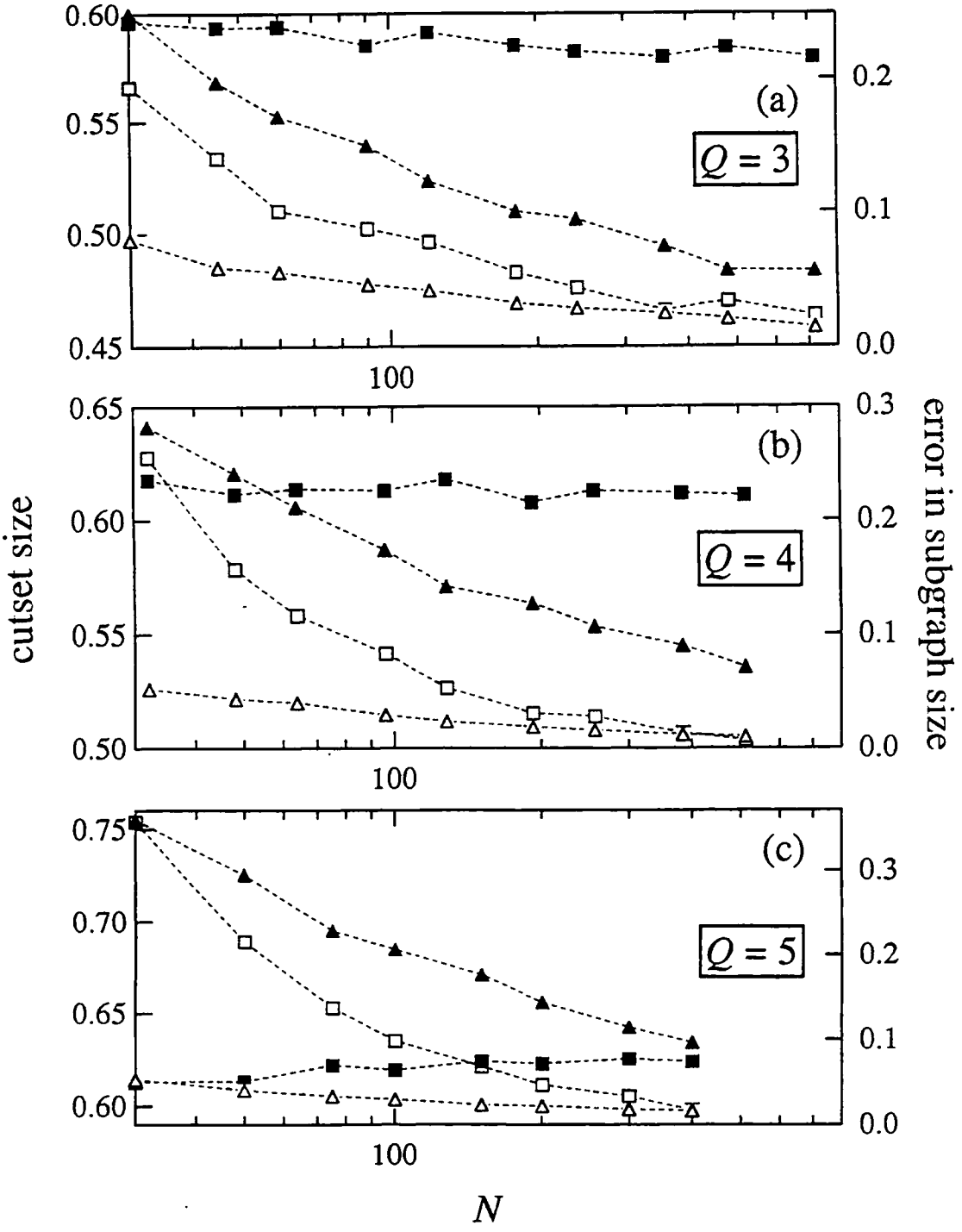


Fig. 1.7 Two performance measures, cutset size and error in subgraph size, as functions of N for standard and competitive graph partitioning networks for (a) $Q = 3$ and (b) $Q = 4$. Competition improves performance by enforcing constraints strictly.

with $F(z) = \tanh(\gamma z)$. In Eq. (1.48), i and a index each neuron's column and row, respectively. The interconnection matrix is

$$J_{ij}^{ab} = \frac{1}{2Np(1-p)+(Q-1)|\alpha|+\sigma} \begin{cases} \sigma, & i = j \text{ and } a = b \\ \alpha, & i = j \text{ and } a \neq b \\ T_{ij} - p, & i \neq j \text{ and } a = b \\ 0, & \text{otherwise.} \end{cases} \tag{1.49}$$

In Eq. (1.49), α implements lateral inhibition, σ is a self-coupling, the connectivity matrix T_{ij} is defined as

$$T_{ij} = \begin{cases} 1, & \text{nodes } i \text{ and } j \text{ connected} \\ 0, & \text{nodes } i \text{ and } j \text{ not connected,} \end{cases} \tag{1.50}$$

and p is the average connectivity of T_{ij}, divided by N. As shown in Fig. 1.6(b), the interconnection matrix (1.49) couples a neuron only to neurons in its same row or column. The prefactor in Eq. (1.49) serves merely to normalize the neuron inputs to approximately unit size. For Fig. 1.7, $\alpha = -2$, $\sigma = 0$, $\gamma = 0.5$, and $Np = 6$. The parameters γ and σ are chosen to avoid oscillation: the cluster gain is $\beta = \gamma = 0.5$ (found by writing the transfer function in terms of two competing neurons), and the most negative eigenvalue is $\hat{\lambda}_{min} \cong -1$ (found numerically), so that the stability criterion (1.47) is not violated.

The update rule for the competitive network is very similar to Eq. (1.48), the important difference being that competition replaces the lateral inhibition term α. The update rule is

$$x_{ia}(t+1) = F\left[\sum_{j=1}^{N} \sum_{b=1}^{Q} J_{ij}^{ab} x_{jb}(t) + B_i(t)\right] \tag{1.51}$$

where $F(z) = \exp(\gamma z) - 1/(Q-1)$, $B_i(t)$ is the competitive bias term, Eq. (1.10), and again i and a index each neuron's column and row, respectively. The interconnection matrix is

$$J_{ij}^{ab} = \frac{1}{[Q/(Q-1)]Np(1-p)+\sigma} \begin{cases} \sigma, & i = j \text{ and } a = b \\ T_{ij} - p, & i \neq j \text{ and } a = b \\ 0, & \text{otherwise,} \end{cases} \tag{1.52}$$

with T_{ij} as defined in Eq. (1.50). For Fig. 1.7 $\sigma = 1$, $\gamma = 5$, and $Np = 6$. Again, γ and σ are chosen to avoid oscillation in accordance with the stability criterion (1.47). Similar networks have been applied to graph partitioning previously [Peterson and Söderberg, 1989; van den Bout, 1990].

1.4.2 Feature detection

This subsection describes competitive networks that detect and classify localized features in a visual scene. Two-dimensional architectures [Noest, 1989, 1990; Coolen, 1990; Marcus, Waugh, and Westervelt, 1991] are considered because they allow efficient mapping of an image onto a network and because they are well-suited for implementation in VLSI circuits. The networks consist of competitive clusters, each acting as a local feature classifier, laid out in a two-dimensional square array [Waugh and Westervelt, 1993a]. The particular application is to detect two features—a triangle and a star—in binary or gray-scale images. To give the networks translation invariance, the features are each divided into four smaller fragments. A cluster's task is to detect these fragments based both on the pixel values in a local image region and on decisions made by neighboring clusters, so that a network "stitches together" whole triangles and stars from the fragments.

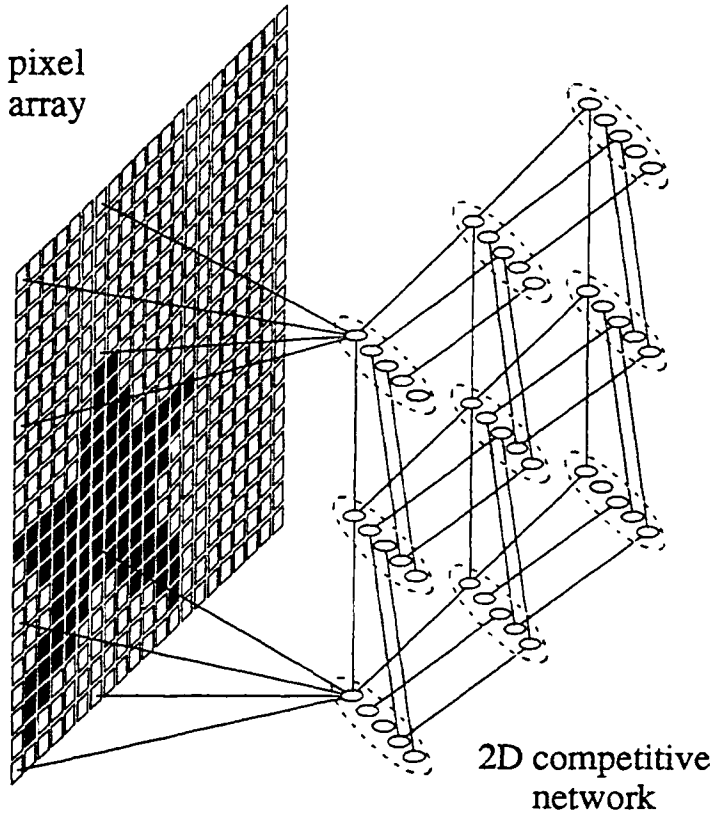


Fig 1.8 Two-dimensional competitive feature detector architecture. Open circles denote neurons; lines denote symmetric interconnections; dashed ellipses indicate competition. Clusters receive inputs from localized 7×7 regions of two-dimensional pixel array and communicate with neighboring clusters on a square lattice. For clarity only some of pixel array inputs are shown.

The network architecture is shown in Fig. 1.8. A competitive cluster is located at each of the N vertices of a square lattice. The clusters each have the same number $Q_i = Q$ of neurons, the same neuron transfer functions $F(z) = \exp(\gamma z) - 1/(Q-1)$, and the same value $R_i = 0$ of the constant appearing in the competitive constraint. Thus each cluster has the same cluster gain $\beta = \gamma/2$. Figure 1.9 shows how each neuron in a cluster represents either a pattern fragment or a uniform background and how excitatory interconnections between neighboring clusters enforce relations among the fragments. For simplicity, Figs. 1.8 and 1.9 show the case $Q = 5$ for four star fragments and the uniform background; the results of Fig. 1.8 are for the case $Q = 9$, which also includes four triangle fragments. Each neuron also receives as input the dot product of (i) the particular pattern fragment it represents and (ii) a localized region of a pixel array. All neurons in a cluster receive this input from the same pixel region, which in Figs. 1.8 and 1.9 consists of a 7×7 square of pixels. Each neuron can also have a self-coupling connection. Similar architectures have been used in other image processors [Fukushima, 1980, 1988].

Typical performance of these networks appears in Fig. 1.10. The figure shows a very noisy image (top left) and a less noisy image in which the star and triangle patterns are occluded (bottom left). Initial conditions are generated from these images, and the update equations are iterated until a fixed point is reached. The fixed points are then converted back to an image by displaying the pattern fragments corresponding to winning neurons. The resulting images appear on the top and bottom right of Fig. 1.10. Clearly, the networks can successfully extract triangle and star patterns from noisy images with occlusion.

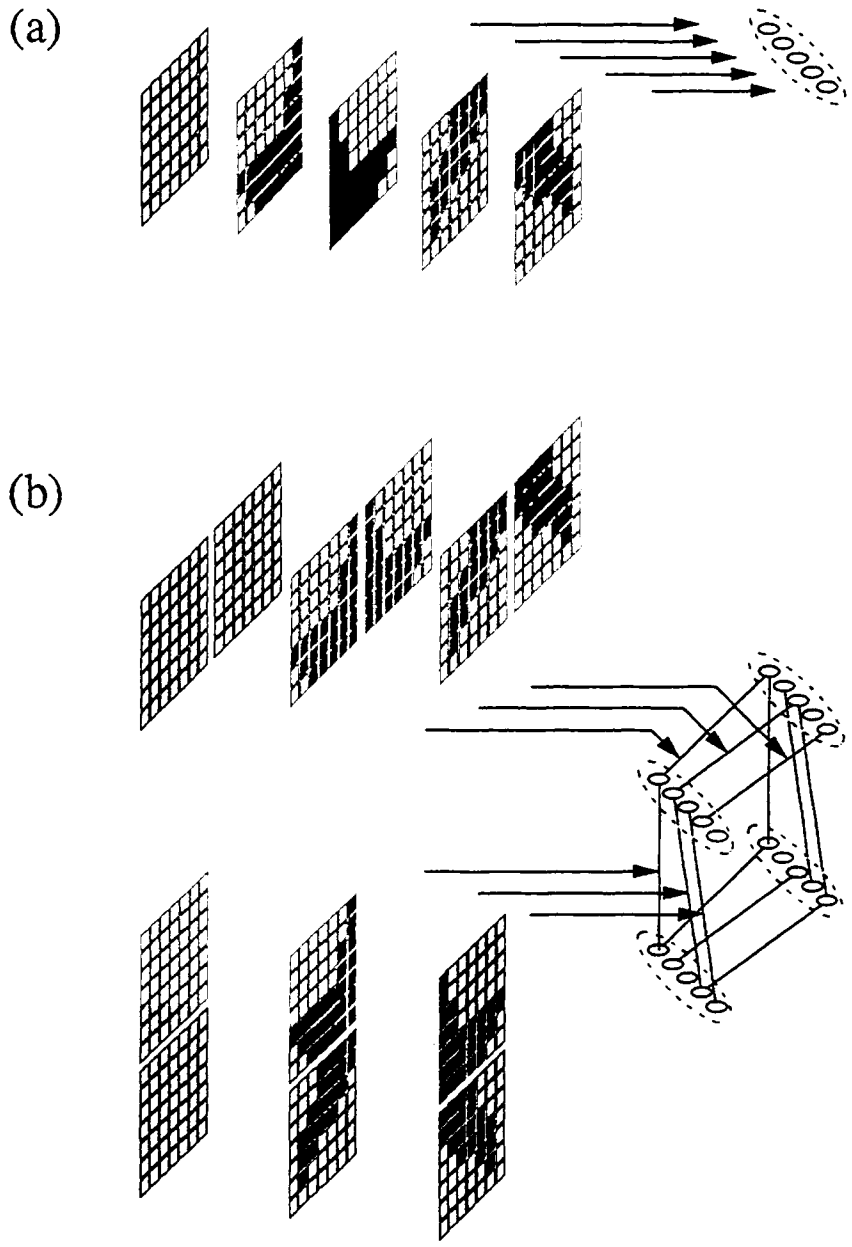


Fig. 1.9 Pattern representation in feature detector. (a) Each neuron in a cluster represents pattern fragment or uniform background. (b) Interconnections between neurons enforce relationships among pattern fragments so that networks "stitch together" whole patterns from fragments with translation invariance.

The translation-invariant interconnection matrix is

$$J_{ij}^{ab} = \begin{cases} \sigma\delta_{ab}/(|\sigma|+4) & \text{if } i=j \\ 1/(|\sigma|+2) & \text{for connected neurons in neighboring clusters} \\ 0 & \text{otherwise.} \end{cases} \tag{1.53}$$

This matrix is easy to diagonalize, allowing the stability criterion (1.47) to be applied. The eigenvalue spectrum is calculated using a standard technique for the classical phonon dispersion spectrum of a crystal lattice with a basis [Madelung, 1981; Marcus, Waugh, and Westervelt, 1991]. Assuming periodic boundary conditions, the eigenvalue equation

$$\lambda x_{ia} = \sum_{j=1}^{N} \sum_{b=1}^{Q} J_{ij}^{ab} x_{jb} \tag{1.54}$$

is expanded in terms of periodic functions $x_{ia} = c_a \exp(i\mathbf{k}\cdot\mathbf{R}_i)$, where $\mathbf{R}_i$ is the vector position of cluster i and $\mathbf{k}$ is a reciprocal lattice vector characterizing a particular eigenmode, to give

$$\lambda c_a = \sum_{b=1}^{Q} J_{ab}(\mathbf{k}) c, \qquad J_{ab}(\mathbf{k}) = \sum_{j=1}^{N} J_{0j}^{ab} \exp\left[i\mathbf{k}\cdot\left(\mathbf{R}_j - \mathbf{R}_0\right)\right]. \tag{1.55}$$

In Eq. (1.55), site 0 is any lattice site. With translation invariance, diagonalizing the $NQ \times NQ$ matrix J_{ij}^{ab} thus reduces to the simpler task of diagonalizing the $Q \times Q$ matrix J_{ab}. For the interconnections (1.53), the 9×9 matrix J_{ab} is already block diagonal since neurons communicate only with neighboring neurons representing fragments of the same

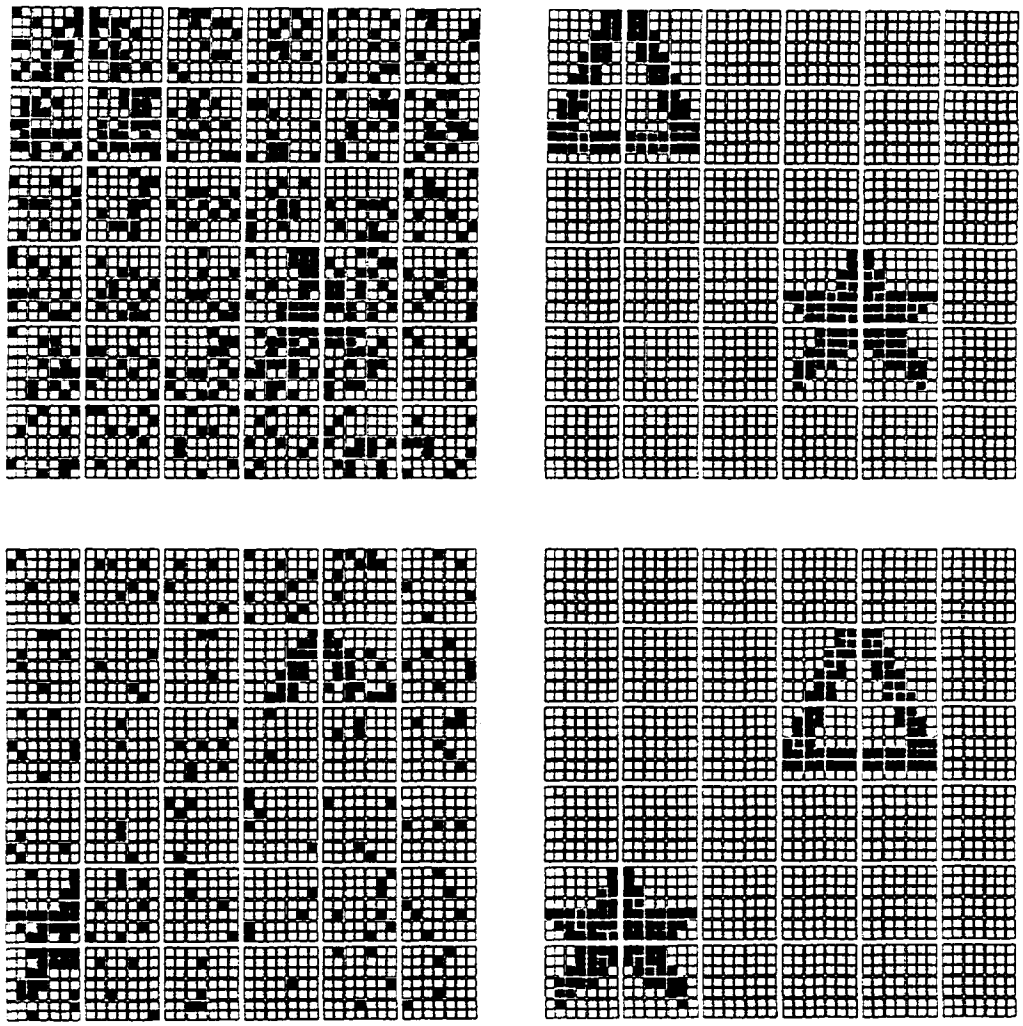


Fig. 1.10 Initial conditions and final attractors of two-dimensional feature detector for (top) very noisy image and (bottom) noisy image with occlusion.

pattern (triangle, star, or uniform background). The procedure yields $\lambda_{min} = (\sigma - 4)/(|\sigma| + 4)$, so that the stability criterion (1.47) for two-dimensional competitive feature detectors is

$$\frac{1}{\gamma} > \frac{4 - \sigma}{2|\sigma| + 8}. \tag{1.56}$$

Two-dimensional feature detectors can also be built using standard neurons. One way is to replace the hard constraint of competition with the soft constraint of lateral inhibition in the network just described, as is done for graph partitioning in Sec. 1.4.1. Since each neuron still represents a pattern fragment, the resulting network is similar to the competitive one, with poorer performance due to soft constraint implementation. Another way is to let each neuron represent one image pixel and to store features using a local version of the *pseudoinverse rule* for associative memories [Personnaz *et al.*, 1985, 1986a, 1986b]. This architecture is quite different from the competitive feature detector. Simulations show, however, that their performance is comparable. The pseudoinverse network has the advantage of being more robust; since each neuron represents a pixel rather than an entire fragment, errors or malfunctions have smaller consequences. On the other hand, a disadvantage of the pseudoinverse network is that it can get stuck on spurious attractors that do not correspond to any of the stored patterns. The spurious attractor problem in associative memories is the subject of Chapter 3.

1.5 SUMMARY

This chapter has introduced a network architecture in which analog neurons compete within localized clusters. Competition occurs when the sum of neuron outputs in a cluster is constrained to equal a constant. Competitive neurons can perform calculations on their inputs—such as winner-take-all or k-winner—that are not possible with standard neurons, making them particularly well-suited for tasks such as feature extraction and pattern classification.

Global stability analysis proves that, with continuous-time updating, competitive networks converge only to fixed points. With discrete-time, parallel updating, they converge to either fixed points or period-two limit cycles, and a stability criterion, Eq. (1.47), guarantees convergence to fixed points when the cluster gain is sufficiently reduced. The cluster gain is a measure of the transfer function steepness for all neurons in a competitive cluster.

Graph partitioning and feature detection are examples of tasks that competitive networks perform well. For both, constraints among neuron outputs are essential for a valid solution. Networks using competition to impose hard constraints show significant performance improvements over standard networks using lateral inhibition to impose soft constraints.

CHAPTER 2*

PHASE DIAGRAMS FOR ANALOG ASSOCIATIVE MEMORIES

2.1 INTRODUCTION: ASSOCIATIVE MEMORIES

Associative memories have long been the proving ground for new neural network ideas [Anderson, 1968; Amari, 1972; Kohonen, 1974; Little, 1974; Hopfield, 1982; Amit, Gutfreund, and Sompolinsky, 1985a,b, 1987], much as the hydrogen atom was essential to the development of quantum theory. Conceptually simple and accessible to analysis, neural associative memories nevertheless exhibit in a rudimentary way some of the richness and complexity of biological neural systems.

In simple terms, associative memories store a set of memories or patterns—for example, vectors of ones and zeroes—and retrieve the best match when presented with inputs that may be noisy or distorted. At first glance this task seems easy, and neural networks or other novel computers seem unnecesary. Certainly any conventional computer can operate as an associative memory, for instance by finding the Hamming distance, or number of mismatched bits, between an input vector and each stored vector and choosing the smallest distance as the best match. Moreover, a parallel conventional computer can speed things up by calculating many Hamming distances simultaneously. A problem with this approach is that Hamming distance is only one of many measures of the correlation between two vectors. Calculating more complicated correlations can be very time-

*A version of this chapter has appeared as *Phys. Rev. E* **47**, 4537 (1993).

consuming, especially when the vectors are long and their number is large; and even knowing which correlations to calculate may not be clear.

Neural networks perform the associative memory task quite differently. A neural associative memory typically has as many neurons as there are elements of the input and pattern vectors. The input vector is used as the initial condition, and the network dynamics flows toward an attractor that corresponds to one of the stored patterns. The correlations that the network uses in matching inputs to patterns are incorporated into the boundaries between basins of attraction for different stored patterns. These boundaries are determined by the network architecture, and in particular by the neuron interconnection matrix. A fascinating area of research involves networks that learn new correlations while they operate and change their basin boundaries accordingly [Hertz, Krogh, and Palmer, 1991, and references therein]. The correlations learned can be complex and subtle; they would be time-consuming to calculate on a conventional computer, assuming the programmer has the foresight to calculate them at all.

This chapter looks at associative memory in competitive networks like those introduced in Chapter 1. As noted there, competitive networks are well-suited for feature extraction or pattern classification applications in which input data is assigned to one or a few of many different categories. In this chapter, competitive networks store patterns in which subsets of neurons in each cluster are chosen to win the competition. Problems for which these networks can be useful occur frequently in image processing, where the different categories may be different colors, depths, textures, elementary image features, or written characters; the networks of Sec. 1.4.2 are in fact simple associative memories. Other possible applications include speech recognition [Bridle and Cox, 1991], analysis of DNA,

proteins, and other complex molecules [Bohr *et al.*, 1990; O'Neill, 1991], and data compression methods such as principal component analysis and vector quantization [Rubner and Schulten, 1990].

The main results of this chapter are analytical phase diagrams describing the attractors of competitive associative memories as a function of the neuron transfer function steepness and the ratio of the number of stored patterns to the number of clusters [Amit, Gutfreund, and Sompolinsky, 1987; Marcus, Waugh, and Westervelt, 1990; Shiino and Fukai, 1990; Kühn, Bös, and van Hemmen, 1991]. The diagrams are calculated with statistical techniques originally developed for spin glasses [Sherrington and Kirkpatrick, 1975; Mezard, Parisi, and Virasoro, 1987] and extended to Ising-spin associative memories [Amit, Gutfreund, and Sompolinsky, 1985a,b, 1987; Gardner, 1988; Amit, 1989; Domany, van Hemmen, and Schulten, 1991]. By indicating regions of parameter space where memory recall is possible as well as regions where spurious fixed points or oscillatory attractors exist, the diagrams provide quantitative guidelines for designing and operating associative memories. Similar diagrams have been calculated for Potts-spin associative memories [Bollé, Dupont, and Huyghebaert, 1992a,b]. Similarities and differences between competitive and Potts memories are noted throughout the chapter. From a practical viewpoint, competitive networks are much more easily implemented in electronic circuitry than are Potts networks.

Section 2.2 discusses how to configure analog competitive networks as associative memories. It also introduces the statistical techniques used to study memory storage and retrieval and defines the different types of attractors—paramagnetic, recall, spin-glass, and oscillatory—the networks can have. The retrieval capability of competitive networks that store a finite number and an extensive number of patterns is analyzed in Secs. 2.3 and 2.4, respectively. The results of this analysis are summarized in bifurcation diagrams for the

memory overlap in finitely-loaded networks (Sec. 2.3) and phase diagrams for several different configurations of extensively-loaded networks (Sec. 2.4). Storage capacities for k-winner networks in the limit of infinite neuron gain are given in Sec. 2.5. Section 2.6 describes numerical investigations that support the analytical results.

2.2 ATTRACTORS IN COMPETITIVE ASSOCIATIVE MEMORIES

2.2.1 Network architectures and dynamics

Competitive networks studied in this chapter evolve according to either the continuous-time differential equations

$$\frac{dx_{ia}(t)}{dt} = -x_{ia}(t) + F_a\left(h_{ia}(t) + B_i(t) \right), \tag{2.1}$$

or the discrete-time, parallel-update equations

$$x_{ia}(t+1) = F_a\left(h_{ia}(t) + B_i(t) \right), \tag{2.2}$$

where

$$h_{ia}(t) = \sum_{j=1}^{N} \sum_{b=1}^{Q} J_{ij}^{ab} x_{jb}(t). \tag{2.3}$$

In Eqs. (2.1) and (2.2), the index i labels the N clusters ($i = 1,\ldots,N$), and the index a labels the Q neurons in each cluster i ($a = 1, \ldots, Q$). The neuron inputs $h_{ia}(t)$, the neuron outputs $x_{ia}(t)$, the analog input-output transfer functions $F_a(z)$, and the interconnection matrix J_{ij}^{ab} are all real-valued. The time-dependent bias terms $B_i(t)$ are

determined implicitly by competitive constraints (see Sec. 1.2.1):

$$\sum_{a=1}^{Q} x_{ia}(t) = 0 \qquad \text{(continuous time)}, \tag{2.4}$$

$$\sum_{a=1}^{Q} x_{ia}(t+1) = 0 \qquad \text{(discrete time)}. \tag{2.5}$$

The Q transfer functions $F_a(z)$, $a = 1,\ldots,Q$, are the same for each cluster. Thus the clusters all have the same cluster gain β as defined in Sec. 1.3.2. The transfer functions are normalized so that outputs of winning neurons are approximately $1/k$. To ensure that all neuron outputs x_{ia} equal zero for sufficiently low cluster gain, all transfer functions must have the same value of $F_a(0)$. Networks in which all transfer functions are given by

$$F_a(z) = \frac{1}{Q-1}\left[\, Q\exp(\gamma z) - 1 \,\right], \tag{2.6}$$

or by

$$F_a(z) = \frac{1}{k(Q-k)}\left[\, Q\big(\, 1+\exp(-\gamma z)\,\big)^{-1} - k \,\right], \qquad 1 \le k \le Q-1, \tag{2.7}$$

are referred to as winner-take-all and k-winner networks, respectively. In winner-take-all networks, the neuron with the largest input in each cluster has a large output, while the other $Q-1$ neuron outputs are suppressed. In k-winner networks, the neurons with the k largest inputs in each cluster have large outputs, while the other $Q-k$ neuron outputs are suppressed. Examples of the transfer functions (2.6) and (2.7) are depicted in Fig.

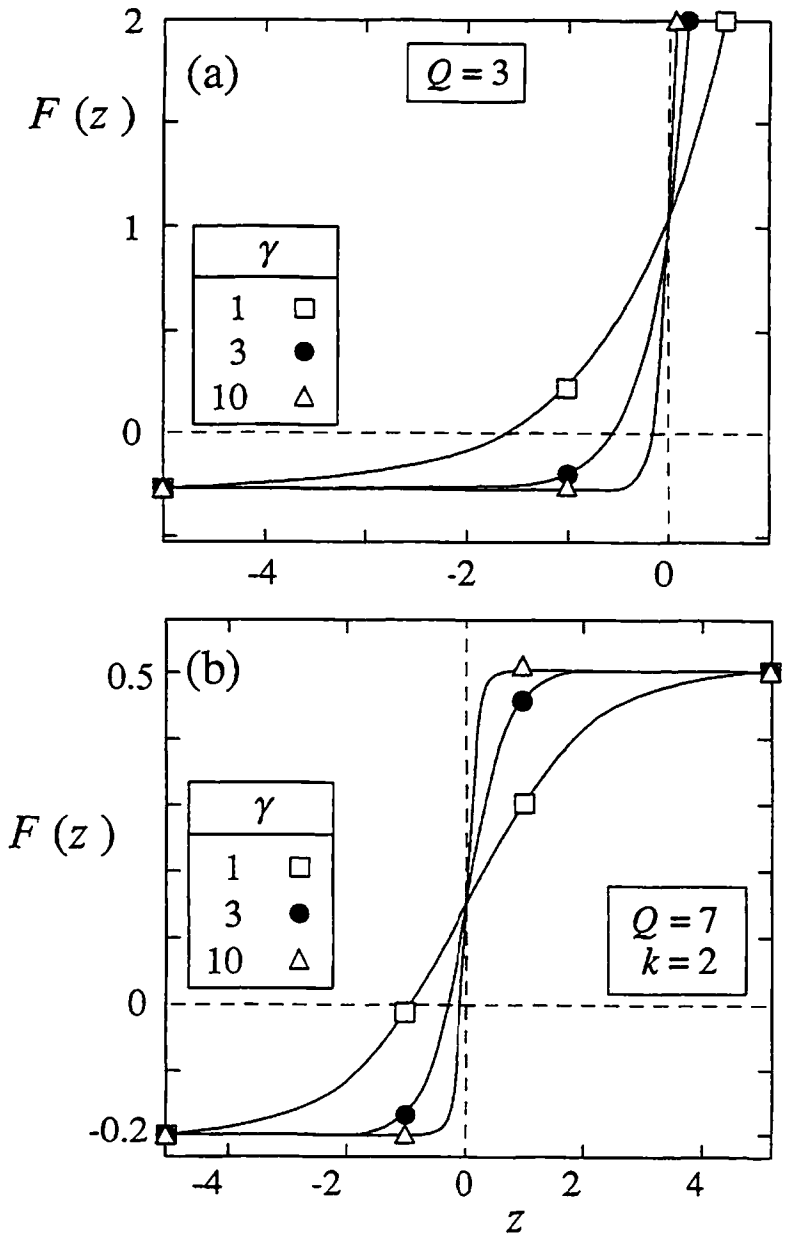


Fig. 2.1 Neuron transfer functions used in competitive associative memories: (a) winner-take-all functions, Eq. (2.6), for cluster of $Q = 3$ neurons with $\gamma = 1$, 3, and 10; (b) k-winner functions, Eq. (2.7), for cluster of $Q = 7$ neurons and $k = 2$ winners with $\gamma = 1$, 3, and 10.

2.1. The *neuron gain* γ (as distinguished from the cluster gain β) controls the slope of both functions.

The interconnection matrix J_{ij}^{ab} couples the output of neuron b in cluster j to the input of neuron a in cluster i. The matrix is constructed to store p patterns ξ_i^μ, where $\mu = 1, \ldots, p$ and $i = 1, \ldots, N$, using a form of the Hebb rule [Hebb, 1949; Kanter, 1988]:

$$J_{ij}^{ab} = \frac{1}{NQ^2} \sum_{\mu=1}^{p} \left(Q\delta_{a,\xi_i^\mu} - k\right)\left(Q\delta_{b,\xi_j^\mu} - k\right), \qquad J_{ii}^{ab} = 0. \tag{2.8}$$

Examples of patterns are shown in Fig. 2.2. Each ξ_i^μ is a set of different integers indicating which neurons are chosen to win the competition in cluster i for pattern μ. The k different integers in this set, where $1 \le k \le Q-1$, are chosen randomly and without bias from the set of integers $\{1, \ldots, Q\}$. For example, the statement $\xi_i^\mu = \{1, 2, \ldots, k\}$ means that, in pattern μ, neurons $1, 2, \ldots, k$ are chosen to win the competition in cluster i. For winner-take-all networks, the value of k in Eq. (2.8) is always 1, while for k-winner networks, it is in the range $1 \le k \le Q-1$. For both network types, the case $Q = 2$ is equivalent to associative memories of standard analog neurons [Marcus, Waugh, and Westervelt, 1990; Shiino and Fukai, 1990; Kühn, Bös, and van Hemmen, 1991]. Sums over the patterns ξ_i^μ in Eq. (2.8) and the rest of this paper imply summation over all k values of ξ_i^μ. Other forms of the Hebb rule that have been studied for Potts networks [Vogt and Zippelius, 1992] are not considered here.

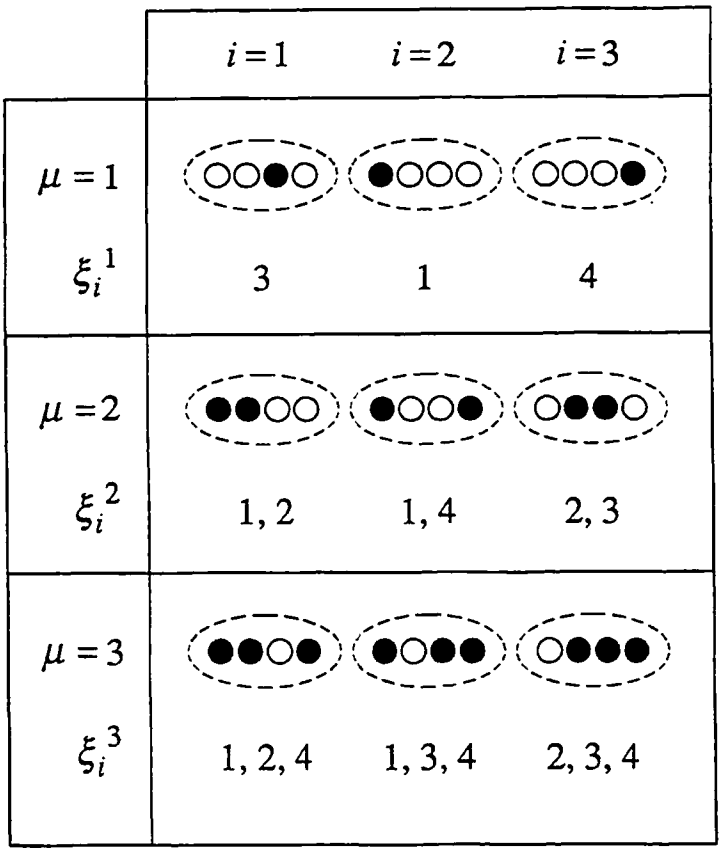


Fig. 2.2 Pattern examples for competitive associative memories, showing how ξ_i^μ and k are defined. For each pattern, $k = 1, 2$, or 3 neurons are chosen to win competition in each cluster. Winner-take-all networks can store patterns like ξ^1; k-winner networks can store patterns like ξ^1, ξ^2, or ξ^3.

2.2.2 Statistical mechanics for analog networks

The Liapunov function $L(t)$ and the stability criterion of Sec. 1.3.2 can be used to analyze the attractors of competitive analog networks. The procedure is to treat $L(t)$ as an energy and to apply standard techniques of statistical mechanics for disordered systems [Amit, Gutfreund, and Sompolinsky, 1987; Amit, 1989]. A crucial step in this procedure is to introduce an auxiliary temperature, which has no physical meaning and is set to zero at the end of the calculation [Kühn, Bös, and van Hemmen, 1991]. The temperature enables the derivation of a free energy f per neuron, which provides information about metastable states. When the temperature is set to zero, the metastable states become the fixed points of the dynamical system for which $L(t)$ is a Liapunov function. For a particular interconnection matrix J_{ij}^{ab}, the free energy per neuron is

$$f = -\frac{1}{\tilde{\beta}N} \ln \int_{-\infty}^{\infty} \prod_{i=1}^{N} \prod_{a=1}^{Q} \left[d\rho(x_{ia}) \right] \exp\left(-\tilde{\beta} L(t) \right), \tag{2.9}$$

where $\tilde{\beta}$ is the inverse of the temperature and $d\rho(x_{ia})$ equals 1 on the range of the transfer function $F_a(z)$ and 0 otherwise [Kühn, Bös, and van Hemmen, 1991].

It should be stressed that these networks are completely deterministic; the auxiliary temperature is simply a mathematical device allowing the use of statistical mechanics to learn about attractors. Thus the free energy at nonzero values of the auxiliary temperature (which describes networks of *noisy* analog neurons) is not considered. Note that this procedure works only for networks in which all cluster gains satisfy the stability criterion, since otherwise $L(t)$ is not a Liapunov function. Other methods can be used to analyze networks in the region in which the stability criterion is violated [Fontanari and Köberle,

1988a; Shiino and Fukai, 1992]. Here period-two limit cycles are considered undesirable, and only networks guaranteed to be free of them are studied.

Important issues for reliable associative memory performance include how many patterns can be stored, what other fixed points exist besides those corresponding to stored patterns, how neuron gain affects pattern retrieval, and how the stability criterion of Sec. 1.3.2 affects discrete-time, parallel-update networks. The next four sections address these issues both analytically, by using the free energy (2.9) to derive phase diagrams summarizing different attractor types, and numerically, by studying attractors of small computer-generated networks. Useful quantities for characterizing attractors are the p pattern overlaps m_μ, defined as

$$m_\mu \equiv \frac{1}{N}\sum_{i=1}^{N} x_i \xi_i^\mu , \qquad \mu = 1,\ldots,p \tag{2.10}$$

and a spin-glass order parameter q, defined in Sec. 2.4 as

$$q \equiv \frac{1}{NQ}\frac{k(Q-k)}{(Q-1)}\sum_{i=1}^{N}\sum_{a=1}^{Q} x_{ia}^2 . \tag{2.11}$$

Successful recall of pattern μ means that m_μ is of order 1, while all other overlaps m_ν, $\nu \neq \mu$, are much less than 1. In the limit of finite memory loading, in which the storage fraction $\alpha \equiv p/N$ approaches zero as $N \to \infty$, two attractor types can occur: (i) a paramagnetic fixed point, for which $q = 0$ and all $m_\mu = 0$, and (ii) memory recall fixed points, for which $q > 0$ and one m_μ is of order 1. When the storage fraction α is greater than zero, two other attractor types can exist in addition to the paramagnetic and recall

attractors: (iii) spin-glass fixed points, for which $q > 0$ but all $m_\mu = 0$, and (iv), period-two limit cycles, which can occur when the stability criterion of Sec. 1.3.2 is violated.

2.3 BIFURCATION DIAGRAMS FOR FINITE MEMORY LOADING

Competitive associative memories are first considered in the limit of finite memory loading, in which the number p of stored patterns remains finite while the number N of clusters becomes large, so that $\alpha = 0$. This limit is particularly simple because the interference between patterns is negligible [Amit, Gutfreund, and Sompolinsky, 1985a,b, 1987]. The analysis shows that a discontinuous, hysteretic transition from paramagnetic to memory recall behavior can occur as neuron gain γ increases.

Figure 2.3, which is the main result of this section, shows bifurcation diagrams for the overlap m when $m_\mu = m\delta_{\mu,1}$ in winner-take-all and k-winner networks with finite memory loading. The diagrams show the overlap as a function of $\hat{\gamma} \equiv \gamma/Q$ for winner-take-all networks and $\hat{\gamma} \equiv \gamma k(Q-k)/Q^2$ for k-winner networks. For both network types, a single solution exists at low $\hat{\gamma}$, and three solutions exist at high $\hat{\gamma}$. The single solution $m = 0$ at low $\hat{\gamma}$ is the paramagnetic solution, for which all neuron outputs x_{ia} equal zero. The three solutions at high $\hat{\gamma}$ are the paramagnetic solution and two recall solutions that approach the values 1 and $-1/(Q-1)$ for winner-take-all networks and 1 and $-1/(\hat{Q}-1)$ for k-winner networks, where $\hat{Q} \equiv \max(Q/k,\ Q/(Q-k))$.

Solutions for m correspond to fixed points of the update equations (2.1) and (2.2) only if they are stable. Stable solutions are indicated by solid curves and unstable solutions by dashed curves in Fig. 2.3. Below it is demonstrated that (i) the paramagnetic solution is stable for $\hat{\gamma} \leq 1$ and unstable for $\hat{\gamma} > 1$; (ii) the positive recall solution is always stable; and

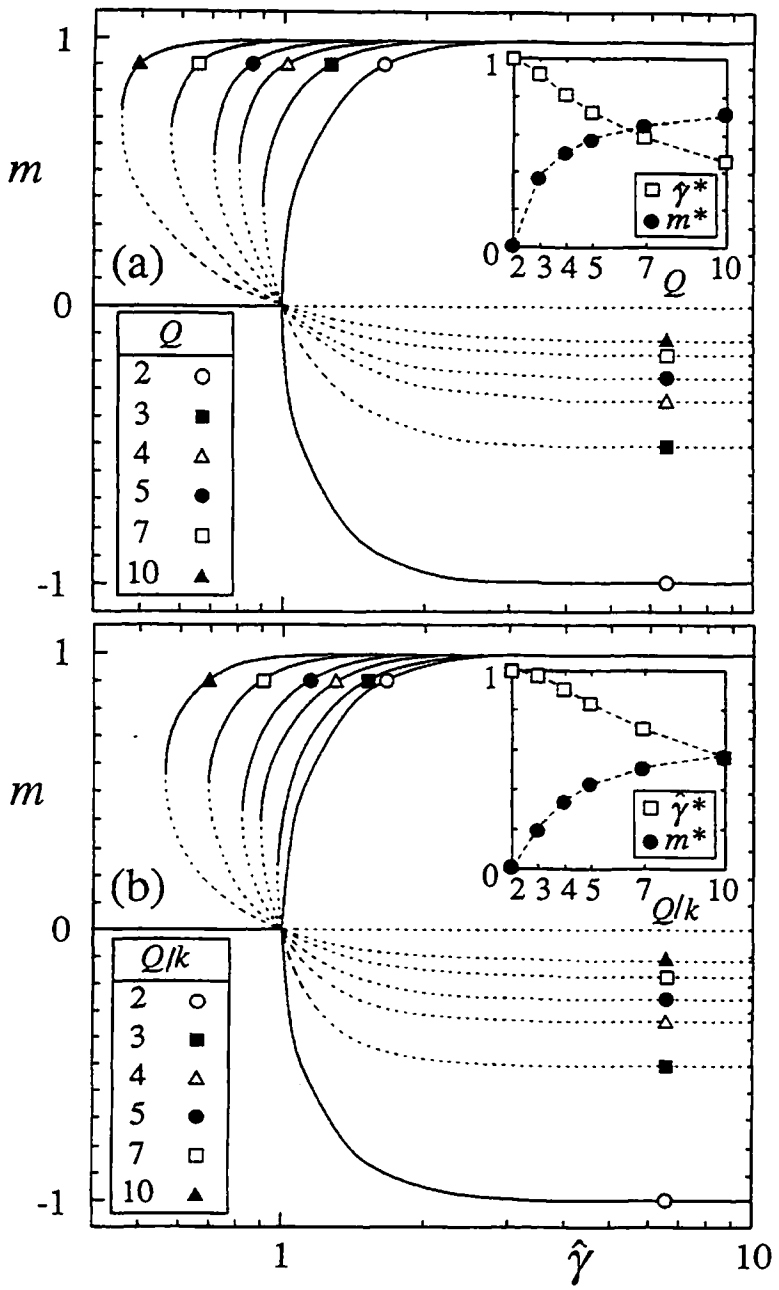


Fig. 2.3 Bifurcation diagrams for overlap m vs. gain parameter $\hat{\gamma}$ for finite loading, showing first-order transition from paramagnetic to recall behavior. Curves are for $Q = 2$, 3, 4, 5, 7, and 10 neurons per cluster in (a) winner-take-all networks (for which $\hat{\gamma} = \gamma/Q$) and (b) k-winner networks with $k = 1$ or $Q-1$ (for which $\hat{\gamma} = \gamma k(Q-k)/Q^2$). Stable solutions are indicated by solid curves, unstable solutions by dashed curves. For k-winner networks, solutions for m depend only on Q/k; however, stability of negative solution is Q-dependent. Insets show values $\hat{\gamma}^*$ of gain parameter and m^* of overlap at onset of first-order transition.

(iii) the negative recall solution is unstable for winner-take-all networks with $Q > 2$ but can be stable at low gain for k-winner networks.

Two distinct types of behavior appear in Fig. 2.3. For $Q = 2$ in winner-take-all networks and $\hat{Q} = 2$ in k-winner networks, a single bifurcation occurs at $\hat{\gamma} = 1$, in which the paramagnetic solution becomes unstable and two stable recall solutions appear. Stable paramagnetic and recall solutions never coexist. For all other values of Q and $\hat{Q}$ tested, two bifurcations occur: one at $\hat{\gamma} = 1$, in which the paramagnetic solution becomes unstable, and another at $\hat{\gamma} = \hat{\gamma}^* < 1$, in which a stable recall solution and another unstable solution appear at a nonzero value $m = m^*$. Stable paramagnetic and recall solutions coexist for $\hat{\gamma}^* < \hat{\gamma} < 1$, and the networks are hysteretic in this region. The values $\hat{\gamma}^*$ and m^* are shown in the insets of Fig. 2.3.

The bifurcation diagrams of Fig. 2.3 are derived from the free energy (2.9) [Amit, Gutfreund, and Sompolinsky, 1985a; Bollé and Mallezie, 1989; Kühn, Bös, and van Hemmen, 1991; Vogt and Zippelius, 1992]. For finite p and Q, the free energy per neuron averaged over patterns in the limit $N \to \infty$ is

$$f = \frac{1}{2}\sum_{\mu=1}^{p} m_\mu^2 - \sum_{\mu=1}^{p} \left\langle m_\mu F_{\xi^\mu}\left(\sum_{\nu=1}^{p} m_\nu \delta_{\xi^\mu,\,\xi^\nu} + B \right) \right\rangle_\xi$$

$$+ \sum_{a=1}^{Q} \left\langle G_a\left(F_a\left(\sum_{\mu=1}^{p} m_\mu \delta_{a,\,\xi^\mu} + B \right) \right) \right\rangle_\xi . \qquad (2.12)$$

In Eq. (2.12), the brackets $\langle \cdots \rangle_\xi$ denote an average over all possible realizations of the stored patterns, the functions $G_a(x)$ are given by Sec. 1.3.1, and the overlaps m_μ obey the saddle-point equations

$$m_\mu = \left\langle F_{\xi^\mu} \left(\sum_{\nu=1}^{p} m_\nu \delta_{\xi^\mu, \xi^\nu} + B \right) \right\rangle_\xi . \tag{2.13}$$

The quantity B in Eqs. (2.12) and (2.13) is determined implicitly by the competitive constraint

$$\sum_{a=1}^{Q} \left\langle F_a \left(\sum_{\mu=1}^{p} m_\mu \delta_{a, \xi^\mu} + B \right) \right\rangle_\xi = 0. \tag{2.14}$$

The derivation of Eqs. (2.12) through (2.14) is outlined in Appendix 2A.

Bifurcation diagrams are constructed by looking for solutions of Eqs. (2.12) through (2.14) with $m_\mu = m\delta_{\mu,1}$ in networks in which all neurons have the same transfer function $F_a(z) = F(z)$. Inserting these simplifications and performing the average over patterns leads to

$$f = \frac{1}{2}m^2 - mF(m + B) + kG\big(F(m + B)\big) + (Q-k)G\big(F(B)\big) \tag{2.15}$$

$$m = kF(m + B) \tag{2.16}$$

$$kF(m + B) + (Q-k)F(B) = 0. \tag{2.17}$$

Combining Eqs. (2.16) and (2.17) gives a self-consistent equation for the overlap m:

$$m = -(Q-k)F\left(F^{-1}\left(\frac{m}{k}\right) - m \right). \tag{2.18}$$

This result is analogous to the equation $m = F(m)$ for standard analog associative memories at finite loading [Kühn, Bös, and van Hemmen, 1991]. For winner-take-all and k-winner networks, Eq. (2.18) reads

$$m = 1 - [\,(Q-1)m + 1\,]\exp(-\gamma m) \qquad \text{(winner-take-all)}, \tag{2.19}$$

$$m = 1 - \frac{Q}{k}\left[1 + \frac{(Q/k-1)(1-m)}{(Q/k-1)m + 1}\exp(\gamma m)\right]^{-1} \qquad (k\text{-winner}). \tag{2.20}$$

Equation (2.19) is identical to the mean-field equation describing Potts associative memories at temperature $T = Q(Q-1)/\gamma$ in the limit of finite loading [Bollé and Mallezie, 1989; Bollé, Dupont, and van Mourik, 1991; Vogt and Zippelius, 1992]. However, as shown below, this equivalence does not persist for extensive loading [Kühn, Bös, and van Hemmen, 1991; Shiino and Fukai, 1992].

Figure 2.4 shows typical behavior of the mean-field equations (2.19) and (2.20). In Figs. 2.4(a) through (c), the left-hand and right-hand sides of the winner-take-all mean-field equation (2.19) are plotted as functions of m for three different values of $\hat{\gamma}$ for $Q = 7$. The three values of $\hat{\gamma}$ are chosen to be less than $\hat{\gamma}^*$, between $\hat{\gamma}^*$ and 1, and greater than 1. Figures 2.4(d) thorugh (f) show similar plots for the k-winner mean-field equation (2.20) for $Q = 7$ and $k = 1$ or 6. Two invariance properties of Eq. (2.20) are apparent in Fig. 2.4. First, because Q and k appear in Eq. (2.20) only in the ratio Q/k, any solution for k winners in Q-neuron clusters is also a solution for nk winners in nQ-neuron clusters, for all positive integers n. Second, the solutions of Eq. (2.20) for k and $(Q-k)$ winners per cluster are identical, despite the fact that, as seen in the figure, the

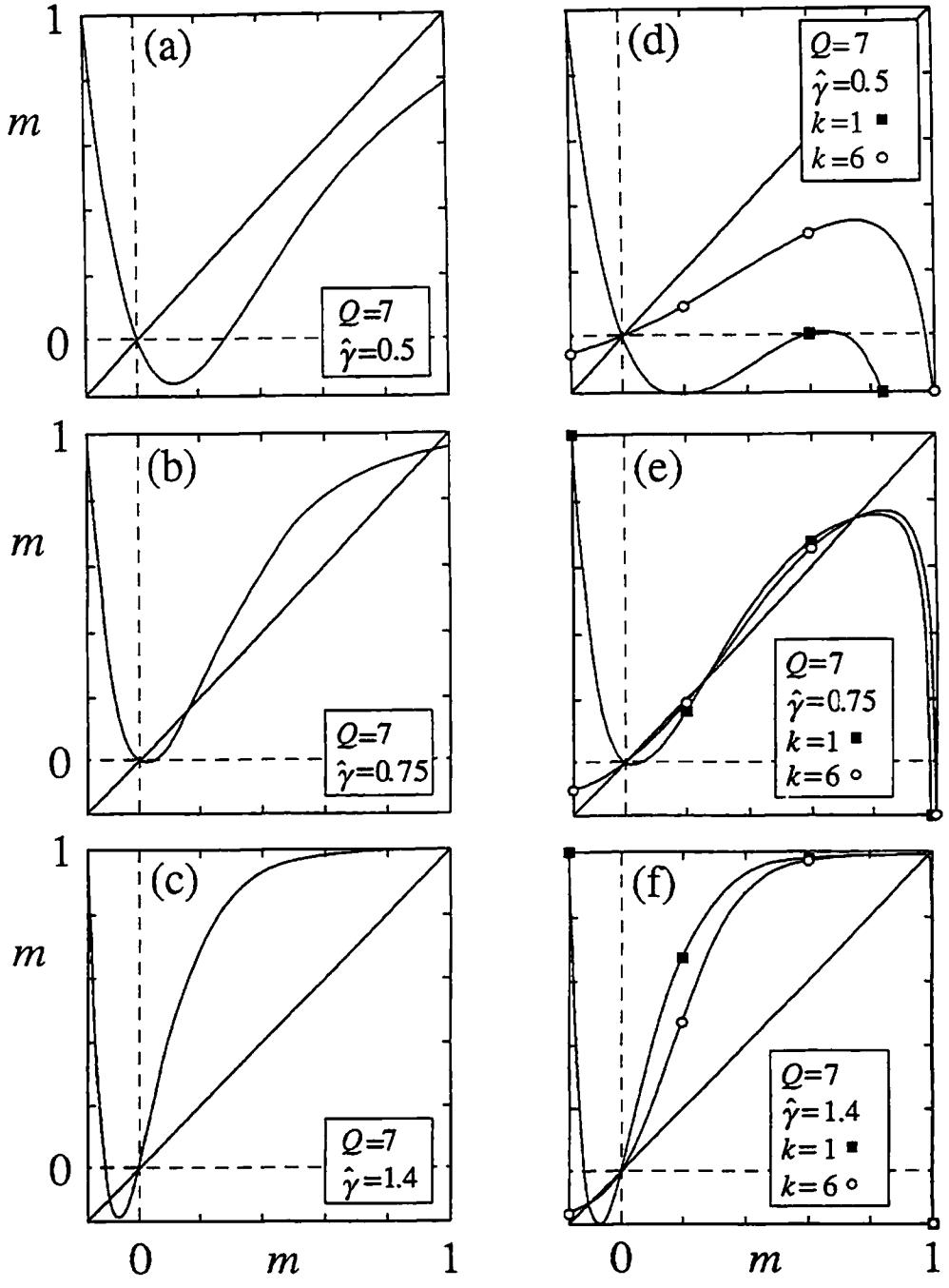


Fig. 2.4 Left-hand side (diagonals) and right-hand side (curves) of mean-field equations (2.19) and (2.20) for finite loading vs. overlap m. Intersections of curves and diagonals determine bifurcation diagrams of Fig. 2.4. (a) through (c) Winner-take-all networks, Eq. (2.19), with $Q = 7$ and $\hat{\gamma}$ =0.5, 0.75, and 1.4; (d) through (f) k-winner networks, Eq. (2.20), with $Q = 7$, $k = 1$ or 6, and $\hat{\gamma}$ =0.5, 0.75, and 1.4. Values of $\hat{\gamma}$ are less than $\hat{\gamma}^*$, between $\hat{\gamma}^*$ and 1, and greater than 1.

right-hand-side of Eq. (2.20) is not invariant under the transformation $k \rightarrow (Q-k)$. Thus solutions of Eq. (2.20) with the same value of $\hat{Q} \equiv \max(Q/k,\ Q/(Q-k))$ are identical.

Stability of solutions of the mean-field equations is determined by the eigenvalues of the matrix $\partial^2 f / \partial m_\rho \partial m_\sigma$ of second derivatives of the free energy with respect to the overlaps. Solutions are stable only if all eigenvalues are positive. The eigenvalues are calculated in Appendix 2B; for the case $m_\mu = m\delta_{\mu,1}$ of a single successfully recalled pattern, the matrix is diagonal, with two eigenvalues. As shown in Fig. 2.4, the positive recall solution is always stable for winner-take-all and k-winner networks, while the negative recall solution is always unstable for winner-take-all networks with $Q > 2$ and for k-winner networks with $Q > 2$ and $k = 1$ or $k = Q-1$. However, the negative recall solution can be stable at low gain for k-winner networks with $1 < k < Q-1$. This result is illustrated in Fig. 2.5, which shows, for various k, the value of $\hat{\gamma}$ at which the negative recall solution becomes unstable as a function of Q/k.

Finally, spin-glass and oscillatory attractors do not appear in networks with finite memory loading. Spin-glass attractors do not appear because, as implied by Eq. (2.13), the overlaps are nonzero whenever the neuron outputs are nonzero. Oscillatory attractors do not appear because, as is shown in Sec. 2.4, the interconnection matrix (2.8) is positive-definite for finite loading so that the stability criterion is satisfied for all values of cluster gain. Thus paramagnetic and recall attractors are the only attractor types.

This section has shown that competitive associative memories with finite memory loading undergo a discontinuous, hysteretic transition from paramagnetic to recall behavior as their transfer functions steepen. This result yields the $\alpha = 0$ axes of the phase diagrams that appear in the next section for extensively loaded networks. The next section shows that the hysteretic transition from paramagnetic to recall behavior persists at low but finite

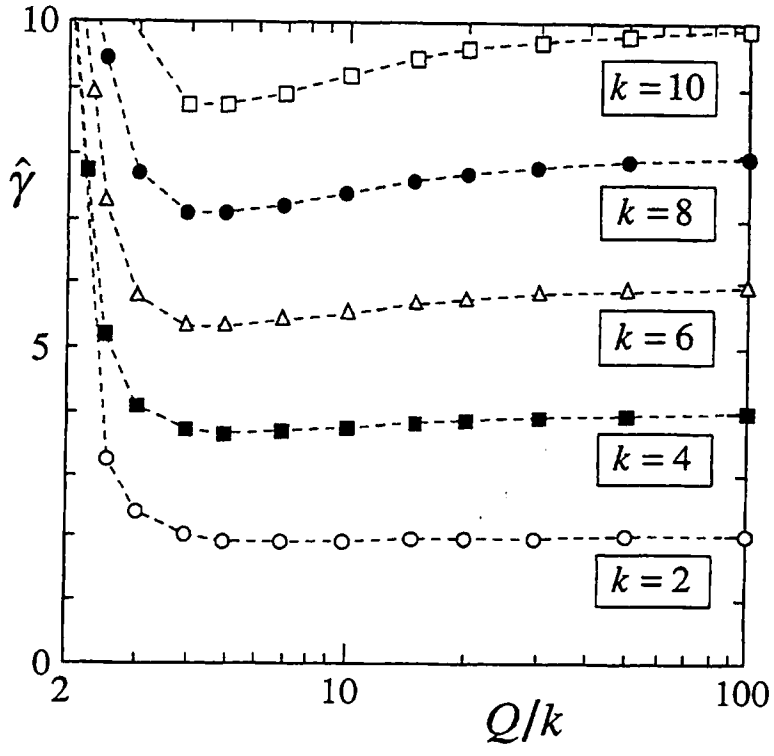


Fig. 2.5 Values of $\hat{\gamma}$ at which negative recall solution becomes unstable in finitely-loaded networks vs. Q/k for k= 2, 4, 6, 8, and 10. Eigenvalues of $\partial^2 f/\partial m_\rho \partial m_\sigma$ determine stability.

α, in contrast to standard analog networks, for which the phase diagram exhibits a spin-glass region between the paramagnetic and recall regions for all $\alpha > 0$ [Marcus, Waugh, and Westervelt, 1990; Shiino and Fukai, 1990; Kühn, Bös, and van Hemmen, 1991]. Another result of this section—that, for finite loading, competitive and Potts associative memories obey the same mean-field equation—is shown *not* to hold at finite α.

2.4 PHASE DIAGRAMS FOR EXTENSIVE MEMORY LOADING

In this section, the number p of patterns varies extensively with the number N of clusters as $p = \alpha N$, so that interference between patterns can no longer be ignored [Amit, Gutfreund, and Sompolinsky, 1987]. Assuming replica symmetry, the replica method can be used to derive a set of self-consistent equations for the overlaps m_μ and the spin-glass order parameter q. Replica-symmetry-breaking effects are expected to be small, as they typically are for Hebb-rule associative memories [Crisanti, Amit, and Gutfreund, 1986; Kohring, 1990b].

Figures 2.6 and 2.7 show several analytical phase diagrams for continuous-time and discrete-time networks with transfer functions given by Eqs. (2.6) and (2.7). These diagrams, which are the main result of this chapter, indicate the types of attractors that the networks can have as a function of the gain parameter $\hat{\gamma}$ and the ratio $\alpha = p/N$ of patterns to clusters. (Recall that $\hat{\gamma} \equiv \gamma/Q$ for winner-take-all networks and that $\hat{\gamma} \equiv \gamma k(Q-k)/Q^2$ for k-winner networks). The diagrams are valid in the limit of large N and finite Q.

The diagrams of Fig. 2.6, which are for continuous-time updating, each contain three regions labelled "pm" (paramagnetic), "recall," and "sg" (spin-glass). In the paramagnetic region, the networks have a single global attractor at the origin of state space, $x_{ia} = 0$ for all a and i. Thus $q = 0$ and all $m_\mu = 0$ in this region. In the spin-glass region, the

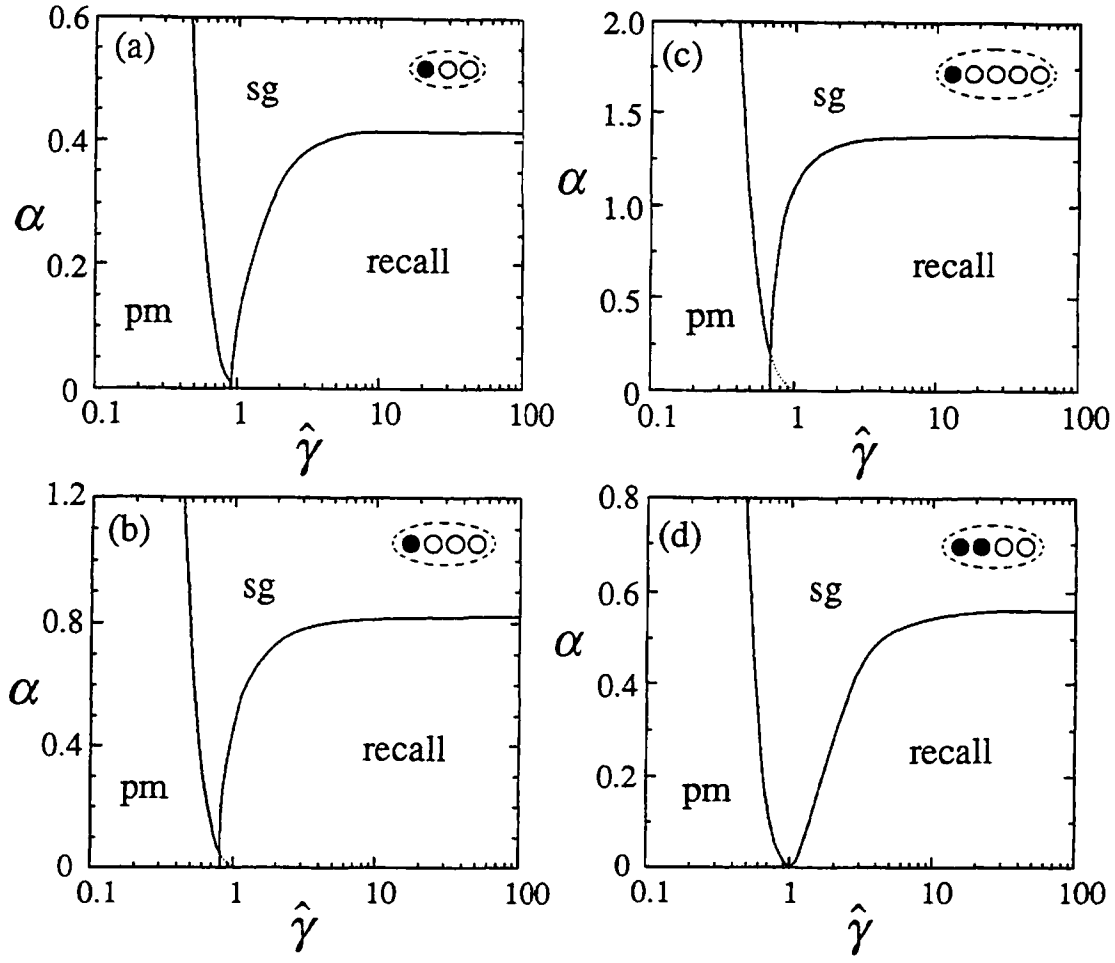


Fig. 2.6 Phase diagrams for extensive loading and continuous-time updating, showing attractor types vs. storage fraction α and gain parameter $\hat{\gamma}$. (a) through (c) are for winner-take-all networks with (a) $Q = 3$, (b) $Q = 4$, and (c) $Q = 5$ neurons per cluster; (d) is for k-winner networks with $Q = 4$ neurons and $k = 2$ winners per cluster. Labels 'pm,' 'sg,' and 'recall' denote paramagnetic, spin glass, and recall regions. Dashed curves within recall region in (a) through (c) show boundary between coexistence of recall attractors with paramagnetic attractor (low gain) and with spin glass attractors (high gain).

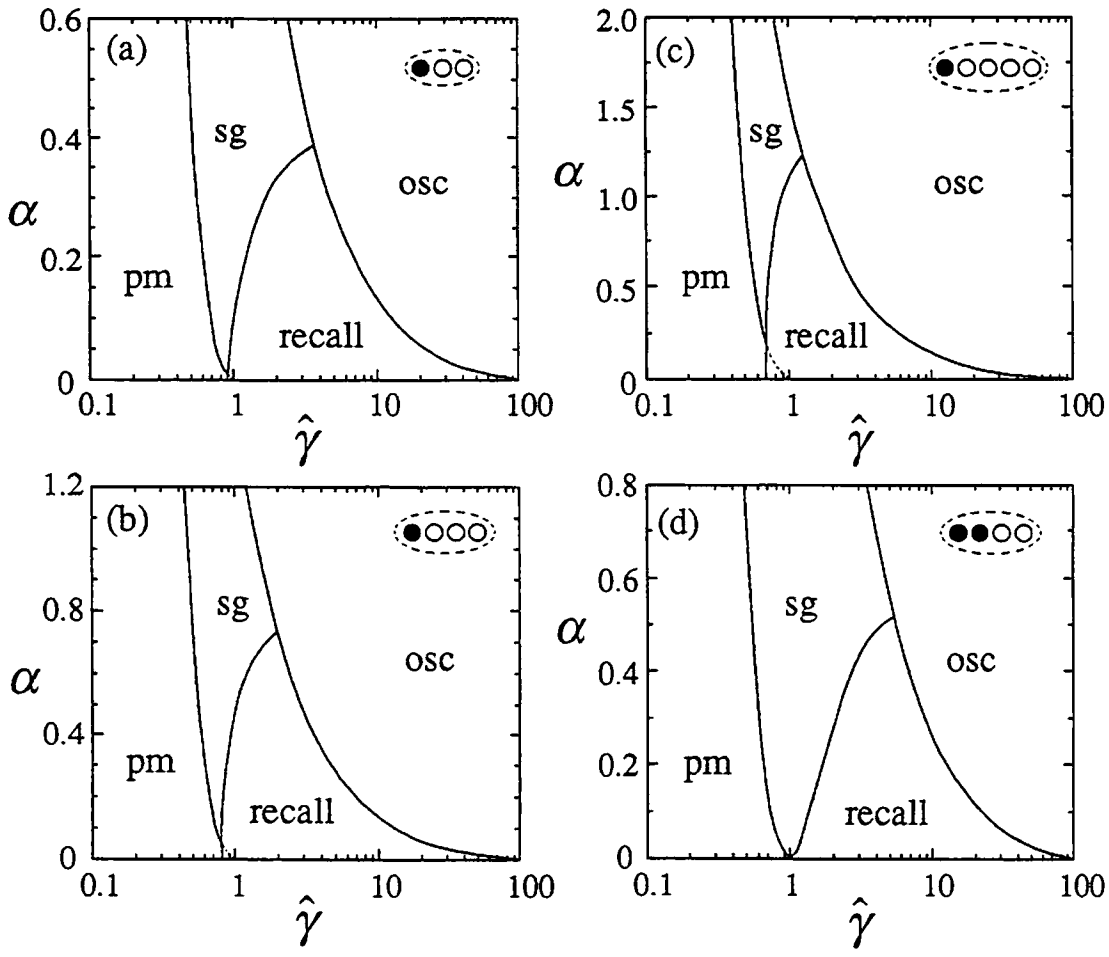


Fig. 2.7 Phase diagrams for extensive loading and discrete-time parallel updating, showing attractor types vs. storage fraction α and gain parameter $\hat{\gamma}$. (a) through (c) are for winner-take-all networks with (a) $Q = 3$, (b) $Q = 4$, and (c) $Q = 5$ neurons per cluster; (d) is for k-winner networks with $Q = 4$ neurons and $k = 2$ winners per cluster. Labels 'pm,' 'sg,' 'recall,' and 'osc' denote paramagnetic, spin glass, recall, and oscillatory regions. Dashed curves within recall region in (a) through (c) show boundary between coexistence of recall attractors with paramagnetic attractor (low gain) and with spin glass attractors (high gain). Outside oscillatory region, diagrams are identical to those for continuous-time updating in Fig. 2.6.

networks have many fixed-point attractors away from the origin. These attractors are characterized by $q > 0$ and all $m_\mu = 0$: the paramagnetic attractor is unstable, but fixed points corresponding to stored patterns have not yet appeared. In the recall region, the networks function reliably as associative memories, with fixed points that have large overlaps with the stored patterns. These fixed points are characterized by $q > 0$ and one or more $m_\mu > 0$.

The diagrams of Fig. 2.7—which are for discrete-time, parallel-updating—also contain paramagnetic, recall, and spin-glass regions. In addition, each contains a region marked "osc" (oscillatory) in which the stability criterion of Sec. 1.3.2 is violated. The function $L(t)$ of Sec. 1.3.1 is not a Liapunov function in this region. Recall and spin-glass attractors may still exist, but the networks can also have period-two limit cycle attractors. Outside the oscillation region, the phase diagrams are identical to those for continuous-time updating. Although no oscillatory regions appear in the phase diagrams of continuous-time networks, unavoidable neural and synaptic delays can lead to oscillatory attractors in electronic implementations [Marcus and Westervelt, 1989a].

In Figs. 2.6(a) through (c) and 2.7(a) through (c), the recall region is itself divided into two parts by a dashed curve. In the smaller, low-gain part, recall fixed points coexist with the paramagnetic fixed point, while in the larger, high-gain part, they coexist with spin-glass fixed points. Similar behavior has been reported in stochastic Potts associative memories for the case $Q = 3$ [Bollé, Dupont, and Huyghebaert, 1992a,b]. The effect is important because it implies that the spurious attractors that often degrade associative memory performance (see Chapter 3) are not present in the low-gain part of the recall region. As seen in Figs. 2.6 and 2.7, the effect is more prominent at higher values of Q. In Figs. 2.6(d) and 2.7(d), recall fixed points coexist only with spin-glass fixed points.

The phase diagrams are computed from the free energy (2.9) and, for discrete-time updating, from the stability criterion Sec. 1.3.2; details of the calculation appear in Appendix 2C. When a network has nonzero overlaps m_ν with finite number s of patterns, $\nu = 1, \ldots, s$, the free energy f per neuron is determined in the limit of large N as a saddle point over the overlaps m_ν, the spin-glass order parameter q, and a third quantity C. The saddle-point equations are

$$m_\nu = \left\langle \hat{x}\xi^\nu \right\rangle_{z,\xi}, \quad \nu = 1, \ldots, s \tag{2.21}$$

$$q = \frac{1}{Q}\frac{k(Q-k)}{Q-1}\left\langle \sum_{a=1}^{Q} \hat{x}_a^2 \right\rangle_{z,\xi} \tag{2.22}$$

$$C = \sqrt{\frac{1}{\alpha r Q}\frac{k(Q-k)}{Q-1}}\left\langle \sum_{a=1}^{Q} z_a \hat{x}_a \right\rangle_{z,\xi}. \tag{2.23}$$

In Eqs. (2.21) through (2.23), the Q quantities $\hat{x}_a$ are determined implicitly by

$$\hat{x}_a = F_a\left(\sum_{\nu=1}^{s} m_\nu \delta_{a,\xi^\nu} + z_a\sqrt{\frac{\alpha r}{Q}\frac{k(Q-k)}{Q-1}} + \hat{x}_a\frac{\alpha}{Q}\frac{k(Q-k)}{Q-1}(\tilde{r}-1) + B \right), \tag{2.24}$$

with B determined by the competitive requirement $\sum_{a=1}^{Q}\hat{x}_a = 0$. The brackets $\langle\cdots\rangle_{z,\xi}$ indicate an average over the patterns ξ^ν and over the Q continuous variables z_a using a Gaussian distribution:

$$\langle\cdots\rangle_{z,\xi} \rightarrow \left\langle \int_{-\infty}^{\infty} \prod_{a=1}^{Q}\left(\frac{dz_a}{\sqrt{2\pi}}\right) \exp\left(-\frac{1}{2}\sum_{a=1}^{Q} z_a^2\right)(\cdots) \right\rangle_{\xi}. \tag{2.25}$$

The quantities r and $\tilde{r}$ are

$$r = \frac{q}{(1-C)^2}, \quad \tilde{r} = \frac{1}{1-C}. \tag{2.26}$$

The boundaries in Figs. 2.6 and 2.7 are calculated as follows. The boundary of the recall region, also known as the storage capacity α_c, is found numerically as the largest value of α for which a solution of Eqs. (2.21) through (2.23) exists with $m_1 \sim 1$ and all other $m_\nu = 0$ (the FORTRAN program for this procedure appears as Appendix A of this thesis). The boundary between the spin-glass and origin regions, which is continuous for all diagrams in Figs. 2.6 and 2.7, is found by expanding Eqs. (2.21) through (2.23) to leading order in the small quantities $\hat{x}_a$. This expansion is carried out in Appendix 2D. For networks in which all neurons have the same transfer function $F(z)$, the boundary is

$$F'\left[F^{-1}(0)\right] = \frac{Q}{k(Q-k)} \frac{1}{1+2\sqrt{\alpha/(Q-1)}}, \tag{2.27}$$

where $F'(z)$ is the derivative of $F(z)$. For both winner-take-all and k-winner networks, Eq. (2.27) yields

$$\hat{\gamma} = \frac{1}{1+2\sqrt{\alpha/(Q-1)}}. \tag{2.28}$$

Finally, the boundary of the oscillation regions of Fig. 2.7 is determined by the stability criterion derived of Sec. 1.3.2. For winner-take-all and k-winner clusters, the cluster gain β is

$$\beta = \frac{Q\gamma}{2(Q-1)} \quad \text{(winner-take-all)}, \tag{2.29}$$

$$\beta = \frac{Q\gamma}{4k(Q-k)} \quad (k\text{-winner}). \tag{2.30}$$

For large N, the minimum eigenvalue of the interconnection matrix (2.8) is

$$\lambda_{min} \cong -\frac{\alpha}{Q}\frac{k(Q-k)}{Q-1}, \tag{2.31}$$

found by numerically computing the eigenvalue spectra of computer-generated interconnection matrices as shown in Fig. 2.8. The results of Fig. 2.8 were generated by constructing 20 matrices according to Eq. (2.8) for network sizes N = 50, 75, 100, 150, and 200 for three sets of Q and α. For large N, the eigenvalue spectra are similar to that of a standard Hebb-rule matrix [Geman, 1980; Crisanti and Sompolinsky, 1987; Le Cun, Kanter, and Solla, 1991], with αN positive eigenvalues forming a continuous distribution, N degenerate eigenvalues equal to zero, and $N(Q-1-\alpha)$ negative eigenvalues grouped about the value $-\alpha/Q$. Combining the stability criterion of Sec. 1.3.2 and Eqs. (2.29) through (2.31) yields the following expressions for the oscillation region boundary:

$$\hat{\gamma} = \frac{1}{\alpha}\frac{2(Q-1)}{Q} \quad \text{(winner-take-all)}, \tag{2.32}$$

$$\hat{\gamma} = \frac{1}{\alpha}\frac{4k(Q-k)(Q-1)}{Q^2} \quad (k\text{-winner}). \tag{2.33}$$

The storage capacity boundary terminates at the oscillation region boundary because $L(t)$ is not a Liapunov function in the oscillation region.

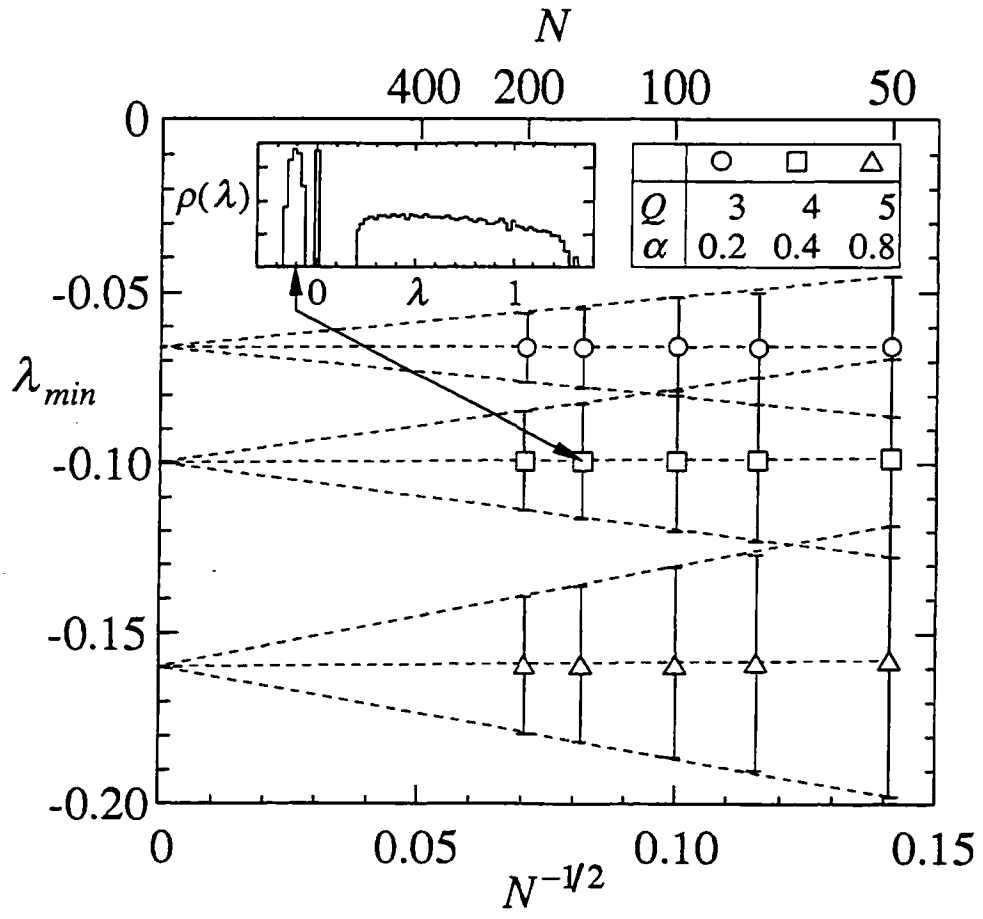


Fig. 2.8 Mean (markers) and standard deviation (error bars) of negative eigenvalues of computer-generated winner-take-all interconnection matrices for various Q and α, showing how distribution of negative eigenvalues approaches a delta-function peak at $\lambda_{min} = -\alpha/Q$ as $N \to \infty$. Inset: eigenvalue histogram for 20 matrices with $N = 150$, $Q = 4$, $k = 1$, and $\alpha = 0.4$, with logarithmic vertical axis.

The winner-take-all phase diagrams, Figs. 2.7(a) through (c), are similar but not identical to phase diagrams for stochastic associative memory networks of Potts spins at finite temperature [Bollé, Dupont, and Huyghebaert, 1992a,b]. Phase boundaries for the two network types are identical in the limits of infinite gain and of finite memory loading. The storage capacity values in the infinite-gain limit—which are α_c = 0.414, 0.828, and 1.375 for $Q = 3$, 4, and 5—are the same as for Potts networks [Kanter, 1988]. The values of $\hat{\gamma}$ at which recall attractors appear in the finite-loading limit—which are $\hat{\gamma}^* = 0.915$, 0.805, and 0.713 for $Q = 3$, 4, and 5—are also the same as for Potts networks [Bollé and Mallezie, 1989; Bollé, Dupont, and van Mourik, 1991; Vogt and Zippelius, 1992].

Aside from these limits, phase diagrams for competitive networks and Potts networks are *not* equivalent, due to differences between deterministic, analog dynamics and stochastic, discrete-state dynamics [Shiino and Fukai, 1990; Takayama and Nemoto, 1990; Kühn, Bös, and van Hemmen, 1991]. The mean-field treatment of Potts systems [Lage and Nunes da Silva, 1984; Gross, Kanter, and Sompolinsky, 1984] yields a reaction field term [Thouless, Anderson, and Palmer, 1977] that subtracts each spin's influence from its own local field, whereas no reaction field appears in the neuron inputs (2.3) of competitive networks. This difference is illustrated in Fig. 2.9, which compares phase diagrams of winner-take-all competitive networks and Potts networks with $Q = 3$. The phase boundaries for Potts networks [Bollé, Dupont, and Huyghebaert, 1992a,b] are plotted as functions of $(Q-1)\beta$, where β is the inverse temperature. The paramagnetic/spin-glass transition and the spin-glass/recall transition both lie further to the right of the diagram for stochastic networks as compared to analog networks. An intuitive explanation is that the reaction field acts as an effective noise source in the stochastic Potts network, decreasing the temperature at which transitions occur. This point arises again in Chapter 3.

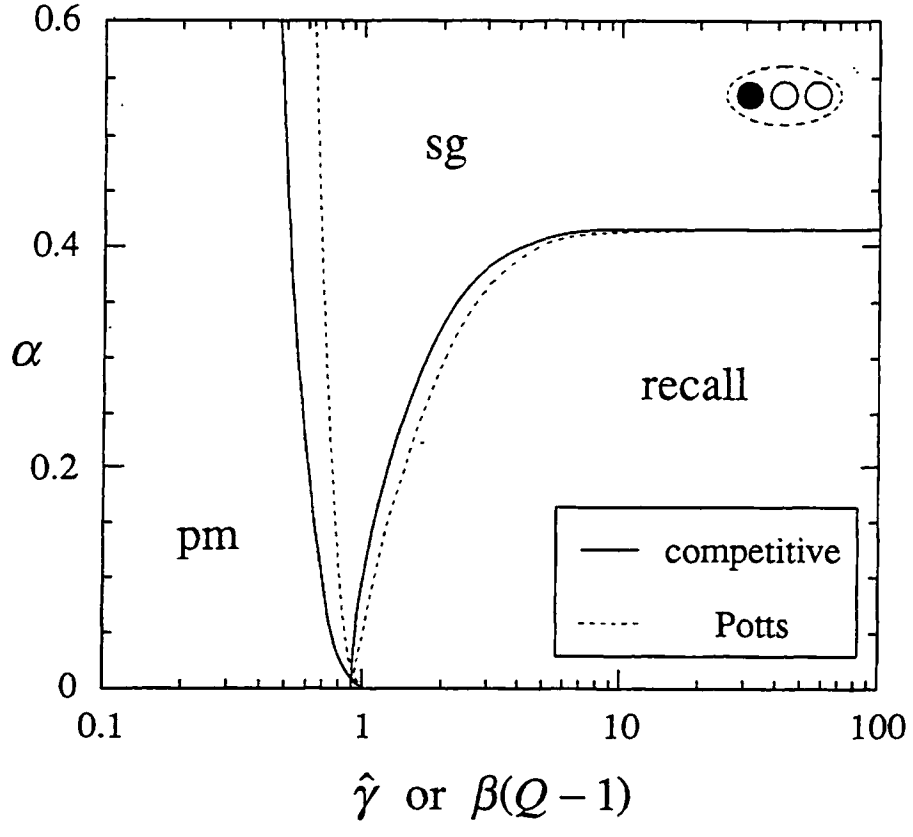


Fig. 2.9 Comparison of phase diagrams for winner-take-all competitive networks (solid curves) and Potts networks (dashed curves) for $Q = 3$. Horizontal axis is $\hat{\gamma}$ for competitive networks and $\beta(Q-1)$ for Potts networks, where β is inverse temperature. Data for Potts networks supplied by J. Huyghebaert [Bollé, Dupont, and Huyghebaert, 1992a,b].

2.5 STORAGE CAPACITY IN INFINITE-GAIN LIMIT

For a given network configuration, the maximum value of the storage capacity α_c is usually achieved in the limit of infinite neuron gain. In this limit, the mean-field equations (2.21) through (2.23) simplify considerably. All but one dimension of the integrals can be done analytically, and the three equations can be reduced to one. These simplifications arise because outputs of high-gain neurons take on one of only two values, which are $1/k$ or $-1/(Q-k)$, respectively, for winning or losing neurons.

	k						
Q	1	2	3	4	5	6	7
2	0.138						
3	0.414						
4	0.828	0.560					
5	1.375	0.799					
6	2.053	1.104	0.946				
7	2.859	1.468	1.159				
8	3.790	1.891	1.418	1.305			
9	4.844	2.371	1.718	1.501			
10	6.019	2.908	2.055	1.734	1.646		
11	7.312	3.502	2.430	1.998	1.831		
12	8.722	4.151	2.840	2.292	2.047	1.976	
13	10.25	4.856	3.287	2.612	2.288	2.153	
14	11.89	5.617	3.770	2.960	2.554	2.356	2.296
15	13.64	6.432	4.287	3.334	2.841	2.581	2.467

Table 2.1 Storage capacities $\alpha_c(Q,k)$ of winner-take-all and k-winner associative memories in limit of infinite neuron gain. Because of the symmetry relation $\alpha_c(Q,k) = \alpha_c(Q,Q-k)$, only capacities for $Q/k \geq 2$ are shown. Capacities are plotted in Fig. 2.10.

Infinite-gain storage capacities $\alpha_c(Q,k)$ are shown in Fig. 2.10 and Table 2.1 for $2 \le Q \le 15$ and $1 \le k \le 7$. The results hold also for zero-temperature Potts networks generalized to k-winner behavior, since the reaction field in these networks vanishes when $T = 0$. The capacities obey the symmetry relation $\alpha_c(Q,k) = \alpha_c(Q,Q-k)$, which is not outwardly apparent in the infinite-gain mean-field equations below. Some of the results for $k = 1$ have been reported previously for Potts networks [Kanter, 1988; Bollé, Dupont, and Huyghebaert, 1992a,b; Vogt and Zippelius, 1992].

Figure 2.10 and Table 2.1 are calculated from the infinite-gain mean-field equations

$$m = -\frac{k}{Q-k} + \frac{Q}{Q-k} I_m \tag{2.34}$$

$$C = \left[\frac{2Q}{\alpha\pi k(Q-k)(Q-1)}\right]^{1/2} I_C \tag{2.35}$$

$$q = \frac{1}{Q-1} \tag{2.36}$$

where

$$I_m = \int_{-\infty}^{\infty} \frac{dz}{\sqrt{\pi}} \exp\left(-z^2\right)$$

$$\times \sum_{m=0}^{k-1} \sum_{n=0}^{Q-k+m} K_{mn} \left[\frac{1}{2}\left(1+\mathrm{erf}(z)\right)\right]^{Q-k+m-n} \left[\frac{1}{2}\left(1+\mathrm{erf}(z+y)\right)\right]^{n} \tag{2.37}$$

and

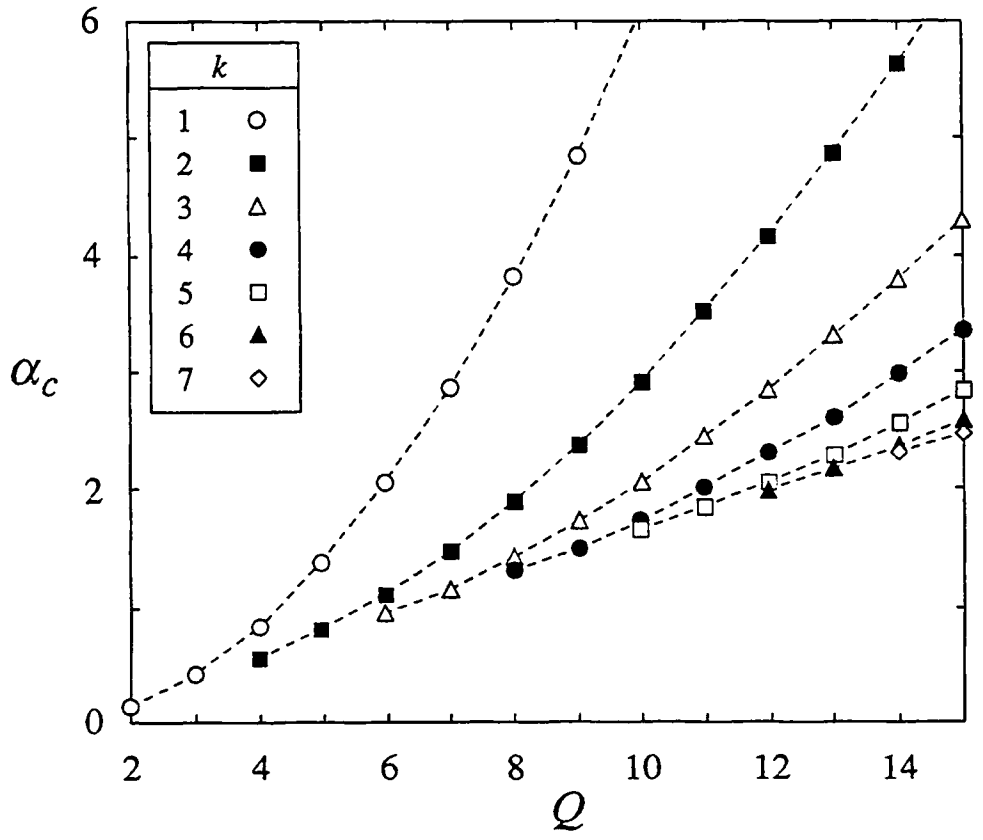


Fig. 2.10 Storage capacities $\alpha_c(Q,k)$ for winner-take-all and k-winner associative memories in infinite-gain limit. Because of symmetry relation $\alpha_c(Q,k) = \alpha_c(Q,Q-k)$, only capacities for $Q/k \geq 2$ are shown. Capacities also appear in Table 1.

$$I_C = \int_{-\infty}^{\infty} \frac{dz}{\sqrt{\pi}}\, z \exp\left(-z^2\right)$$

$$\times \sum_{m=0}^{k-1} \sum_{n=0}^{Q-k+m} \left\{ Q K_{mn} \left[\frac{1}{2}\left(1+\operatorname{erf}(z)\right)\right]^{Q-k+m-n} \left[\frac{1}{2}\left(1+\operatorname{erf}(z+y)\right)\right]^{n} \right.$$

$$\left. + \; (Q-k) L_{mn} \left[\frac{1}{2}\left(1+\operatorname{erf}(z)\right)\right]^{n} \left[\frac{1}{2}\left(1+\operatorname{erf}(z-y)\right)\right]^{Q-k+m-n} \right\}. \qquad (2.38)$$

The coefficients K_{mn} and L_{mn} are

$$K_{mn} = \binom{k-1}{Q-k+m-n}\binom{Q-k}{n}(-1)^m \frac{1}{m!}\prod_{p=1}^{m}(Q-k+p-1) \qquad (2.39)$$

$$L_{mn} = \binom{k}{Q-k+m-n}\binom{Q-k-1}{n}(-1)^m \frac{1}{m!}\prod_{p=1}^{m}(Q-k+p-1), \qquad (2.40)$$

and the quantity y is defined as

$$y \equiv m\left[\frac{Q(Q-1)}{2\alpha r k (Q-k)}\right]^{1/2}. \qquad (2.41)$$

Equations (2.34) through (2.36) reduce to a single equation for y:

$$y = \frac{Q-1}{Q-k}(QI_m - k)\left[\, 2I_C + \sqrt{2\alpha k(Q-k)/Q}\, \right]^{-1}. \qquad (2.42)$$

The infinite-gain storage capacities α_c of Fig. 2.10 and Table 2.1 indicate the value of α

at which the nonzero solution of Eq. (2.42) disappears. When $k = 1$, Eq. (2.42) reduces to previously reported results for Potts networks [Kanter, 1988; Vogt and Zippelius, 1992]. Because the replica-symmetric assumption breaks down for very large neuron gain, the storage capacities derived from Eq. (2.42) are not exact. By analogy with known results for Ising and Potts associative memories [Lage and Nunes da Silva, 1984; Crisanti, Amit, and Gutfreund, 1986; Bollé, Dupont, and Huyghebaert, 1992a,b], the exact storage capacities should be slightly higher than those appearing in Fig. 2.10 and Table 2.1.

2.6 VERIFYING THE PHASE DIAGRAMS

This section presents tests of the phase diagrams derived in Sec. 2.4 using computer-generated competitive associative memories with discrete-time, parallel updating. Results are shown in Fig. 2.11. Each panel of Fig. 2.11 tests one of the phase diagrams in Fig. 2.7 along a horizontal line. The panels show, as a function of the gain parameter $\hat{\gamma}$, the fraction of randomly-generated initial conditions that flow to each of the four possible types of attractors—the paramagnetic attractor, a recall attractor, a spin-glass attractor, or an oscillatory attractor. The first three panels in Fig. 2.11 are for three different winner-take-all network configurations: Fig. 2.11(a) is for networks with $N = 100$ clusters, $Q = 3$ neurons per cluster, and storage fraction $\alpha = 0.2$; Fig. 2.11(b) is for networks with $N = 75$, $Q = 4$, and $\alpha = 0.4$; and Fig. 2.11(c) is for networks with $N = 60$, $Q = 5$, and $\alpha = 0.8$. The last panel, Fig. 2.11(d), is for k-winner networks with $N = 75$ clusters, $Q = 4$ neurons per cluster, $k = 2$ winning neurons per cluster, and storage fraction $\alpha = 0.2$.

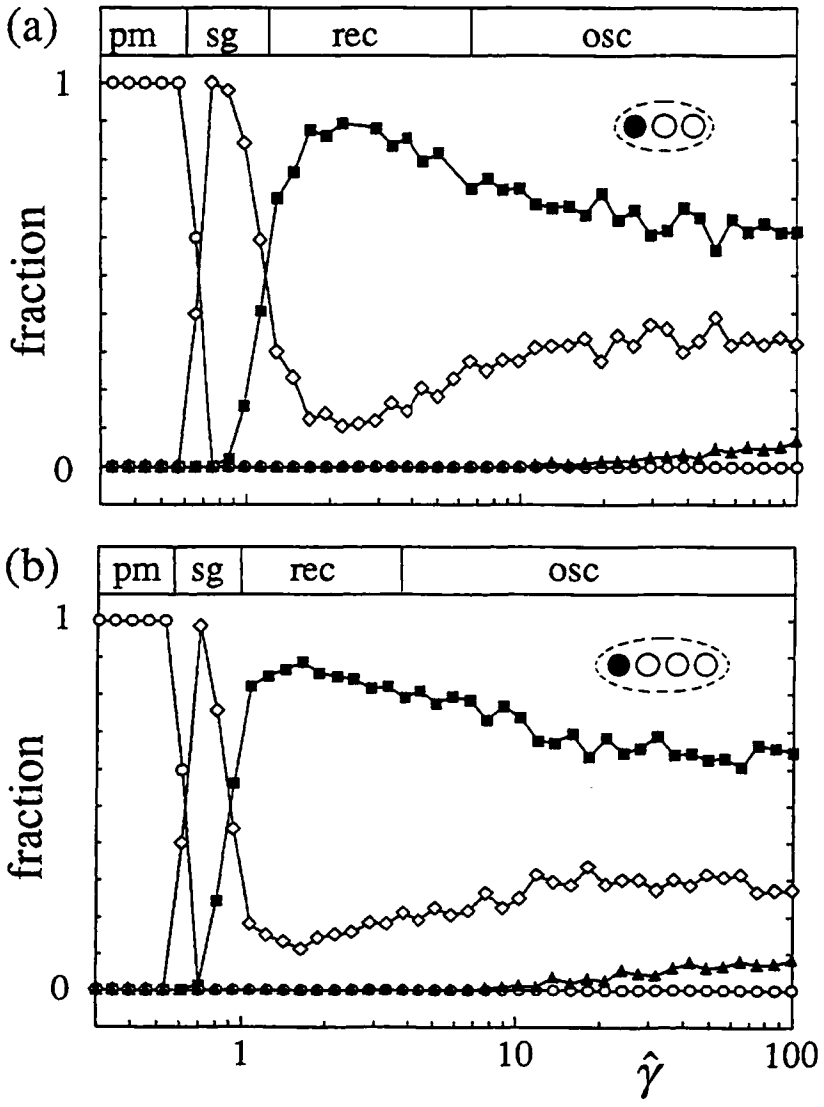


Fig. 2.11 Fraction of randomly generated initial conditions flowing to paramagnetic (circles), spin glass (diamonds), recall (squares), and oscillatory (triangles) attractors in small, computer-generated competitive networks with discrete-time parallel updating. (a) Winner-take-all networks with $N = 100$, $Q = 3$, and $\alpha = 0.2$; (b) winner-take-all networks with $N = 75$, $Q = 4$, and $\alpha = 0.4$. Each panel tests one phase diagram of Fig. 2.7 along a horizontal line; at top of each panel, boxes labelled "pm," "sg," "recall," and "osc" indicate corresponding phase diagram regions. Data strongly support phase diagrams.

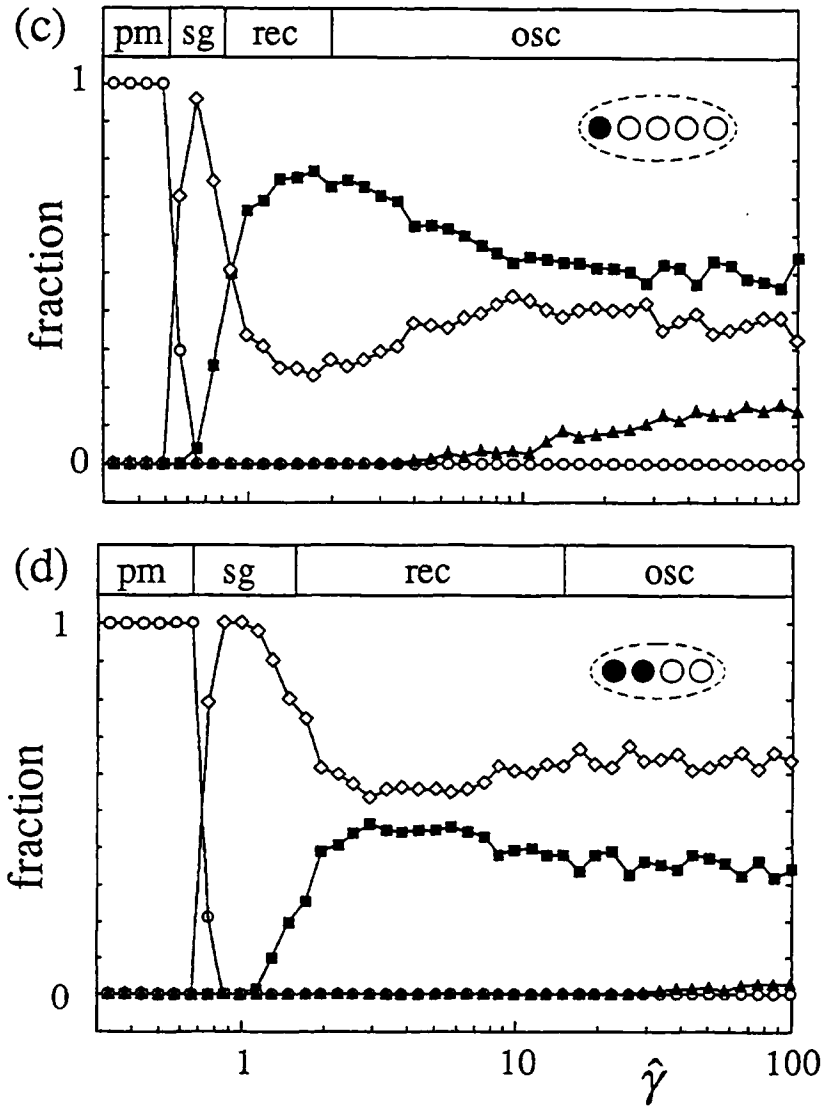


Fig. 2.11 Fraction of randomly generated initial conditions flowing to paramagnetic (circles), spin glass (diamonds), recall (squares), and oscillatory (triangles) attractors in small, computer-generated competitive networks with discrete-time parallel updating. (c) Winner-take-all networks with $N = 60$, $Q = 5$, and $\alpha = 0.8$; (d) k-winner networks with $N = 75$, $Q = 4$, $k = 2$, and $\alpha = 0.2$. Each panel tests one phase diagram of Fig. 2.7 along a horizontal line; at top of each panel, boxes labelled "pm," "sg," "recall," and "osc" indicate corresponding phase diagram regions. Data strongly support phase diagrams.

The data in Fig. 2.11 were generated by starting from random initial conditions

$$x_{ia}(0) = \frac{1}{k(Q-k)}\left(Q\delta_{a,b_i} - k\right), \tag{2.43}$$

where each b_i, $i = 1, \ldots, N$, is a set of k different integers chosen randomly and without bias from the set $\{1, \ldots, Q\}$. (In applications, the initial conditions would not be random but rather would have some overlap with on of the stored patterns.) A total of 500 initial conditions were used for each gain value, 20 from each of 25 different interconnection matrices constructed according to Eq. (2.8). The update equations (2.2) were then iterated until convergence to a fixed point or period-two attractor. The fixed points were classified into three categories: recall attractors, for which one of the thresholded overlaps

$$m_\mu^{thr} \equiv \frac{1}{Nk}\sum_{i=1}^{N} \text{sign}\left(x_{i\xi_i^\mu}\right), \tag{2.44}$$

is greater than 0.9; paramagnetic attractors, for which $x_{ia} = 0$ for all i and a; and spin-glass attractors, which are all other fixed-point attractors.

At the top of each panel are four boxes containing the names of the different attractor types—"recall," "pm," "sg," and "osc"—that can occur in discrete-time networks. These boxes indicate the location on the appropriate phase diagram in Fig. 2.7 of the four phase regions for the particular value of α used in that panel. The paramagnetic/spin-glass transitions and the spin-glass/recall transitions occur at values of $\hat{\gamma}$ predicted by the phase diagrams; the lack of sharpness in these transitions is the result of finite-size effects. Oscillatory attractors appear at values of $\hat{\gamma}$ somewhat higher than that given by the stability criterion, which is however a worst-case result. Delayed appearance of oscillatory

attractors in competitive networks can be understood using geometric arguments [Waugh and Westervelt, 1993a]. One notable feature of Fig. 2.11 is that recall ability decreases as $\hat{\gamma}$ increases within the recall region. Improved performance at lower gain, sometimes referred to as deterministic annealing, has been observed in a variety of analog systems [Hopfield, 1984; Koch, Marroquin, and Yuille, 1986; Hopfield and Tank, 1986; Durbin and Willshaw, 1987; Marcus, Waugh, and Westervelt, 1990] and has been investigated analytically in standard analog associative memories [Waugh, Marcus, and Westervelt, 1990, 1991; Fukai and Shiino, 1990]. Deterministic annealing is the topic of Chapter 3.

2.7 SUMMARY

Despite some rather complicated calculations, the main results of this chapter—the phase diagrams of Sec. 2.4—are simple and useful. The diagrams map out the regions of parameter space where recall, paramagnetic, spin glass, and oscillatory attractors occur for competitive associative memories with extensive memory loading ($p = \alpha N$). The most important attractors, of course, are the recall states, and the diagrams indicate that these exist over a large range of storage fraction and transfer function gain. The diagrams can be used as guidelines for designing competitive associative memory networks of resistively coupled nonlinear amplifiers that either evolve in continuous time or are clocked externally. Other results of the chapter include bifurcation diagrams for the memory overlap in finitely-loaded networks and storage capacities for k-winner networks with infinite neuron gain.

As suggested by the numerical results of Sec. 2.6, however, the phase diagrams do not tell the whole story. Within the recall region, the ability to arrive at a stored memory rather than a spurious spin-glass state varies widely as a function of gain. The basins for spurious attractors tend to crowd out those for recall attractors as gain increases, decreasing

the probability for recalling a stored pattern. A way of improving network performance, then, is *deterministic annealing*, a process similar to simulated annealing for stochastic systems [Kirkpatrick, Gelatt, and Vecchi, 1983] except that neuron gain rather than temperature is changed. In deterministic annealing, the clusters start with low gains and end with high ones for each initial condition, and the phase diagrams place limits on how low or high the gains can be without leaving the recall region. The next chapter shows how deterministic annealing can lead to an exponential decrease in spurious attractors in analog associative memories.

APPENDIX 2A: MEAN-FIELD EQUATIONS FOR FINITE LOADING

This appendix derives the mean-field equations (2.12) through (2.14) for finite memory loading. Assuming all clusters to have the same Q transfer functions $F_a(z)$, $a = 1, \ldots, Q$, and using Eq. (2.8) for the interconnection matrix, the free energy per neuron (2.9) averaged over all possible realizations of the patterns ξ_i^μ is

$$f = -\frac{1}{\tilde{\beta}N}\left\langle \ln \int \prod_{i,a} [d\rho(x_{ia})] \exp\tilde{\beta}\left[\frac{1}{2N}\sum_\mu \left(\sum_i x_{i\xi_i^\mu}\right)^2 - \frac{1}{2N}\sum_{i,\mu}\left(x_{i\xi_i^\mu}\right)^2 - \sum_{i,a} G_a(x_{ia})\right]\right\rangle_\xi . \qquad (2A.1)$$

The brackets $\langle \cdots \rangle_\xi$ indicate pattern averaging. The squared sum in the exponent is made linear in the neuron outputs using a Gaussian identity, which introduces the p overlaps m_μ, $\mu = 1, \ldots, p$ [Amit, Gutfreund, and Sompolinsky, 1987; Amit, 1989]. In the limit $N \to \infty$, the second term in the exponent vanishes, the integrals over the overlaps can be done by saddle-point integration, and sums over the cluster index i are self-averaging. The resulting free energy is

$$f = \frac{1}{2}\sum_\mu m_\mu^2 - \frac{1}{\tilde{\beta}}\left\langle \ln \int \prod_a [d\rho(x_a)] \exp\tilde{\beta}\left[\sum_\mu m_\mu x_{\xi^\mu} - \sum_a G_a(x_a)\right]\right\rangle_\xi , \qquad (2A.2)$$

where the overlaps satisfy the saddle-point equations

$$m_\mu = \left\langle \left\langle x_\xi{}^\mu \right\rangle_x \right\rangle_\xi . \tag{2A.3}$$

The brackets $\langle \cdots \rangle_x$ in (2A.3) indicate an average over the integrand appearing in the free energy:

$$\langle \cdots \rangle_x \rightarrow \frac{\int \prod_a [d\rho(x_a)]\, I\, (\cdots)}{\int \prod_a [d\rho(x_a)]\, I}, \tag{2A.4}$$

where

$$I \equiv \exp \tilde{\beta} \left[\sum_\mu m_\mu \, x_\xi{}^\mu - \sum_a G_a(x_a) \right]. \tag{2A.5}$$

As discussed in Sec. 2.2, only the $\tilde{\beta} \rightarrow \infty$ limit of the free energy is of interest. In this limit, the integrals over the neuron outputs x_a can be done by saddle-point integration. The saddle-point equations are determined by maximizing I with respect to the x_a, subject to the competitive constraint $\sum_a x_a = 0$. Using a Lagrange multiplier B to enforce this constraint leads to the following saddle-point equations:

$$x_a = \left\langle F_a \left(\sum_\mu m_\mu \delta_{a,\xi^\mu} + B \right) \right\rangle_\xi , \quad a = 1, \ldots, Q. \tag{2A.6}$$

Inserting (2A.6) into the free energy (2A.2) and the overlaps (2A.3) leads to Eqs. (2.12) and (2.13).

APPENDIX 2B: STABILITY OF FINITE LOADING SOLUTIONS

The stability of the overlap solutions in finitely-loaded networks is determined by the eigenvalue spectrum of the matrix $\partial^2 f / \partial m_\rho \partial m_\sigma$, where f is the free energy (2.12). A solution is stable if all eigenvalues of this matrix are positive. When all neurons have the same transfer function $F(z)$, the matrix is

$$\frac{\partial^2 f}{\partial m_\rho \partial m_\sigma} = \left\langle \delta_{\rho,\sigma} - \delta_{\xi^\rho,\xi^\sigma}\, F'\left(h_\xi^\rho + B\right) + \frac{F'\left(h_\xi^\rho + B\right) F'\left(h_\xi^\sigma + B\right)}{\sum_a F'\left(h_a + B\right)} \right\rangle_\xi , \qquad (2B.1)$$

where $h_a \equiv \sum_\mu m_\mu \delta_{a,\xi^\mu}$ and $F'(z)$ is the transfer function derivative. In the case $m_\mu = m\delta_{\mu,1}$ of a single successfully recalled pattern, the matrix is diagonal, with the following two eigenvalues:

$$\lambda_1 = 1 - \frac{1}{S} k(Q-k)\, F'(m+B)\, F'(B) \qquad (2B.2)$$

$$\lambda_2 = 1 - \frac{1}{QS}\Big[k(k-1)\, F'(m+B)^2 + (Q-k)(Q-k-1)F'(B)^2 + 2k(Q-k)F'(m+B)F'(B)\Big], \qquad (2B.3)$$

where

$$S \equiv kF'(m+B) + (Q-k)F'(B). \qquad (2B.4)$$

The eigenvalue λ_1 indicates stability of a solution with respect to perturbations of the nonzero overlap m_1, while λ_2 indicates stability with respect to perturbations of the other $p-1$ overlaps m_μ, $\mu > 1$. Note that the eigenvalues λ_1 and λ_2, unlike the mean-field equation (2.20) for k-winner networks, are not invariant to replacing Q and k by nQ and nk for positive integers n. Thus the stability of solutions of Eq. (2.20) depends on Q, even though the solutions themselves depend only on the ratio Q/k.

APPENDIX 2C: MEAN-FIELD EQUATIONS FOR EXTENSIVE LOADING

This appendix sketches the derivation of the mean-field equations (2.21) through (2.23) for extensive memory loading. The derivation follows the standard replica approach [for details, see Amit, Gutfreund, and Sompolinsky, 1987; Amit, 1989; Kühn, Bös, and van Hemmen, 1991] which uses the identity

$$\ln z = \lim_{n\to 0}\left(z^n - 1\right) \tag{2C.1}$$

to write the free energy (2.9) per neuron, averaged over all possible realizations of the patterns, as

$$f = \lim_{n\to 0} -\frac{1}{\tilde{\beta} n N}\left(\left\langle \int \prod_{i,a,\sigma}\left[d\rho\left(x_{ia}^{\sigma}\right)\right] \exp\tilde{\beta}\left[\frac{1}{2N}\sum_{\mu,\sigma}\left(\sum_i x_{i\xi_i^\mu}^{\sigma}\right)^2 - \frac{1}{2N}\sum_{i,\mu,\sigma} x_{i\xi_i^\mu}^{\sigma} x_{i\xi_i^\mu}^{\sigma} - \sum_{i,a,\sigma} G_a\left(x_{ia}^{\sigma}\right)\right]\right\rangle_{\xi} - 1\right), \tag{2C.2}$$

where the brackets $\langle\cdots\rangle_\xi$ indicate pattern averaging. The index σ, which labels the replicas, runs from 1 to n. To make the squared sum in (2C.2) linear in the neuron outputs, the p pattern overlaps m_μ^σ, $\mu = 1, \ldots, p$, are introduced using a Gaussian identity. The patterns are separated into a finite number s of "condensed" patterns with nonzero overlaps and an infinite number $p - s$ of "uncondensed" patterns with vanishing overlaps. The average over uncondensed patterns is performed assuming that each pattern ξ_i^μ occurs with equal probability $k!(Q-k)!/Q!$, and the overlaps corresponding to these patterns are integrated out. After using self-averaging to remove the site index i and taking the $N \to \infty$ limit, the free energy is

$$
\begin{aligned}
f \;=\; \lim_{n\to 0} \frac{1}{n} \exp \Bigg\{ & \frac{1}{2} \sum_{\nu,\sigma} \left(m_\nu^\sigma\right)^2 + \frac{\alpha}{2\tilde{\beta}} \operatorname{Tr} \ln\left(\mathbf{I} - \tilde{\beta}\mathbf{q}_0 - \tilde{\beta}\mathbf{q}\right) \\
& + \frac{\alpha\tilde{\beta}}{2}\left(\sum_\sigma q_0^\sigma r_0^\sigma + \sum_{\sigma\neq\sigma'} q^{\sigma\sigma'} r^{\sigma\sigma'} \right) \\
& - \frac{1}{\tilde{\beta}} \Bigg\langle \ln \int \prod_{a,\sigma} \left[d\rho\left(x_a^\sigma\right) \right] \exp \Bigg[\tilde{\beta} \sum_{\nu,\sigma} m_\nu^\sigma x_{\xi^\nu}^\sigma - \tilde{\beta} \sum_{a,\sigma} G_a\left(x_a^\sigma\right) \\
& + \frac{\alpha\tilde{\beta}}{2Q} \frac{k(Q-k)}{Q-1} \left\{ \sum_{a,\sigma} \left(\tilde{\beta} r_0^\sigma - 1\right)\left(x_a^\sigma\right)^2 + \tilde{\beta} \sum_{a,\sigma\neq\sigma'} r^{\sigma\sigma'} x_a^\sigma x_a^{\sigma'} \right\} \Bigg] \Bigg\rangle_\xi . \qquad \text{(2C.3)}
\end{aligned}
$$

The quantities m_ν^σ, q_0^σ, r_0^σ, $q^{\sigma\sigma'}$, and $r^{\sigma\sigma'}$ are order parameters. The matrices $\mathbf{I}$, $\mathbf{q}_0$, and $\mathbf{q}$ in Eq. (2C.3) are $n \times n$; $\mathbf{I}$ is the identity matrix, $\mathbf{q}_0$ is a diagonal matrix with diagonal elements equal to q_0^σ, and $\mathbf{q}$ is a symmetric matrix with diagonal elements equal to

0 and off-diagonal elements equal to $q^{\sigma\sigma'}$. The pattern average $\langle\cdots\rangle_\xi$ is now over the s condensed patterns ξ^ν, $\nu = 1, \ldots, s$.

At this point replica symmetry is assumed, meaning that the values of the order parameters in Eq. (2C.3) are independent of replica index σ. Physically, replica symmetry means that there is only one fixed point in the vicinity of a stored memory. By analogy with Ising associative memories, for which replica-symmetry-breaking effects are small, the replica-symmetric solution should be correct over a wide range of neuron gain but should break down for very high gain [Crisanti, Amit, and Gutfreund, 1986; Kohring, 1990].

Applying replica symmetry and taking the limit $n \to 0$ yields

$$f = \frac{1}{2}\sum_\nu m_\nu^2 + \frac{1}{2}\alpha\left\{\frac{1}{\tilde\beta}\ln\left[1-\tilde\beta(q_0-q)\right] - \frac{q}{1-\tilde\beta(q_0-q)} + q_0\tilde r + \tilde\beta(q_0-q)r\right\}$$

$$- \frac{1}{\tilde\beta}\left\langle \ln\int\prod_a\left[d\rho(x_a)\right]\exp\tilde\beta\left[\sum_\nu m_\nu x_{\xi^\nu} - \sum_a G_a(x_a)\right.\right.$$

$$\left.\left. + \frac{\alpha}{2Q}\frac{k(Q-k)}{Q-1}(\tilde r - 1)\sum_a x_a^2 + \sqrt{\frac{\alpha r}{Q}\frac{k(Q-k)}{Q-1}}\sum_a z_a x_a\right]\right\rangle_{z,\xi} \qquad \text{(2C.4)}$$

where $\tilde r \equiv \tilde\beta(r_0 - r)$. The brackets $\langle\cdots\rangle_{z,\xi}$ stand for an average over both the s condensed patterns ξ^ν and the Q continuous variables z_a using a Gaussian distribution:

$$\langle\cdots\rangle_{z,\xi} \to \left\langle\int\prod_a\left(\frac{dz_a}{\sqrt{2\pi}}\right)\exp\left(-\frac{1}{2}\sum_a z_a^2\right)(\cdots)\right\rangle_\xi . \qquad \text{(2C.5)}$$

The saddle-point equations for m_ν, q_0, q, r, and $\tilde{r}$ are calculated by setting the partial derivatives of f with respect to these variables equal to zero:

$$m_\nu = \left\langle\left\langle x_\xi{}^\nu \right\rangle_x\right\rangle_{z,\xi} \tag{2C.6}$$

$$\tilde{\beta}(q_0 - q) = \sqrt{\frac{1}{\alpha r Q}\frac{k(Q-k)}{Q-1}}\left\langle\left\langle \sum_a z_a x_a \right\rangle_x\right\rangle_{z,\xi} \tag{2C.7}$$

$$q_0 = \frac{1}{Q}\frac{k(Q-k)}{Q-1}\left\langle\left\langle \sum_a x_a^2 \right\rangle_x\right\rangle_{z,\xi} \tag{2C.8}$$

$$r = \frac{q}{\left[1-\tilde{\beta}(q_0-q)\right]^2} \tag{2C.9}$$

$$\tilde{r} = \frac{1}{1-\tilde{\beta}(q_0-q)} \tag{2C.10}$$

where

$$\langle \cdots \rangle_x \rightarrow \frac{\int \prod_a [d\rho(x_a)]\, I\,(\cdots)}{\int \prod_a [d\rho(x_a)]\, I} \tag{2C.11}$$

and I is the integrand appearing in double brackets in f:

$$I = \exp\tilde{\beta}\left[\sum_{\nu} m_{\nu} x_{\xi}^{\nu} - \sum_{a} G_a(x_a)\right.$$

$$\left. + \frac{\alpha}{2Q}\frac{k(Q-k)}{Q-1}(\tilde{r}-1)\sum_{a} x_a^2 + \sqrt{\frac{\alpha r}{Q}\frac{k(Q-k)}{Q-1}}\sum_{a} z_a x_a\right]. \quad \text{(2C.12)}$$

At the saddle point, the free energy can be written as

$$f = \frac{1}{2}\sum_{\nu} m_{\nu}^2 + \frac{1}{2}\alpha\left\{\frac{1}{\tilde{\beta}}\ln\left[1-\tilde{\beta}(q_0-q)\right] + (q_0-q)\tilde{r} + \tilde{\beta}(q_0-q)r\right\}$$

$$- \frac{1}{\tilde{\beta}}\left\langle \ln\int\prod_{a}\left[d\rho(x_a)\right] I\right\rangle_{z,\xi}. \quad \text{(2C.13)}$$

Only the $\tilde{\beta}\to\infty$ limit of the saddle-point equations (2C.6) through (2C.10) is of interest. In this limit, the integrals over the neuron outputs x_a can themselves be carried out by saddle-point integration. This entails finding the values $\hat{x}_a$ for which the argument of the exponent in Eq. (2C.12) is a maximum, subject to the competitive constraint $\sum_a \hat{x}_a = 0$. Using a Lagrange multiplier B to enforce this constraint leads to the Q saddle-point equations for the $\hat{x}_a$ that appear as Eq. (2.24). These values of $\hat{x}_a$ are then inserted into (2C.6) through (2C.8) to yield the saddle-point equations (2.21) through (2.23). Note that, since the right-hand side of Eq. (2C.7) is finite as $\tilde{\beta}\to\infty$, q_0 must approach q in such a way that $\tilde{\beta}(q_0-q)\equiv C$ approaches a finite value.

APPENDIX 2D: PARAMAGNETIC/SPIN-GLASS BOUNDARY

This appendix derives the boundary (2.27) between the paramagnetic and spin-glass regions when the transition between these regions is continuous. The procedure is to expand Eq. (2.24) to leading order in the quantities $\hat{x}_a$, with all overlaps m_ν equal to zero. Inserting the result into the saddle-point equations (2.22) and (2.23) allows the integrals in these equations to be done analytically.

Expanding Eq. (2.24) around $\hat{x}_a = 0$ when all neurons have the same transfer function $F(z)$ leads to

$$\hat{x}_a \;=\; \Phi\left[\alpha\left(\tilde{r}-1\right)U\hat{x}_a \;+\; \alpha r V z_a \;+\; \delta B\right], \tag{2D.1}$$

where $\Phi \equiv F'\left(F^{-1}(0)\right)$ and

$$U \;\equiv\; \frac{1}{Q}\frac{k(Q-k)}{Q-1}, \qquad V \;\equiv\; \sqrt{\frac{1}{\alpha r Q}\frac{k(Q-k)}{Q-1}}. \tag{2D.2}$$

The quantity δB, which is first-order in $\hat{x}_a$ and z_a, is determined by the competitive constraint $\sum_a \hat{x}_a = 0$:

$$\delta B \;=\; -\frac{1}{Q}\sum_a \left[\alpha\left(\tilde{r}-1\right)U\hat{x}_a \;+\; \alpha r V z_a\right]. \tag{2D.3}$$

Inserting (2D.3) into (2D.1) and solving for $\hat{x}_a$ yields

$$\hat{x}_a = \frac{\alpha r V \Phi}{1 - \alpha(\tilde{r}-1)U\Phi} \sum_b M_{ab}\, z_b , \tag{2D.4}$$

where $M_{ab} \equiv \delta_{ab} - 1/Q$. The expansion (2D.4) is then inserted into the saddle-point equations (2.22) and (2.23) for q and C:

$$q = U\left(\frac{\alpha r V \Phi}{1 - \alpha(\tilde{r}-1)U\Phi}\right)^2 \int \prod_a \left(\frac{dz_a}{\sqrt{2\pi}}\right) \exp\left(-\frac{1}{2}\sum_a z_a^2\right) \sum_{abc} M_{ab} M_{ac} z_b z_c$$

$$= U\left(\frac{\alpha r V \Phi}{1 - \alpha(\tilde{r}-1)U\Phi}\right)^2 (Q - 1) \tag{2D.5}$$

$$C = V\left(\frac{\alpha r V \Phi}{1 - \alpha(\tilde{r}-1)U\Phi}\right) \int \prod_a \left(\frac{dz_a}{\sqrt{2\pi}}\right) \exp\left(-\frac{1}{2}\sum_a z_a^2\right) \sum_{ab} M_{ab} z_a z_b$$

$$= V\left(\frac{\alpha r V \Phi}{1 - \alpha(\tilde{r}-1)U\Phi}\right) (Q - 1) \tag{2D.6}$$

Inserting the expressions (2.26) for r and $\tilde{r}$ into Eqs. (2D.5) and (2D.6) produces two equations in the two quantities Φ and C; eliminating C yields the boundary curve of Eq. (2.27).

CHAPTER 3*

FIXED-POINT ATTRACTORS AND DETERMINISTIC ANNEALING

3.1 INTRODUCTION: OPTIMIZATION

Optimization problems arise in physics, mathematics, engineering, and a variety of other fields. Typically, they involve searching a large state space for a solution that best satisfies a set of conflicting constraints as embodied in some cost function. The search is made difficult by the size of the state space and by the fact that the constraints give rise to many local minima in the cost function, the vast majority of which correspond to poor solutions. Systems exhibiting these features include spin glasses, neural networks, data clustering algorithms, problems in artificial vision such as edge detection and stereoscopic depth perception, and more traditional optimization problems such as the travelling salesman and the graph partitioning problems (see Sec. 1.4.1). [An extensive literature exists on the statistical mechanics of optimization: for spin glasses see Tanaka and Edwards, 1980; De Dominicis *et al.*, 1980; Bray and Moore, 1980, 1981; Gutfreund, Reger, and Young, 1988; Takayama and Nemoto, 1990; for neural networks see Gardner, 1986; Treves and Amit, 1988; Fukai, 1990; for data clustering see Rose, Gurewitz, and Fox, 1990; for artificial vision see Koch, Marroquin, and Yuille, 1986; Blake and Zisserman, 1987; Hentschel and Fine, 1989; for travelling salesman and graph partitioning

*Versions of this chapter have appeared as *Phys. Rev. Lett.* **64**, 1986 (1990) and *Phys. Rev. A* **43**, 3131 (1991).

problems see Hopfield and Tank, 1985, 1986; Basharan, Fu, and Anderson, 1986; Fu and Anderson, 1986; Burgess and Moore, 1989.]

A variety of heuristic methods have been developed that can quickly find locally optimal solutions to these problems. A popular approach is to regard the cost function as the energy of a high-dimensional dynamical system and to interpret steady-state configurations of the system as solutions. In this approach, the details of the system's dynamics often strongly affect its performance in practical applications. Stochastic techniques like the Metropolis algorithm [Metropolis *et al.*, 1953] or simulated annealing [Kirkpatrick, Gelatt, and Vecchi, 1982] can find good solutions because they allow a system to climb out of local minima. However, the time required to reach the global optimum can be prohibitively long [Geman and Geman, 1984]. Moreover, such algorithms are computationally expensive when implemented on a computer and can be difficult to implement in electronics because of the necessity of local noise generators [Alspector, Gupta, and Allen, 1989; Alspector *et al.*, 1991]. In contrast, systems with deterministic dynamics, the focus of this thesis, are generally faster and more suitable for hardware implementation, but they may yield poor solutions because they lack the ability to climb out of local minima.

As has already been discussed in Chapters 1 and 2, neural networks are well-suited for finding solutions of difficult optimization problems. Their performance depends not only on the dynamical considerations discussed above but also on the input/output characteristics of individual neurons. Several authors [Hopfield and Tank, 1985, 1986; Bilbro and Snyder, 1989; Marcus, Waugh, and Westervelt, 1990] have observed that decreasing the maximum slope, or gain, of the neuron transfer function improves the quality of solutions found by a neural network in a way that qualitatively resembles the results of stochastic techniques. The benefits of using a reduced gain have been observed numerically in a

variety of other optimization problems [Soukoulis, Levin, and Grest, 1982, 1983; Koch, Marroquin, and Yuille, 1986; Blake and Zisserman, 1987; Durbin and Willshaw, 1987].

This chapter shows both analytically and numerically that using analog neurons dramatically reduces the number of local minima in the energy landscape of a standard associative memory neural network with deterministic dynamics. Specifically, the number of local minima averaged over realizations of the interconnection matrix is shown to increase exponentially with the number of neurons N as $\exp(NF)$. The chapter's main result is the calculation of the scaling exponent F as a function of the number of stored memories and of the neuron gain. For a standard sigmoidal transfer function, the scaling exponent (and therefore the number of local minima) decreases as the neuron gain is reduced. For associative memories, this behavior implies that reducing the gain leads to considerable improvement in computational performance, since a vast majority of the local minima eliminated are spurious states, which have negligible overlap with the stored memory patterns (see Fig. 3.1). As discussed at the end of Chapter 2, reducing neuron gain to improve performance is known as *deterministic annealing*, by analogy to simulated annealing for stochastic systems [Kirkpatrick, Gelatt, and Vecchi, 1983]. The phase diagrams of that chapter are essential guides for implementing deterministic annealing in associative memories.

The rest of this chapter is organized as follows. In Sec. 3.2, the scaling exponent F is calculated for analog associative memories using techniques previously developed to count solutions of the Thouless-Anderson-Palmer (TAP) equations for infinite-range Ising spin glasses [Thouless, Anderson, and Palmer, 1977; Bray and Moore, 1980] and to count metastable states of neural networks with two-state neurons [Gardner, 1986; Bruce, Gardner, and Wallace, 1987]. Section 3.3 considers the limits of high memory loading

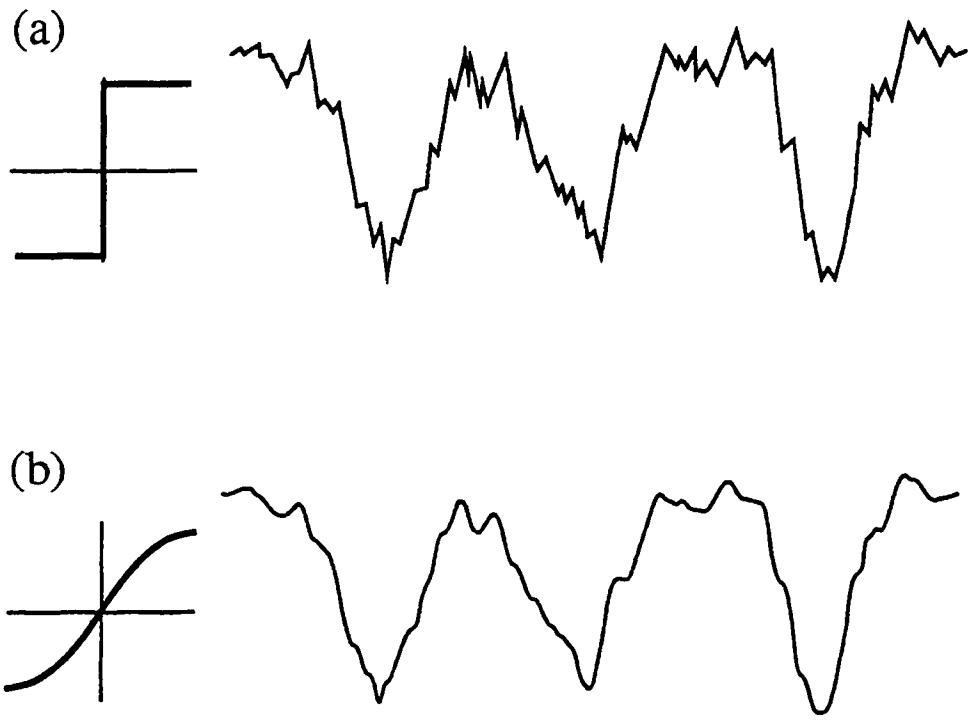


Fig. 3.1 Schematic landscapes showing energy or a Liapunov function (vertically) vs. abstract phase space coordinate (horizontally) for networks with (a) two-state neurons; (b) analog neurons with finite gain. Insets depict neuron transfer functions. Decreasing neuron gain smooths energy landscape, greatly reducing number of spurious states.

and high neuron gain. Numerical counts of fixed points in small networks, presented in Sec. 3.4, agree well with the analytical calculations.

3.2 FIXED POINTS IN ANALOG ASSOCIATIVE MEMORIES

3.2.1 Network architectures and dynamics

In contrast to the last two chapters, this chapter focuses on standard analog networks like those discussed in Sec. 1.1, which may be regarded as competitive networks with all $Q_i = 2$. The networks are defined by either the continuous-time system,

$$\frac{dm_i(t)}{dt} = -m_i(t) + f\left(\beta \sum_{j=1}^{N} J_{ij} m_j(t)\right), \quad i=1,\ldots,N, \tag{3.1}$$

or the discrete-time system

$$m_i(t+1) = f\left(\beta \sum_{j=1}^{N} J_{ij} m_j(t)\right), \quad i=1,\ldots,N. \tag{3.2}$$

The updating rule of Eq. (3.2) may be either serial or parallel. The real-valued quantity $m_i(t)$ denotes the output of neuron i at time t. All neurons have the same neuron gain β and the same transfer function $f(x)$, which may be any single-valued, continuous function; the commonly used form $f(x) = \tanh(x)$ is often used. In keeping with the literature, the notation used in (3.1) and (3.2) is slightly different from the previous

chapters: the neuron outputs are m_i instead of x_i, and the transfer function is $f(\beta z) = F(z)$ (the steepest slope of $f(z)$ is 1).

The interconnection matrix is given by the Hebb rule for associative memory,

$$J_{ij} = \left(N\sqrt{\alpha}\right)^{-1} \sum_{\mu=1}^{\alpha N} \xi_i^\mu \xi_j^\mu, \quad i \neq j; \quad J_{ii} = 0, \tag{3.3}$$

where the memory patterns ξ_i^μ are assumed to take on the values ± 1 with equal probability. The normalization $\left(N\sqrt{\alpha}\right)^{-1}$ is chosen to make the magnitude of the J_{ij}'s independent of the ratio α of stored memories to neurons for large N. (In Eq. (3.3) and below, Roman indices range from 1 to N and Greek indices range from 1 to αN unless explicitly stated otherwise.)

Fixed points of Eqs. (3.1) and (3.2) satisfy the condition

$$G_i \equiv f^{-1}(m_i) - \beta \sum_j J_{ij} m_j = 0. \tag{3.4}$$

Although Eqs. (3.1) and (3.2) have the same fixed points, in general their dynamical behavior can be different. Under the conditions of sigmoidal transfer functions and symmetric interconnection matrices with zero diagonals, Eq. (3.1) has only fixed point attractors, as does Eq. (3.2) when updated serially. Under the same conditions when updated in parallel, Eq. (3.2) has only fixed points if $1/\beta > -\lambda_{min}$, where λ_{min} is the smallest eigenvalue of the matrix J_{ij} ($\lambda_{min} = -\sqrt{\alpha}$ for the Hebb rule for $\alpha < 1$), but may have period-two limit cycles in addition to fixed points if $1/\beta < -\lambda_{min}$ [see Sec. 1.3.2 and Marcus and Westervelt, 1989; Marcus, Waugh, and Westervelt, 1990]. Moreover, the

basin boundaries are likely to be different for Eqs. (3.1) and (3.2). However, these differences are irrelevant to the analysis, which concerns the number of stable fixed points and not the dynamics far from fixed points.

The fixed-point condition of the analog associative memory network is similar in form to the TAP mean-field equations for the infinite-range Ising spin glass at finite temperature, which are [Thouless, Anderson, and Palmer, 1977]

$$\tanh^{-1}(m_i) - \beta \sum_j J_{ij} m_j + m_i \beta \sum_j J_{ij}^2 \left(1 - m_j^2\right) = 0. \tag{3.5}$$

In Eq. (3.5), m_i is the thermally averaged magnetization at site i, β is the inverse temperature, and J_{ij} is the coupling between spins at sites i and j, usually chosen from a gaussian probability distribution. The third term on the left side of (3.5), known as the reaction field, removes the influence of each spin on itself to lowest order. Omitting the reaction field from Eq. (3.5) leads to a set of equations, known as the "naive" mean field equations [Soukoulis, Levin, and Grest, 1982, 1983; Bray, Sompolinsky, and Yu, 1986], that are similar in form to the fixed-point conditions (3.4). However, Eqs. (3.4) are the correct equations for the analog associative memory neural network, which by design has no reaction field.

3.2.2 Counting fixed points

Calculating the average number of fixed points in analog associative memories employs methods used previously to count solutions of the TAP equations [Bray and Moore, 1980] and to count metastable states of neural networks with two-state neurons [Gardner, 1986; Bruce, Gardner, and Wallace, 1987]. The results of these previous studies suggest that the

number N_{fp} of fixed points is of the form $N_{fp} = \exp[NF(\alpha,\beta)]$, and that it is dominated by spurious states, which have a vanishing overlap with any of the memory patterns as $N \to \infty$.

The number of fixed points for a single realization of the interconnection matrix **J** is found by integrating a product of delta functions of the quantities G_i of Eq. (3.4) over state space:

$$N_{fp} = \int \prod_i (dm_i) \prod_i [\delta(G_i)] \, |\det \mathbf{A}|. \tag{3.6}$$

The integrals over m_i extend over the range of the neuron transfer function. The quantity $|\det \mathbf{A}|$ is the Jacobian normalizing the delta functions. The elements of **A** are given by

$$A_{ij} = \frac{\partial G_i}{\partial m_j} = a(m_i)\delta_{ij} - \beta J_{ij} \tag{3.7}$$

where δ_{ij} is the Kronecker delta function and

$$a(m_i) \equiv \frac{d}{dm}\left[f^{-1}(m)\right]_{m_i}. \tag{3.8}$$

For the case $f(x) = \tanh(x)$, Eq. (3.8) gives $a(m_i) = \left(1 - m_i^2\right)^{-1}$.

In addition to normalizing the integrand of Eq. (3.6), the matrix **A** is the Hessian of the Liapunov function for Eq. (3.1) and for Eq. (3.2) with serial updating at all values of gain [Hopfield, 1984], and for Eq. (3.2) with parallel updating when the gain satisfies $1/\beta > -\lambda_{min}$ [see Sec. 1.3.2 and Marcus and Westervelt, 1989; Waugh, Marcus, and Westervelt, 1990]. Thus **A** characterizes the curvature of these energy landscapes and can

be used to identify the stability of fixed points: when **A** is positive definite at a fixed point, then that point is a local minimum of the energy landscape—that is, it is stable. Since only local minima rather than saddle points or local maxima are of interest, the analysis below uses an approximate method to restrict the integration over state space in Eq. (3.6) to regions where **A** is positive definite.

The next step is to average the number of fixed points over all realizations of the interconnection matrix. Such an average is permissible if the averaged quantity scales linearly with network size [Brout, 1959]. Since the number of fixed points is expected to scale exponentially with N, the correct procedure is to average the logarithm of the number of fixed points, not the number itself. It appears, however, that for neural networks and Ising spin glasses with infinite-range coupling, the two averages give the same result for large N, indicating that the number of fixed points in each realization, and not just the logarithm of that number, approaches the ensemble average as $N \to \infty$ [Bray and Moore, 1980; Gardner, 1986; Bruce, Gardner, and Wallace, 1987]. This agreement is not found in systems with short-range interactions, and for such systems averages must be done over extensive quantities [Bray and Moore, 1981; Burgess and Moore, 1989].

Integration of Eq. (3.6) proceeds by using an integral representation of the delta function,

$$\delta(G_i) = \int_{-\infty}^{\infty} \frac{dx_i}{2\pi} \exp\left(iG_i x_i\right), \tag{3.9}$$

to write the average number of fixed points as

$$\langle N_{fp} \rangle = \left\langle \int \prod_i (dm_i) \int_{-\infty}^{\infty} \prod_i \left(\frac{dx_i}{2\pi} \right) \right.$$

$$\left. \times \exp\left[i \sum_i x_i f^{-1}(m_i) - \frac{i\beta}{N\sqrt{\alpha}} \sum_{i,j,\mu} x_i \xi_i^\mu \xi_j^\mu m_j + i\beta\sqrt{\alpha} \sum_i x_i m_i \right] |\det \mathbf{A}| \right\rangle. \quad (3.10)$$

The angled brackets in Eq. (3.10) denote an average over the patterns ξ_i^μ, and Eq. (3.3) is used to write out explicitly the dependence of the interconnection matrix on the patterns. It is now assumed that the determinant of **A** can be averaged separately from the rest of the integral in (3.10). This approximation, made previously for spin glasses [Bray and Moore, 1980; Takayama and Nemoto, 1990; Nishimura, Nemoto, and Takayama, 1990], assumes that the vast majority of local minima have identical local curvatures, so that the average can be used as the multiplier for each. This assumption is reasonable in light of the fact that the distributions of other features of the local minima, such as their energies and overlaps with the stored patterns, are dominated by a single value at large N.

The second term of the exponential in Eq. (3.10) can be averaged over the memory patterns ξ_i^μ to yield

$$\langle N_{fp} \rangle = \int \prod_i (dm_i) \int_{-\infty}^{\infty} \prod_i \left(\frac{dx_i}{2\pi} \right) \int_{-\infty}^{\infty} \prod_\mu \left(\frac{da_\mu \, db_\mu}{2\pi} \right)$$

$$\times \exp\left\{ i \sum_i x_i \left[f^{-1}(m_i) + \beta\sqrt{\alpha} m_i \right] + i \sum_\mu a_\mu b_\mu \right\}$$

$$\times \left\langle \exp\left[-i\left(\frac{\beta}{N\sqrt{\alpha}}\right)^{1/2}\sum_{i,\mu}\left(a_\mu x_i + b_\mu m_i\right)\xi_i^\mu\right]\right\rangle \left\langle |\det \mathbf{A}|\right\rangle \tag{3.11}$$

where the identity

$$\exp\left[-i\left(A_\mu B_\mu\right)\right] = \int_{-\infty}^{\infty}\frac{da_\mu\, db_\mu}{2\pi}\exp\left[i\left(a_\mu b_\mu - A_\mu a_\mu - B_\mu b_\mu\right)\right] \tag{3.12}$$

is used. The average of the exponential in (3.11) is carried out by noting that

$$\left\langle \exp\left(-ia\xi_i^\mu\right)\right\rangle = \cos(a) \approx \exp\left(-a^2/2\right) \quad \text{for} \quad a << 1 \tag{3.13}$$

with the result

$$\left\langle N_{fp}\right\rangle = \int\prod_i (dm_i)\int_{-\infty}^{\infty}\prod_i\left(\frac{dx_i}{2\pi}\right)\ \int_{-\infty}^{\infty}\prod_\mu\left(\frac{da_\mu\, db_\mu}{2\pi}\right)$$

$$\times \exp\left[i\sum_i x_i\left[f^{-1}(m_i) + \beta\sqrt{\alpha}\, m_i\right] + i\sum_\mu a_\mu b_\mu\right]$$

$$\times \exp\left[-\frac{\beta}{2N\sqrt{\alpha}}\left(\sum_i x_i^2\sum_\mu a_\mu^2 + \sum_i m_i^2\sum_\mu b_\mu^2 + 2\sum_\mu a_\mu b_\mu\sum_i x_i m_i\right)\right]$$

$$\times \left\langle |\det \mathbf{A}|\right\rangle. \tag{3.14}$$

Further progress can be made by defining the three order parameters

$$u = \frac{1}{N}\sum_i m_i^2, \quad v = \frac{1}{N}\sum_i x_i^2, \quad w = \frac{i}{N}\sum_i x_i m_i, \tag{3.15}$$

and by introducing their conjugate fields U, V, and W through the identities

$$1 = \int_{-\infty}^{\infty} du \int_{-i\infty}^{i\infty} \frac{N\,dU}{2\pi i} \exp\left[U\left(Nu - \sum_i m_i^2\right)\right] \tag{3.16}$$

$$1 = \int_{-\infty}^{\infty} dv \int_{-i\infty}^{i\infty} \frac{N\,dV}{2\pi i} \exp\left[V\left(Nv - \sum_i x_i^2\right)\right] \tag{3.17}$$

$$1 = \int_{-\infty}^{\infty} dw \int_{-i\infty}^{i\infty} \frac{N\,dW}{2\pi i} \exp\left[W\left(Nw - i\sum_i x_i m_i\right)\right]. \tag{3.18}$$

Inserting Eqs. (3.15) through (3.18) into Eq. (3.14) allows the integrals over a_μ and b_μ to be evaluated by straightforward gaussian integration, giving

$$\begin{aligned}\langle N_{fp}\rangle = {} & \int_{-\infty}^{\infty} du \int_{-i\infty}^{i\infty} \frac{N\,dU}{2\pi i} \int_{-\infty}^{\infty} dv \int_{-i\infty}^{i\infty} \frac{N\,dV}{2\pi i} \int_{-\infty}^{\infty} dw \int_{-i\infty}^{i\infty} \frac{N\,dW}{2\pi i} \exp(NG) \\ & \times \int \prod_i (dm_i) \int_{-\infty}^{\infty} \prod_i \left(\frac{dx_i}{2\pi}\right) \exp\left\{i\sum_i x_i\left[f^{-1}(m_i) + \left(\beta\sqrt{\alpha} - W\right)m_i\right]\right\} \\ & \times \exp\left[-U\sum_i m_i^2 - V\sum_i x_i^2\right] \langle|\det \mathbf{A}|\rangle \end{aligned} \tag{3.19}$$

where

$$G \equiv uU + vV + wW - \frac{\alpha}{2} \ln\left[\frac{\beta^2 uv}{\alpha} + \left(1 + \frac{\beta w}{\sqrt{\alpha}}\right)^2\right]. \tag{3.20}$$

Because u, v, and w now appear only in the $\exp(NG)$ factor, integrals over these variables can be evaluated easily using steepest descent by setting partial derivatives of G with respect to these variables equal to zero. This yields the solutions

$$u = \frac{2V\alpha}{4UV + W^2}; \quad v = \frac{2U\alpha}{4UV + W^2}; \quad w = \frac{W\alpha}{4UV + W^2} - \frac{\sqrt{\alpha}}{\beta}; \tag{3.21}$$

$$G = \alpha - \frac{W\sqrt{\alpha}}{\beta} - \frac{\alpha}{2} \ln\left(\frac{\beta^2 \alpha}{4UV + W^2}\right). \tag{3.22}$$

The integrals over x_i are gaussian and can be evaluated to give

$$\left\langle N_{fp} \right\rangle = \max_{U,V,W} \left\{ \left(\frac{1}{\sqrt{4\pi V}}\right)^N \exp\left[N\left(\alpha - \frac{W\sqrt{\alpha}}{\beta} - \frac{\alpha}{2}\ln\left(\frac{\beta^2\alpha}{4UV + W^2}\right)\right)\right] \right.$$

$$\left. \times \int \prod_i (dm_i) \exp\left[-\sum_i U m_i^2 - \sum_i \frac{\left[f^{-1}(m_i) + \left(\beta\sqrt{\alpha} - W\right)m_i\right]^2}{4V}\right] \right\} \left\langle |\det \mathrm{A}| \right\rangle \tag{3.23}$$

where max indicates that the integrals over U, V, and W are evaluated by steepest descent by numerically maximizing the expression in curly brackets.

Next the quantity $\langle|\det \mathbf{A}|\rangle$ is evaluated using the following general property of multi-dimensional gaussian integrals:

$$(\det \mathbf{A})\prod_i\{\theta[\lambda_i(\mathbf{A})]\} = \left[\int_{-\infty}^{\infty}\prod_i\left(\frac{d\rho_i}{\sqrt{2\pi}}\right)\exp\left(-\frac{1}{2}\sum_{i,j}\rho_i A_{ij}\rho_j\right)\right]^{-2} \tag{3.24}$$

where λ_i, $i = 1, ..., N$ are the eigenvalues of $\mathbf{A}$, and where $\theta(x) = 1$ for $x > 0$ and $\theta(x) = 0$ otherwise. The right hand side of Eq. (3.24) equals det $\mathbf{A}$ if $\mathbf{A}$ is positive definite but is zero otherwise. Because $\mathbf{A}$ is the Hessian of the Liapunov function for the network dynamics specified in Eqs. (3.1) and (3.2), replacing $|\det \mathbf{A}|$ in Eq. (3.23) with $(\det \mathbf{A})\prod_i\{\theta[\lambda_i(\mathbf{A})]\}$ picks out only the local minima of the energy landscape.

The right hand side of Eq. (3.24) can be averaged by introducing replicas of the quantity inside the square brackets and later setting the number m of replicas to -2 [Bray and Moore, 1980]. With the understanding that the count now includes only stable fixed points, Eq. (3.24) can be written as

$$\det \mathbf{A} = \lim_{m\to -2}\int_{-\infty}^{\infty}\prod_i\prod_{\gamma=1}^{m}\left(\frac{d\rho_{i\gamma}}{\sqrt{2\pi}}\right)\exp\left(-\frac{1}{2}\sum_{i,j}\sum_{\gamma=1}^{m}\rho_{i\gamma}A_{ij}\rho_{j\gamma}\right). \tag{3.25}$$

The calculation of the average $\langle\det \mathbf{A}\rangle$ from Eq. (3.25) is presented in Appendix 3A. The result, which assumes replica symmetry, is

$$\langle \det \mathbf{A} \rangle = \min_R \left\{ \exp\left[N\left(-\alpha + \frac{2R\sqrt{\alpha}}{\beta} + \alpha \ln\left(\frac{2R}{\beta\sqrt{\alpha}} \right) \right) \right] \prod_i \left(a(m_i) + \beta\sqrt{\alpha} - 2R \right) \right\} \quad (3.26)$$

where $a(m_i)$ is defined in Eq. (3.8).

The quantity $\left(a(m_i) + \beta\sqrt{\alpha} - 2R\right)$ in Eq. (3.26) may become negative in certain regions of state space. For $f(x) = \tanh(x)$, the quantity is negative near the center of state space and positive near the corners. Negative values of the quantity are interpreted as indicating that **A** is not positive definite in these regions and that the replica treatment has failed to return a zero for the determinant average. Because only the stable fixed points—for which **A** is positive definite—are of interest, integrals over state space are limited to the sub-region of the range of f where $\left(a(m_i) + \beta\sqrt{\alpha} - 2R\right)$ is positive. This is indicated by a "+" marking such integrals. Limiting the integrals in this way is necessary to obtain meaningful solutions of the saddle-point equations below.

Substituting Eq. (3.26) into Eq. (3.23) and making the change of variables $B = \beta\sqrt{\alpha} - 2R$, $\lambda = -U$, $q = 2V/\beta^2$, and $\Delta = W - \beta\sqrt{\alpha}$ gives

$$\left\langle N_{fp} \right\rangle = \max_{q,\lambda,\Delta} \min_B \left\{ \exp\left[N\tilde{F}(\alpha,\beta,q,\lambda,\Delta,B) \right] \right\} \quad (3.27)$$

where

$$\tilde{F} = -\frac{\sqrt{\alpha}}{\beta}(B+\Delta) + \frac{\alpha}{2} \ln\left[\frac{\left(\Delta/\beta + \sqrt{\alpha}\right)^2 - 2\lambda q}{\left(B/\beta - \sqrt{\alpha}\right)^2} \right] + \ln\left(\frac{I}{\beta\sqrt{2\pi q}} \right) \quad (3.28)$$

and

$$I = \int_{+} dm(a(m)+B)\ \exp\left[-\frac{\left(f^{-1}(m)-\Delta m\right)^2}{2\beta^2 q}+\lambda m^2\right] \tag{3.29}$$

and the "+" on the integral means that the region of integration is restricted to the range of f where $(a(m)+B)>0$.

Extrema of Eq. (3.27) with respect to q, λ, B, and Δ are found by setting partial derivatives of $\tilde{F}(\alpha,\beta,q,\lambda,\Delta,B)$ equal to zero, which gives a set of four saddle-point equations:

$$q = \frac{1}{\alpha}\left[\left(\frac{\Delta}{\beta}+\sqrt{\alpha}\right)^2-2\lambda q\right]\left\langle\left\langle m^2\right\rangle\right\rangle \tag{3.30}$$

$$\lambda = -\frac{1}{2\alpha q}\left[\left(\frac{\Delta}{\beta}+\sqrt{\alpha}\right)^2-2\lambda q\right]\left(1-\frac{1}{\beta^2 q}\left\langle\left\langle\left(f^{-1}(m)-\Delta m\right)^2\right\rangle\right\rangle\right) \tag{3.31}$$

$$0 = 1+\frac{1}{\alpha}\left[\left(\frac{\Delta}{\beta}+\sqrt{\alpha}\right)^2-2\lambda q\right]\left[\frac{\left\langle\left\langle m f^{-1}(m)\right\rangle\right\rangle}{\beta\sqrt{\alpha}\,q}-1\right] \tag{3.32}$$

$$B = \left(\frac{\beta B}{\sqrt{\alpha}}-\beta^2\right)\left\langle\left\langle\left(a(m)+B\right)^{-1}\right\rangle\right\rangle. \tag{3.33}$$

The double brackets in Eqs. (3.30) through (3.33) indicate a weighted average with the weight function $W(m)$ given by the integrand of I:

$$\langle\langle\ g(m)\ \rangle\rangle \equiv \frac{\int_+ dm\ g(m)\ W(m)}{\int_+ dm\ W(m)}; \tag{3.34}$$

$$W(m) = (a(m) + B)\exp\left(-\frac{\left(f^{-1}(m) - \Delta m\right)^2}{2\beta^2 q} + \lambda m^2\right). \tag{3.35}$$

For given values of α and β, self-consistent solutions for q, λ, Δ, and B are found by solving Eqs. (3.30) through (3.33) numerically (see Appendix 3B). The solutions are then inserted into Eqs. (3.27) through (3.29) to yield a value for the quantity of interest, the scaling exponent $F(\alpha,\beta)$.

Values of $F(\alpha,\beta)$ are plotted in Fig. 3.2 for the case $f(x) = \tanh(x)$. The result shows that for any value of the storage fraction α, the function $F(\alpha,\beta)$ decreases as the neuron gain is lowered. Since the number of fixed points depends exponentially on the product $NF(\alpha,\beta)$, even small changes in $F(\alpha,\beta)$ dramatically affect the number of fixed points, especially for large N. As an example, consider the effect of lowering the neuron gain from $\beta = 100$ to $\beta = 10$ in an analog associative memory network with storage fraction $\alpha = 0.1$. Using the values $F(0.1,100) = 0.059$ and $F(0.1,10) = 0.040$, the average number of (predominantly spurious) fixed points is reduced by approximately 97% for $N = 200$ and by eight orders of magnitude for $N = 1000$.

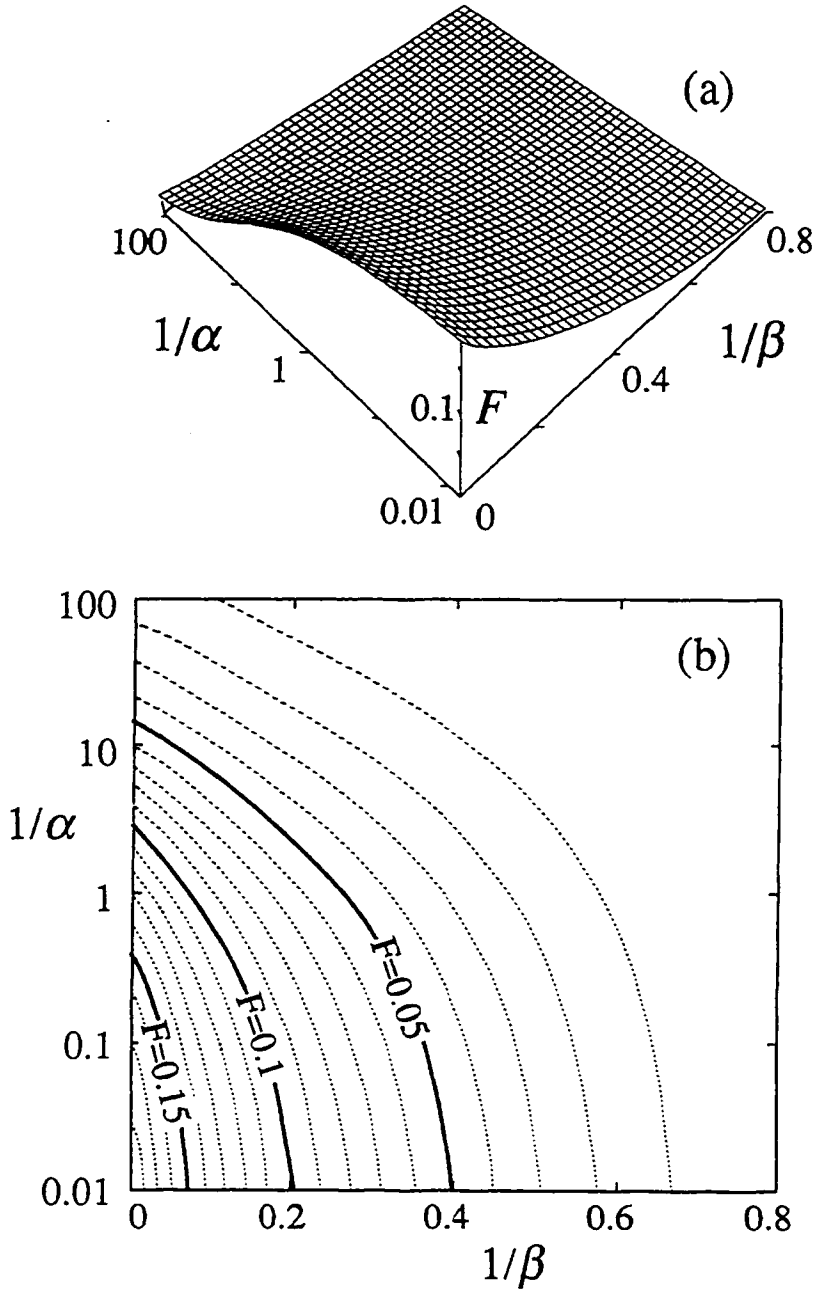


Fig. 3.2 Theoretical values of $F(\alpha,\beta)$ vs. storage ratio α and inverse neuron gain $1/\beta$, for neuron transfer function $f(x) = \tanh(x)$. Expected number of fixed points is $e^{NF(\alpha,\beta)}$. All quantities are dimensionless. Contour lines of surface (a) are shown in (b) for every 0.01 (dashed) and every 0.05 (solid).

3.3 LIMITING CASES

3.3.1 Finite-temperature spin glasses

The limit $\alpha \to \infty$ of infinite pattern storage is related to the finite-temperature Sherrington-Kirkpatrick spin glass [Sherrington and Kirkpatrick, 1975]. The limit $\alpha \to \infty$ of the saddle-point equations is calculated by considering the leading terms in $1/\alpha$ of Eqs. (3.30) through (3.33):

$$\lim_{\alpha\to\infty} q = \left\langle\left\langle m^2 \right\rangle\right\rangle \tag{3.36}$$

$$\lim_{\alpha\to\infty} \lambda = -\frac{1}{2q}\left[1-\frac{1}{\beta^2 q}\left\langle\left\langle \left(f^{-1}(m)-\Delta m\right)^2 \right\rangle\right\rangle\right] \tag{3.37}$$

$$\lim_{\alpha\to\infty} \Delta = \frac{1}{2q}\left\langle\left\langle m f^{-1}(m) \right\rangle\right\rangle \tag{3.38}$$

$$\lim_{\alpha\to\infty} B = -\beta^2\left\langle\left\langle \left(a(m)+B\right)^{-1} \right\rangle\right\rangle. \tag{3.39}$$

The meaning of the double brackets in Eqs. (3.36) through (3.39) is still given by Eq. (3.34) and is unaffected by the limiting process. To calculate $\tilde{F}$ from Eq. (3.28) in the limit $\alpha \to \infty$, it is necessary to expand the first logarithm in Eq. (3.28) to order $1/\alpha^2$, with the result

$$\lim_{\alpha\to\infty} \tilde{F} = -\lambda q + \frac{1}{2\beta^2}\left(B^2 - \Delta^2\right) + \ln\left(\frac{I}{\beta\sqrt{2\pi q}}\right) \tag{3.40}$$

where I is still given by Eq. (3.29).

The expressions (3.36) through (3.40) obtained by letting $\alpha \to \infty$ can also be obtained by starting with a random symmetric interconnection matrix with zero diagonals and off-diagonals chosen from the gaussian probability distribution

$$P\left(J_{ij}\right) = (N/2\pi)^{1/2} \exp\left(-N J_{ij}^{\,2}/2\right). \tag{3.41}$$

For the transfer function $f(x) = \tanh(x)$, Eqs. (3.36) through (3.40) can therefore be interpreted as the number of solutions of the naive mean-field equations for the infinite-range Ising spin glass at finite temperature. Also, the quantity q appearing in Eqs. (3.36) through (3.40) corresponds to the Edwards-Anderson spin-glass order parameter, though for finite α this correspondence is invalid. The function $\tilde{F}$ of Eq. (3.40) differs from the corresponding function $\tilde{F}_{TAP}$ for the number of solutions of the TAP mean-field equations (3.5), which is [Bray and Moore, 1980; Takayama and Nemoto, 1990; Nishimura, Nemoto, and Takayama, 1990]

$$\tilde{F}_{TAP} = -\lambda q + \frac{1}{2\beta^2}\left(B^2 - \Delta^2\right) - (B+\lambda)(1-q) + \ln\left(\frac{I}{\beta\sqrt{2\pi q}}\right). \tag{3.42}$$

The extra term $(B+\lambda)(1-q)$ in Eq. (3.42) arises from the reaction field of the TAP equations. Figure 3.3 compares the curves $F(\alpha \to \infty, \beta)$ and $F_{TAP}(\beta)$ calculated by numerically solving the saddle-point equations associated with Eqs. (3.40) and (3.42).

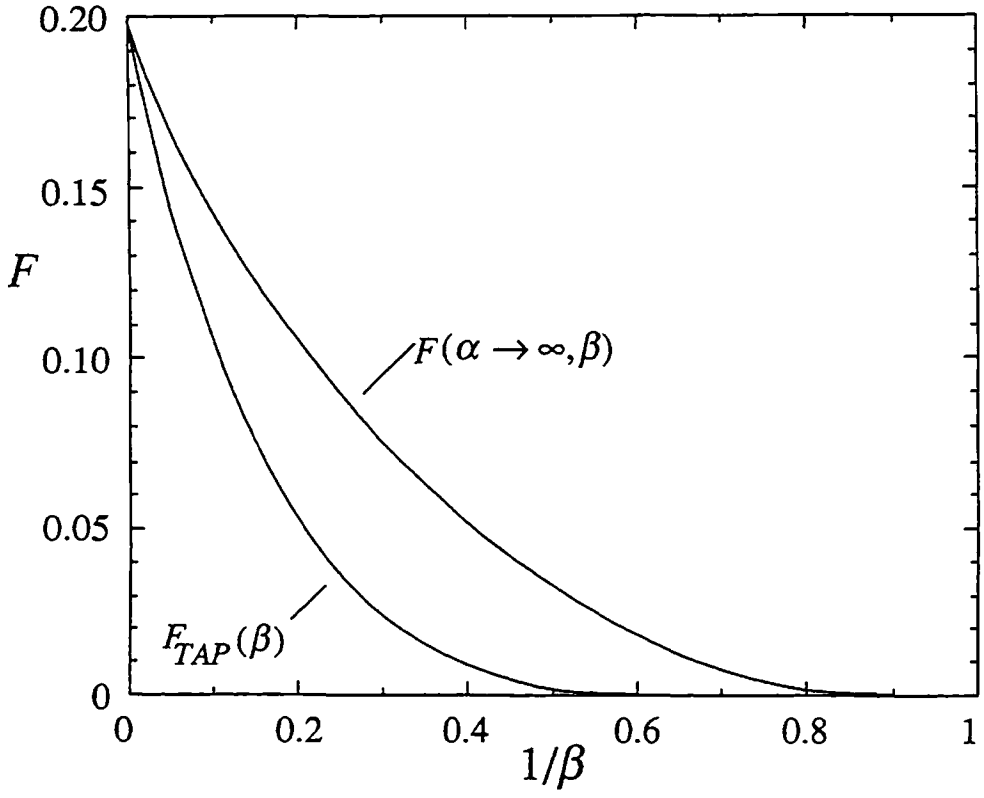


Fig. 3.3 Theoretical values of $F(\alpha \to \infty, \beta)$, scaling exponent in limit of high memory loading, vs. inverse gain $1/\beta$. Scaling exponent $F_{TAP}(\beta)$ for TAP equations, from Eq. (3.42), is also shown for comparison.

Note that the reaction field has a greater effect for spin glasses than it does in Chapter 2 for associative memories (see Fig. 2.9 and Sec. 2.4). In both cases, the reaction field acts as an effective noise source, decreasing the number of fixed points for a given value of β.

3.3.2 Associative memories of two-state neurons

The limit $\beta \to \infty$ of the saddle-point equations (3.30) through (3.33) depends on the specific form of the neuron transfer function $f(x)$. The form considered here is $f(x) = \tanh(x)$, which, as $\beta \to \infty$, makes neurons equivalent to Ising spins at $T = 0$. For this transfer function, the limits of each integral in the saddle-point equations are ± 1. The limit requires the observation that, for large β, the entire contribution to these integrals comes from regions within $\varepsilon << 1$ of the limits of integration. Physically, this arises because all minima move to the corners of the state-space hypercube as $f(x)$ approaches a two-state function with increasing β.

The value of ε is determined by the point at which the quantity $(a(m)+B)$ changes sign and is given by $\varepsilon \approx -(2B)^{-1}$ for $f(x) = \tanh(x)$. The integral I and the other integrals appearing in the saddle-point equations are then given by the correction terms of Appendix 3B, with the substitution $\delta = \varepsilon \approx -(2B)^{-1}$. For example, the result for I is

$$I \underset{\beta\to\infty}{\to} \beta\sqrt{2\pi q}\, e^{\lambda}\, \mathrm{erfc}(y), \tag{3.43}$$

where $\mathrm{erfc}(y) = 2\pi^{-1/2}\int_y^\infty dt\, e^{-t^2}$ is the complementary error function and

$$y = \frac{1}{\beta\sqrt{2q}}\left[\frac{1}{2}\ln(-4B) - \Delta\right]. \tag{3.44}$$

The other integrals can be done similarly, leading to the following asymptotic forms for the saddle-point equations for large β:

$$q \approx \frac{1}{\alpha}\left[\left(\frac{\Delta}{\beta}+\sqrt{\alpha}\right)^2 - 2\lambda q\right]\left\{1-\frac{1}{2}\beta^{-3/2}\left[\frac{\exp\left(-\Delta^2/2\beta^2 q\right)}{\sqrt{2\pi q}\ \mathrm{erfc}\left(-\Delta/\beta\sqrt{2q}\right)}\right]^{1/2}\right\} \tag{3.45}$$

$$\lambda \approx -\frac{\Delta \exp\left(-\Delta^2/2\beta^2 q\right)}{\beta\sqrt{2\pi q}\ \mathrm{erfc}\left(-\Delta/\beta\sqrt{2q}\right)} \tag{3.46}$$

$$\Delta \approx -\beta\sqrt{\alpha}\,(1-q) - \frac{\beta\sqrt{2q}\exp\left(-\Delta^2/2\beta^2 q\right)}{\sqrt{\pi}\ \mathrm{erfc}\left(-\Delta/\beta\sqrt{2q}\right)} \tag{3.47}$$

$$B \approx -\beta^{1/2}\left[\frac{\exp\left(-\Delta^2/2\beta^2 q\right)}{\sqrt{2\pi q}\ \mathrm{erfc}\left(-\Delta/\beta\sqrt{2q}\right)}\right]^{1/2} . \tag{3.48}$$

Substituting these results into Eq. (3.28) and taking the $\beta \to \infty$ limit yields

$$\lim_{\beta\to\infty} \tilde{F} = \alpha - \sqrt{2\alpha q(\lambda+\alpha/2)} + \lambda + \frac{1}{2}\alpha \ln q + \ln\left[\mathrm{erfc}\left(\sqrt{\alpha/2q} - \sqrt{\lambda+\alpha/2}\right)\right]. \tag{3.49}$$

With the change of variables $a = q$, $b = 1-\sqrt{2q(\lambda+\alpha/2)/\alpha}$, Eq. (3.49) may be written

$$\lim_{\beta\to\infty} \tilde{F} = \alpha\left[-\frac{1}{2} + \frac{(b-1)^2}{2a} + b + \frac{1}{2}\ln a\right] + \ln\left[\mathrm{erfc}\left(\sqrt{\frac{b^2\alpha}{2a}}\right)\right]. \tag{3.50}$$

Equation (3.50) is identical to the result for the total number of spurious states in a Hebb rule network of two-state neurons [Gardner, 1986; Bruce, Gardner, and Wallace, 1987]. This equivalence is remarkable given that the calculation for two-state neurons proceeds quite differently from that for analog neurons. In particular, counting local minima for two-state neurons involves summation over a discrete state space rather than integration over a continuous one and thus avoids the difficulties associated with the normalizing determinant. The curve $F(\alpha, \beta \to \infty)$ obtained by solving the saddle-point equations derived from Eq. (3.50) is shown in Fig. 3.4.

3.3.3 Zero-temperature spin glasses

The limit of both $\alpha \to \infty$ and $\beta \to \infty$ corresponds to a spin glass at $T = 0$. To calculate $\tilde{F}$ in this limit, consider the $\alpha \to \infty$ limit of Eq. (3.50). Since the saddle-point equations derived from (3.50) imply $a = (1-b)/(1+b)$ for all values of α, Eq. (3.50) may be written

$$\lim_{\beta\to\infty} \tilde{F} = \alpha\left[b - \frac{1}{2}b^2 + \frac{1}{2}\ln\left(\frac{1-b}{1+b}\right)\right] + \ln\left[\operatorname{erfc}\left(\sqrt{\frac{b^2\alpha(1+b)}{2(1-b)}}\right)\right]. \tag{3.51}$$

For this expression not to diverge as $\alpha \to \infty$, it is necessary that $b \propto 1/\sqrt{\alpha}$. Writing $b = t/\sqrt{\alpha}$ and expanding the first logarithm in (3.51) to order $1/\alpha^2$ leads to

$$\lim_{\alpha\to\infty,\beta\to\infty} \tilde{F} = -\frac{1}{2}t^2 + \ln\left[\operatorname{erfc}\left(\sqrt{\frac{t^2}{2}}\right)\right]. \tag{3.52}$$

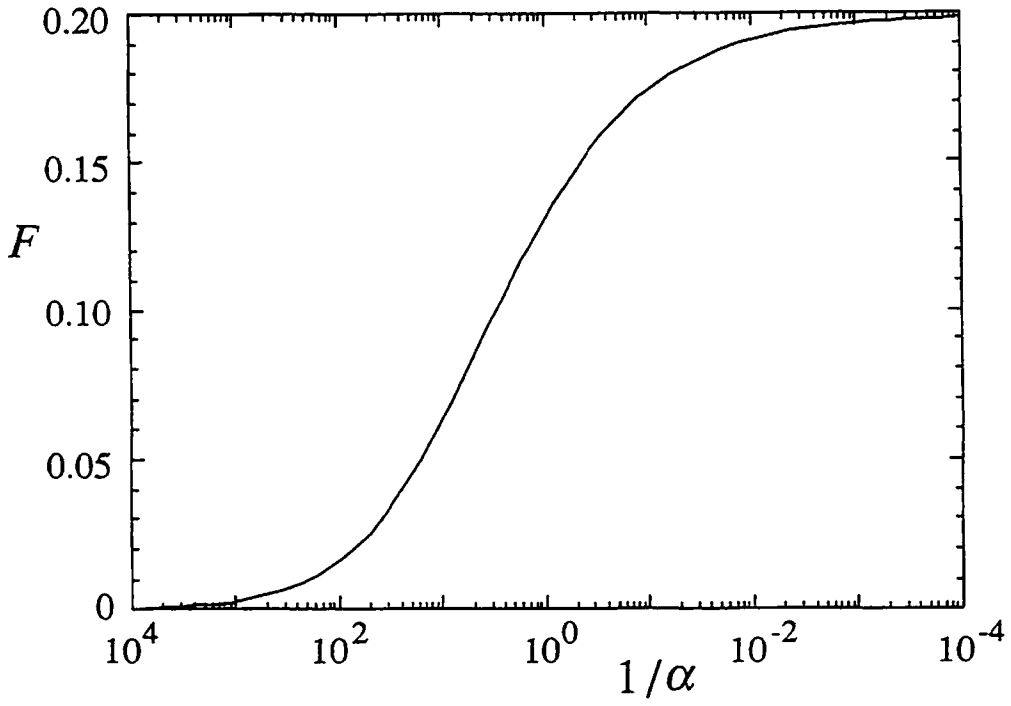


Fig. 3.4 Theoretical values of $F(\alpha, \beta \to \infty)$, scaling exponent in limit of high neuron gain, vs. inverse storage fraction $1/\alpha$.

The saddle-point equation associated with Eq. (3.52) may be solved numerically with the result

$$\lim_{\alpha\to\infty,\,\beta\to\infty} F \cong 0.1992, \tag{3.53}$$

which has been calculated previously [Tanaka and Edwards, 1980; De Dominicis *et al.*, 1980].

For $\alpha \to \infty$ and large but finite β, the coefficients in Eqs. (3.47) through (3.50) can be solved analytically, leading to the asymptotic forms $q \approx 1 - 0.252\beta^{-3/2}$, $\lambda \approx -0.128$, $\Delta \approx 0.506\beta$, and $B \approx -0.503\beta^{1/2}$. The $\beta^{-3/2}$ dependence of q differs from other analytical results for the naive mean-field equations [Takayama and Nemoto, 1990] but is observed in numerical studies of the naive mean-field equations for the infinite-range Ising spin glass [Nishimura, Nemoto, and Takayama, 1990].

3.4 VERIFYING THE RESULTS

The analytical results of Secs. 3.2 and 3.3 have been tested numerically for the nonlinearity $f(x) = \tanh(x)$ by counting the stable fixed points in small computer-generated analog neural networks. Numerical data were obtained by the following procedure. At several pairs of values of α and β, 20 realizations were generated for each of 5 or 6 values of N using the Hebb rule, Eq. (3.3), with $\xi_i^\mu = \pm 1$ randomly and without bias. The values of N for given α and β were chosen so that the number of fixed points was roughly in the range 20 to 400. The number of fixed points in each network was counted by choosing random initial conditions $m_i(0) = \pm 1$ and iterating the discrete time,

sequential-update network of Eq. (3.2) until convergence to a fixed point was reached. A network was considered to have converged to a fixed point at time t if the quantity $N^{-1}\sum_i |m_i(t) - m_i(t-1)|$ was less than 10^{-6}. As noted previously, under sequential updating of state variables, a network with symmetric interconnections and zero diagonals converges to a fixed point for all values of gain β. For each realization, the search for new fixed points was terminated after 10^5 initial conditions or when no new fixed points had been found for 10^4 consecutive initial conditions and for every fixed point found, the inverse point ($m_i \rightarrow -m_i$ for all i) had also been found.

Next, for each set of parameters (N,α,β), the mean $\overline{N_{fp}}$ and the variance of the observed number of fixed points were calculated for the 20 realizations. From these were computed numerical values $F_{num}(\alpha,\beta)$ for the scaling exponent, defined by the line

$$F_{num}(\alpha,\beta) = \frac{1}{N}\ln\left[\overline{N_{fp}}(N,\alpha,\beta)\right] + \text{const.}, \tag{3.54}$$

using weighted least-square fits. The number of observed fixed points in each realization, rather than the logarithm of the number, is averaged simply because $\ln\langle N_{fp}\rangle$, not $\langle \ln N_{fp}\rangle$, is calculated in Sec. 3.2. The difference in the two types of averages is negligible, as it should be for fully connected networks in the large-N limit.

The least-square fits are shown in Fig. 3.5(a). Figure 3.5(b) compares the resulting experimental values of $F_{num}(\alpha,\beta)$ to the theoretical curves for $\alpha = 10$, 1, and 0.1 as computed in Sec. 3.2. Agreement is very good at large values of α and β, and reasonably good, though outside the range of the error bars, for smaller α and β.

Similar techniques can be used to find $F_{num}(\alpha \rightarrow \infty,\beta)$ for networks with interconnections chosen from the probability distribution of Eq. (3.41). The results are shown in Figs. 3.6(a) and 3.6(b). The data of Fig. 3.6(b) agree well with the analytical

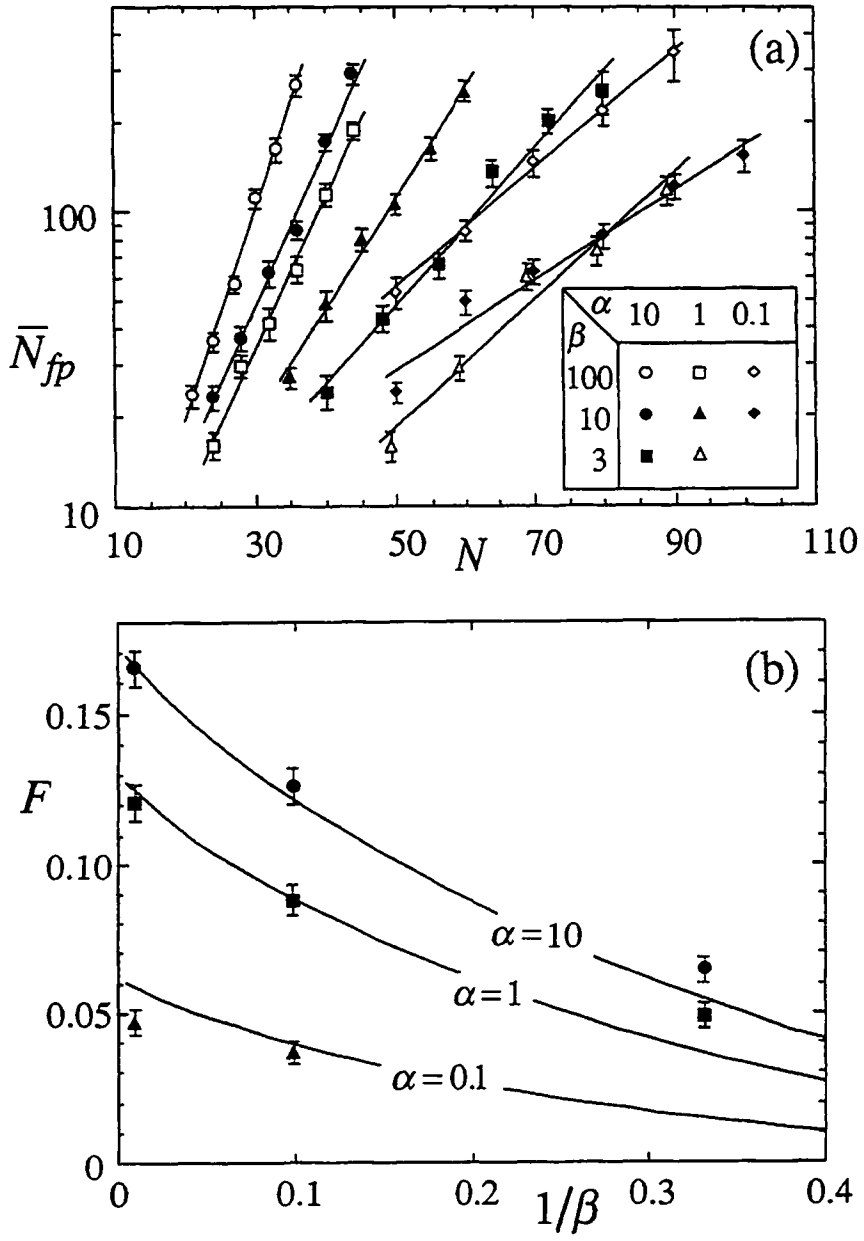


Fig. 3.5 (a) Numerical counts of stable fixed points for analog associative memory for several values of α and β, vs. number of neurons N. Lines are weighted exponential fits to data as described in text. Numerical values $F_{num}(\alpha,\beta)$ for scaling exponent are given by slopes of lines using logarithmic scaling. (b) Comparison of theoretical and numerical results for scaling exponent. Numerical results for $\alpha = 10$ (circles), $\alpha = 1$ (squares), and $\alpha = 0.1$ (triangles) agree well with theory (curves) as calculated from Eqs. (3.27) through (3.29).

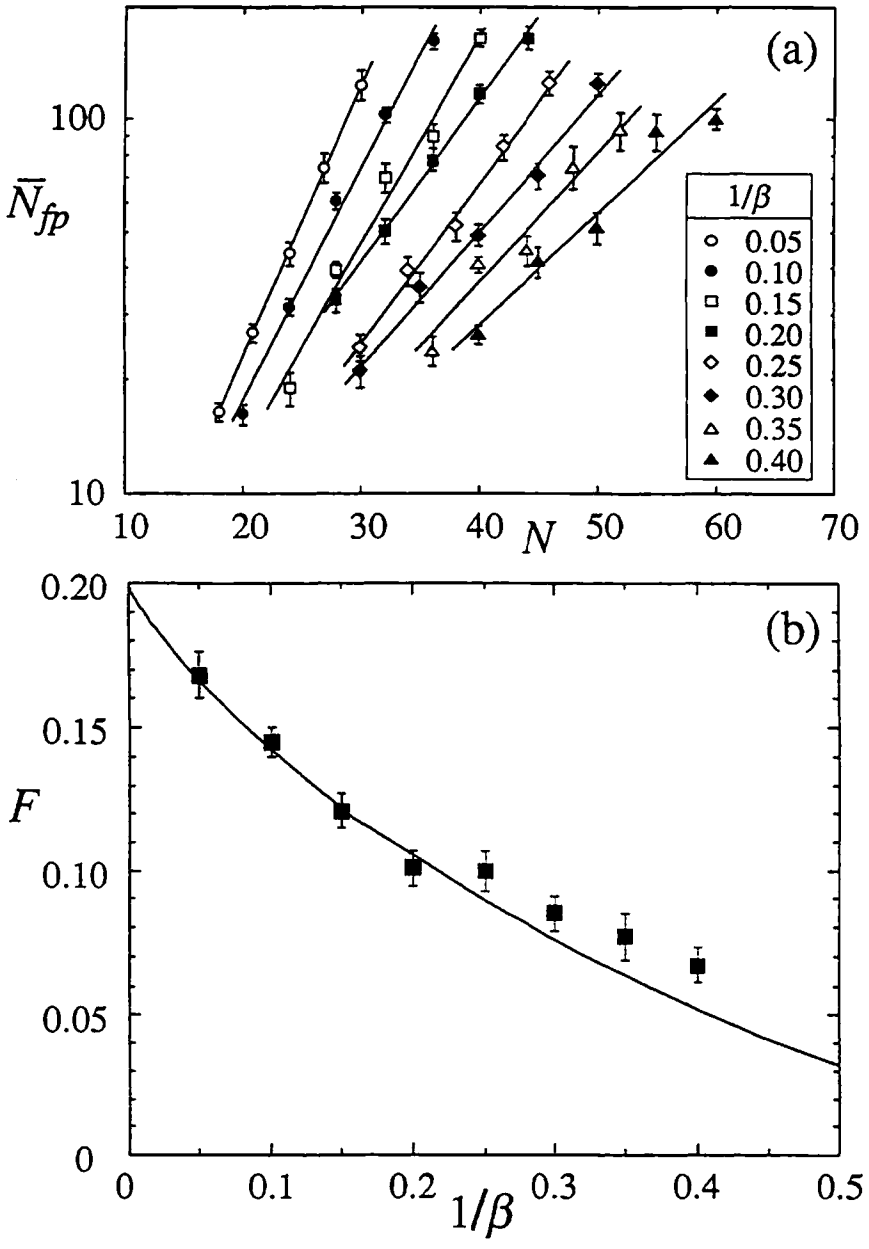


Fig. 3.6 (a) Numerical counts of stable fixed points for analog associative memory in high storage fraction limit $\alpha \to \infty$ for several values of β, vs. number of neurons N. Lines are weighted exponential fits to data as described in text. Numerical values $F_{num}(\alpha \to \infty, \beta)$ for scaling exponent are given by slopes of lines using logarithmic scaling. (b) Comparison of theoretical and numerical results for scaling exponent $F(\alpha \to \infty, \beta)$ as a function of inverse neuron gain $1/\beta$. Numerical results (squares) are in good agreement with theory (curve) as calculated from Eq. (3.40).

results of Sec. 3.3.1, especially at large values of β. Similar numerical results have been reported elsewhere [Takayama and Nemoto, 1990; Nishimura, Nemoto, and Takayama, 1990]. Apparently, there are no data verifying the comparable curve for the TAP equations away from $T = 1/\beta = 0$, due in part to the complicated dynamics exhibited by the TAP equations when solved iteratively [Bray and Moore, 1979].

3.5 SUMMARY

This chapter has presented analytical and numerical results showing quantitatively that lowering the gain of the neuron transfer function in an analog associative memory neural network greatly reduces the number of local minima in the energy landscape. This phenomenon provides one mechanism for the observed improvement of performance in analog neural networks compared to networks of two-state neurons. Because the use of analog state variables to eliminate spurious attractors is a fully deterministic procedure, the method can be implemented in electronic hardware more easily than stochastic techniques. Using analog state variables with reduced gain is not suggested as a replacement for stochastic methods such as simulated annealing, since it does not necessarily find global minima; rather, the method may be better suited for applications where the speed in finding a solution is more important than the optimality of the solution found.

Other deterministic strategies have been proposed for eliminating the spurious fixed points in a neural network. For the zero-temperature Ising spin glass, adding asymmetry to the interconnection matrix reduces the number of fixed points as well as the total number of attractors [Gutfreund, Reger, and Young, 1988; Parisi, 1986]. Similar effects have been observed numerically for neural networks [Crisanti and Sompolinsky, 1987]. However,

calculations of the distribution of fixed points in asymmetrically diluted Hebb-rule neural networks with binary neurons find an exponential number of spurious fixed-point attractors for *all* values of asymmetry [Trevis and Amit, 1988], suggesting that asymmetry weakens the effect of spurious states in neural networks not by eliminating them completely but by shrinking their basins of attraction. These remaining spurious states can be eliminated by adding a self-inhibition term $J_{ii} < 0$ [Kepler, 1989], but only at the expense of creating limit cycles. In contrast to these methods, smoothing the landscape using analog neurons eliminates fixed points without introducing any new attractors.

APPENDIX 3A: CALCULATING THE DETERMINANT

This appendix derives the expression (3.26) for $\langle \det \mathbf{A} \rangle$, the average determinant of the Hessian matrix $\mathbf{A}$ defined in Eq. (3.7). The average of Eq. (3.25) is taken over realizations of Hebb matrices each storing αN memory patterns. The patterns ξ_i^μ (i=1,...,N; μ=1,...,αN) take on the values ± 1 with equal probability. Starting from Eq. (3.25),

$$\langle \det \mathbf{A} \rangle = \lim_{m \to -2} \left\langle \int_{-\infty}^{\infty} \prod_{i,\gamma} \left(\frac{d\rho_{i\gamma}}{\sqrt{2\pi}} \right) \exp\left(-\frac{1}{2} \sum_{i,j} \sum_{\gamma=1}^{m} \rho_{i\gamma} A_{ij} \rho_{j\gamma} \right) \right\rangle. \tag{3A.1}$$

Inserting A_{ij} from Eq. (3.7) and J_{ij} from Eq. (3.3) into (3A.1) gives

$$\langle \det \mathbf{A} \rangle = \lim_{m \to -2} \left\langle \int_{-\infty}^{\infty} \prod_{i,\gamma} \left(\frac{d\rho_{i\gamma}}{\sqrt{2\pi}} \right) \right.$$

$$\left. \times \exp\left\{ -\frac{1}{2} \sum_{i,\gamma} \rho_{i\gamma}{}^2 \left[a(m_i) + \beta\sqrt{\alpha} \right] + \frac{1}{2} \frac{\beta}{N\sqrt{\alpha}} \sum_{\gamma,\mu} \left(\sum_i \rho_{i\gamma} \, \xi_i^\mu \right)^2 \right\} \right\rangle. \tag{3A.2}$$

The square in the last term can be reduced to a linear form through the identity

$$\exp\left(\frac{a^2}{2} \right) = (2\pi)^{-1/2} \int_{-\infty}^{\infty} dx \exp\left(-\frac{x^2}{2} + ax \right) \tag{3A.3}$$

with $a = \left(\beta/N\sqrt{\alpha}\right)^{1/2} \sum_i \xi_i^\mu \rho_{i\gamma}$. This introduces a new set of integration variables, $\sigma_{\gamma\mu}$ ($\gamma = 1, ..., m$; $\mu = 1, ..., \alpha N$) and gives

$$\langle \det \mathbf{A} \rangle = \lim_{m \to -2} \int_{-\infty}^{\infty} \prod_{\gamma,\mu} \left(\frac{d\sigma_{\gamma\mu}}{\sqrt{2\pi}} \right) \int_{-\infty}^{\infty} \prod_{i,\gamma} \left(\frac{d\rho_{i\gamma}}{\sqrt{2\pi}} \right)$$

$$\times \exp\left[-\frac{1}{2} \sum_{i,\gamma} \rho_{i\gamma}{}^2 \left[a(m_i) + \beta\sqrt{\alpha} \right] - \frac{1}{2} \sum_{\gamma,\mu} \sigma_{\gamma\mu}{}^2 \right]$$

$$\times \left\langle \exp\left[\left(\frac{\beta}{N\sqrt{\alpha}} \right)^{1/2} \sum_{i,\gamma,\mu} \rho_{i\gamma} \sigma_{\gamma\mu} \xi_i^\mu \right] \right\rangle . \qquad \text{(3A.4)}$$

Averaging over the ξ_i^μ can now be done immediately using Eq. (3.13), which leads to

$$\left\langle \exp\left[\left(\frac{\beta}{N\sqrt{\alpha}} \right)^{1/2} \sum_{i,\gamma,\mu} \rho_{i\gamma} \sigma_{\gamma\mu} \xi_i^\mu \right] \right\rangle \equiv \exp\left[\frac{\beta}{2N\sqrt{\alpha}} \sum_{i,\mu} \left(\sum_\gamma \rho_{i\gamma} \sigma_{\gamma\mu} \right)^2 \right] . \qquad \text{(3A.5)}$$

The assumption $\left(\sum_\gamma \rho_{i\gamma} \sigma_{\gamma\mu} \right)^2 \equiv \sum_\gamma \rho_{i\gamma}^2 \sigma_{\gamma\mu}^2$ is now made. Ignoring the cross terms in this square is valid when there are no correlations among replicas, as is expected when evaluating $\langle \det \mathbf{A} \rangle$ at spurious fixed points. A similar assumption has been made in the spin-glass problem, where it has been shown that keeping the cross terms does not change the result significantly [Bray and Moore, 1980; Takayama and Nemoto, 1990]. Assuming replica symmetry by setting $\sigma_{\gamma\mu} = \sigma_\mu$ and $\rho_{i\gamma} = \rho_i$ for all γ allows (3A.4) to be written as a single-site integral (in replica space) raised to the power m:

$$\langle \det \mathbf{A} \rangle = \lim_{m \to -2} \left\{ \int_{-\infty}^{\infty} \prod_{\mu} \left(\frac{d\sigma_\mu}{\sqrt{2\pi}} \right) \int_{-\infty}^{\infty} \prod_{i} \left(\frac{d\rho_i}{\sqrt{2\pi}} \right) \right.$$

$$\left. \times \exp \left[-\frac{1}{2} \sum_{i} \rho_i{}^2 \left[a(m_i) + \beta\sqrt{\alpha} \right] - \frac{1}{2} \sum_{\mu} \sigma_\mu{}^2 + \frac{\beta}{2\sqrt{\alpha}} \left(\frac{1}{N} \sum_{i} \rho_i{}^2 \right) \left(\sum_{\mu} \sigma_\mu{}^2 \right) \right] \right\}^m . \tag{3A.6}$$

Next, the order parameter

$$r = \frac{1}{N} \sum_{i} \rho_i{}^2 \tag{3A.7}$$

is introduced, along with its conjugate field R, through the identity

$$1 = \int_{-\infty}^{\infty} dr \int_{-i\infty}^{i\infty} \frac{N\, dR}{2\pi i} \exp \left[-R \left(Nr - \sum_{i} \rho_i{}^2 \right) \right]. \tag{3A.8}$$

Inserting (3A.7) and (3A.8) into (3A.6) gives

$$\langle \det \mathbf{A} \rangle = \lim_{m \to -2} \left\{ \int_{-\infty}^{\infty} dr \int_{-i\infty}^{i\infty} \frac{N\, dR}{2\pi i} \int_{-\infty}^{\infty} \prod_{\mu} \left(\frac{d\sigma_\mu}{\sqrt{2\pi}} \right) \int_{-\infty}^{\infty} \prod_{i} \left(\frac{d\rho_i}{\sqrt{2\pi}} \right) \exp\left(-NRr\right) \right.$$

$$\left. \times \exp \left[-\frac{1}{2} \sum_{i} \rho_i{}^2 \left[a(m_i) + \beta\sqrt{\alpha} - 2R \right] - \frac{1}{2} \left(1 - \frac{\beta r}{\sqrt{\alpha}} \right) \sum_{\mu} \sigma_\mu{}^2 \right] \right\}^m , \tag{3A.9}$$

which, after gaussian integration of σ_μ and ρ_i, becomes

$$\langle \det \mathbf{A} \rangle = \lim_{m \to -2} \left\{ \int_{-i\infty}^{i\infty} \frac{dr\, dR}{2\pi i/N} \exp\left\{ N\left[-rR - \frac{\alpha}{2} \ln\left(1 - \frac{\beta r}{\sqrt{\alpha}} \right) \right] \right\} \right.$$

$$\left. \times \prod_i \left[a(m_i) + \beta\sqrt{\alpha} - 2R \right]^{-1/2} \right\}^m . \tag{3A.10}$$

The integrals over r and R are evaluated by steepest descent, which is justified for large N. Each integral produces a factor $(2\pi/N)^{1/2}$ to cancel the existing prefactor proportional to N; other numerical prefactors of O(1) are ignored. The steepest descent integral over r can be done explicitly by setting

$$\frac{\partial}{\partial r}\left[rR + \frac{\alpha}{2} \ln\left(1 - \beta r/\sqrt{\alpha} \right) \right] = 0 \tag{3A.11}$$

with the result

$$r = \left(\frac{\sqrt{\alpha}}{\beta} - \frac{\alpha}{2R} \right). \tag{3A.12}$$

The steepest descent integral over R is eventually done numerically. Setting the number of replicas to –2 before performing this integral makes the minimum, not the maximum, with respect to R the valid solution [Bray and Moore, 1980]. Inserting (3A.12) into (3A.10) and setting $m = -2$ yields Eq. (3.26).

APPENDIX 3B: INTEGRAL EXPANSIONS

The saddle-point equations (3.30) through (3.33) each contain integrals of the form shown in Eq. (3.34) which must be evaluated numerically. For the transfer function $f(x) = \tanh(x)$, all but one of the integrands diverge at the endpoints of the domain of integration, ± 1, making numerical integration difficult. This problem may be avoided by splitting the domain of integration into three regions:

$$\int_{-1}^{1} = \int_{-1}^{-1+\delta} + \int_{-1+\delta}^{1-\delta} + \int_{1-\delta}^{1} , \tag{3B.1}$$

which may be evaluated as

$$\int_{-1}^{1} = 2\left(\int_{0}^{1-\delta} + \int_{1-\delta}^{1} \right) \tag{3B.2}$$

by virtue of the even symmetry of all of the integrands. The integrals with limits at zero and $1-\delta$ can be evaluated accurately using a standard numerical integrator. The remaining integrals extending from $1-\delta$ to 1 can be evaluated to leading order in δ for $\delta << 1$, as shown below. This appendix suppresses the "+" markers on the integrals indicating that the range of integration is limited to a sub-region where $\langle \det \mathbf{A} \rangle > 0$. The excluded region is in the region of state space covered by the integrals that are done numerically.

First, consider the integral I defined in Eq. (3.29). For $f(x) = \tanh(x)$,

$$I \equiv 2\int_{0}^{1-\delta} dm \; W(m) + \beta\sqrt{2\pi q}\, e^{\lambda}\, \mathrm{erfc}\,(y), \tag{3B.3}$$

where

$$y = \frac{1}{\beta\sqrt{2q}}\left[\frac{1}{2}\ln(2/\delta) - \Delta\right] \tag{3B.4}$$

and erfc(y) is the complementary error function,

$$\text{erfc}(y) \;=\; \frac{2}{\sqrt{\pi}}\int_y^\infty dt\, e^{-t^2}. \tag{3B.5}$$

The integrals in double brackets in Eqs. (3.30) through (3.33) have the following expansions:

$$\left\langle\left\langle m^2 \right\rangle\right\rangle \cong \frac{1}{I}\left(2\int_0^{1-\delta} dm\, m^2 W(m) \;+\; \beta\sqrt{2\pi q}\; e^{\lambda}\; \text{erfc}(y)\right) \tag{3B.6}$$

$$\left\langle\left\langle m f^{-1}(m) \right\rangle\right\rangle \cong \frac{1}{I}\left[2\int_0^{1-\delta} dm\left(m \tanh^{-1}(m)\right) W(m)\right.$$

$$\left. +\; \frac{e^{\lambda}}{2}\left(\beta\sqrt{2q}\; e^{-y^2} + \Delta\sqrt{\pi}\,\text{erfc}\,(y)\right)\right] \tag{3B.7}$$

$$\left\langle\left\langle\, f^{-1}(m) - \Delta m \right\rangle\right\rangle \cong \frac{1}{I}\left[2\int_0^{1-\delta} dm\, \left(\tanh^{-1}(m) - \Delta m\right) W(m)\right.$$

$$\left. +\; \left(\beta\sqrt{2q}\right)^3 e^{\lambda}\left(y e^{-y^2} + \frac{\sqrt{\pi}}{2}\text{erfc}(y)\right)\right]. \tag{3B.8}$$

Finally, the integrand in $\left\langle\left\langle\, (a(m) + B)^{-1} \right\rangle\right\rangle$ is well-behaved at ±1, so it is not necessary to expand this integral.

CHAPTER 4

THEORY OF NANOELECTRONIC DEVICES

4.1 INTRODUCTION: NANOELECTRONICS

Von Neumann computation and silicon VLSI circuity are the foundations upon which today's computers are built. The last three chapters investigated how neural networks may provide an alternative to the first. The next three chapters consider nanoelectronics, a possible alternative to the second. To the extent that computer architectures and their physical realizations are not always independent, there may be synergies that make the relation between neural networks and nanoelectronics still deeper. For example, as dimensions shrink, difficulties in long range communication between devices may favor cellular automaton or parallel-pipeline architectures, which are similar to neural networks having only local connections [Wolfram, 1984; Langton, 1990; Yang and Chiang, 1990; Averin and Likharev, 1992; Lent, Tougaw, and Porod, 1993; Tougaw, Lent, and Porod, 1993]. In addition, as the number of gates per chip exceeds ten or a hundred million, neural network learning may become an increasingly attractive alternative for configuring circuits to work despite design or fabrication errors.

The term "nanoelectronics" covers a broad range of devices small enough that quantum effects of tunneling, interference, or energy or charge quantization are important [for reviews, see Harris, Pals, and Woltjer, 1989; Datta and McLennan, 1990; Altshuler, Lee, and Webb, 1991; Beenakker and van Houten, 1991; Washburn and Webb, 1992; Buot, 1993]. A common feature of these devices is electron confinement to a low-dimensional

region. Another common, though not necessary, feature is operation at temperatures below 1K. In the devices considered here, which are fabricated in GaAs/AlGaAs heterostructures, confinement is achieved using the *split-gate technique* depicted schematically in Fig. 4.1. First, electrons are confined at low temperatures to a two-dimensional GaAs/AlGaAs interface a few hundred angstroms beneath the heterostructure surface. There they form a high-mobility two-dimensional electron gas (2DEG), shown as shaded in Fig. 4.1. Indium ohmic contacts make electrical contact to the 2DEG. Next, electrons are further confined by applying negative voltage to surface metal gates that deplete the areas beneath them. The gates, shown as hatched in Fig. 4.1, are fabricated with electron-beam lithography and Cr/Au evaporation (see Sec. 5.2) and can have features and separations smaller than 100 nm. Figure 4.1 shows two devices that can be formed in this way: a narrow constriction, or *quantum point contact* (Fig. 4.1(a)); and an isolated electron island, or *quantum dot* (Fig. 4.1(b)), consisting of two series point contacts with two additional confining walls. As is discussed below, electrons confined in point contacts and dots are effectively one- and zero-dimensional, respectively. These two devices are building blocks for the experiments of Chapter 6.

Figure 4.2 introduces several types of quantum-mechanical phenomena arising from electron confinement that are discussed in detail in this chapter. Figures 4.2(a) and (b) show quantization and interference in the low-temperature (T = 0.4 K) conductance of two series quantum point contacts that are about 100 nm wide and about 1 μm apart in a GaAs heterostructure device. Figure 4.2(a) shows the conductance G_1 and G_2 of each point contact individually vs. gate voltage, with the other point contact grounded. The plateaus occuring when the conductance is a multiple of $2e^2/h$ arise from one-dimensional confinement within the point contact channel. Figure 4.2(b) compares the measured conductance G_{12} of both point contacts in series to the value $G_1G_2/(G_1 + G_2)$ calculated from the curves of Fig. 4.2(a). To account for crosstalk between gates, which decreases

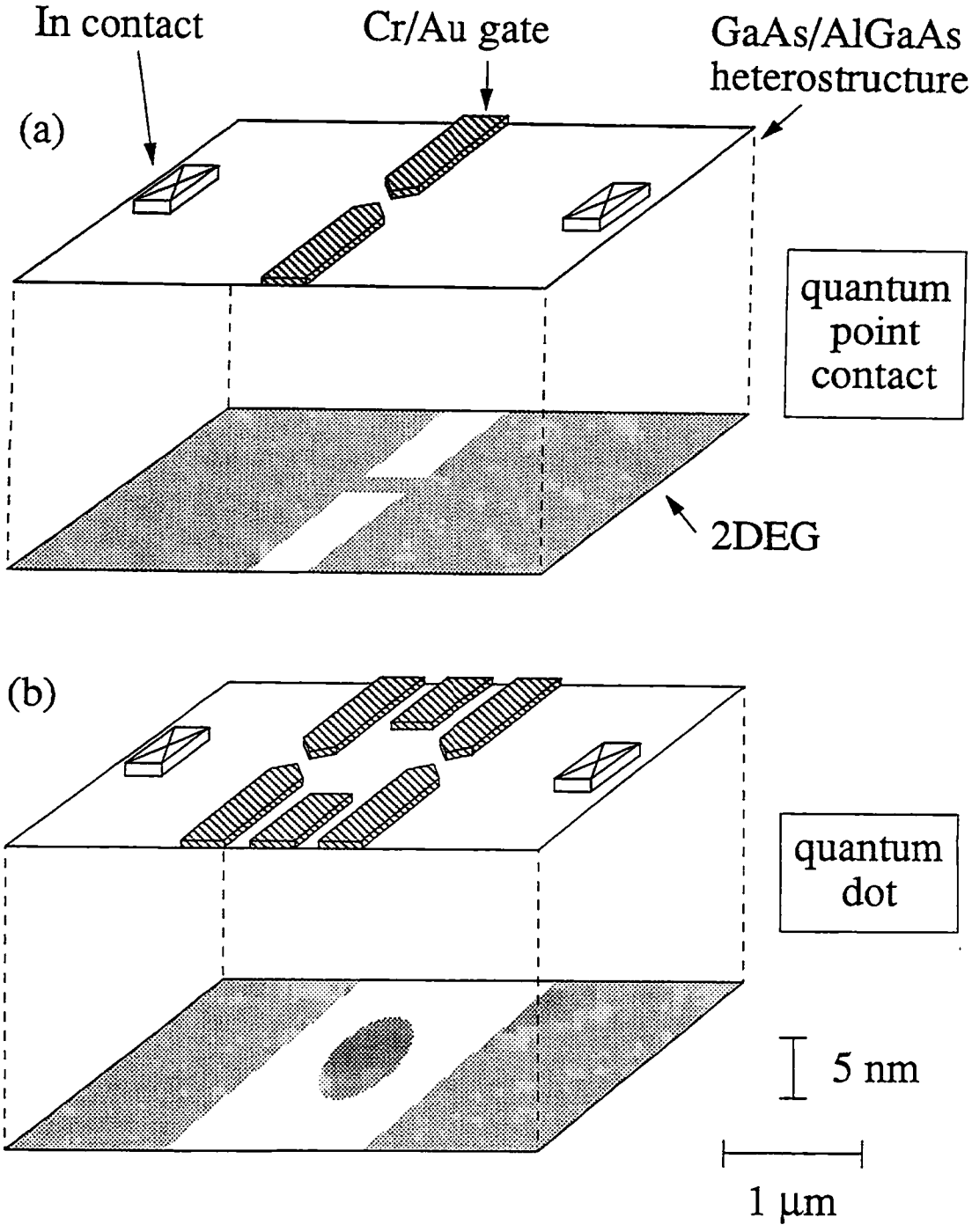


Fig. 4.1 Schematic examples of GaAs/AlGaAs heterostructure split-gate devices, showing Cr/Au metal gates (hatched) and In ohmic contacts (crossed) on surface and 2DEG (shaded) underneath at GaAs/AlGaAs interface. Depleted regions form in 2DEG when negative bias is applied to gates. (a) Quantum point contact or narrow constriction with width 100 nm. (b) Quantum dot or isolated electron island with diameter 500 nm. Note different horizontal and vertical scales.

the conductance of one point contact as the gate voltage of the other becomes more negative, the measured curve G_{12} is shifted horizontally to the left. Strong, reproducible interference and collimation effects make the measured conductance strikingly different from the simple series addition. Finally, Fig. 4.2(c) shows how the conductance G_d of a quantum dot at $T = 30$ mK varies with the voltage of a nearby gate. The dot, about 500 nm in diameter, is formed by constricting the region between two quantum point contacts that are nearly pinched off. Sharp conductance peaks at regular gate voltage intervals occur when the average number of electrons in the confined region changes by one.

Whether effects like those of Fig. 4.2 prove useful in wider device applications or remain laboratory novelties is an open question [Bate, 1988; Keyes, 1993]. Devices including electrometers, photodetectors, electron turnstiles, and precision current sources have already been built using single-electron charging phenomena as in Fig. 4.2(c) [Lafarge *et al.*, 1991; Cleland *et al.*, 1992; Kouwenhoven *et al.*, 1992; Martinis and Nahum, 1993; Martinis, Nahum, and Jensen, 1994; Dresselhaus *et al.*, 1994], and resonant tunneling diodes show promise for multi-valued logic [Capasso *et al.*, 1989; Buot, 1993]. At the same time, significant obstacles to applications remain [Landauer, 1989].

This chapter outlines the theory needed for understanding quantum effects such as those of Fig. 4.2, particularly the phenomenon of quantum dot conductance peaks, Fig. 4.2(c). The purpose is to provide a background for the experimental results of Chapter 6, rather than to attempt to survey the field. In Sec. 4.2, basic properties of two-dimensional electron gases formed in GaAs heterostructures are reviewed. Section 4.3 covers models of quantum point contacts. Section 4.4 discusses single-electron charging in single quantum dots using capacitive charging models and Anderson models. Finally, Sec. 4.5 looks at capacitive charging models and Hubbard models for coupled quantum dots. The capacitive charging model for coupled dots is investigated numerically in Sec. 6.6.

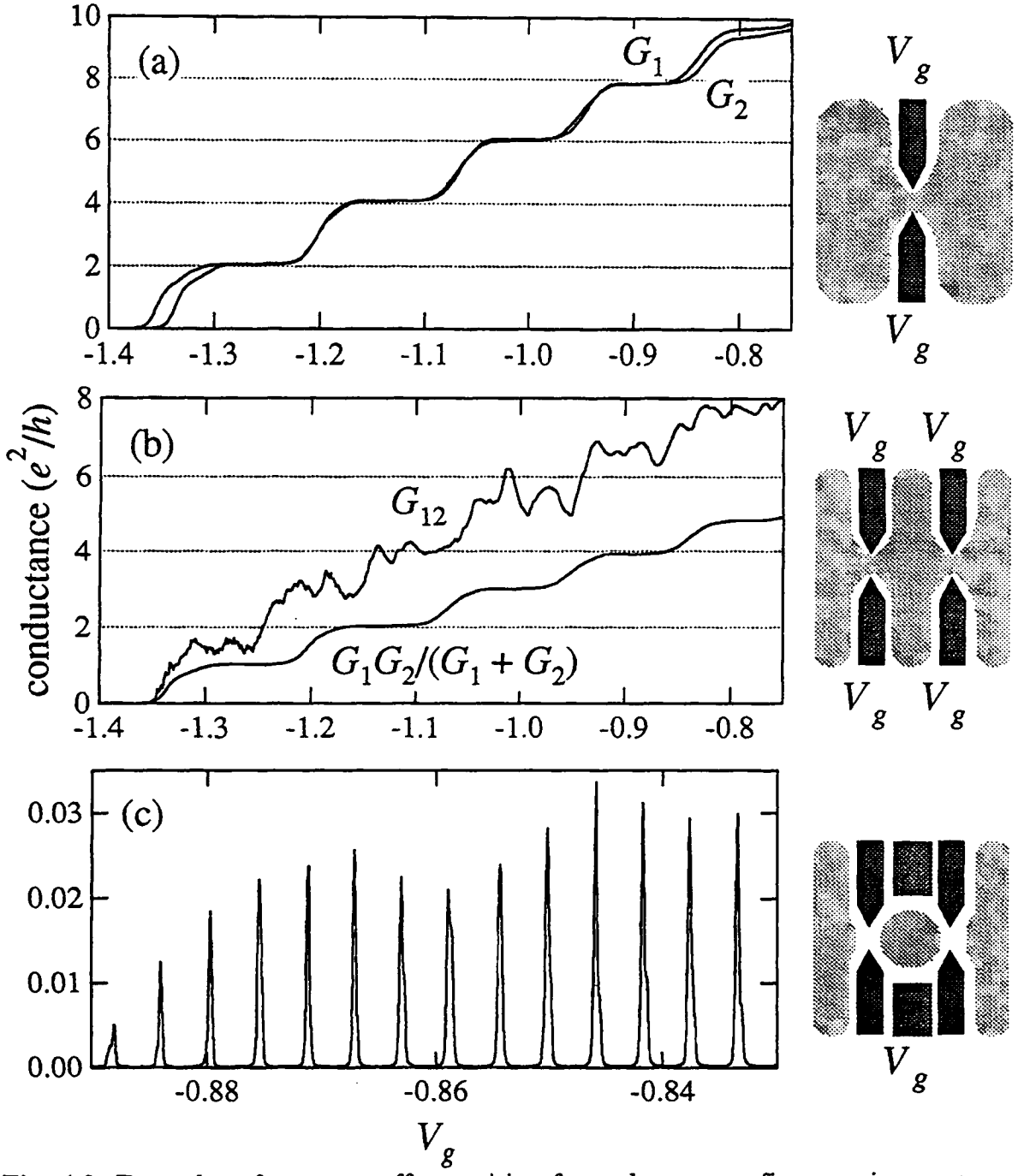


Fig. 4.2 Examples of quantum effects arising from electron confinement in quantum point contacts and quantum dots. Device schematics to right show 2DEG depletion and indicate swept gate voltage V_g. (a) Conductance G_1 and G_2 of two quantum point contacts individually vs. V_g at $T = 0.4$ K, showing quantization in units of $2e^2/h$. (b) Conductance G_{12} of two quantum point contacts in series vs. V_g at $T = 0.4$ K, compared to series addition of data from (a). Interference and collimation make G_{12} strikingly different from series addition. (c) Conductance of quantum dot vs. V_g at $T = 30$ mK, showing single-electron charging peaks.

4.2 TWO-DIMENSIONAL ELECTRON GASES

Two-dimensional electron gases (2DEGs) enable electron confinement by potential barriers to regions that are effectively two, one, or zero dimensional [for reviews, see Ando, Fowler, and Stern, 1982; Harris, Pals, and Woltjer, 1989]. Dimensionality is not an absolute property but is related to length scales—such as the mean free path l, the phase coherence length l_φ, the de Broglie wavelength λ_F, or the cyclotron radius l_B in a magnetic field—that characterize the electron motion. The 2DEGs most often studied form at material interfaces in Si MOSFETs and in GaAs/AlGaAs heterostructures. GaAs/AlGaAs heterostructures, which can have significantly higher quality 2DEGs due to a smoother interface and physical separation of carriers from donors, are considered exclusively here. Typical length scales for heterostructures at dilution refrigerator temperatures are $l \cong 5$ μm, $l_\varphi \cong 50$ μm, and $\lambda_F \cong 40$ nm, so that electrons confined in sub-micron split-gate devices are effectively zero dimensional and move coherently and ballistically, scattering almost entirely from confining gates rather than from impurities or phonons [Beenaker and van Houten, 1991].

A 2DEG forms in a GaAs heterostructure at an interface between GaAs and $Al_xGa_{1-x}As$, which at room temperature have band gaps differing by about 0.4 eV for $x = 0.3$. Electrons supplied by n-type dopants, usually Si, implanted in a δ-doped partial monolayer on the AlGaAs side of the interface transfer into the lower-energy GaAs conduction band. The electrons are confined in a nearly triangular well about 5 nm wide formed by the conduction band discontinuity and the electrostatic potential of the ionized donors. Within this well, the energy spectrum for motion perpendicular to the interface is discrete, and at low temperatures only one electric subband typically is populated. (GaAs heterostructure fabrication is described in Sec. 5.2, and Fig. 5.1 shows how the band discontinuity leads to electron confinement.)

As the name 2DEG suggests, the electron energy in the absence of a magnetic field is

$$E\left(k_x,k_y\right) = \frac{\hbar\left(k_x^2 + k_y^2\right)}{2m^*}, \tag{4.1}$$

where x and y are the directions parallel to the interface and $m^* = 0.067m_e$ is the electron effective mass in GaAs. Equation (4.1) implies a number density $n(E) = m^* E/\pi\hbar^2$ of electron states per unit area, giving a density of states $\rho(E)$ independent of energy:

$$\rho(E) = \frac{dn}{dE} = \frac{m^*}{\pi\hbar^2}. \tag{4.2}$$

One result of Eq. (4.2) is that, at low temperatures, the Fermi energy E_F is simply proportional to sheet density n_s:

$$E_F = n_s/\rho. \tag{4.3}$$

A metal gate fabricated on the heterostructure surface changes the density beneath it according to the plate capacitor formula

$$\delta n_s = \frac{\varepsilon\varepsilon_o}{ed}\delta V_g, \tag{4.4}$$

where $\varepsilon = 13$ is the GaAs dielectric constant, d is the 2DEG depth beneath the surface, and V_g is the gate voltage. Together, Eqs. (4.3) and (4.4) imply that the Fermi energy varies linearly with gate potential.

In the split-gate technique, negatively biased surface metal gates deplete the 2DEG beneath them entirely, reversibly forming regions of virtually any desired shape. This

confinement modifies the electron energy of Eq. (4.1). For example, the energy spectrum of a one-dimensional channel in the x direction is

$$E_n(k_x) = E_n + \frac{\hbar k_x^2}{2m^*}, \tag{4.5}$$

where the electric subbands E_n depend on the confining potential. Similarly, the spectrum of a zero-dimensional island can be fully discrete. The rest of this chapter examines the physics of split-gate devices.

4.3 QUANTUM POINT CONTACTS

Quantum point contacts like the one depicted schematically in Fig. 4.1(a) are building blocks for many split-gate devices [for reviews see van Wees, 1990; Beenakker and van Houten, 1991]. As shown in Fig. 4.2(a), the point contact conductace G_{qpc} has a step-like dependence on the voltage of the gates that form it [van Wees *et al.*, 1988]. This conductance quantization can be understood by modeling the point contact as a one-dimensional channel with transverse electric subbands, as in Eq. (4.5), and using the Landauer formula [Landauer, 1957, 1988; Büttiker, 1986]

$$G_{qpc} = \frac{2e^2}{h} \sum_{i=1}^{N} T_i \tag{4.6}$$

for the channel conductance. In Eq. (4.6), N is the number of electric subbands and T_i is the transmission probability of the ith subband. Making the gate voltage more negative decreases the channel width, causing the energy of transverse modes to pass through the Fermi energy one at a time. As the ith mode passes through the Fermi energy, its

transmission probability changes from $T_i \cong 1$ to $T_i \cong 0$, decreasing the conductance by $2e^2/h$ according to Eq. (4.6).

A more complete description of point contacts involves solving the Schrödinger equation for the wave functions in the constriction and the leads and then matching the solutions and their derivatives [Szafer and Stone, 1989; Haanappel and van der Marel, 1989]. These calculations show that, in the absence of disorder, the conductance has inflection points at multiples of $2e^2/h$ when the constriction length is small compared to the Fermi wavelength. These inflection points broaden into plateaus as channel length increases. For channels much longer than the Fermi wavelength, additional oscillations appear on conductance plateaus due to interference resonances between the front and back of the channel. On the other hand, if disorder effects are included [Haanappel and van der Marel, 1989; Nixon, Davies and Baranger, 1991; Laughton *et al.*, 1991; Timp, 1992] conductance plateaus become *less* pronounced as channel length increases. Finally, increasing temperature improves plateau quantization by smearing out the plateau oscillations arising from interference resonances and from disorder. Many of these effects are observed in experiments [van Wees, 1990; Timp, 1992].

Point contacts can be used in experiments to control the number of modes entering and leaving a region of 2DEG. The following two sections consider devices in which point contacts transmit much less than one mode, so that $G << 2e^2/h$. Used this way, they serve as tunnel barriers that enable the study of quantum dots that are very weakly coupled to their leads.

4.4 SINGLE QUANTUM DOTS

Single-electron charging occurs in a variety of systems: it was first observed in films of small metal granules [Giaver and Zeller, 1968] and arises also in the interaction of a scanning tunneling microscope tip with a small metal particle [van Bentum, Smokers, and van Kempen, 1988]. This section discusses the theory of single-electron charging in split-gate quantum dots [Kastner, 1992; van Houten, Beenakker, and Staring, 1992]. Quantum dots are formed in split-gate devices by a pair of series quantum point contacts with additional confining walls between them (see Fig. 4.1(b)). When its point contacts are biased in the tunneling regime, a sub-micron-sized dot is effectively zero-dimensional and has a discrete energy spectrum. Coulomb blockade or single-electron charging, the origin of the prominent conductance peaks in Fig. 4.2(b), occurs in dots under two conditions [for a review see Grabert and Devoret, 1992]. First, the tunnel barrier conductance between dot and leads must be much less than e^2/h, so that the number of electrons on the dot is a good quantum number. Second, the total capacitance C_Σ must be small enough that the charging energy for adding or removing a single electron is large compared to $k_B T$.

Coulomb blockade conductance peaks in single dots can be understood using a capacitive charging model that ignores electron energy levels [Kulik and Shekhter, 1975; Glazman and Shekhter, 1989; Averin and Likharev, 1991, 1992; Beenakker, 1991; van Houten, Beenakker, and Staring, 1992]. As shown in Fig. 4.3(a), this model treats a quantum dot as a site with no spatial extent that holds an integer number N of electrons. The dot has total capacitance $C_\Sigma = C_g + C$, where C_g is the capacitance to a nearby gate at voltage V_g and C is the sum of all other capacitances. Tunnel junctions, represented in Fig. 4.3(a) by split boxes, isolate the dot from each lead; in the simplest case both junctions

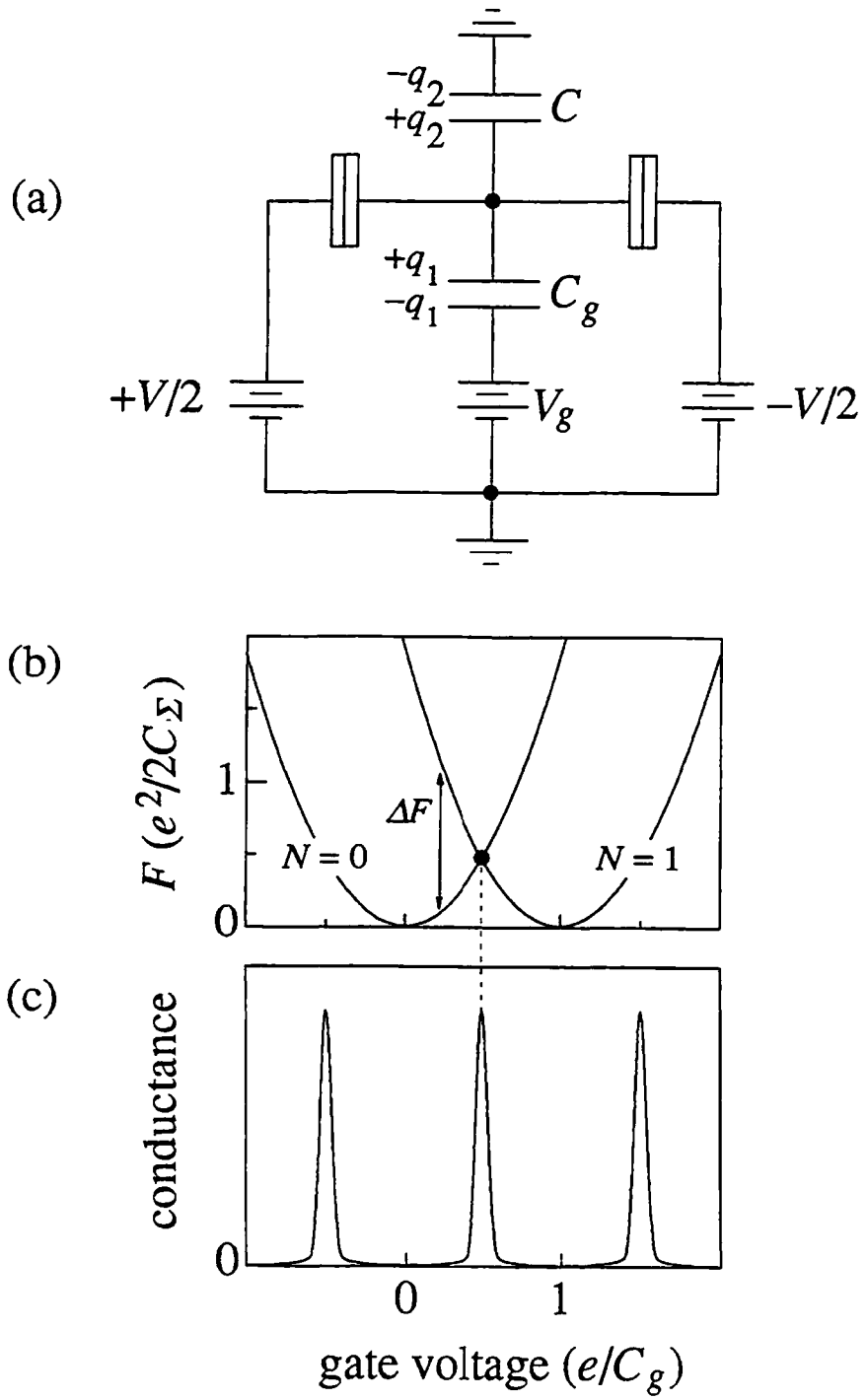


Fig. 4.3 (a) Capacitive charging model of single quantum dot. Dot has capacitance C to ground and C_g to gate at voltage V_g and is connected through tunnel junctions (split boxes) to leads which provide source-drain bias V. (b) Single dot electrostatic energy F_1, Eq. (4.8), vs. gate voltage for 0 and 1 electrons per dot. Also shown is energy barrier ΔF_1 of Eq. (4.9). (c) Schematic dot conductance vs. gate voltage. Conductance peaks occur at parabola intersections, where $F(N) = F(N+1)$.

have the same tunneling rate Γ, and their capacitance is neglected. A dc voltage V biases the dot symmetrically.

A simple picture of how this model leads to conductance peaks is shown in Fig. 4.3(b) and (c). Neglecting the voltage bias V, the electrostatic energy F_1 of a single dot is

$$F_1 = \frac{{q_1}^2}{2C_g} + \frac{{q_2}^2}{2C} + q_1 V_g. \tag{4.7}$$

The first two terms of Eq. (4.7) are the charging energy of the capacitors C_g and C; the last term is the work done by the battery providing the gate voltage V_g. The capacitor charges q_1 and q_2 can be eliminated through the constraint $q_1 + q_2 = -eN$ using the method of Lagrange multiplers [Ruzin *et al.*, 1992]. Up to a constant independent of N, the result is

$$F_1(N) = \frac{\left(eN - C_g V_g\right)^2}{2C_\Sigma}. \tag{4.8}$$

In Fig. 4.3(b), the energy (4.8) is plotted vs. gate voltage for different N. For a given gate voltage, it is energetically favorable for N to take on the value corresponding to the lowest parabola, and an energy barrier

$$\Delta F_1 = F_1(N+1) - F_1(N) = \frac{e^2}{C_\Sigma}\left(N + \frac{1}{2} - \frac{C_g V_g}{e}\right) \tag{4.9}$$

must be overcome to add an electron (a similar barrier exists for removing an electron). The barrier leads to Coulomb blockade or suppression of dot conductance, as shown schematically in Fig. 4.3(c). However, the barrier vanishes when the gate voltage equals

$e(N+1/2)/C_g$ and the two parabolas intersect, making the configurations N and N+1 equally favorable. In this case, the Coulomb blockade is lifted and the dot conductance is large, as shown in Fig. 4.3(c). The separation ΔV_p in gate voltage between neighboring peaks is simply

$$\Delta V_p = e/C_g. \tag{4.10}$$

As discussed below, thermal and lifetime effects broaden the conductance peaks.

This model can be modified to include electron energy levels [Averin and Likharev, 1991, 1992; Averin, Korotkov, and Likharev, 1991; Beenakker, 1991; van Houten, Beenakker, and Staring, 1992]. The appropriate energy is now

$$F(N) = \frac{\left(eN - C_g V_g\right)^2}{2C_\Sigma} + \sum_{k=1}^{\infty} n_k E_k, \qquad \sum_{k=1}^{\infty} n_k = N, \tag{4.11}$$

where E_k is the energy of level k and n_k equals 1 if level k is occupied and 0 otherwise. Including spin degeneracy effects, the spacing between peaks alternates between the two values

$$\Delta V_{p1} = \frac{e^2/C_\Sigma}{e^2/C_\Sigma + \Delta E/2}\left(\frac{\Delta E}{e} + \frac{e}{C_g}\right) \tag{4.12}$$

$$\Delta V_{p2} = \frac{e^2/C_\Sigma + \Delta E}{e^2/C_\Sigma + \Delta E/2}\left(\frac{\Delta E}{e} + \frac{e}{C_g}\right), \tag{4.13}$$

where $\Delta E = 2E_F/N$ is the spacing between electron energy levels, assumed to be uniform. Note that the average spacing between peaks remains e/C_g as in Eq. (4.10).

Linear response treatment of this model yields the conductance peak shape, which depends on the relative sizes of the temperature T, the level spacing ΔE, and the charging energy e^2/C_Σ [Kulik and Shekhter, 1975; Glazman and Shekhter, 1989; Beenakker, 1991]. For the case $k_BT < \Delta E < e^2/C_\Sigma$ relevant to the experiments of Chapter 6, the appropriate peak shape is the derivative of the Fermi function:

$$\frac{G}{G_{max}} = \frac{1}{\cosh^2(x/\gamma_T)}, \tag{4.14}$$

where

$$x = \mathrm{int}\left(\frac{C_g V_g}{e}\right) - \frac{1}{2}. \tag{4.15}$$

In Eqs. (4.14) and (4.15), G/G_{max} is the ratio of the dot conductance to its peak value, C_g and V_g are gate capacitance and gate voltage, and $\gamma_T = C_\Sigma k_B T/e^2$. Peak broadening can also occur due to the finite lifetime $1/\Gamma$ of electrons on a dot, which leads to energy level broadening via the uncertainty principle. Electron lifetimes become smaller and lifetime broadening more pronounced as a dot's coupling to its leads increases. No general theory exists when lifetime broadening and Coulomb blockade are both present [Beenakker, 1991]. For *noninteracting* electrons, meaning that $e^2/C_\Sigma = 0$, the short-lifetime limit $k_BT << \hbar\Gamma << \Delta E$ yields the Breit-Wigner peak shape [Büttiker, 1988]

$$\frac{G}{G_{max}} = \frac{1}{1+(x/\gamma_\Gamma)^2}, \tag{4.16}$$

where x is defined in Eq. (4.15) and $\gamma_\Gamma = \hbar\Gamma C_\Sigma/2e^2$. As is shown in Sec. 6.3, the thermal broadening shape (4.14) gives the better fit to experiments. A convolution of the

peak shapes (4.14) and (4.16) has been used to estimate the relative sizes of thermal and lifetime broadening [Foxman *et al.*, 1993].

A drawback of this charging model is its failure to treat both the single-electron energy levels and the charging energy using quantum mechanics, despite the fact that in experiments they have comparable magnitude (see Secs. 6.2 and 6.3). One way to treat both energies equally is to use an Anderson Hamiltonian for a single magnetic impurity [Gruner and Zawadowski, 1974; Kulik and Shekhter, 1975; Meir, Wingreen, and Lee, 1991, 1993; Wang, Zhang, and Bishop, 1994]. Modifying these Hamiltonians to describe a dot leads to

$$H = \sum_n E_n c_n^+ c_n + \frac{1}{2} UN(N-1) + \sum_{k \in L,R} E_k a_k^+ a_k + \sum_{\substack{n \\ k \in L,R}} \left(V_{kn} a_k^+ c_n + \text{H.c.} \right) . \qquad (4.17)$$

The operators c_n^+ and a_k^+ create electrons in the dot and leads, respectively, and $N = \sum_n c_n^+ c_n$ is the dot electron number operator. The quantum numbers n and k include spin. The four sums in Eq. (4.17) are, respectively, the dot energy levels E_n, the Coulomb repulsion $U = e^2/C_\Sigma$, the energy levels E_k in the left (L) and right (R) leads, and the tunneling matrix elements V_{kn} between the dot and the leads. Neglecting the leads and assuming a single spin-degenerate level E, the one-electron and two-electron ground-state energies for Eq. (4.17) are $E_g(1) = E$ and $E_g(2) = 2E + U$, respectively. In experiments in which a gate is used to tune the dot's Fermi energy $E_F(N) = E_g(N) - E_g(N-1)$, conductance peaks occur at the values $E_F(1) = E$ and $E_F(2) = E + U$ for which single-particle transitions can alter the number of electrons on the

dot. The separation in gate voltage between neighboring peaks thus corresponds to the charging energy U.

In addition to describing periodic conductance peaks, Anderson models predict a variety of other phenomena, including alternating peak height and width due to spin effects [Wang, Zhang, and Bishop, 1994] and Kondo peaks for nonzero bias voltage [Meir, Wingreen, and Lee, 1993]. Though Anderson models treat confinement and charging energies equally, they still use a constant Coulomb interaction U, ignoring the dependence of electron-electron interaction on electron separation. Models treating the electron-electron interaction explicitly have also been proposed [Johnson and Payne, 1993].

4.5 COUPLED QUANTUM DOTS

While Coulomb blockade in single quantum dots has been extensively investigated both theoretically and experimentally, less is known about larger arrays of coupled dots. Inter-dot coupling can be expected to give rise to new phenomena when the associated energy Δ becomes comparable to the charging energy U and the average level spacing ΔE. Chapter 6 describes experiments in which a gate voltage tunes the inter-dot coupling in small arrays of two and three dots. As shown there, the most prominent effect of inter-dot coupling is that it splits peaks into two and three peaks for double and triple dots, respectively, with the separation between split peaks approximately proportional to the inter-dot tunneling rate.

Peak splitting in coupled dots can be understood using a simple extension of the single-dot capacitive charging model, illustrated in Fig. 4.4(a). This model is investigated in greater detail in Sec. 6.6 for two and three coupled dots. The electrostatic energy F_M of an array of M noninteracting dots each containing N_i electrons, with gate capacitance C_{gi}, gate voltage V_{gi}, and total capacitance $C_{\Sigma i}$ is

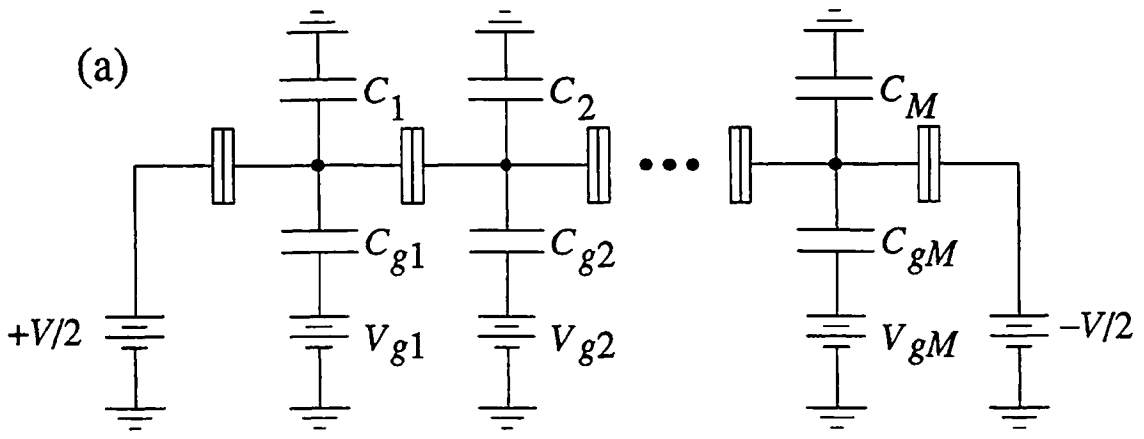


Fig. 4.4 (a) Capacitive charging model of coupled quantum dots. Each dot i, $i = 1,\ldots,M$, has capacitance C_i to ground and C_{gi} to gate at voltage V_{gi} and is connected through tunnel junctions (split boxes) to neighboring dots; end dots are connected to leads which provide source-drain bias V.

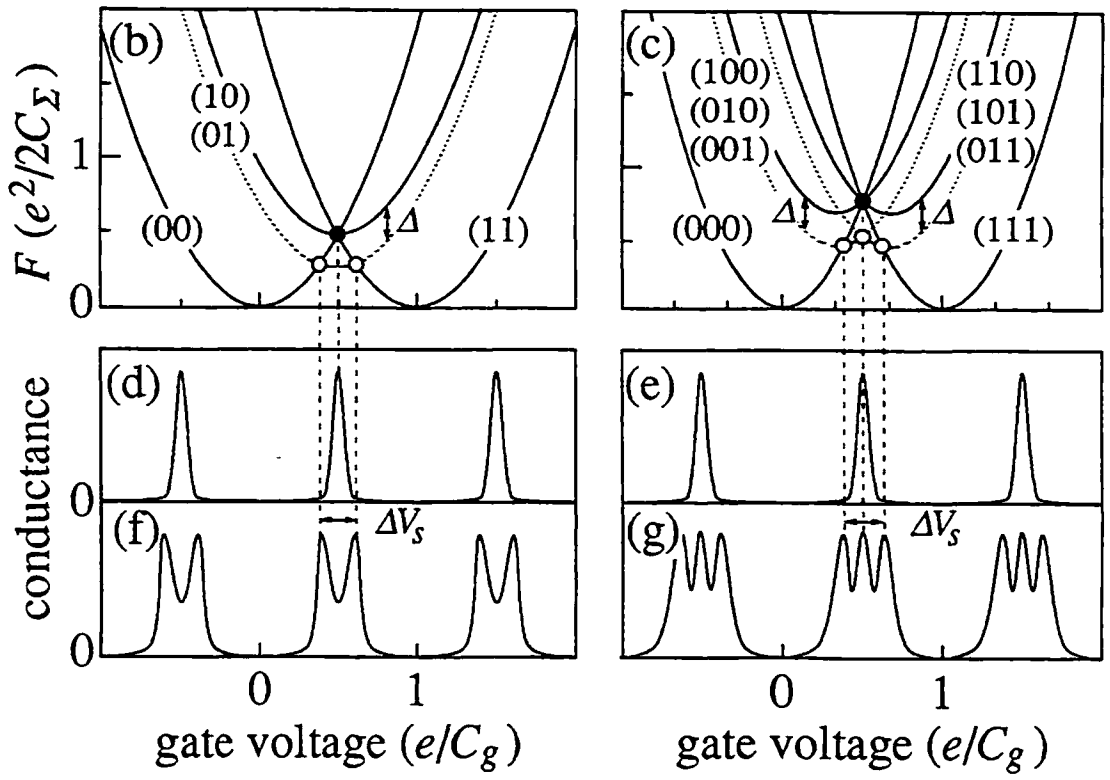


Fig. 4.4 (b) Double-dot ($M = 2$) and (c) triple-dot charging energy ($M = 3$), Eq. (4.18), vs. gate voltage for indicated numbers ($N_1N_2...$) of electrons on each dot when all $C_i = C$, $C_{gi} = C_g$, and $V_i = V$. Without inter-dot coupling, parabolas with unequal N_i are degenerate (solid curves). Coupling removes degeneracy, shifting lowest parabola down by Δ (dotted curves). (d) Double-dot and (e) triple-dot conductance vs. gate voltage without coupling (schematic). Conductance peaks occur at filled markers in (b) and (c). (f) Double-dot and (g) triple-dot conductance vs. gate voltage with coupling (schematic). Conductance peaks occur at open markers in (b) and (c). Coupling splits peaks, with split peak separation ΔV_S proportional to Δ (Eq. (4.19)).

$$F_M = \sum_{i=1}^{M} \left(eN_i - C_{gi}V_{gi}\right)^2 \Big/ 2C_{\Sigma i}\,. \tag{4.18}$$

This expression, like Eq. (4.7), neglects tunnel junction capacitance and assumes zero bias voltage. The total energy F is shown as a function of gate voltage for two and three dots in Figs. 4.4(b) and (c) respectively for the special case where C_{gi}, V_{gi}, and $C_{\Sigma i}$ are the same for all dots. In the absence of inter-dot coupling, the number N_i of electrons on each dot is a good quantum number and can be indicated using the notation $(N_1N_2\cdots)$. Parabolas with all N_i equal [(00) and (11) for Fig. 4.4(b)] touch the horizontal axis; parabolas with different N_i [(10) and (01) for Fig. 4.4(b)] have higher energy and are degenerate. As in the single-dot diagram of Fig. 4.3(a), the most energetically favorable configuration (N_1N_2...) for a given gate voltage corresponds to the lowest parabola, and a Coulomb blockade energy barrier must be overcome to add or remove an electron on any dot. This energy barrier vanishes at gate voltages for which the parabolas intersect. The resulting conductance, shown in Fig. 4.4(d) and (e), looks similar to that of a single dot, with peaks occuring when all dots add an electron simultaneously.

With coupling between dots—arising, for example, from inter-dot tunneling matrix elements or inter-dot capacitance—the electron eigenstates are no longer confined to each dot, and electrons can be shared between dots. The main effect is to lift the degeneracy of the raised parabolas of Fig. 4.4(b) and (c). How this happens depends on the details of the interaction, but one effect is that one of the raised parabolas is pushed downward by some energy Δ, as indicated by the dotted curves in Figs. 4.4(b) and (c). The interaction may affect the lower parabolas as well; for small Δ this effect is of order Δ^2 and is ignored in this argument. The result is that two new parabola intersections appear for the double dot and three for the triple dot, implying that each conductance peak splits in two (Fig. 4.4(f)) or three (Fig. 4.4(g)) by an amount

$$\Delta V_s = \left(2C/C_g e\right)\Delta. \tag{4.19}$$

The ratio C/C_g acts as a capacitive lever arm: in experiments, $C/C_g >> 1$ so that $\Delta V_s >> \Delta/e$.

Other phenomena are predicted by the capacitive charging model of coupled dots when the dot capacitances C_i in Eq. (4.18) are not equal. Gate capacitance mismatch tends to suppress some conductance peaks through the process of stochastic Coulomb blockade [Ruzin *et al.*, 1992]. For double quantum dots, peak suppression results from the quasiperiodic beating between the two conductance peak periods e/C_1 and e/C_2, leading to peak height modulation with period approximately $e/|C_1 - C_2|$. Peak suppression in double dots becomes increasingly strong as temperature decreases because, with no dot interaction, conductance peaks occur only when the condition for a peak in each dot separately lines up to within $k_B T/e$ in gate voltage. Similar effects occur for more than two dots (see Sec. 6.6). With inter-dot coupling, peaks not suppressed by the stochastic Coulomb blockade split through the same mechanism described above.

Quantum mechanical treatments of coupled dots also predict conductance peak splitting. Hubbard models [for a review, see Adler, 1968], which are many-site generalizations of the Anderson model used in Sec. 4.4 for single dots, have been investigated for coupled dots in a number of configurations [Stafford and Das Sarma (1994); Klimeck, Chen, and Datta (1994)]. A simple generalization of the Anderson Hamiltonian Eq. (4.17) to an array of M dots is [Kulik and Shekhter, 1975]

$$H = \sum_{i,n} E_{in}\, c_{in}^{+} c_{in} + U \sum_{i} N_i(N_i - 1) + t \sum_{\substack{n,n' \\ \langle i,j \rangle}} \left(c_{in}^{+} c_{jn'} + \text{H.c.} \right) + W \sum_{\langle i,j \rangle} N_i N_j$$

$$+ \sum_{k \in L,R} E_k\, a_k^{+} a_k + \sum_{\substack{n \\ k \in L}} \left(V_{kn}\, a_k^{+} c_{1n} + \text{H.c.} \right) + \sum_{\substack{n \\ k \in R}} \left(V_{kn}\, a_k^{+} c_{Mn} + \text{H.c.} \right). \tag{4.20}$$

The index i, $i = 1,\ldots,M$, labels the dots, and $\langle i,j \rangle$ indicates nearest neighbor summation. The first two and the last three terms of Eq. (4.20) are simple extensions of Eq. (4.17) to more than one dot. The third and fourth terms are new: they describe dot interaction via tunneling matrix elements t and via Coulomb interaction W. In more complicated models, t and W can also depend on the indices i and n.

To see how dot interaction leads to peak splitting in Hubbard models, consider how electrons fill a double dot when $t << U$, $W = 0$, each dot has a single spin-degenerate level E, and the leads are neglected [Ashcroft and Mermin, 1976; Klimeck, Chen, and Datta, 1994]. The one-electron ground-state energy is $E_g(1) = E - t$; the electron is shared by both dots equally, lowering its energy by the coupling t. When a second electron is added, it is energetically favorable for each electron to be confined to a single dot to avoid incurring the charging energy U. Thus the two-electron ground-state energy is $E_g(2) = 2E + O\left(t^2\right)$. The charging energy cannot be avoided when a third electron is added, and the three-electron ground-state energy is $E_g(3) = 3E + U - t$. Finally, adding a fourth electron causes two electrons to be confined to each dot so that the four-electron ground-state energy is $E_g(4) = 4E + 2\,U$. In experiments in which a gate is used to tune the dot's Fermi energy $E_F(N)$, conductance peaks occur at the four values $E_F(1) = E - t$, $E_F(2) = E + t$, $E_F(3) = E + U - t$, and $E_F(4) = E + U + t$ for which single-particle transitions can occur. For $t << U$, the conductance thus consists of two double peaks each having splitting proportional to the coupling t and separated by an amount proportional to the charging energy U.

More detailed simulations of Hamiltionians like Eq. (4.20) show that peak splitting persists in the presence of inelastic scattering and disorder in the dot levels or inter-dot tunneling matrix elements [Stafford and Das Sarma (1994); Klimeck, Chen, and Datta (1994)]. Inter-dot capacitive coupling, described by the term proportional to W in Eq. (4.20), is also predicted to yield peak splitting [Klimeck, Chen, and Datta (1994)].

4.6 SUMMARY

This chapter summarizes theories relevant to the experiments of Chapter 6, focusing on single-electron charging in single and coupled quantum dots formed in GaAs/AlGaAs heterostructures with split gates. Simple capacitive charging models adequately describe the most prominent charging effects in these devices. For single dots, the models predict conductance peaks with separation determined mainly by the charging energy U and, to a lesser extent, by the average level spacing ΔE. For coupled dots, they predict conductance peak splitting, with separation between split peaks proportional to the inter-dot coupling energy Δ, as well as peak suppression via the stochastic Coulomb blockade when gate capacitances are unequal. Anderson and Hubbard models, more rigorous in that they treat all the relevant Hamiltonian terms with quantum mechanics, make similar predictions.

CHAPTER 5

FABRICATION AND MEASUREMENT OF NANOELECTRONIC DEVICES

5.1 INTRODUCTION

At Harvard as at many other research facilities, fabrication and measurement of nanoelectronic devices has become routine in recent years. Starting from GaAs/AlGaAs heterostructures provided by the Gossard group at U. C. Santa Barbara, devices are prepared at Harvard with a class 100 clean room and an electron beam lithography system and measured in one of several He evaporation and dilution refrigerators using commercial low-noise electronics. Under ideal conditions, a device can be fabricated and cooled to millikelvin temperatures in a few days. Of course, designing and optimizing a good experiment takes much longer.

Many fabrication and measurement procedures used in the Westervelt laboratory have been described with great detail and redundancy by recent graduates [Hopkins, 1990; Rimberg, 1992; Baskey, 1994; Mar, 1994; Berry, 1994]. However, the single-electron charging experiments of Chapter 6 involved some techniques not previously used in the laboratory. Sections 5.2 and 5.3 of this chapter briefly describe fabrication and measurement for single-electron charging experiments, focusing on these techniques.

5.2 DEVICE FABRICATION

This section reviews how GaAs/AlGaAs heterostructures are made (Sec. 5.2.1) and how they are fashioned into split-gate nanoelectronic devices (Sec. 5.2.2).

5.2.1 Molecular beam epitaxy

GaAs/AlGaAs heterostructure wafers are prepared in the Gossard laboratory at U. C. Santa Barbara using a Varian Gen II ultra-high vacuum molecular beam epitaxy (MBE) chamber. This machine uses Knudsen effusion cells—essentially crucibles of molten material—to deposit "sandwiches" of Ga, As, Si, In, or Be on the surface of a GaAs substrate, with atomically precise interfaces between each layer.

Figure 5.1(a) shows schematically a cross section of the wafer used in the experiments of Chapter 6. Starting from the substrate, the wafer consists of the following layers: GaAs (100 nm), AlGaAs (100 nm), a 30-period GaAs/AlGaAs smoothing superlattice (120 nm total), GaAs (1000 nm), an AlGaAs spacer (22 nm), a Si delta doping layer with sheet density 8×10^{12} cm^{-2}, an AlGaAs barrier (15 nm), and a GaAs cap (5 nm). GaAs is shown as white, AlGaAs as shaded, and Si as hatched in the figure. All AlGaAs layers have Al fraction $x = 0.31$. The region near the heterostructure surface is shown in expanded view in Fig. 5.1(b), and the conduction band edge of this region is shown in Fig. 5.1(c) vs. depth from the surface. As described also in Sec. 4.2, a two-dimensional electron gas (2DEG) forms at the interface between the thick GaAs layer and the AlGaAs spacer, where electrons are confined in a nearly triangular well about 5 nm wide formed by the conduction band discontinuity and the electrostatic potential of the ionized donors. The electron gas in this wafer has sheet density 3.7×10^{11} cm^{-2} and mobility 5×10^{5} cm^2/Vs at 10 K, as measured by Ken Campman at U. C. Santa Barbara. The mobility increases to 7×10^{5} cm^2/Vs at 0.4 K, as measured by Jordan Katine at Harvard.

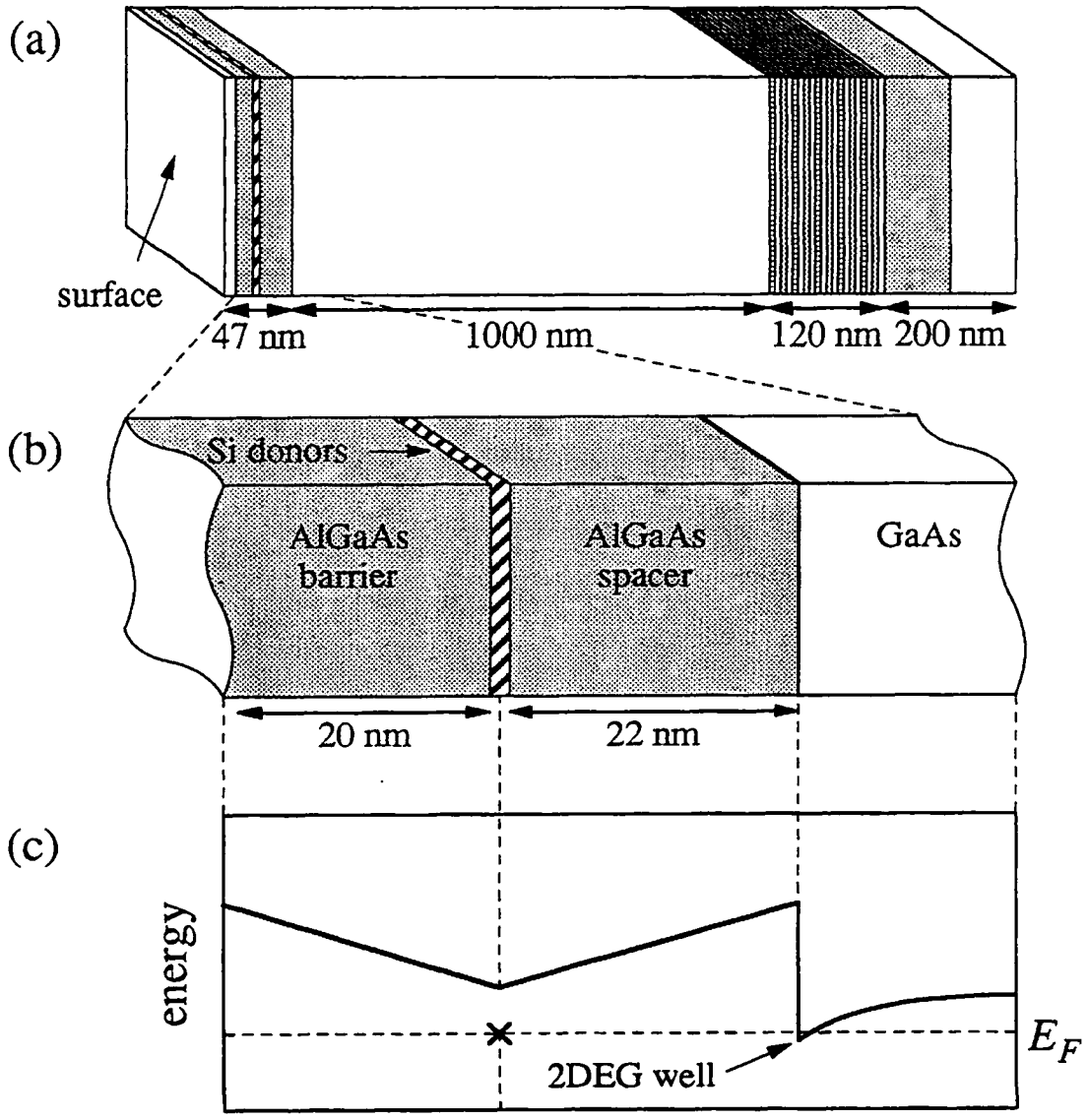


Fig. 5.1 (a) Cross-section of GaAs/AlGaAs heterostructure, showing GaAs as white, AlGaAs as shaded, and Si dopants as hatched. (b) Expanded view showing Si dopant layer and GaAs/AlGaAs interface where 2DEG forms. (c) Conduction band edge vs. distance from surface for (b). 2DEG forms in triangular well formed by interface conduction band discontinuity and electrostatic potential of ionized donors.

A number of considerations must be taken into account in choosing the layer widths near the heterostructure surface, shown in Fig. 5.1(b). Perhaps the most important is the tradeoff between high mobility and small feature size, both desirable characteristics for nanoelectronic devices. The problem is that both mobility and minimum feature size increase as cap, barrier, and spacer layers are made thicker; mobility improves the further the 2DEG is from dopants and surface states, while minimum feature size for split-gate devices is roughly equal to the distance from the 2DEG to surface gates. A second consideration is that the 2DEG sheet density decreases as the spacer layer is made thicker and as the cap and barrier layers are made thinner, since the 2DEG interface well and the surface states compete for the fixed amount of dopant charge. These tradeoffs are evident in Table 5.1, which compares the wafer described above to two nearly identical ones; all three wafers were fabricated in immediate succession. The table shows that both mobility and sheet density decrease as the AlGaAs barrier thickness decreases from 47 to 37 nm. The wafer with 47 nm AlGaAs barrier was used for the experiments of Chapter 6, since feature size was not crucial.

AlGaAs barrier thickness (nm)	sheet density (cm^{-2})	mobility (cm^2/Vs)
→ 47	3.7×10^{11}	5×10^5
42	3.6×10^{11}	4×10^5
37	2.6×10^{11}	3×10^5

Table 5.1 Sheet density and mobility at $T = 10\text{K}$ for three wafers differing only in AlGaAs barrier thickness. Decreasing barrier thickness improves minimum feature size but lowers mobility and sheet density. Arrow indicates wafer used in experiments of Chapter 6. Wafers were fabricated and measured by Ken Campman at U. C. Santa Barbara.

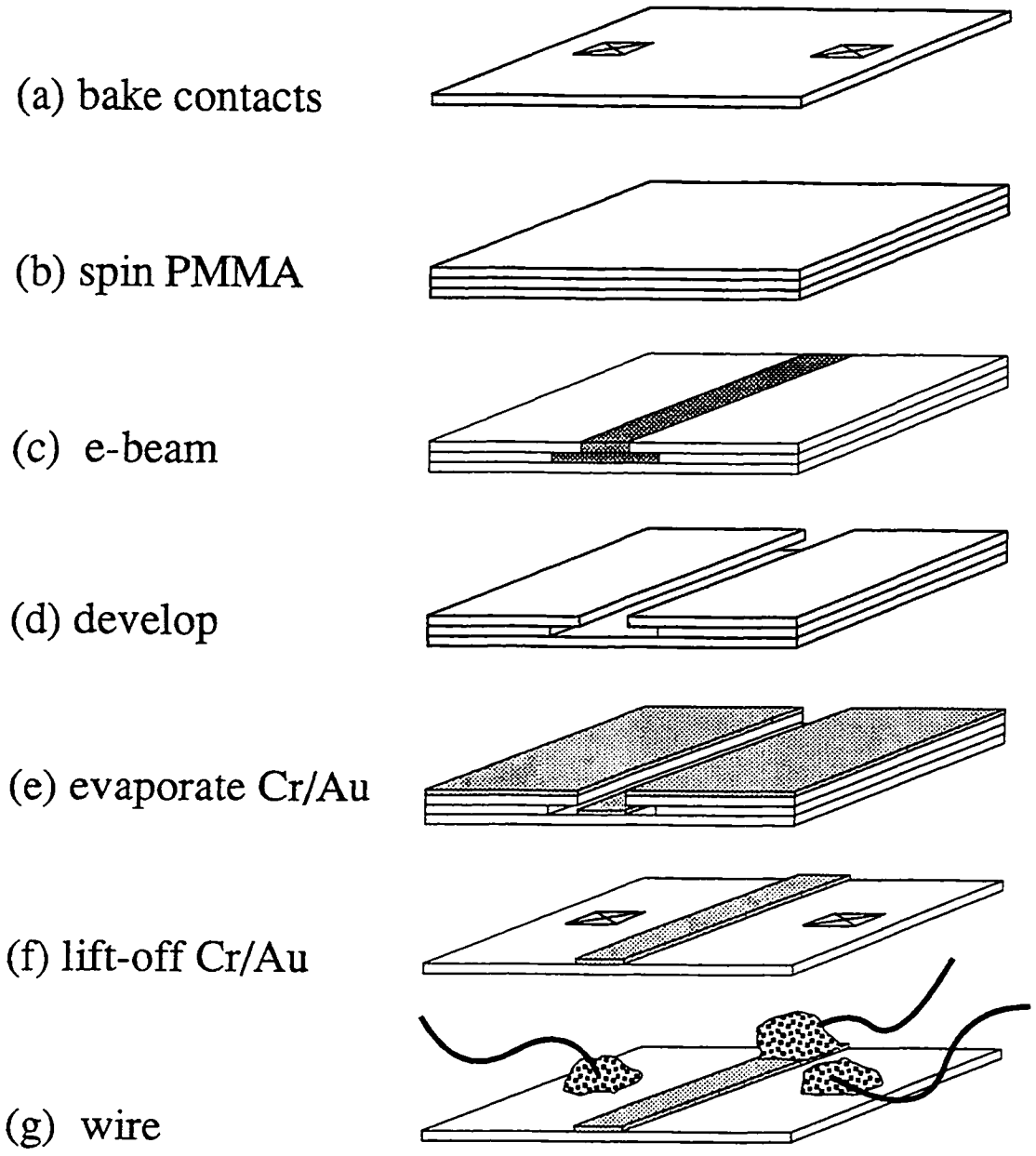


Fig. 5.2 Fabrication steps for split-gate nanoelectronic devices, described fully in text.

5.2.2 Contacting, lithography, and metallization

Finished wafers are shipped from U. C. Santa Barbara to Harvard, where the remaining fabrication is performed. The fabrication steps for split-gate devices are depicted in Fig. 5.2 and described below. The guiding principle of each step is reproducibility: whenever possible, parameters should be kept to the same values from one fabrication run to the next.

(a) **bake contacts** Electrical contact is made to the 2DEG when In, placed on the sample surface and heated to melting, punches through to the 2DEG interface. In contacts are shown as crosses in Fig. 5.2(a). The process is uncontrolled, leading to large variations in contact quality. Fortunately, contact quality is considerably less important for single-electron charging than for other experiments: since the device resistance can be several MΩ in charging experiments, little voltage is dropped over the contacts.

The contacting process is as follows. First a wafer is cleaved into samples approximately 3×5 mm^2 and cleaned for 10 minutes each in boiling trichloroethylene (TCE), acetone (ACE) with ultrasound, and methanol (METH) with ultrasound. Next small pieces of In are cut and scraped with a clean scalpel and pressed firmly onto the heterostructure surface with clean tweezers and probes. The sample is then placed in a sealed chamber through which forming gas (80% He, 20% H) flows. The sample is heated at 100 C for 2 minutes, 180 C for 2 minutes, 400 C for 50 to 80 seconds, and 180 C for 2 minutes, and is allowed to cool to room temperature before opening the chamber. The crucial concerns are that (i) the In and sample surface are very clean and (ii) heating at 400 C is stable and precisely timed.

Contacts are tested by dunking samples in a liquid He storage dewar and measuring contact resistance at $T = 4.2$ K with a two-probe, 1 μA current bias. Good contacts can have two-probe resistances below 100 Ω; typical contacts have resistances of 300 to 400 Ω. Another important consideration is contact diodicity, meaning resistance

asymmetry when current direction is reversed. Good contacts have less than 1% asymmetry for 1 μA current bias.

(b) **spin PMMA** An e-beam resists of poly(methyl methacrylate) (PMMA) is applied to the sample surface in Harvard's class 100 clean room. The sample is cleaned for 10 minutes each in ACE with ultrasound and METH with ultrasound. A layer each of 496 K and 950 K PMMA, both diluted to 2% by weight in chlorobenzene, are spun onto the sample surface at 4000 rpm for 1 minute. After each spin the sample is baked for 15 to 20 minutes on a hot plate at 180 C. Profilometry indicates that the resulting bilayer is approximately 150 nm thick. The important considerations are that the bilayer is free of dirt and that it has uniform thickness. Thickness gradients, visible to the eye as interference fringes, lead to nonuniform e-beam exposure.

(c) **e-beam** The PMMA is patterned using a JEOL 6400 scanning electron microscope with lithography pattern generation software and hardware obtained from J. Nabity Lithography Systems. Since PMMA is a positive e-beam resist (though it can be negative when very heavily exposed), the beam exposes those areas where metal is to be placed. Exposed areas are shown as darkly shaded in Fig. 5.2(c). Note the intentional undercut in the first PMMA layer compared to the second, arising from differences in exposure sensitivity. This undercut facilitates the lift-off step described below.

The guiding principle of reproducibility is especially important for e-beam lithography. One reason is that the pattern generation software assumes unsoundly that the exposure process is linear, meaning that only the total dose (the product of dwell time and beam current) is important. To improve reproducibility, beam current should be adjusted so that the same currents are always used, rather than using whatever current the machine puts out. Stage height should always be set to the same value, rather than keeping the last user's setting. Beam lenses should be cleared regularly to minimize hysteresis. Finally, extra care should be taken in focusing and stigmating the beam.

(d) **develop** The exposed sample is developed for 60 seconds in 1:3 methyl isobutyl ketone: isopropanol (MIBK:IPA) with 1.3% methyl ethyl ketone (MEK) by volume, then immediately rinsed in two washes of IPA and blown dry with ultrapure N_2 gas. Care should be taken that the develop time is as close to 60 seconds as possible since under- or over-developing can be indistinguishable from under- or over-exposure. It is also extremely important that all carbon paint (used to mount the sample for e-beam lithography) be scraped off; it dissolves in IPA and adheres strongly to the exposed GaAs surface.

(e) **evaporate Cr/Au** Developed samples should be placed as quickly as possible in the general-purpose thermal evaporator; exposure to air makes metal films less likely to adhere. The metallization process consists of depositing 100 Å of Cr (evaporated from a Cr-coated rod) and 200 to 250 Å of Au (evaporated from a Mo boat coated with alumina). The sample is pressed to the evaporator stage with a clip and the stage cooled with liquid N_2 to –20 to –40 C (others have successfully air-cooled the stage). The sources should be cleaned by heating before cooling the stage, in order to prevent dirt from cryopumping onto the sample. Important considerations are that (i) the evaporator arms are kept free from grease and dirt (very slight amounts of which can cause metals films not to stick) and (ii) the Mo boat is kept clean and handled carefully to avoid bending (which can increase its resistance, leading to excessive sample heating during evaporation and reducing film quality). The resulting metal film is shown as the lightly shaded layer in Fig. 5.2(e).

(f) **lift-off Cr/Au** The excess metal film is removed by soaking the sample in ACE for several hours (preferably overnight). Patience is crucial: the excess metal virtually always comes off, and slow lift-off, indicating that the film has adhered well, is always preferable to fast lift-off. Some ultrasound is almost always necessary—again an indication that the film has adhered well—and a sample that does not survive it most likely was not good anyway. It may help to soak the sample in a near-vertical orientation, so that metal sloughs off as the PMMA under it dissolves.

(g) **wire** This can be the most excruciating step. The samples of Chapter 6 were hand-wired, meaning that small pieces of In were scraped clean and pressed onto Cr/Au pads to form gate contacts, to which Au wires were then attached. The In pieces and Au wires are depicted in Fig. 5.2(g). Typical pad sizes and separations are 150 to 200 μm, so that caution, concentration, and a caffeine-free bloodstream are essential. The difficulty is compounded by the danger of inadvertently destroying the sample with static electricity. To reduce this danger, the wiring process should be done on a metal surface (for example an Al sheet) with the sample, surface, and all tools grounded to one point. Using an anti-static mat, taking off one's shoes, and running a dehumidifier in the room can also help. Even with these precautions, static can still destroy the sample, especially when grounded tools are forcibly unstuck from In contacts or when devices are wired onto the He evaporation refrigerator probe.

5.3 LOW-TEMPERATURE MEASUREMENTS

5.3.1 Cryogenics

Typically, quantum effects in nanoelectronic devices are associated with one or more energy scales E. Observing an effect then requires that $k_B T_e < E$, where k_B is Boltzmann's constant and T_e is the electron temperature. For the Coulomb blockade experiments of Chapter 6, the relevant energy scales are the Coulomb charging energy $U \cong 70$ μeV, the single-electron level spacing $\Delta E \cong 20$ μeV, and the dot interaction energy $0 \leq \Delta \leq 20$ μeV (see Sec. 4.4). These energies dictate that coupled dot experiments be performed in a He dilution refrigerator (or "fridge"), for which $T_e \cong 5$ μeV. In a He evaporation refrigerator, for which $T_e \cong 40$ μeV, charging effects are observable but dot interactions cannot be resolved.

The dilution refrigerator is an Oxford 200 top-loading model, manufactured by Oxford Instruments Ltd. Usually, the fridge consumes less than 20 liters of liquid He per day and maintains a base temperature of about 14 mK. The top-loading feature, when working properly, allows samples to be loaded and unloaded without warming the mixing chamber above about 1.5 K. Other fridge features not used in the experiments of Chapter 6 are a 7 T superconducting magnet and a temperature control system. Additional information on the fridge can be found in the Harvard-annotated version of the operating manual and in previous theses [Hopkins, 1990; Rimberg, 1992].

Radiation shielding, always important for dilution refrigerators, is crucial for single-electron charging experiments. Principal concerns are (i) radiation traveling down the leads to the sample and (ii) ambient black-body radiation from warmer parts of the fridge or from outside the fridge. The tell-tale sign of radiation leaks in single-electron charging experiments is a peak-to-trough ratio lower than expected from thermal activation alone, indicating that photon-assisted tunneling is occurring [Martinis and Nahum, 1993]. The fridge's pre-existing lead shielding—consisting of copper-powder microwave filters heat-sunk to the mixing chamber and 5 kΩ metal-film resistors attached to the sample slug about 1 cm from the sample—is sufficient for Coulomb blockade experiments. On the other hand, the pre-existing shielding for ambient radiation—consisting primarily of a cylindrical brass sheath that fits snugly over the sample slug—is *not* sufficient. The shielding can be improved by blocking the larger gaps between slug and sheath with microwave-absorbing sponge and copper tape and by painting the smaller ones shut with silver paint. Care must be taken not to increase the slug outer diameter, as there is very little tolerance in the top-loading process. This additional shielding was found to increase the Coulomb blockade peak-to-trough ratio from less than 10:1 to greater than 100:1 for experiments at base fridge temperature. [Shielding and photon-assisted tunneling are discussed further in Martinis,

Devoret, and Clarke, 1987; Cleland, Schmidt and Clarke, 1992; Martinis and Nahum, 1993; Martinis, Nahum, and Jensen, 1994; Dresselhaus *et al.*, 1994.]

Because the current signal can sometimes be less than 100 fA in Coulomb blockade experiments, electrical pick-up from nearby electronic or mechanical equipment is also a concern, and preventative measures must be taken that were not necessary for previous fridge experiments. Generally, *all* equipment in the fridge vicinity should be turned off and unplugged, and *all* wires or cables attached to the fridge disconnected, if they are not essential to the experiment. The N_2 jacket level meter is an especially flagrant noise source that should always be turned off and unplugged. Other noise sources include the He level meter and the thermometry and temperature control electronics. In addition, signal-to-noise improves after disconnecting various cables attached to the fridge whose functions are tangential to its operation. Other noise considerations related to the measurement electronics are discussed in the following subsection.

5.3.2 Electronics

Coulomb blockade measurements are made with commercial and custom-made low-noise electronics integrated with a microcomputer for recording data. A typical experimental arrangement is shown in Fig. 5.3. A 5 to 10 μV, 11 Hz ac voltage is generated by voltage-dividing a signal from the built-in oscillator of an EG&G Princeton Applied Research (PAR) 124A lockin amplifier. This signal voltage-biases the sample, and the resulting current is measured with a battery-powered Ithaco 1211 current preamplifier (borrowed from the Tinkham group) and detected by the same PAR 124A. A Fluke 8842A digital multimeter measures and digitizes the PAR 124A output. Simultaneously, another Fluke 8842A reads a gate voltage generated by a custom battery-powered voltage source. Both digitized signals are passed along a GPIB interface bus to an Apple Macintosh computer, where they are read and recorded using Labview, an instrumentation control

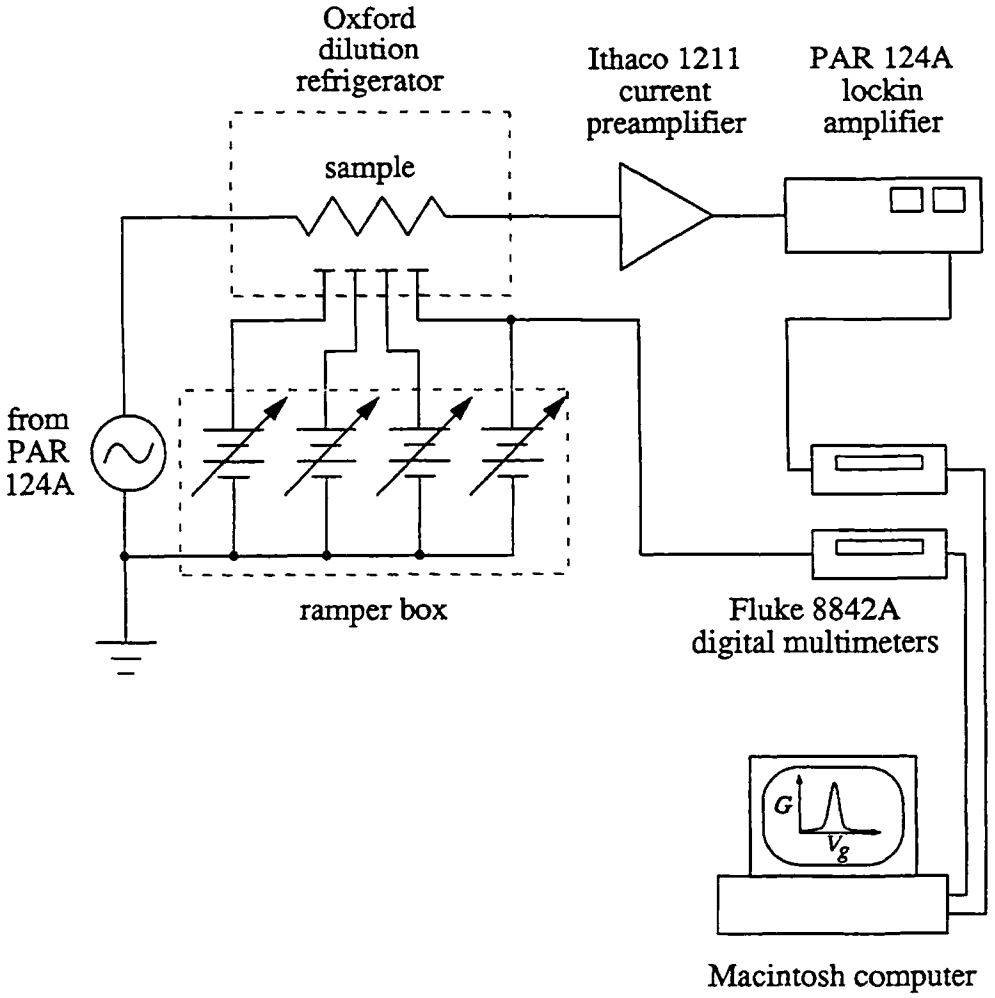


Fig. 5.3 Electronics for single-electron charging experiments. Voltage bias is applied across sample; resulting current is amplified by current preamp and detected by lockin amp. Circuitry built by Jordan Katine supplies constant and ramped gate voltages. Lockin signal and ramped gate voltage are digitized by multimeters and recorded onto Macintosh microcomputer using Labview program written by Doug Mar.

program with graphical interface. The battery-powered voltage source, built by Jordan Katine, consists of one ramped voltage and seven adjustable constant voltages. Its circuit diagrams appear in Fig. 5.4. The Labview program was designed by Doug Mar [Mar, 1994].

This procedure incorporates a number of considerations specific to Coulomb blockade experiments. First, samples are voltage-biased rather than current-biased because current-biasing can lead to voltage drops as large as the Coulomb charging gap U/e, strongly altering conductance peak shape [Foxman *et al.*, 1993]. The bias amplitude is made as small as possible to prevent electron heating [Rimberg, 1992] while still providing a clean signal. An added complication is that large tunnel junction resistances improve the Coulomb blockade peak-to-trough ratio even though they decrease the overall signal amplitude. The reason is that larger tunnel junction resistances R_T reduce co-tunneling, which occurs at a rate proportional to $(R_Q/R_T)^N$ for N tunnel junctions in series, where $R_Q = h/e^2 = 25.8$ kΩ [Averin and Likharev, 1990, 1992; Martinis and Nahum, 1993; Martinis, Nahum, and Jensen, 1994; Dresselhaus *et al.*, 1994]. For a device whose resistance ranges from 100 kΩ to 100 MΩ, it is found empirically that a bias voltage of 5 to 10 μV produces a clean signal with a high peak-to-trough ratio, implying minimal heating and co-tunneling.

A number of simple precautions can help to reduce noise. One is to place the current preamplifier and the battery-powered gate voltage source as close to the fridge electrical feedthroughs as possible. Similarly, the ac bias voltage should be divided to the desired amplitude immediately before entering the fridge. A second precaution is to filter the constant and ramped gate voltages using RC low-pass filters with 0.1 sec time constants. A third precaution is to turn off any multimeters not directly needed for taking data, such as those used to measure constant gate voltages. Of course, all signals are carried on BNC

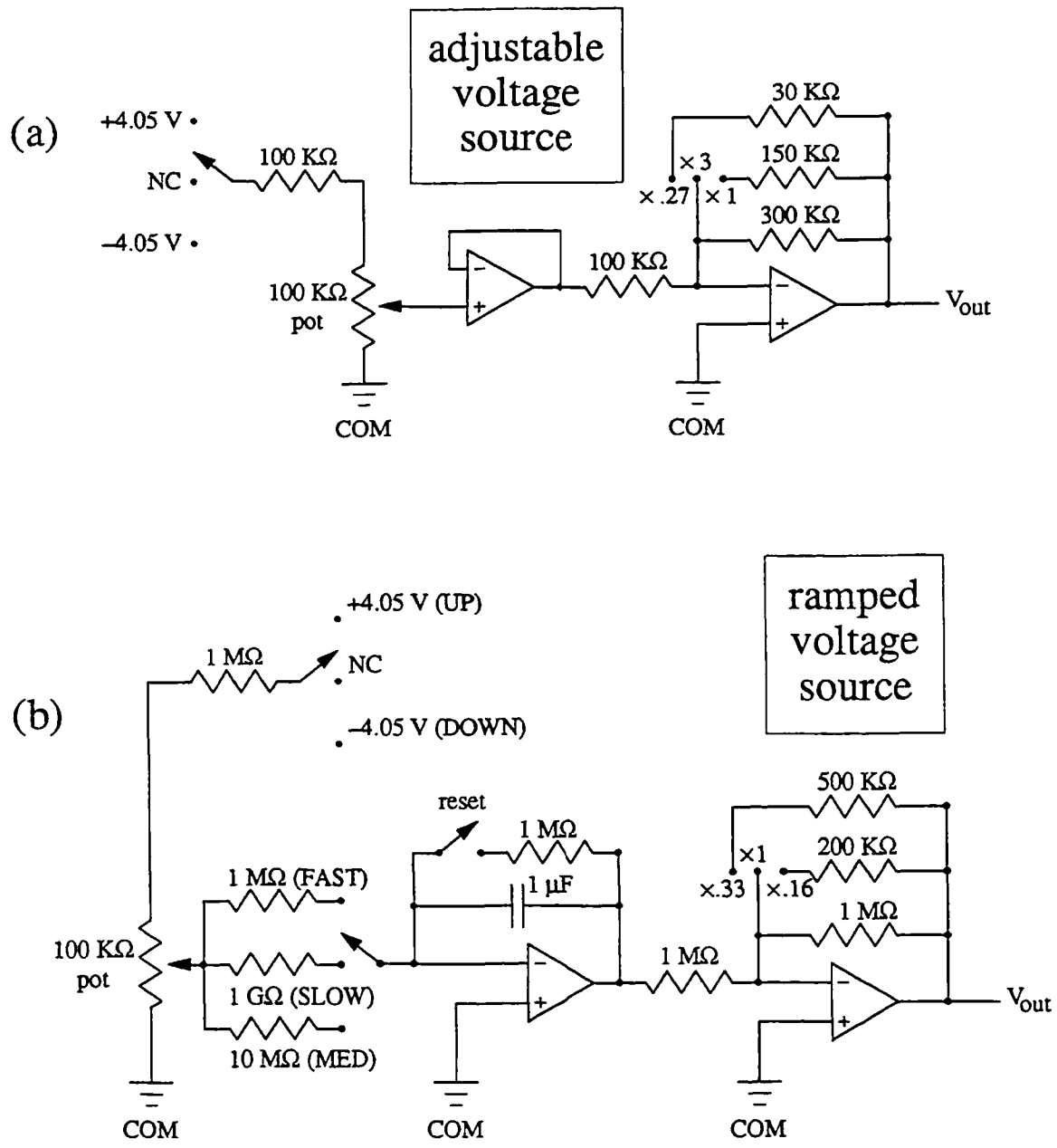


Fig. 5.4 Circuit diagrams for ramper box of Fig. 5.3: (a) adjustable voltage source and (b) ramped voltage source. All voltages are with respect to COM; all OP 97 op amps are powered by ±6 V. Ramper box used in coupled quantum dot experiments, built by Jordan Katine, contains one ramped and seven adjustable voltage sources.

coaxial cables with grounded shields, and all custom-made electrical components are shielded in Pomona boxes.

A drawback of this measurement procedure is that it is rather slow, requiring roughly 20 to 30 minutes to complete a 10 mV gate voltage sweep. Despite all efforts to eliminate noise, it is likely that several noise-induced charging events will disrupt such a long sweep. These events—whose frequency ranges widely from sample to sample and even from day to day—appear as a phase slip in a sequence of otherwise periodic single-electron charging conductance peaks. They are a nuisance in experiments that track how conductance peak locations and amplitudes vary as some parameter (such as inter-dot coupling) changes. For such experiments, long sweeps that record the detailed shape of each conductance peak are unnecessary, and it is therefore desirable to develop data-taking methods that quickly raster gate voltage and inter-dot coupling (or other parameters of interest), picking out only the peak locations and amplitudes.

5.4 OTHER GOOD THINGS TO KNOW

This section gathers together other experimental details and observations that may be helpful to group members who continue coupled dot experiments.

(i) Frequently split-gate devices have "renegade" gates, which show no visible defects yet still do not behave properly. In single-electron charging experiments, the signature of such a gate is unreproducible conductance noise as the gate is ramped. Both samples described in Chapter 6 have this problem: for sample A, gate 5 (the "top" confining gate) is only slightly a renegade, but still quite useful; for sample B, gate 5 (the "bottom" confining gate) is very strongly a renegade. Interestingly, these gates behave properly when held at constant voltage; only when ramped does their renegade nature come forth. Figure 6.5(d) gives a good illustration of this problem.

(ii) A frustrating aspect of single-electron charging experiments using split-gate devices is that nearby gates strongly affect tunnel barrier conductances. Measurements of this effect are important, since they can be used to estimate the barrier conductance for a given gate voltage configuration. Assuming the main result is to shift the point contact characteristic horizontally, the conductance of point contact i is estimated as $G_i(V_i + \Delta V)$, where $G_i(V_i)$ is the characteristic with all gates grounded. The horizontal shift ΔV is given by $\Delta V = \sum_{j \neq i} V_j \left(\partial V_i^{(po)} / \partial V_j \right)$, where V_j is the voltage of a nearby gate j and $\partial V_i^{(po)} / \partial V_j$ is found by recording the pinch-off conductance characteristic for point contact i with gate j (i) grounded and (ii) set to some negative voltage. For the triple-dot device of Chapter 6, this matrix is 4×6 and has 20 non-zero elements.

(iii) Attractive SEM pictures are a vital aspect of any nanoelectronic experiment. One trick is to use a 20 kV accelerating voltage with a 15 mm working distance. This yields better contrast and less detail than higher accelerating voltages, making the device more visible while smoothing over fabrication blemishes. Another trick for improving contrast is to tilt the stage a few degrees (3 to 10 degrees is sufficient). A point not often appreciated is that the optimal beam current changes with magnification. Condenser lens 14 usually works well for high magnification pictures (approximately 10,000× to 30,000×). For lower magnification (approximately 1,000× to 3,000×), however, using condenser lens 14 leads to unattractive haloes around Cr/Au gates. A larger beam current—condenser lens 10 or 11, for example—eliminates haloes.

CHAPTER 6*

SINGLE-ELECTRON CHARGING IN DOUBLE AND TRIPLE QUANTUM DOTS

6.1 INTRODUCTION

Quantum dots are often described as "artificial atoms" [for reviews, see Beenakker and van Houten, 1991; Kastner, 1992; van Houten, Beenakker, and Staring, 1992] and as "single-electron transistors" [Averin and Likharev, 1991, 1992; Tucker, 1992]. Both labels imply their use as building blocks in larger structures. In the first case, quantum dot arrays may form custom-engineered artificial molecules or artificial crystals [Kouwenhoven *et al.*, 1990; Haug, Hong, and Lee, 1992; Sakamoto *et al.*, 1994]. These differ from their real counterparts in that leads may be attached and conductance spectroscopy performed as gates tune their size, shape, electron density, or coupling. In the second case, quantum dot arrays may be fashioned into logic circuits in which individual electrons represent bits of information. Experiments with sub-micron metal-island electrometers and current sources already approach the accuracy needed for metrology and possibly also for computing applications [Martinis, Nahum, and Jensen, 1994; Dresselhaus *et al.*, 1994].

Crucial to both uses of quantum dots is an understanding of how coupled dots interact. Arrays of semiconductor quantum dots and metal islands often do not allow control over coupling. For metal islands and vertical semiconductor dots, couplings are fixed in the fabrication process. For lateral semiconductor quantum dots, couplings can in principle be

*Versions of this chapter have been submitted to *Phys. Rev. Lett.* and *Phys. Rev. B*.

individually tuned with gates, but many device architectures forego this flexibility [Kouwenhoven *et al.*, 1990; Haug, Hong, and Lee, 1992]. The many phenomena predicted for quantum dot arrays in the tunneling regime—including conductance peak splitting, peak suppression, single-electron solitons, and quasiperiodicity—can be difficult to observe experimentally without control over inter-dot coupling [Bakhvalov *et al.*, 1989; Castaño, Kirczenow, and Ulloa, 1990; Ruzin *et al.*, 1992; Bryant, 1993; Stafford and Das Sarma, 1994; Klimeck, Chen, and Datta, 1994].

This chapter describes experiments that explicitly probe inter-dot coupling in arrays of two and three quantum dots. The arrays are defined in a two-dimensional electron gas by tuneable gates that permit separate control of the tunnel barriers and confining walls. Tuneability provides two important advantages over previously studied devices: it allows compensation for disorder due to the impurity potential and to lithographic imperfections, and it enables new experiments in which the conductance is measured as inter-dot coupling varies. In terms of the "artificial atom" and "single-electron transistor" pictures, the arrays can be described as two-atom and three-atom artificial molecules with tuneable inter-atom coupling, or as single-electron memory devices [Averin and Likharev, 1991, 1992; Bock and Hartnagel, 1993; Yano *et al.*, 1993; Dresselhaus *et al.*, 1994].

The experiments consist of low-temperature tunneling conductance measurements in zero magnetic field. As the quantum point contacts between dots are opened, the inter-dot coupling increases in a continuous transition from isolated dots to one large dot. Isolated dot arrays show strong Coulomb blockade conductance peaks vs. gate voltage which split into two (double dot) and three (triple dot) peaks as the coupling increases. The splitting is proportional to the measured barrier conductance and experimentally determines the dot interaction energy. For dot arrays with unequal gate capacitance, the conductance peaks exhibit beating and quasiperiodicity.

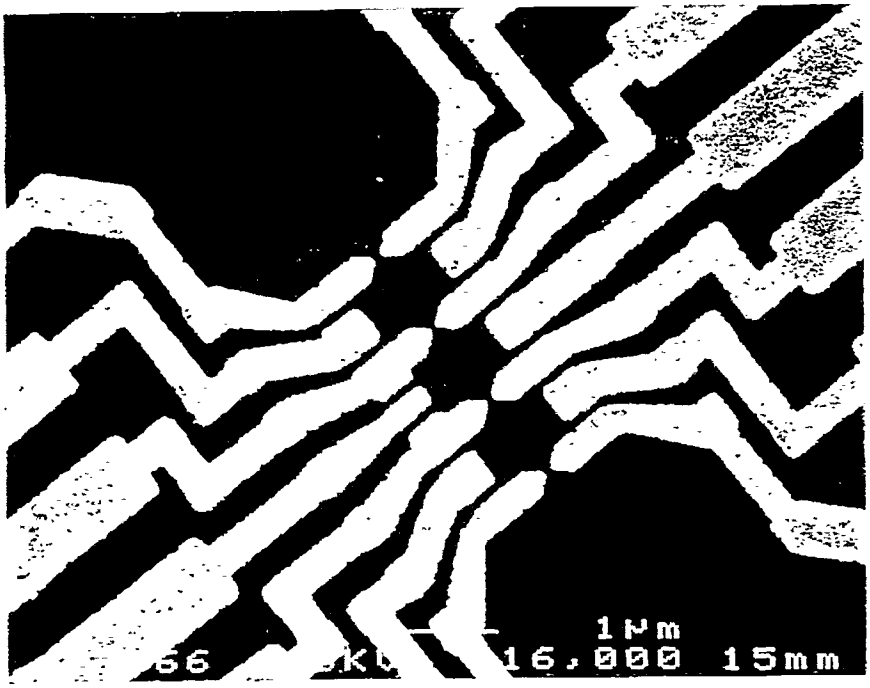
The rest of this chapter is organized as follows. Section 6.2 describes coupled quantum dot devices, and Sec. 6.3 presents data on the quantum point contacts and individual quantum dots that form the arrays. Double-dot and triple-dot experimental data are presented in Secs. 6.4 and 6.5, respectively. A capacitive charging model for coupled dots is introduced in Sec. 6.6, and computer simulations of the model are shown to be in qualitative agreement with the experimental data.

6.2 DEVICES

Two devices were studied: a triple dot (device A), used also for double-dot experiments by not energizing all gates; and a similar double dot (device B). Scanning electron micrographs of the two devices are shown in Figs. 6.1 and 6.2 respectively. Both devices were fabricated using the same GaAs/AlGaAs heterostructure, which contains a two-dimensional electron gas located 470 Å beneath the surface with sheet density $n_s = 3.7 \times 10^{11}$ cm^{-2} and mobility $\mu = 5 \times 10^5$ cm^2/Vs at 10 K (see Sec. 5.2.1).

Device A consists of 14 Schottky gates fabricated on the heterostructure surface with electron-beam lithography and chrome/gold evaporation (see Sec. 5.2.2). As shown in Fig. 6.1(a), eight gates form the four quantum point contacts used as tunnel barriers and six gates form the dot confining walls, when sufficient negative voltage is applied to deplete the electron gas underneath. The lithographic area of each dot is $A_{dot} = 0.5 \times 0.8\ \mu\text{m}^2$, implying approximately $n_s A_{dot} = 1500$ electrons per dot. The actual number is somewhat less, because the depletion region extends out from under each gate and because the electron density is reduced beyond the depletion region. As depicted in Fig. 6.1(b), the device is wired with six independently tuneable gate voltages: one for each tunnel barrier (V_1 through V_4) and one each for each set of confining walls at the top

(a)



(b)

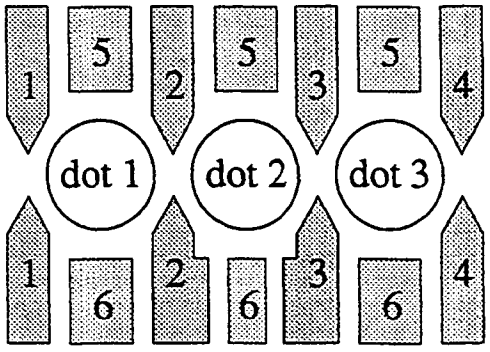
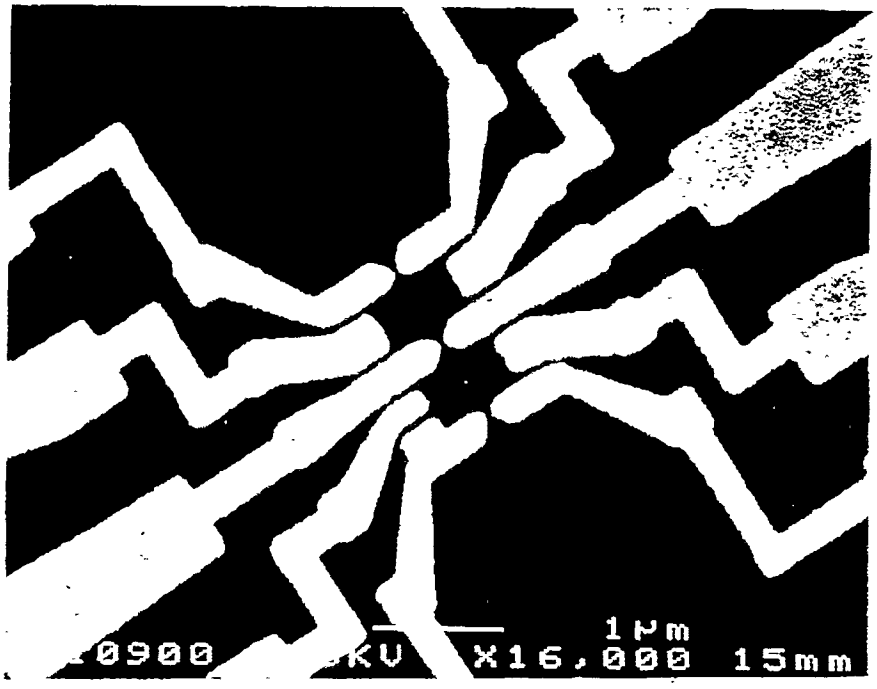


Fig. 6.1 (a) SEM micrograph at 16,000× magnification of three coupled quantum dots (device A) with tuneable tunneling barriers in GaAs/AlGaAs heterostructure. Scale bar is 1 μm; dots are $0.5 \times 0.8\ \mu m^2$. (b) Dot and gate labeling for device A.

(a)



(b)

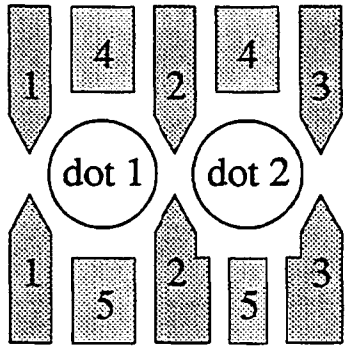


Fig. 6.2 (a) SEM micrograph at 16,000× magnification of two coupled quantum dots (device B) with tuneable tunneling barriers in GaAs/AlGaAs heterostructure. Scale bar is 1 μm; dots are $0.5 \times 0.8\ \mu m^2$. (b) Dot and gate labeling for device B.

(V_5) and bottom (V_6) of the array. Note that the confining wall on dot 2 and gate 6 is intentionally made smaller than the others. Double dot experiments were conducted using device A by not energizing V_1. Device B is similar to device A but consists of 10 Schottky gates forming two coupled quantum dots, also with area $A_{dot} = 0.5 \times 0.8\ \mu m^2$.

The dot geometry determines two important energy scales, the charging energy U and the single-electron level spacing ΔE. The charging energy can be estimated from the total dot capacitance C_Σ. Treating a dot as a metal disk embedded in a dielectric [Glattli *et al.*, 1991] gives the estimate

$$U = \frac{e^2}{C_\Sigma} \cong \frac{e^2}{\pi\varepsilon\varepsilon_0 L} = \frac{35\ \mu eV}{L\,[\mu m]}, \tag{6.1}$$

where $\varepsilon = 13$ is the GaAs dielectric constant, $\varepsilon_0 = 1.1 \times 10^{-12}$ F/cm, and L is a characteristic dot length. For the dots in Figs. 6.1 and 6.2, Eq. (6.1) gives $C_\Sigma \cong 2$ fF and a charging energy $U \cong 70\ \mu eV$. Assuming equally spaced levels, ΔE can be estimated from the dot area:

$$\Delta E \cong \frac{2E_F}{n_s A_{dot}} = \frac{2\pi\hbar^2}{m^* A_{dot}} = \frac{6.7}{A_{dot}\left[\mu m^2\right]}\ \mu eV. \tag{6.2}$$

The factor of 2 in Eq. (6.2) accounts for spin degeneracy, and $m^* = 0.067m$ is the electron effective mass in GaAs. With $A_{dot} = 0.4\ \mu m^2$, Eq. (6.2) gives $\Delta E = 20\ \mu eV$.

The samples were cooled in a He dilution refrigerator at the base temperature $T = 14$ mK; care was taken to shield them from external electromagnetic radiation (see Sec. 5.3.1). The tunneling conductance of dot arrays was measured by applying a small

(typically 5 to 10 μV) ac voltage at 11 Hz and recording the current with a current preamplifier and lockin amplifier (see Sec. 5.3.2).

6.3 QUANTUM POINT CONTACTS AND SINGLE QUANTUM DOTS

An important advantage of lateral quantum dot devices like those of Figs. 6.1 and 6.2 is that each component is controlled by separate gates and can be individually tested and adjusted. This tuneability permits compensation for disorder, particularly important for the tunnel barriers, and distinguishes these devices from previously studied chains of semiconductor dots [Kouwenhoven *et al.*, 1990; Haug, Hong, and Lee, 1992] and metal islands [Martinis, Nahum, and Jensen, 1994; Dresselhaus *et al.*, 1994].

For device A, the four nominally identical point contacts, separately measured, each show high quality characteristics G_{qpc} vs. gate voltage with up to 10 quantized conductance plateau, as shown in Fig. 6.3(a). However, their pinchoff voltages range from –0.92 V to –1.02 V, demonstrating the need for independent tuneability. The three point contacts of device B also show high quality conductance curves, plotted in Fig. 6.3(b). While their characteristics are more uniform than those of device A, there is still a considerable range of pinchoff voltages, from –0.77 V to –0.82 V. For both devices, the point contact nonuniformity arises predominantly from the disordered impurity potential [Haanappel and van der Marel, 1989; Nixon, Davies and Baranger, 1991; Laughton *et al.*, 1991; Timp, 1992].

When separately energized, each dot of devices A and B shows regularly spaced conductance peaks corresponding to adding a single electron as the confining wall gate voltages are swept [Kastner, 1992; van Houten, Beenakker, and Staring, 1992]. Figures 6.4 and 6.5 show the single-dot conductance G_{sd} vs. gate voltage for devices A and B.

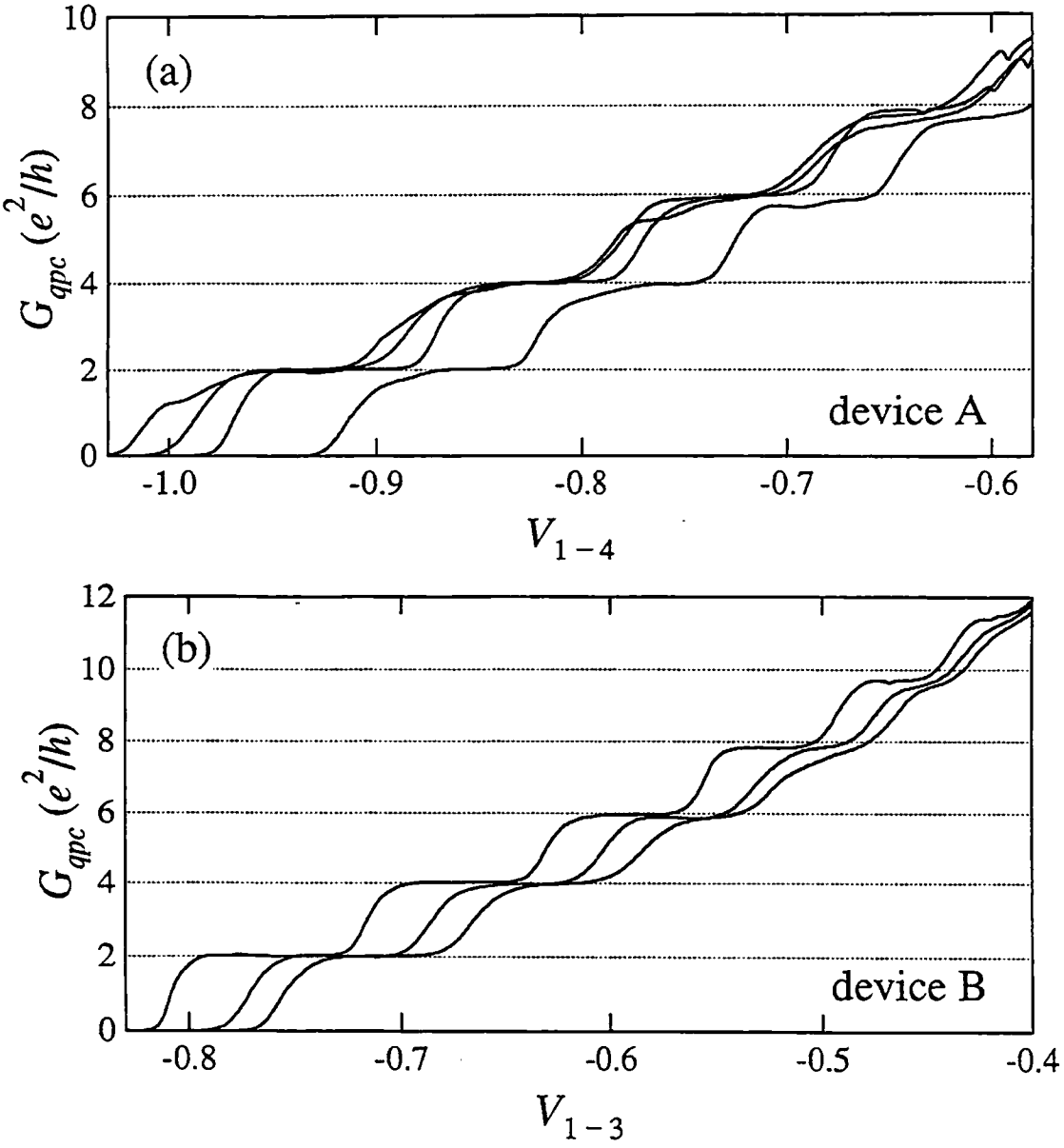


Fig. 6.3 (a) Four quantum point contact characteristics G_{qpc} vs. gate voltage, with all other gates grounded, for device A. (b) Three quantum point contact characteristics G_{qpc} vs. gate voltage for device B.

The three panels of Fig. 6.4 correspond to dots 1, 2, and 3, respectively, of device A as gate voltage V_5 is swept; the four panels of Fig. 6.5 correspond to dots 1 and 2 of device B as gate voltage V_4 is swept (Fig. 6.5(a) and (b)) and as gate voltage V_5 is swept (Fig. 6.5(c) and (d)). As discussed in Sec. 4.4, both the gate capacitance C_{ia} and the level spacing ΔE determine the peak spacing for dot i when gate a is swept (see Eqs. (4.12) and (4.13)). Because the value $\Delta E = 20$ μeV from Eq. (6.2) is considerably smaller than the observed peak separation of about 4 mV in Figs. 6.4 and 6.5, the level spacing can be neglected and the peak spacing experimentally determines the gate capacitances C_{ia}. For device A, the gate capacitances are $C_{15} = 38$ aF, $C_{25} = 43$ aF, $C_{35} = 41$ aF, $C_{16} = 41$ aF, $C_{26} = 32$ aF, and $C_{36} = 39$ aF. For device B, the gate capacitances are $C_{14} = 43$ aF, $C_{24} = 42$ aF, $C_{15} = 44$ aF, and $C_{25} = 25$ aF.

While gate capacitance determines the separation between peaks, a dot's total capacitance C_Σ determines the peak width [Glazman and Shekhter, 1989; Beenakker, 1991; van Houten, Beenakker, and Staring, 1992]. As discussed in Sec. 4.4, the peak shape depends on the relative sizes of the electron temperature T, the average level spacing ΔE, and the Coulomb charging energy e^2/C_Σ; the experimental values are 5 μeV, 20 μeV, and 70 μeV, respectively. Figure 6.6 shows fits to the thermally broadened shape, Eq. (4.14), and the lifetime broadened shape, Eq. (4.16), for conductance peaks of dot 3 on device A. The peak widths are the only fitting parameters. Peaks are best fitted by the thermally broadened shape. Assuming both broadening mechanisms are present, the fits place upper bounds on the electron temperature and lifetime. The peak width $w_T = e\gamma_T/C_3 = 170$ μV found from fitting Eq. (4.14), together with the measured gate capacitance $C_3 = 41$ aF, relates the electron temperature and the total capacitance:

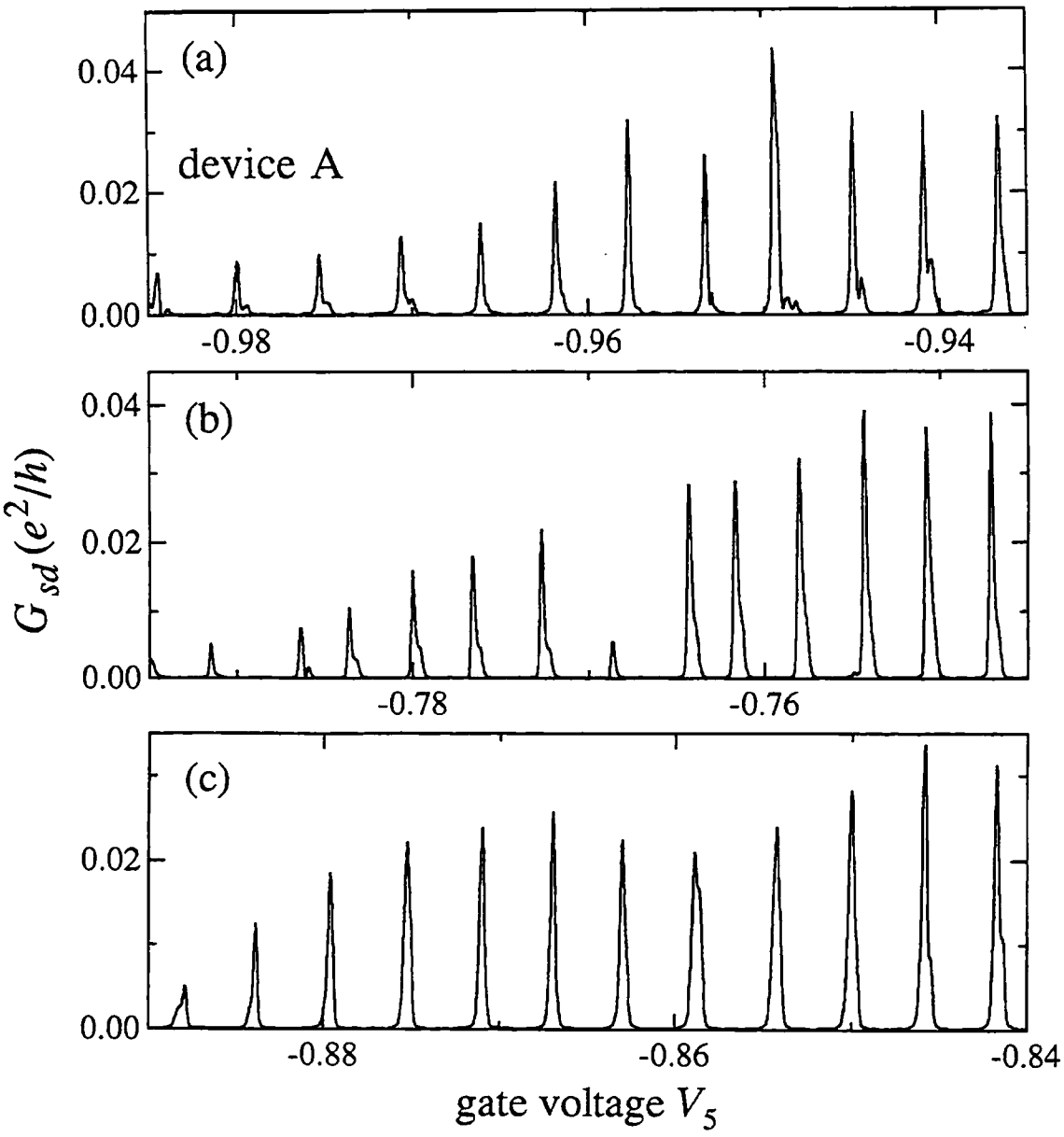


Fig. 6.4 Single-dot conductance G_{sd} vs. gate voltage V_5 of device A: (a) dot 1; (b) dot 2; (c) dot 3. Peak separation determines gate capacitances $C_{15} = 38$ aF, $C_{25} = 43$ aF, and $C_{35} = 41$ aF. Similar sweeps of gate voltage V_6 determine $C_{16} = 41$ aF, $C_{26} = 32$ aF, and $C_{36} = 39$ aF.

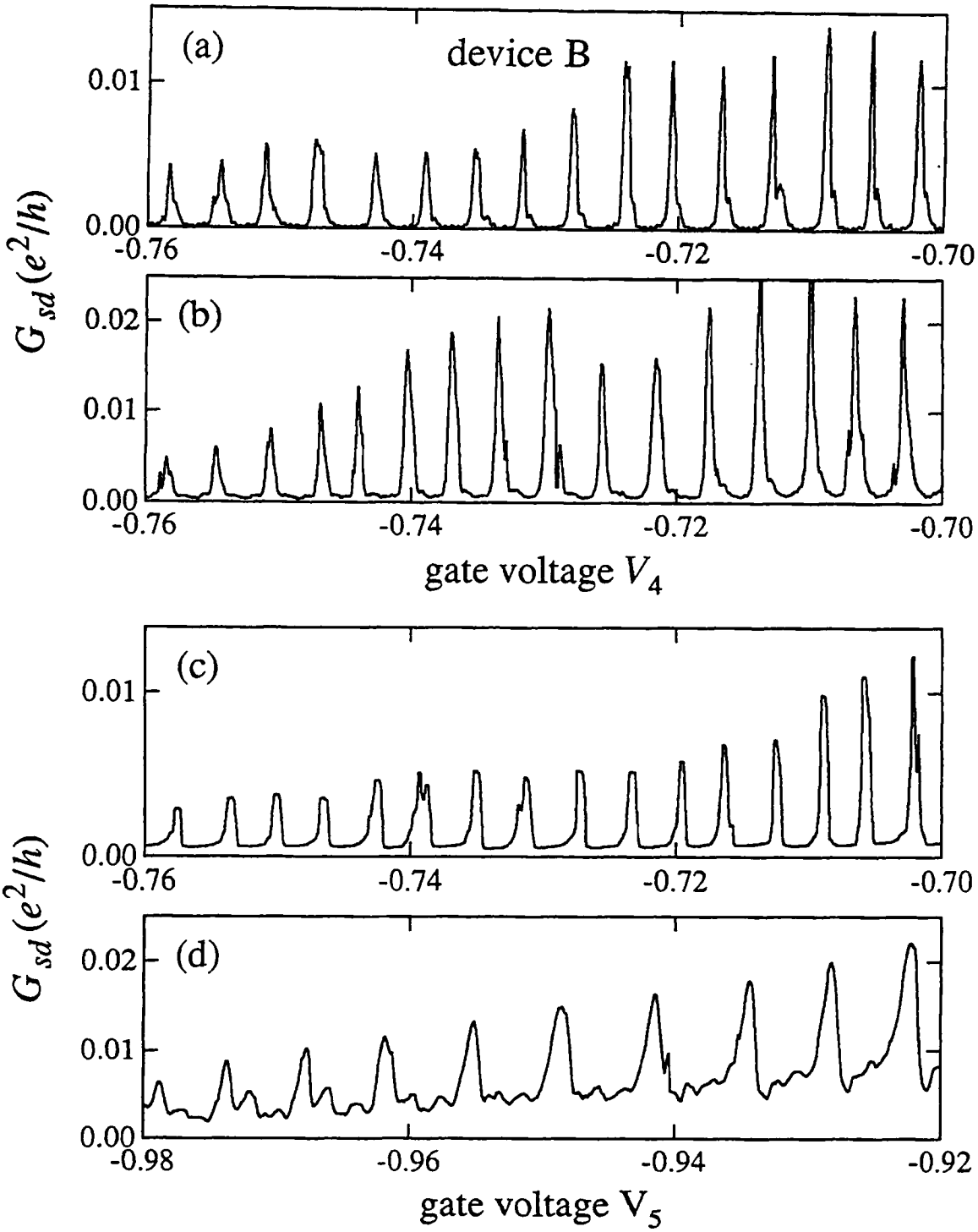


Fig. 6.5 Single-dot conductance G_{sd} vs. gate voltages V_4 and V_5 of device A: (a) dot 1 with gate 4 swept; (b) dot 2 with gate 4 swept; (c) dot 1 with gate 5 swept; (d) dot 2 with gate 5 swept. Peak separation determines gate capacitances C_{14} = 43 aF, C_{24} = 42 aF, C_{15} = 44 aF, and C_{25} = 25 aF.

$$k_B T < \frac{e w_T C_3}{C_\Sigma} = 0.04 \frac{e^2}{C_\Sigma}. \tag{6.3}$$

Using the value C_Σ = 2 fF from Eq. (6.1) yields an electron temperature T < 30 mK. Similarly, the peak width $w_\Gamma = e\gamma_\Gamma/C_3$ = 140 μV found from fitting the lifetime broadened shape, Eq. (4.16), relates the electron lifetime $1/\Gamma$ and the total capacitance:

$$\hbar\Gamma < \frac{2e w_\Gamma C_3}{C_\Sigma} = 0.07 \frac{e^2}{C_\Sigma}. \tag{6.4}$$

With the estimate C_Σ = 2 fF, Eq. (6.4) gives $\hbar\Gamma$ < 11 μeV, or $\Gamma < 1.5 \times 10^9$ sec^{-1}.

Larger dots form when the conductance of the point contacts between two or three of the single dots is made greater than $2e^2/h$. Figure 6.7 shows that the gate capacitance and total capacitance of these larger dots both scale approximately linearly with the number of merged dots. Figures 6.7(a) through (c) compare the conductance of device A vs. gate voltage V_5 for a single dot, a larger dot formed by merging two dots, and a still larger dot formed by merging three dots. The gate capacitances determined from peak separation are (a) 38 aF, (b) 73 aF, and (c) 110 aF. As shown in Fig. 6.7(d), the peak shape remains predominantly thermally broadened for the large dot of (c): the peak width w_T changes much less than the gate capacitance, increasing from w_T = 170 μV for one dot to w_T = 270 μV for three merged dots, implying that the total capacitance scales roughly linearly with the number of merged dots.

Visible on a number of the conductance peaks in Figs. 6.4 and 6.5 (and on double-dot and triple-dot conductance peaks shown below) are smaller shoulders and side peaks. Similar peak asymmetries have been observed by other researchers [Foxman *et al.*, 1993]. Separation $\Delta V \cong$ 500 μV between main peaks and side peaks is observed for dots of both

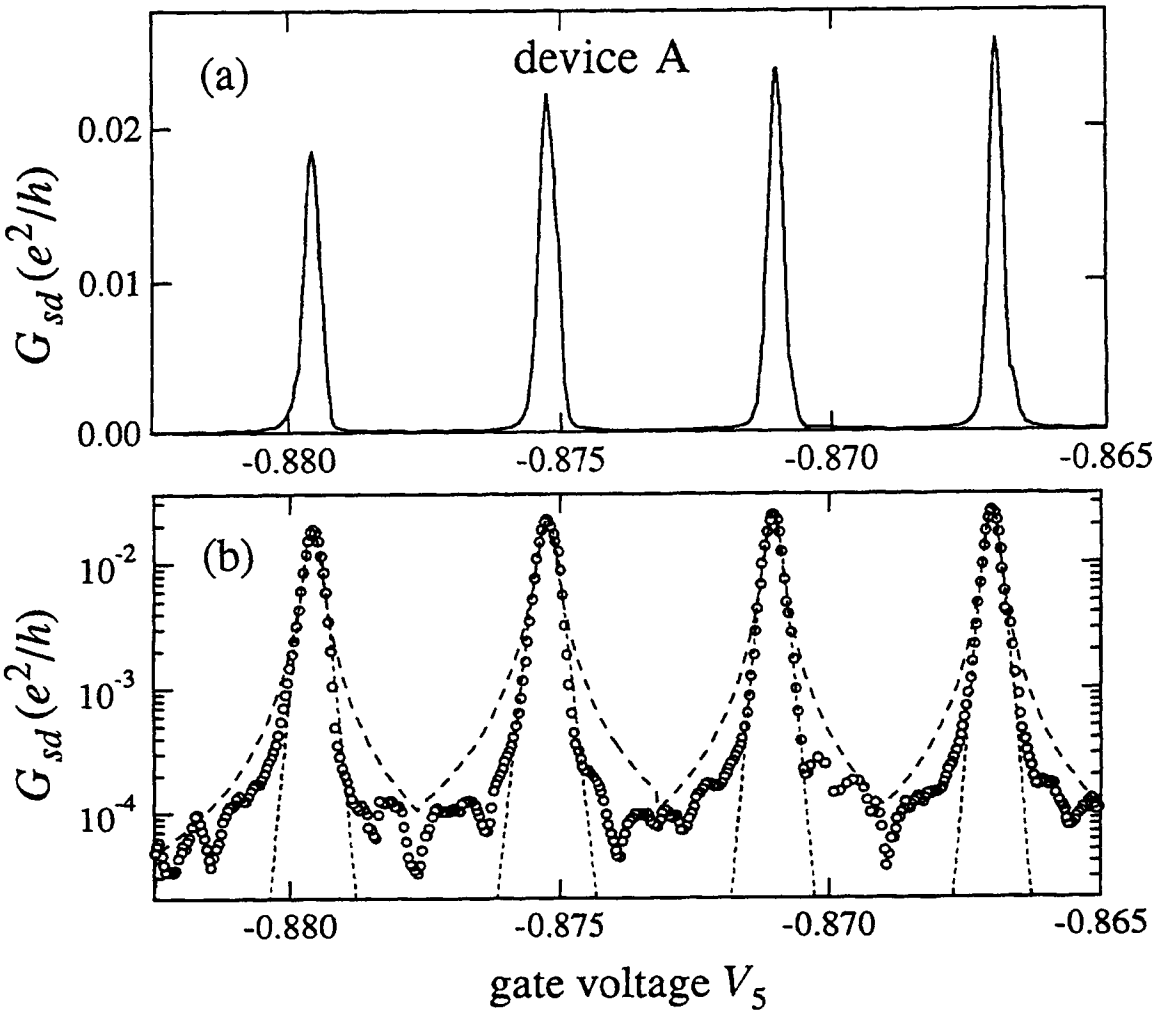


Fig. 6.6 (a) Conductance peaks of dot 1 on device A vs. gate voltage V_5. (b) Same peaks on log scale (open circles), together with fits to thermally broadened peak shape of Eq. (4.14) (short dashes) and lifetime broadened peak shape of Eq. (4.16) (long dashes). Broadening is predominantly thermal.

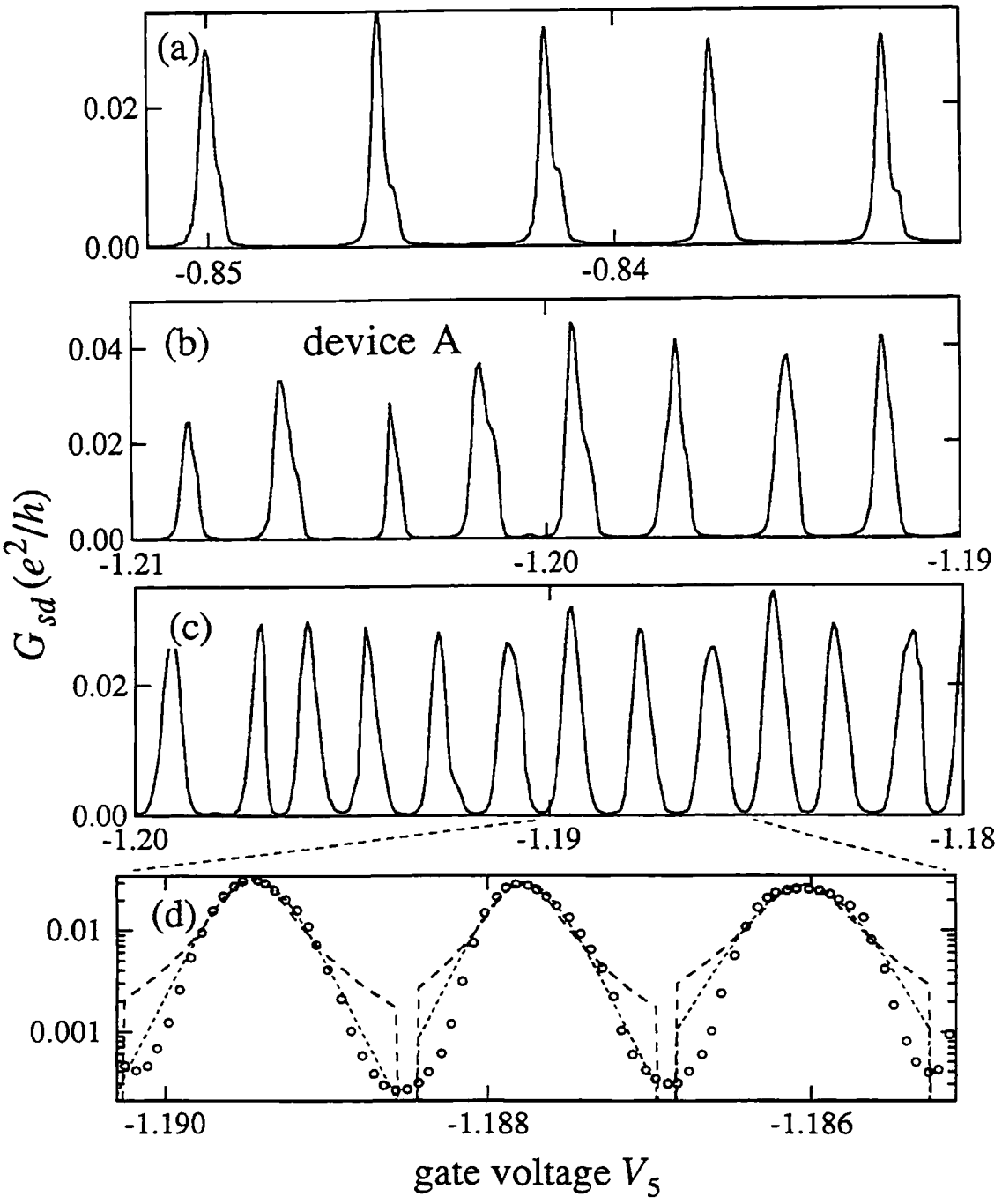


Fig. 6.7 Single-dot conductance G_{sd} vs. gate voltage V_5 on device A for (a) dot 3, (b) larger dot formed by merging dots 2 and 3, and (c) still larger dot formed by merging dots 1, 2, and 3. Gate capacitances are (a) 38 aF, (b) 73 aF, and (c) 110 aF. (d) Fits to thermally broadened peak shape of Eq. (4.14) (short dashes) and lifetime broadened peak shape of Eq. (4.16) (long dashes) for several peaks from (c).

devices A and B and implies an energy of $e\Delta V\left(C_g/C_\Sigma\right) \cong 10$ μeV. Possible explanations of these structures include the Kondo effect [Wan, Phillips, and Li, 1994] and photon-assisted tunneling [Kouwenhoven *et al.*, 1994] due to ambient radiation (see Sec. 5.3). For the Kondo effect, off-peak resonances are associated with the Coulomb exchange energy between two electron levels, while for photon-assisted tunneling, they are associated with the photon energy. Further measurements are needed to determine the origin of these structures.

An issue raised by this section is the nature of the transition from two or three noninteracting dots in series, each having conductance like Figs. 6.4 or 6.5, to a single large dot with conductance like Fig. 6.7(b) or (c). Is the transition gradual or sharp? How do the extra peaks emerge? What is the signature of dot interaction on the conductance? Theoretical answers to these questions, based on capacitive charging and Hubbard models, appear in Chapter 4. In the next two sections, these questions are addressed experimentally for double and triple dots.

6.4 DOUBLE QUANTUM DOTS

Double quantum dots are the simplest system for studying coupling between dots. This section describes experiments on double dots that show how inter-dot coupling leads to conductance peak splitting. The splitting determines the total dot interaction energy Δ as well as the relationship between Δ and the conductance G_b of the inter-dot tunnel barrier. Attention is focused on the role of unequal gate capacitances, because—as shown in the previous section—lithographically identical dots often have gate capacitances varying by as much as 10%.

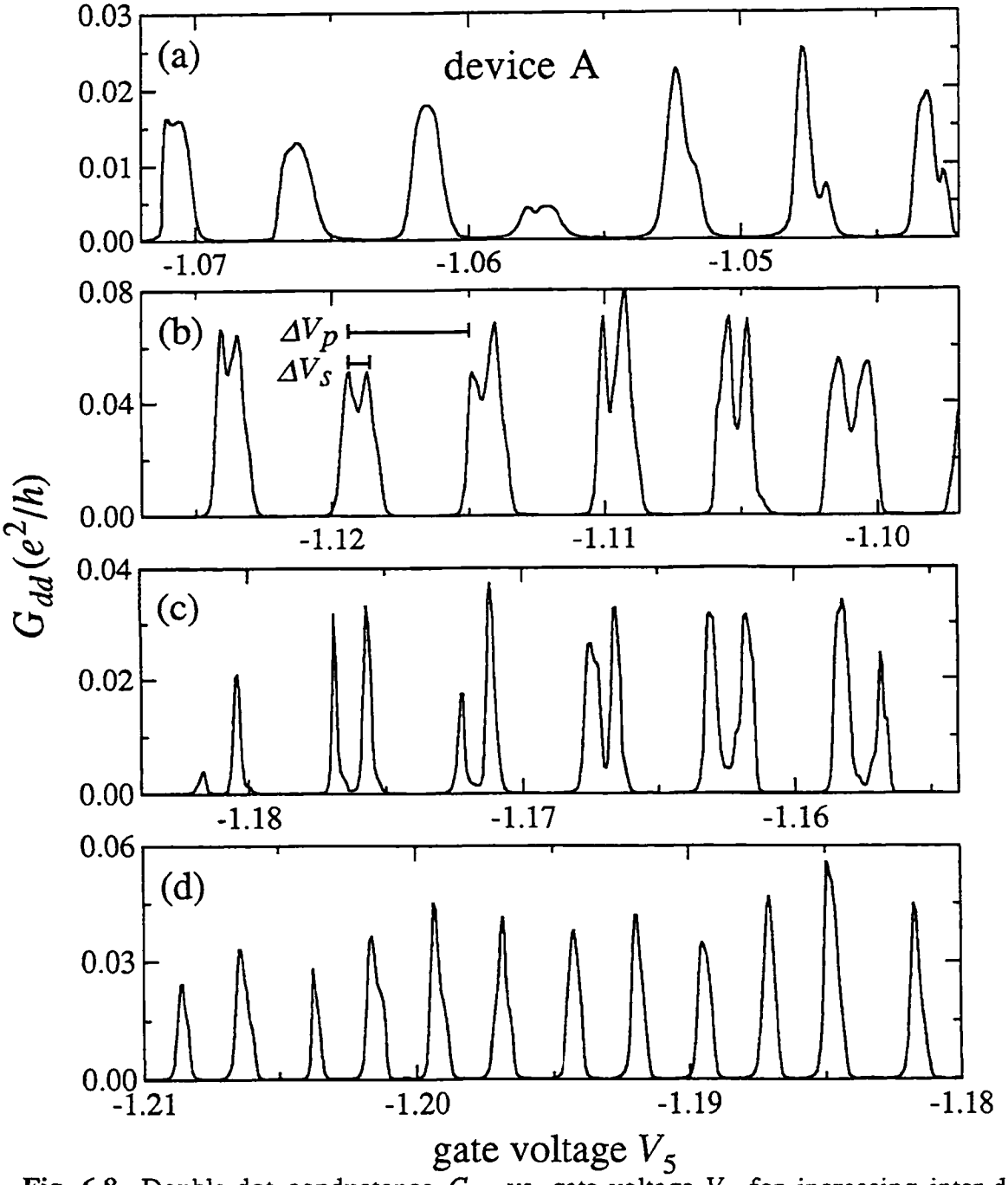


Fig. 6.8 Double-dot conductance G_{dd} vs. gate voltage V_5 for increasing inter-dot coupling; double dot formed from dots 2 and 3 of device A. Coupling splits conductance peaks, with split peak separation ΔV_S proportional to interaction energy Δ. Gate voltage V_3 controlling inter-dot tunnel barrier is (a) –0.912 V, (b) –0.897 V, (c) –0.891 V, and (d) –0.860 V. Barrier conductance in units e^2/h is (a) 0.03, (b) 0.88, (c) 1.37, and (d) 1.94; estimated by shifting measured point contact conductance (see Fig. 6.3) by 72 mV to account for influence of nearby gates.

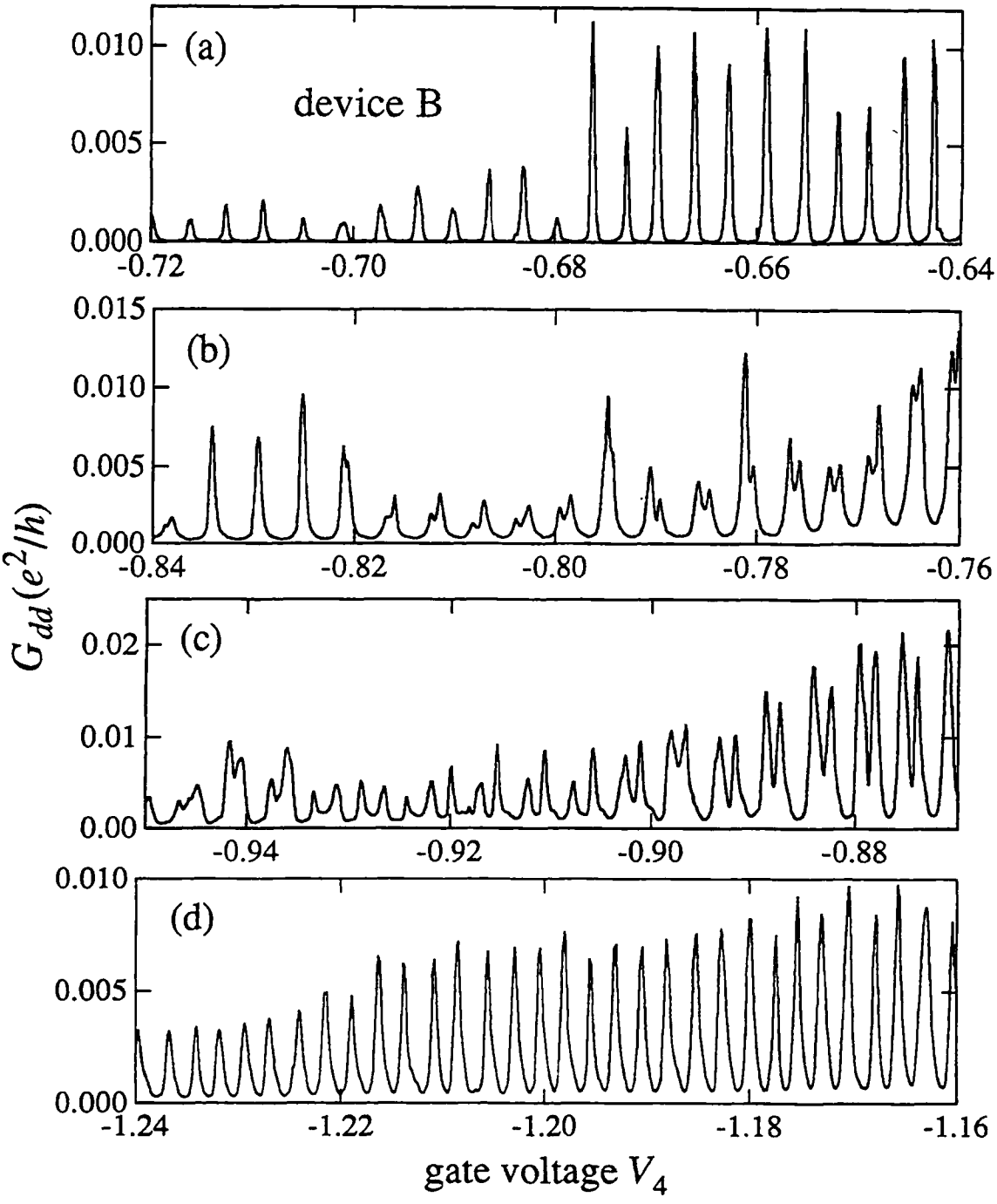


Fig. 6.9 For device B, double-dot conductance G_{dd} vs. gate voltage V_4 for increasing inter-dot coupling. Gate voltage V_2 controlling inter-dot tunnel barrier is (a) –0.779 V, (b) –0 770 V, (c) –0 760 V, and (d) –0.700 V.

Figures 6.8(a) through (d) show changes in the conductance G_{dd} of a double dot vs. gate voltage with increasing inter-dot coupling. The double dot is formed in device A by grounding gate 1 and energizing all other gates; gate voltage V_5 is swept, and gate voltage V_3 controls the inter-dot coupling. In Fig. 6.8(a), the inter-dot coupling is weak. The double-dot conductance consists of weakly split peaks, with the same average separation ΔV_p in gate voltage that is observed for dots 2 and 3 individually (see Fig. 6.4). Each conductance peak in Fig. 6.8(a) corresponds to adding two electrons to the double dot, one to each dot. In Figs. 6.8(b) and (c), each peak clearly splits into two peaks whose separation increases with inter-dot coupling. Finally in Fig. 6.8(d), the tunnel barrier between dots 2 and 3 is removed, and the conductance is that of a single large dot with peak separation about half that in Fig. 6.8(a). The gate voltage V_3 controlling tunnel barrier 3 is (a) −0.912 V, (b) −0.897 V, (c) −0.891 V, and (d) −0.860 V. Similar peak splitting with increased inter-dot coupling is seen in Fig. 6.9 for device B; for this figure the gate voltage V_2 controlling tunnel barrier 2 is (a) −0.779 V, (b) −0.770 V, (c) −0.760 V, and (d) −0.700 V. While uncontrolled peak splitting attributed to disorder has been observed in dots and metal islands [Chandrasekhar and Webb, 1991; Staring, van Houten, and Beenakker, 1992; Hwang, Tsui, and Shayegan, 1994], peak splitting controlled via tuneable gates has not previously been reported.

Figures 6.10(a) and (b) demonstrate the strong correlation between inter-dot coupling and peak splitting. Figure 6.10(a) plots the fractional peak splitting $F = 2\Delta V_s/\Delta V_p$ (see Fig. 6.8(b)) vs. gate voltage V_3 for device A. Each triangle represents the average of F in a single sweep of gate voltage V_5. The separately measured tunnel barrier conductance G_b (see Fig. 6.3(a)) is also plotted in Fig. 6.10(a), offset by 72 mV to account for the influence of other gates. The fractional peak splitting and tunnel conductance track each other closely. This correlation is shown in another way in Fig. 6.10(b), which plots the

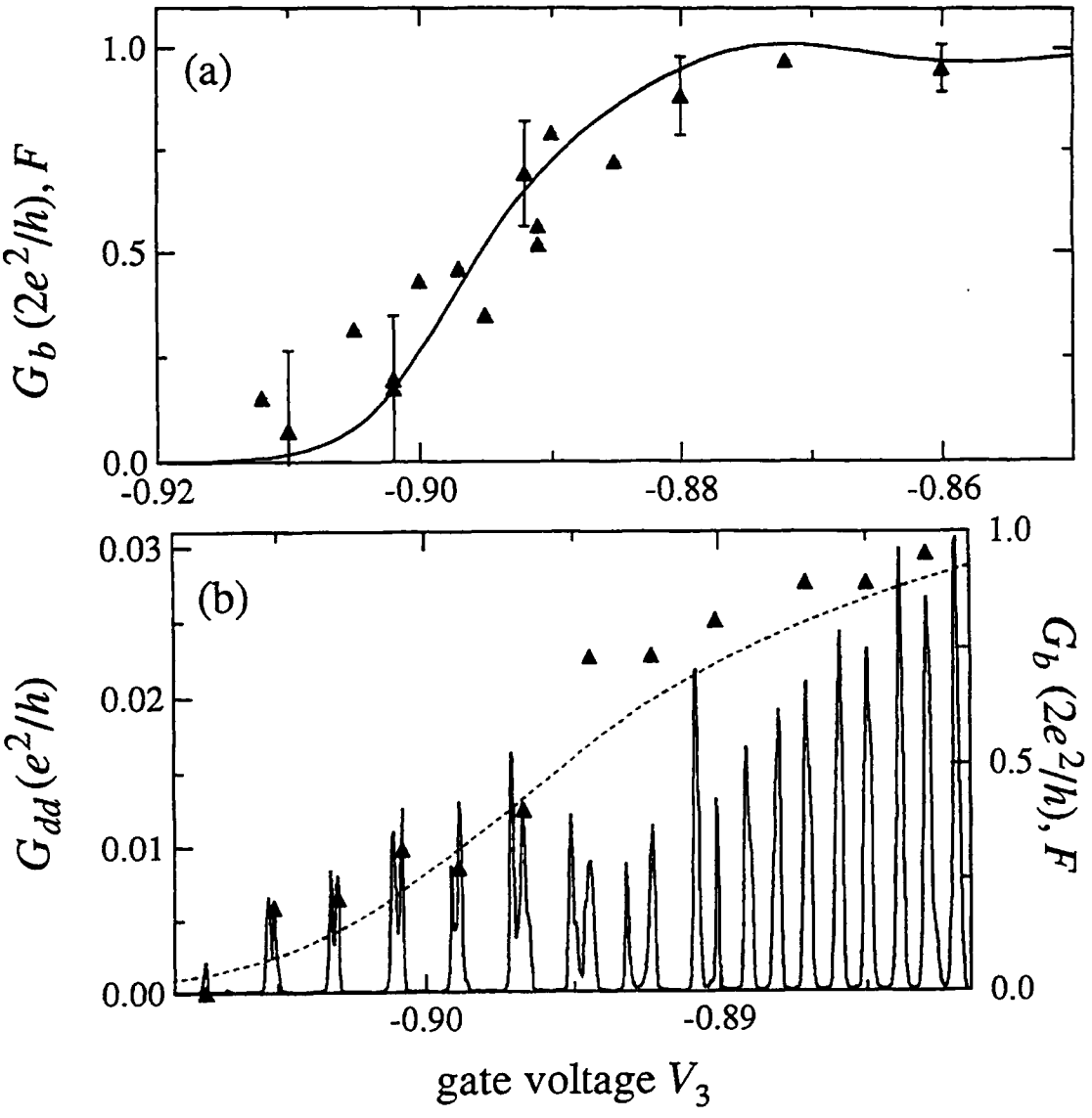


Fig. 6.10 (a) For device A, double-dot fractional splitting $F = 2\Delta V_s/\Delta V_p$ (triangles), averaged over 16 sweeps of V_5, and measured inter-dot barrier conductance G_b (curve) vs. gate voltage V_3. ΔV_s and ΔV_p are defined in Fig. 6.8(b). (b) Double-dot conductance G_{dd} (solid curve, left scale), F (triangles, right scale), and G_b (dotted curve, right scale) vs. gate voltage V_3.

double-dot conductance G_{dd} vs. barrier voltage V_3 for fixed gate voltages V_5 and V_6 together with measured tunnel conductance G_b; the fractional peak splittings are shown as triangles.

In Hubbard model calculations [Kulik and Shekhter, 1975; Stafford and Das Sarma, 1994; Klimeck, Chen, and Datta, 1994], the tunneling rate controls the inter-dot coupling energy Δ, and both tunneling and charging contribute to Δ when tunneling is not weak (see Sec. 4.5). As shown in Fig. 6.10(a), the peak splitting saturates when the inter-dot conductance reaches $2e^2/h$. For this value the two dots merge, and the number of electrons on dot 2 or dot 3 separately is no longer well defined. The measured value $C_g \cong 40$ aF and the estimate $C \cong 2$ fF from Eq. 6.1 imply that Δ ranges from 0 to 25 μeV (see Sec. 4.5).

Figure 6.11 illustrates the conductance G_{dd} of the double dot of device A using intentionally mismatched gate capacitances C_{26} and C_{36}, both controlled by V_6. The four panels plot G_{dd} vs. gate voltage V_6; the gate voltage V_3 controlling tunnel barrier 3 is (a) −0.901 V, (b) −0.895 V, (c) −0.890 V, and (d) −0.880 V. As in Figs. 6.8 and 6.9, the conductance peaks show splitting that increases with inter-dot coupling. Additional phenomena arising from unequal gate capacitance are also evident. Figure 6.11(a) demonstrates the phenomenon of "stochastic Coulomb blockade" predicted by theory [Ruzin *et al.*, 1992] in which conductance peaks through double dots with different gate capacitance become increasingly sparse at low temperatures. The stochastic Coulomb blockade is lifted with increasing inter-dot coupling as shown in Figs. 6.11(b) through (d).

Quasiperiodic modulation of the peak height and spacing due to capacitance mismatch is also apparent in Fig. 6.11. This is in contrast to the case of nearly identical gate capacitances, Fig. 6.9, for which the peak splitting is uniform. The measured beat period of 23 mV for Fig. 6.11(c) equals the period $e/|C_{36} - C_{26}| = 23$ mV calculated with the

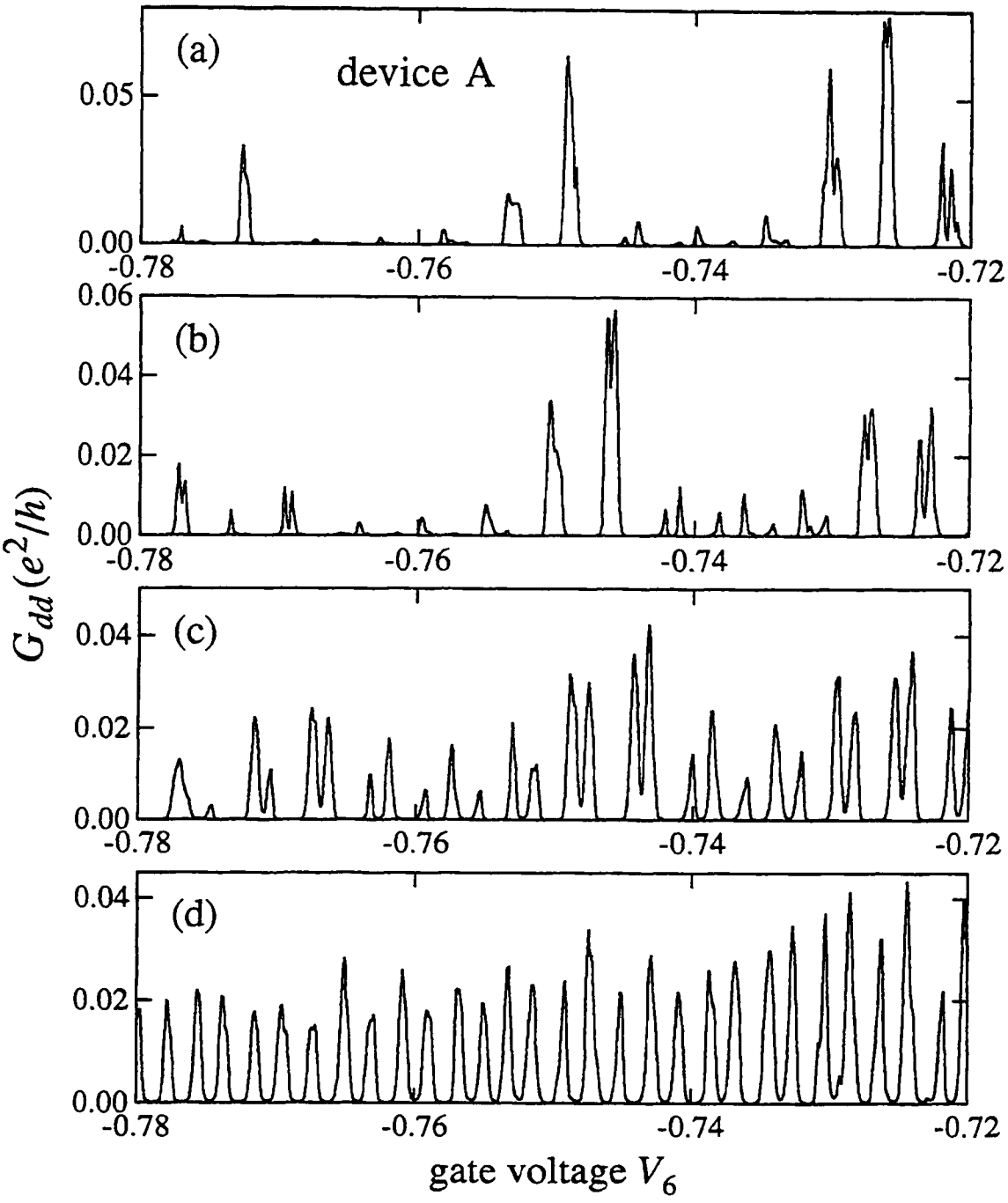


Fig. 6.11 For device A, double-dot conductance G_{dd} vs. gate voltage V_6 for increasing inter-dot coupling. Gate voltage V_3 controlling inter-dot tunnel barrier is (a) –0.901 V, (b) –0.895 V, (c) –0.890 V, and (d) –0.880 V. Barrier conductance in units e^2/h is (a) 0.29, (b) 0.96, (c) 1.42, and (d) 1.90. Gate capacitance mismatch causes peak suppression in (a) and quasiperiodic beating in (c).

gate capacitances measured for the singly energized dots. When the gate capacitances are more nearly matched, the beat period is longer and these phenomena become less prominent. For the top gates, designed with similar capacitances, beating is sometimes observed with a period 70 mV, in good agreement with the expected period $e/|C_{35} - C_{25}| = 80$ mV. No beating is observed in device B, for which the measured gate capacitances are equal to within a few percent.

6.5 TRIPLE QUANTUM DOTS

Peak splitting, stochastic Coulomb blockade, and quasiperiodicity in peak height and spacing are also observed in the triple dot formed by energizing all gates of device A. However, there are important differences between double and triple dots: conductance peaks split into triple peaks when all gate capacitances are nearly equal, and gate capacitance mismatch leads to more complicated beating with a mix of single, double and triple peaks.

The change in triple-dot conductance G_{td} vs. gate voltage V_5 as inter-dot coupling increases is plotted in Fig. 6.12. The figure is analogous to Figs. 6.8 and 6.9 for double dots. Weakly coupled triple dots, Fig. 6.12(a), show conductance peaks with uniform spacing equal to the single-dot spacing. As inter-dot coupling increases in Figs. 6.12(b) and (c), peaks split in two and in three; triply-split peaks are more prevalent for stronger coupling. The mismatch of gate capacitances C_{15}, C_{25}, and C_{35} is observable as an asymmetry of the split peaks. Finally, Fig. 6.12(d) shows the conductance when the tunnel barriers have been removed to create a single, large dot. The gate voltages V_2 and

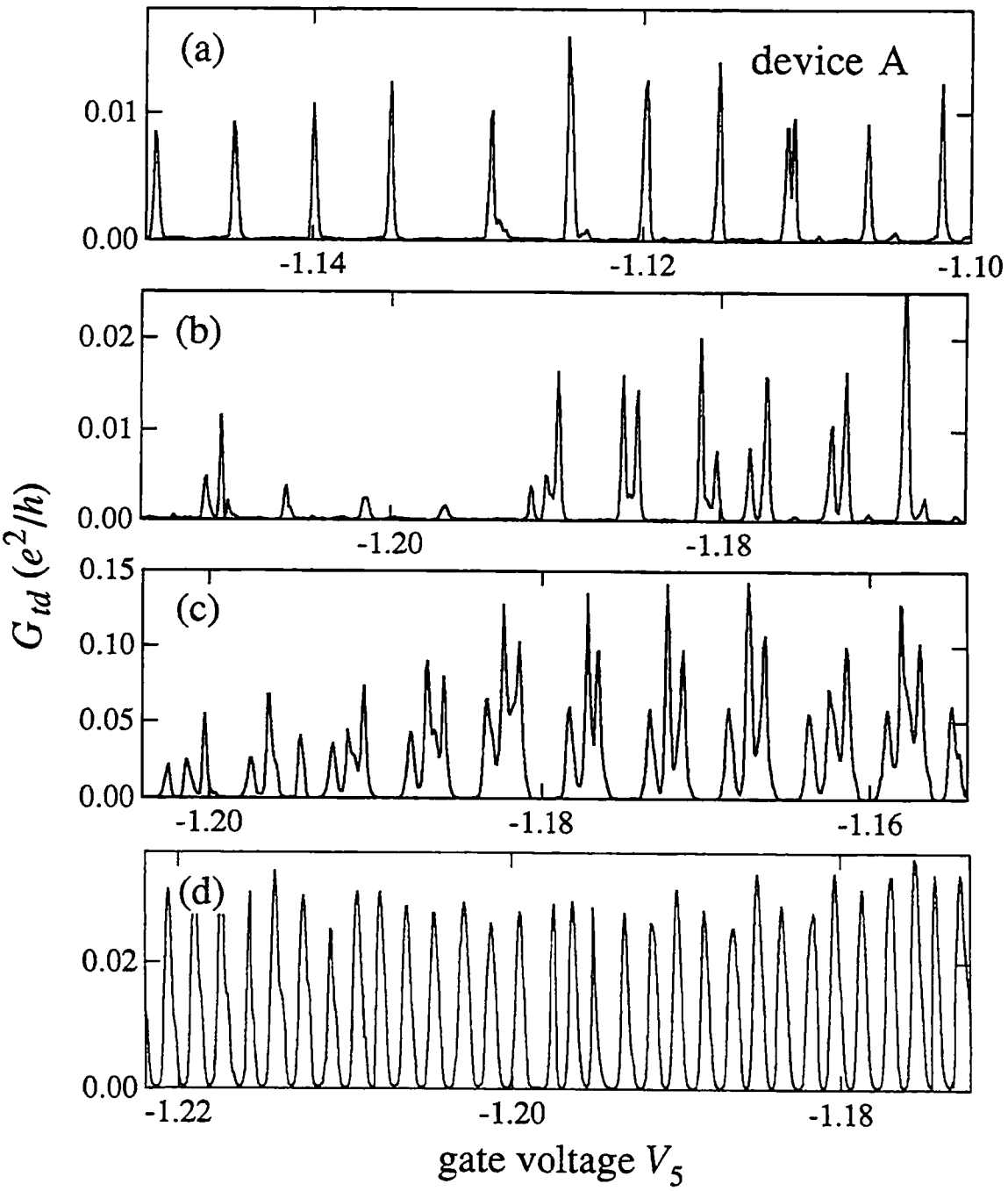


Fig. 6.12 Triple-dot conductance G_{td} vs. gate voltage V_5 for increasing inter-dot coupling. Gate voltages V_2 and V_3 controlling tunnel barriers 2 and 3 are (a) –0.839 V, –0.892 V; (b) –0.833 V, –0.887 V; (c) –0.828 V, –0.883V; and (d) –0.795 V, –0.840 V. Coupling splits conductance peaks into three in (c).

V_3 controlling tunnel barriers 2 and 3 are (a) −0.839 V, −0.892 V; (b) −0.833 V, −0.887 V; (c) −0.828 V, −0.883 V; and (d) −0.795 V, −0.840 V.

Figure 6.13 shows the triple-dot conductance vs. gate voltage V_6; this figure is analogous to Fig. 6.11. As in Fig. 6.12, increased coupling causes peaks to split into double and triple peaks, with triple peaks predominant for stronger coupling. In addition, quasiperiodic beating appears due to the intentional mismatch of gate capacitance C_{26}; the observed beat period in Figs. 6.13(b) and (c) is 26 mV, in excellent agreement with the capacitances measured from single-dot data and with the double-dot beat period when gate voltage V_6 is swept. As for double dots, increasing coupling destroys the stochastic Coulomb blockade. The gate voltages V_2 and V_3 controlling tunnel barriers 2 and 3 are (a) −0.835 V, −0.890 V; (b) −0.833 V, −0.888 V; (c) −0.823 V, −0.878 V; and (d) −0.784 V, −0.838 V.

Beating among three frequencies leads to substantially more structure in the conductance peaks for triple dots than for double dots. Some further examples of triple-dot beating are shown in Fig. 6.14 vs. gate voltage V_5. Just as gate capacitance mismatch leads to a mix of single and double peaks for double dots (see Fig. 6.11), for triple dots it leads to a mix of single, double, and triple peaks, with complicated peak height modulation. The gate voltages V_2 and V_3 controlling tunnel barriers 2 and 3 are (a) −0.833 V, −0.887 V; (b) −0.830 V, −0.883 V; and (c) −0.815 V, −0.870 V.

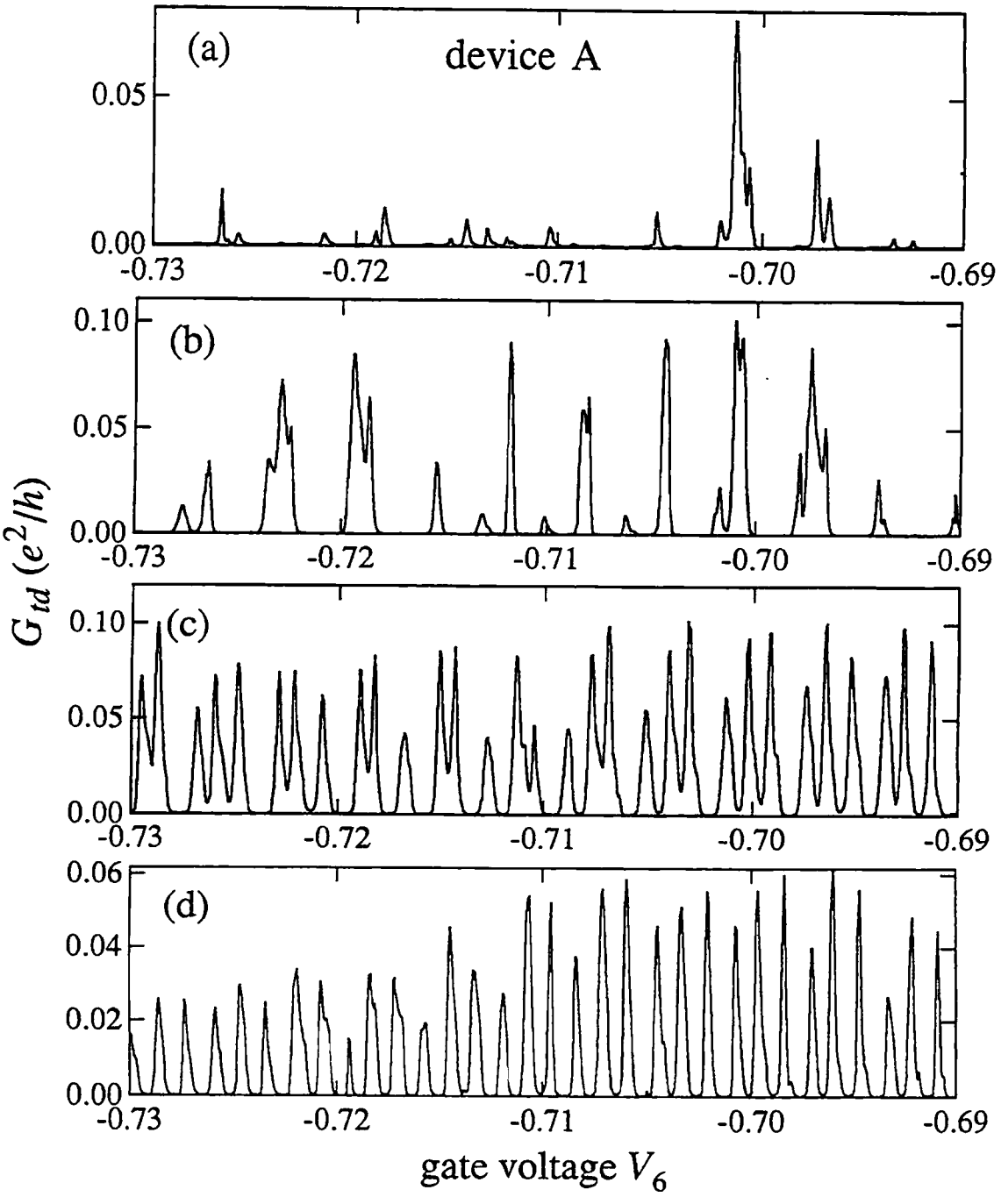


Fig. 6.13 Triple-dot conductance G_{td} vs. gate voltage V_6 for increasing inter-dot coupling. Gate voltages V_2 and V_3 controlling tunnel barriers 2 and 3 are (a) –0.835 V, –0.890 V; (b) –0.833 V, –0.888 V; (c) –0.823 V, –0.878V; and (d) –0.784 V, –0.838 V. Gate capacitance mismatch causes peak suppression in (a) and quasiperiodicity in (c) with measured beat period 26 mV.

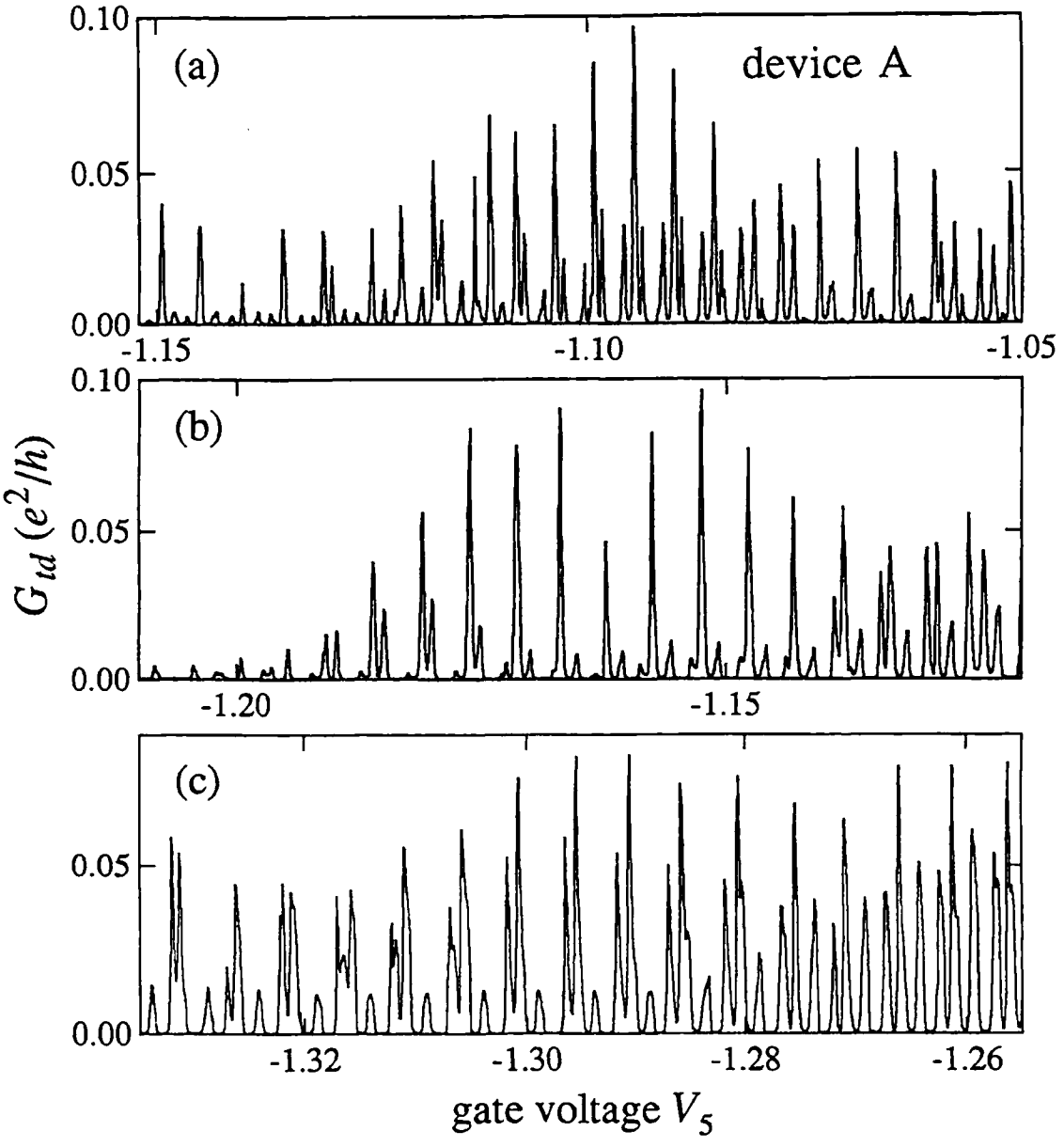


Fig. 6.14 Examples of complicated beat structure in triple-dot conductance G_{td} vs. gate voltage V_5.

6.6 CHARGING MODEL FOR COUPLED QUANTUM DOTS

Conductance peak splitting occurs in theoretical and numerical studies of quantum dot arrays in which coupling arises from inter-dot charging [Ruzin *et al.*, 1992], inter-dot tunneling [Stafford and Das Sarma, 1994], or both [Klimeck, Chen, and Datta, 1994]. All approaches yield qualitative agreement with experiments, which measure the total splitting due to all interactions. This section presents computer simulations of a classical charging model for the conductance of double and triple quantum dots [Bakhvalov *et al.*, 1989; Ruzin *et al.*, 1992]. The model, depicted schematically in Figure 6.15, treats electrons as point particles and inter-dot capacitances as the "knob" that tunes the interaction energy. The computer algorithm is described briefly in Sec. 6.6.1 and in more detail in Appendix 6A. Simulations are presented in Sec. 6.6.2 for double dots and in Sec. 6.6.3 for triple dots. The FORTRAN source code for triple dots appears as Appendix B of this thesis.

6.6.1 The model

Classical charging models of quantum dots in the tunneling regime treat electrons as point particles with charge e and quantum dots as sites that can hold integer numbers of electrons [Kulik and Shekhter, 1975; Glazman and Shekhter, 1989; Averin and Likharev, 1991, 1992; Averin, Korotkov, and Likharev, 1991; Beenakker, 1991; van Houten, Beenakker, and Staring, 1992]. Dots may be coupled through tunnel junctions to other dots and to leads, and may in addition have capacitance to nearby gates. Figure 6.15 shows this model for double and triple dots. Each dot i, $i = 1,\ldots,M$, has capacitances C_i to a gate at voltage V_g, C_{int} to neighboring dots, and C to ground; capacitance of dots to leads is neglected. A dc voltage V biases the arrays symmetrically. Tunnel junctions are

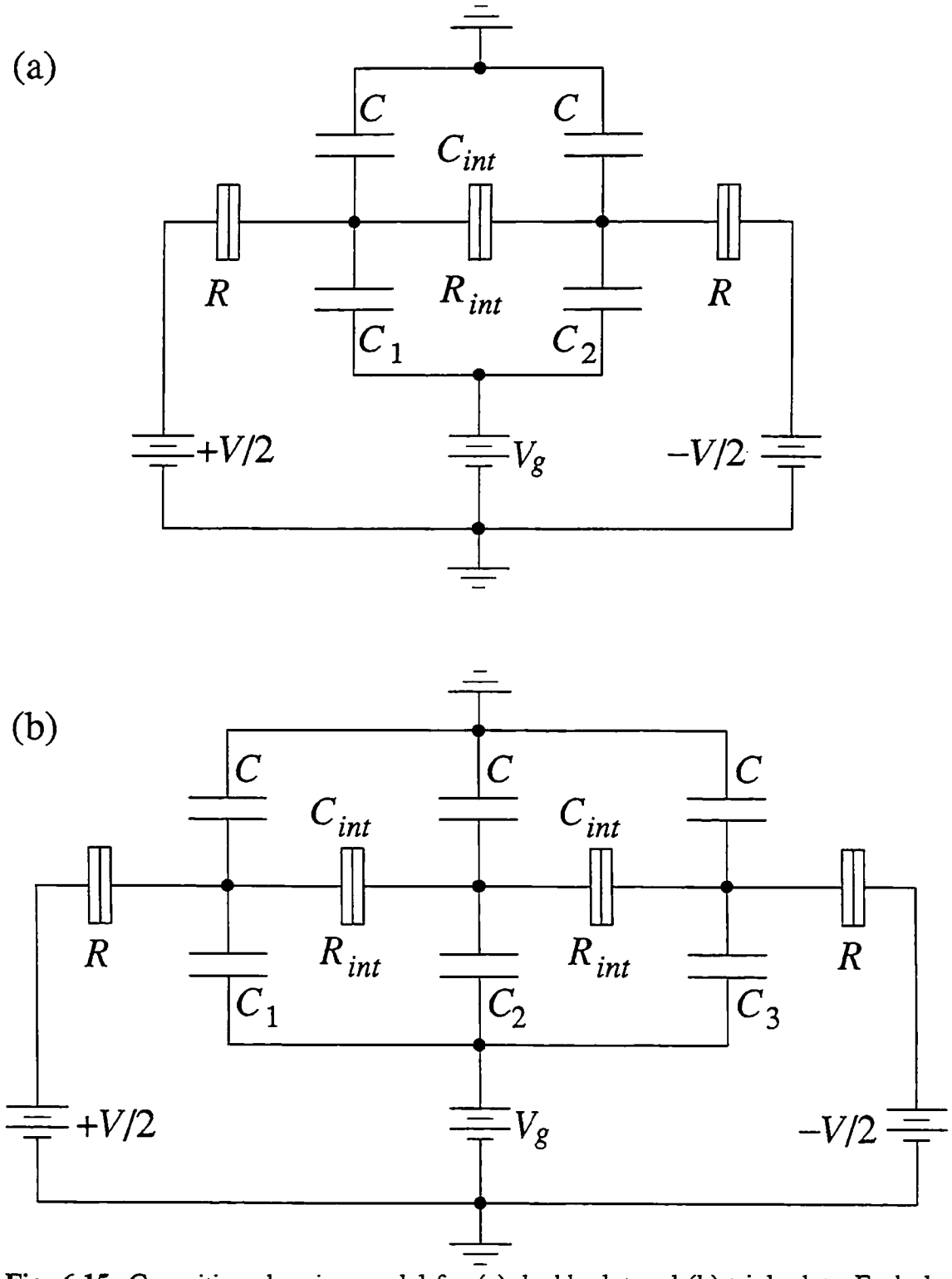


Fig. 6.15 Capacitive charging model for (a) double dot and (b) triple dot. Each dot i has capacitance C_i to gate at voltage V_g, capacitance C to ground, and capacitance C_{int} to neighboring dots. Tunnel junctions (split boxes) have resistance R_{int} between dots and R to leads. A dc voltage V biases arrays symmetrically.

represented in Fig. 6.15 by split boxes. Junctions between dots have resistance R_{int}, while those connected to the leads have resistance R. As explained in Appendix 6A, each tunneling event is treated as a Markov process whose rate is determined by the electrostatic energy difference before and after tunneling occurs.

It is important to emphasize that this *capacitive charging model* is phenomenological: in experiments the largest contribution to the dot interaction energy Δ most likely comes from tunneling matrix elements, not capacitive charging. The model's main advantage is that it is simple to calculate compared to more realistic though more complicated Hubbard models. As shown below, the capacitive charging model reproduces many of the phenomena observed experimentally, sometimes with greater success than Hubbard models, implying that the origin of the interaction energy is not crucial.

The following results focus mainly on how the simulated conductance $\tilde{G}$ vs. gate voltage V_g changes with inter-dot capacitance C_{int}. (The symbol $\tilde{G}$ is used to distinguish the simulated conductance from the experimentally measured conductance G.) Also considered is the dependence of $\tilde{G}$ vs. V_g on capacitance C to ground and on the resistance R_{int} of the inter-dot tunnel barrier. When not explicitly varied, the following parameters are set to their approximate experimental values: $C = 1$ fF, $R = R_{int} = 1\,\mathrm{M}\Omega$, $T = 50$ mK, and $V = 10\,\mu\mathrm{V}$. For both double and triple dots, simulation results are given when the gate capacitances are all equal ($C_i = C_g$, $i = 1,\ldots,M$) and when the gate capacitances values are those determined from single-dot experiments and listed in Sec. 6.3.

6.6.2 Double-dot simulations

Figures 6.16 through 6.18 each show simulated double-dot conductance curves $\tilde{G}_{dd}$ vs. gate voltage V_g for six inter-dot capacitance values ranging from $C_{int} = 0$ aF (top) to $C_{int} = 10$ fF (bottom). The capacitance to ground is $C = 1$ fF in all three figures. This choice of C makes $C_\Sigma = C + C_g + C_{int}$ approximately equal to its experimental value when $C_{int} = C$, which is when the most interesting behavior occurs. The difference between the three figures is that the gate capacitances are $C_1 = C_2 = 40$ aF in Fig. 6.16; $C_1 = 43$ aF and $C_2 = 41$ aF in Fig. 6.17; and $C_1 = 32$ aF and $C_2 = 39$ aF in Fig. 6.18. The gate capacitance values for Figs. 6.17 and 6.18 are chosen to equal the experimentally measured values for device A given in Sec. 6.2, and Fig. 6.16 is included to show what happens in the absence of gate capacitance mismatch. Thus Figs. 6.16 through 6.18 correspond to the experimental data in Figs. 6.9, 6.8 and 6.11, respectively. Note that electron-hole symmetry gives each simulated curve mirror symmetry through the vertical line $V_g = 0$.

Many of the phenomena observed in double-dot experiments appear also in Figs. 6.16 through 6.18: conductance peaks split into double peaks with increasing capacitive coupling C_{int}, and quasiperiodic beating in both peak height and peak spacing occurs due to unequal gate capacitance. Comparisons of particular experimental and simulated curves—for example, Fig. 6.11(c) and the fourth curve from the top of Fig. 6.18—show remarkable qualitative agreement.

It is notable that the fractional peak splitting saturates in the limit of large C_{int} for the capacitive coupling model (bottom curves of Figs. 6.16 through 6.18), as it does experimentally when the inter-dot barrier is removed (Figs. 6.8(d), 6.9(d), and 6.11(d)). Saturation occurs in experiments when the two dots merge into a single, large dot, so that the number N_i of electrons on each dot is no longer a good quantum number. The ability

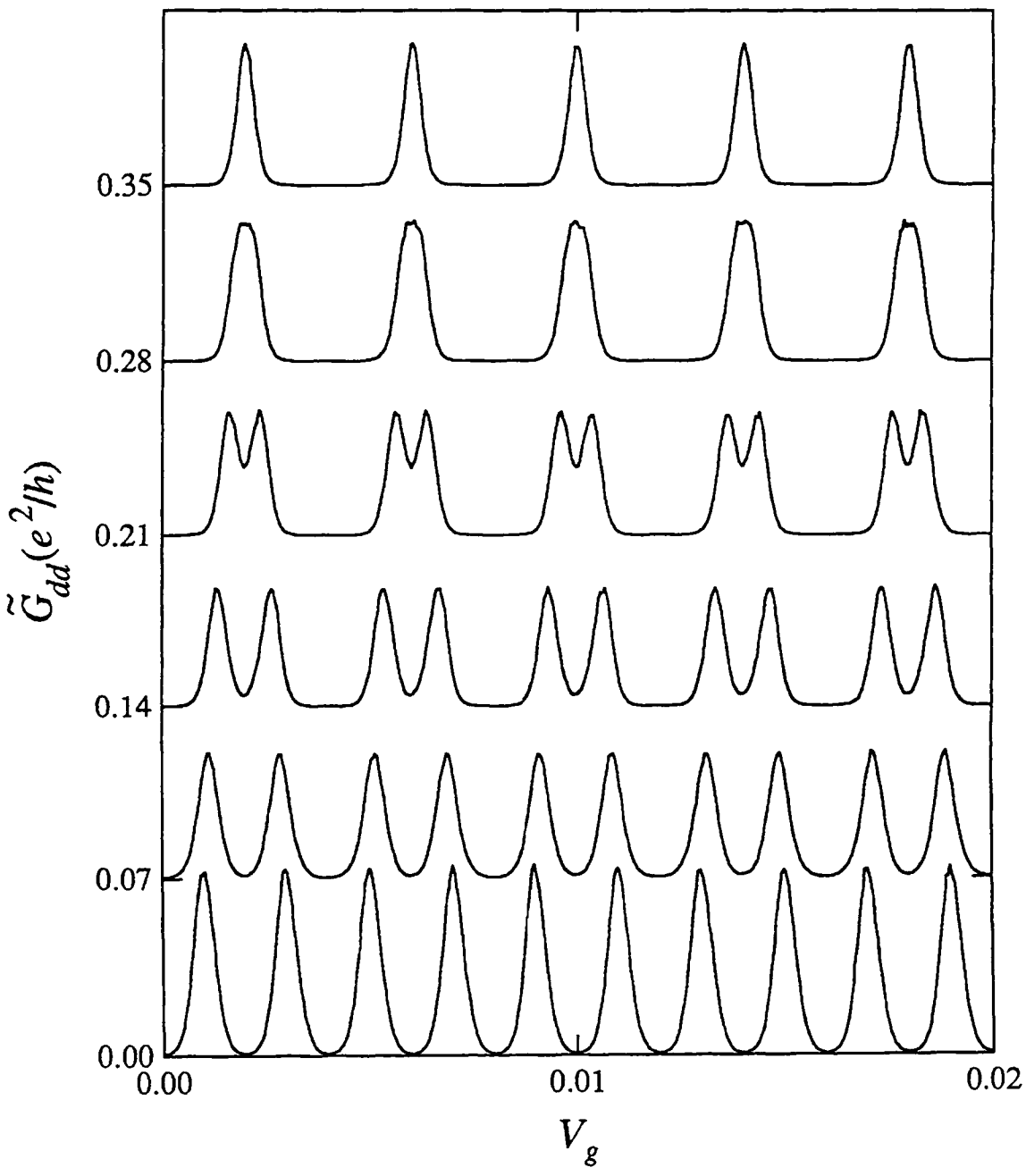


Fig. 6.16 Simulated double-dot conductance $\tilde{G}_{dd}$ vs. gate voltage V_g for increasing inter-dot capacitance C_{int} with gate capacitances $C_1 = C_2 = 40$ aF. Capacitive coupling splits conductance peaks into two peaks. From top to bottom, $C_{int} = 0$ aF, 100 aF, 300 aF, 1 fF, 3 fF, and 10 fF. For all curves, $C = 1$ fF, $T = 50$ mK, and R = 1 MΩ. Curves are offset by 0.07 e^2/h.

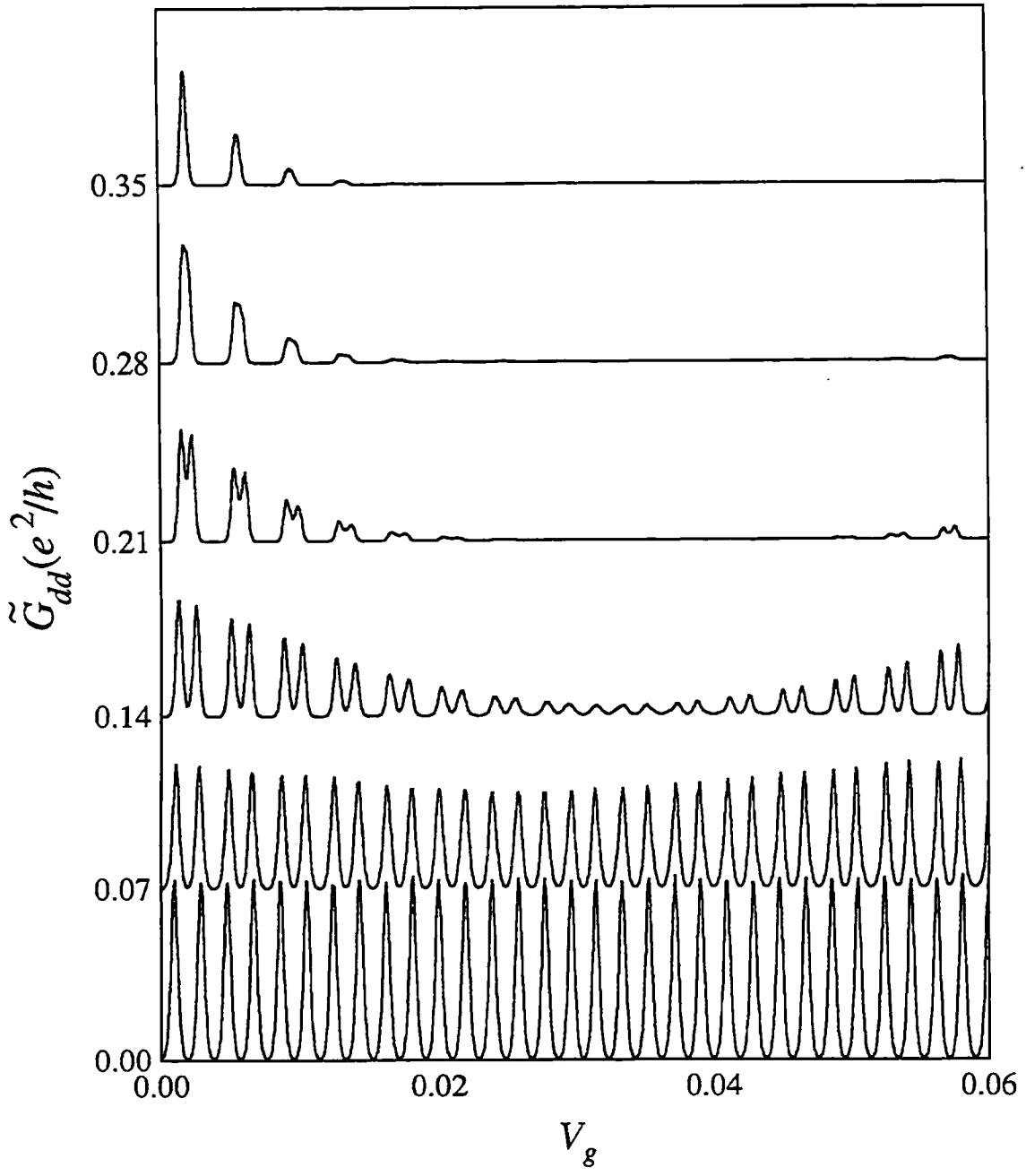


Fig. 6.17 Simulated double-dot conductance $\tilde{G}_{dd}$ vs. gate voltage V_g for increasing inter-dot capacitance C_{int}, with same gate capacitances C_1 = 43 aF and C_2 = 41 aF as for experimental data in Fig. 6.8. Gate capacitance mismatch leads to peak suppression and quasiperiodic beating. From top to bottom, C_{int} = 0 aF, 100 aF, 300 aF, 1 fF, 3 fF, and 10 fF. For all curves, C = 1 fF, T = 50 mK, and R = 1 MΩ. Curves are offset by 0.07 e^2/h.

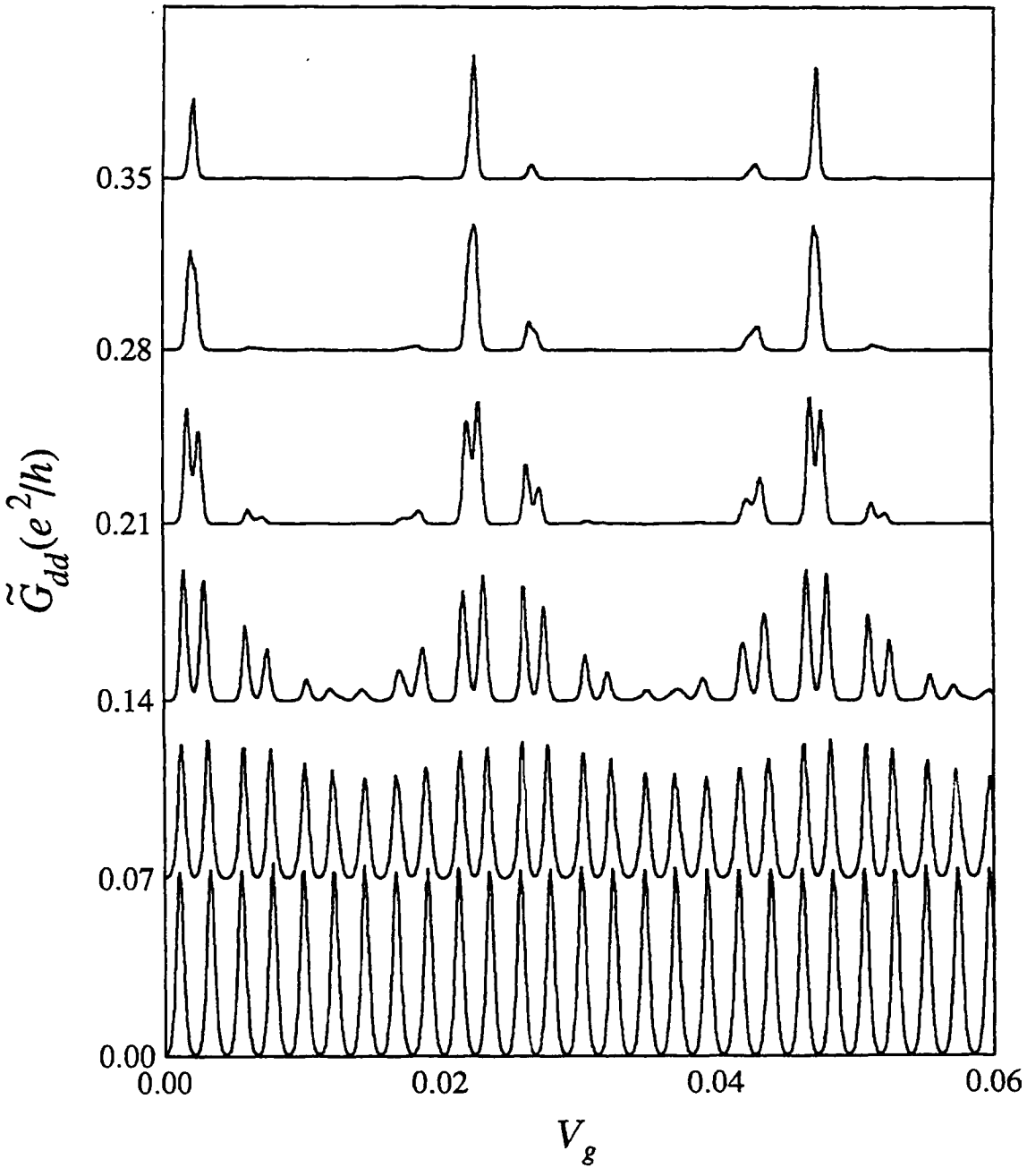


Fig. 6.18 Simulated double-dot conductance $\tilde{G}_{dd}$ vs. gate voltage V_g for increasing inter-dot capacitance C_{int}, with gate same capacitances C_1 = 32 aF and C_2 = 39 aF as for experimental data in Fig. 6.11. Gate capacitance mismatch leads to peak suppression and quasiperiodic beating. From top to bottom, C_{int} = 0 aF, 100 aF, 300 aF, 1 fF, 3 fF, and 10 fF. For all curves, C = 1 fF, T = 50 mK, and R = 1 MΩ. Curves are offset by 0.07 e^2/h.

of the capacitive charging model to reproduce this saturation is in contrast with some Hubbard model calculations, in which split peaks continue to move apart as tunneling matrix elements become stronger, eventually overlapping with neighboring split peaks [Stafford and Das Sarma, 1994].

The dependence of the simulated conductance $\tilde{G}_{dd}$ on capacitance C to ground is investigated in Fig. 6.19. The figure shows how $\tilde{G}_{dd}$ changes as C is increased from 500 aF to 5 fF for the case C_1 = 32 aF, C_2 = 39 aF, and C_{int} = 1 fF (fourth curve from the top of Fig. 6.18). The values of C are (a) 5 fF, (b) 2 fF, (c) 1 fF, and (d) 500 aF. The main effects of increasing C are to broaden the peaks and to increase the baseline conductance. Similar behavior occurs as temperature T is increased, consistent with thermal peak broadening having width $w_T \sim C_\Sigma T$ [Glazman and Shekhter, 1989; Beenakker, 1991; van Houten, Beenakker, and Staring, 1992].

In the simulations discussed so far, all three tunnel junctions have the same resistance R_{int} = 1 MΩ, so that differences in tunneling rates are determined entirely by energetics (see Eq. (6A.16) of Appendix 6A). In experiments, the resistance of the inter-dot tunnel junction changes over a wide range in the transition from weakly coupled to merged dots. Figure 6.20 shows what happens in the capacitive charging model when the resistance R_{int}, rather than the capacitance C_{int}, is used as the "knob" that varies dot interaction. In Fig. 6.20(a), three double-dot conductance curves $\tilde{G}_{dd}$ are compared vs. gate voltage V_g for the case of equal gate capacitance, $C_1 = C_2 = 40$ aF. The solid curve (for which $C_{int} = 0$ fF and $R_{int} = 1$ MΩ) and the short-dashed curve (for which $C_{int} = 1$ fF and $R_{int} = 1$ MΩ) reproduce the top and fourth-from-top curves of Fig. 6.16, respectively. In contrast, the long-dashed curve has no capacitive interaction, $C_{int} = 0$ fF, but the inter-dot resistance is reduced to $R_{int} = 1$ kΩ, well below the value $h/2e^2$ = 12.9 kΩ at which the dots merge in experiments. Note that reducing R_{int} increases peak amplitude but does not

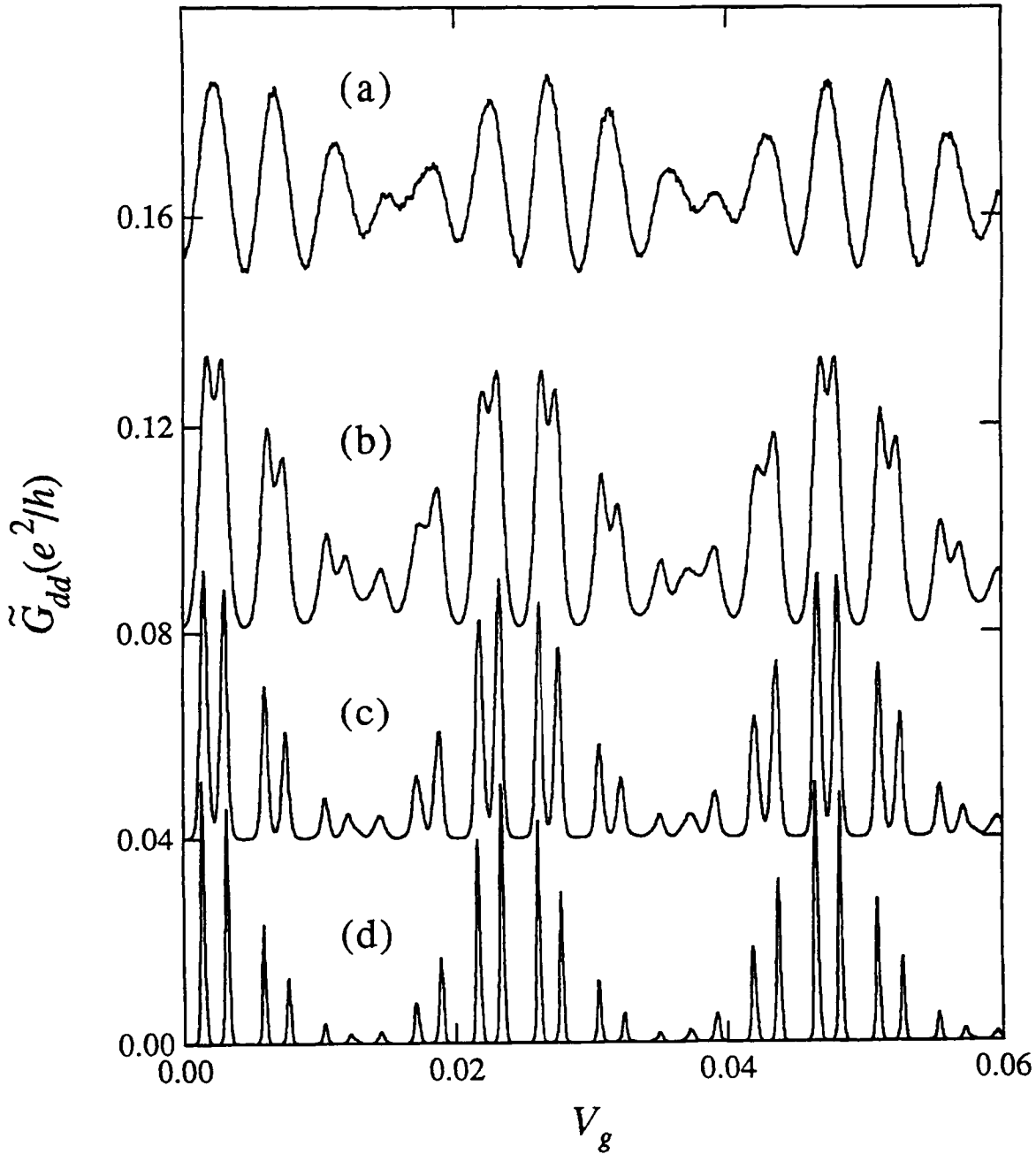


Fig. 6.19 Simulated double-dot conductance $\tilde{G}_{dd}$ vs. gate voltage V_g as capacitance C to ground varies, with same gate capacitances C_1 = 32 aF and C_2 = 39 aF as for experimental data in Fig. 6.11. Increasing capacitance to ground broadens peaks and lifts baseline. From bottom to top, C = 500 aF, 1 fF, 2 fF, and 5 fF. For all curves, C_{int} = 1 fF, T = 50 mK, and R = 1 MΩ. Curves are offset by 0.04 e^2/h.

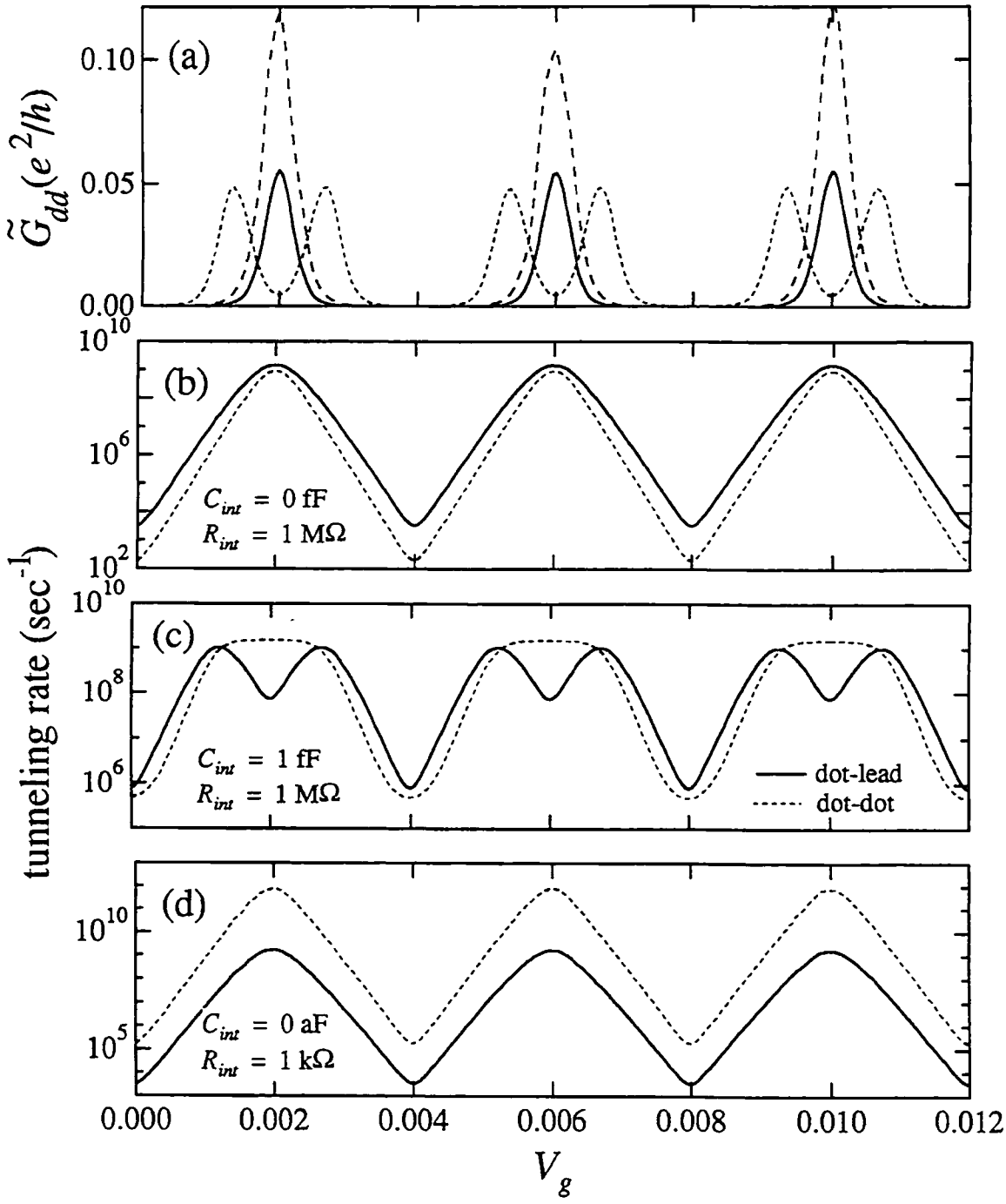


Fig. 6.20 (a) Simulated double-dot conductance $\tilde{G}_{dd}$ vs. gate voltage V_g with $C_1 = C_2 = 40$ aF and $T = 50$ mK. Solid curve is for $C_{int} = 0$ fF and $R_{int} = 1\,\text{M}\Omega$; short-dashed curve, $C_{int} = 1$ fF and $R_{int} = 1\,\text{M}\Omega$; long-dashed curve, $C_{int} = 0$ fF and $R_{int} = 1\,\text{k}\Omega$. Decreasing tunnel resistance increases peak heights but does not split peaks. (b) through (d) Dot-lead (solid curves) and dot-dot (dashed curves) tunneling rates vs. gate voltage V_g for each curve in (a).

split peaks. Changing the inter-dot resistance is *not* equivalent to changing the inter-dot capacitance in the capacitive charging model.

Figures 6.20(b) through (d) show the dot-lead and dot-dot tunneling rates—solid and dashed curves, respectively—for each conductance curve of Fig. 6.20(a). Figure 6.20(b) corresponds to the solid curve of Fig. 6.20(a), Fig. 6.20(c) to the sort-dashed curve, and Fig. 6.20(d) to the long-dashed curve. In each case, the conductance is determined by the tunnel junctions with the lowest tunneling rates, which act as bottlenecks. Comparing (b) and (c) shows that increasing C_{int} broadens peaks in the dot-dot rate and introduces dips into peaks in the dot-lead rate, resulting in conductance peak splitting. On the other hand, comparing (b) and (d) shows that decreasing R_{int} merely increases the dot-dot rate by a multiplicative factor but otherwise leaves the tunneling rates virtually unchanged.

6.6.3 Triple-dot simulations

Figures 6.21 through 6.23 each show simulated triple-dot conductance curves $\tilde{G}_{td}$ vs. gate voltage V_g. As for double dots, there are six curves with inter-dot capacitance values ranging from C_{int} = 0 aF (top) to C_{int} = 10 fF (bottom), and the capacitance to ground is C = 1 fF in all three figures. Gate capacitances are $C_1 = C_2 = C_3$ = 40 aF in Fig. 6.21; C_1 = 38 aF, C_2 = 43 aF, and C_3 = 41 aF in Fig. 6.22; and C_1 = 41 aF, C_2 = 32 aF, and C_3 = 39 aF in Fig. 6.23. Figures 6.22 and 6.23 correspond to the experimental data in Figs. 6.12 and 6.13, respectively. Again, electron-hole symmetry gives each simulated curve mirror symmetry through the vertical line $V_g = 0$.

The experimental differences between double dots and triple dots—peak splitting into triple rather than double peaks, and more complicated beating structure—appear as well in the simulations. Splitting into three peaks is clearest for equal gate capacitances (Fig. 6.21). Strong peak suppression via the stochastic Coulomb blockade is visible in Figs.

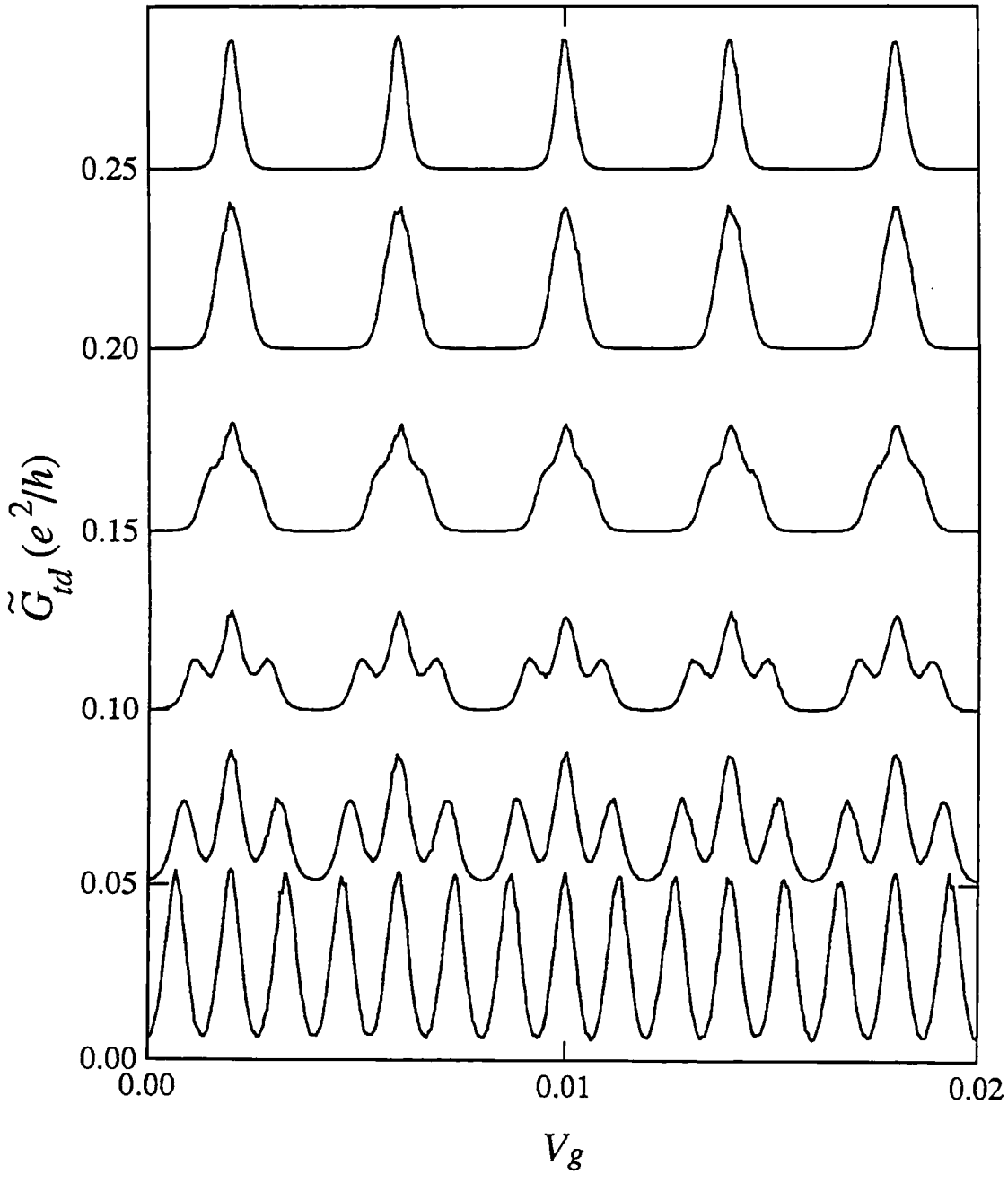


Fig. 6.21 Simulated triple-dot conductance $\tilde{G}_{td}$ vs. gate voltage V_g for increasing inter-dot capacitance C_{int}, with gate capacitances $C_1 = C_2 = C_3 = 40$ aF. Capacitive coupling splits conductance peaks into three peaks. From top to bottom, $C_{int} = 0$ F, 100 aF, 300 aF, 1 fF, 3 fF, and 10 fF. For all curves, $C = 1$ fF, $T = 50$ mK, and R = 1 MΩ. Curves are offset by 0.06 e^2/h.

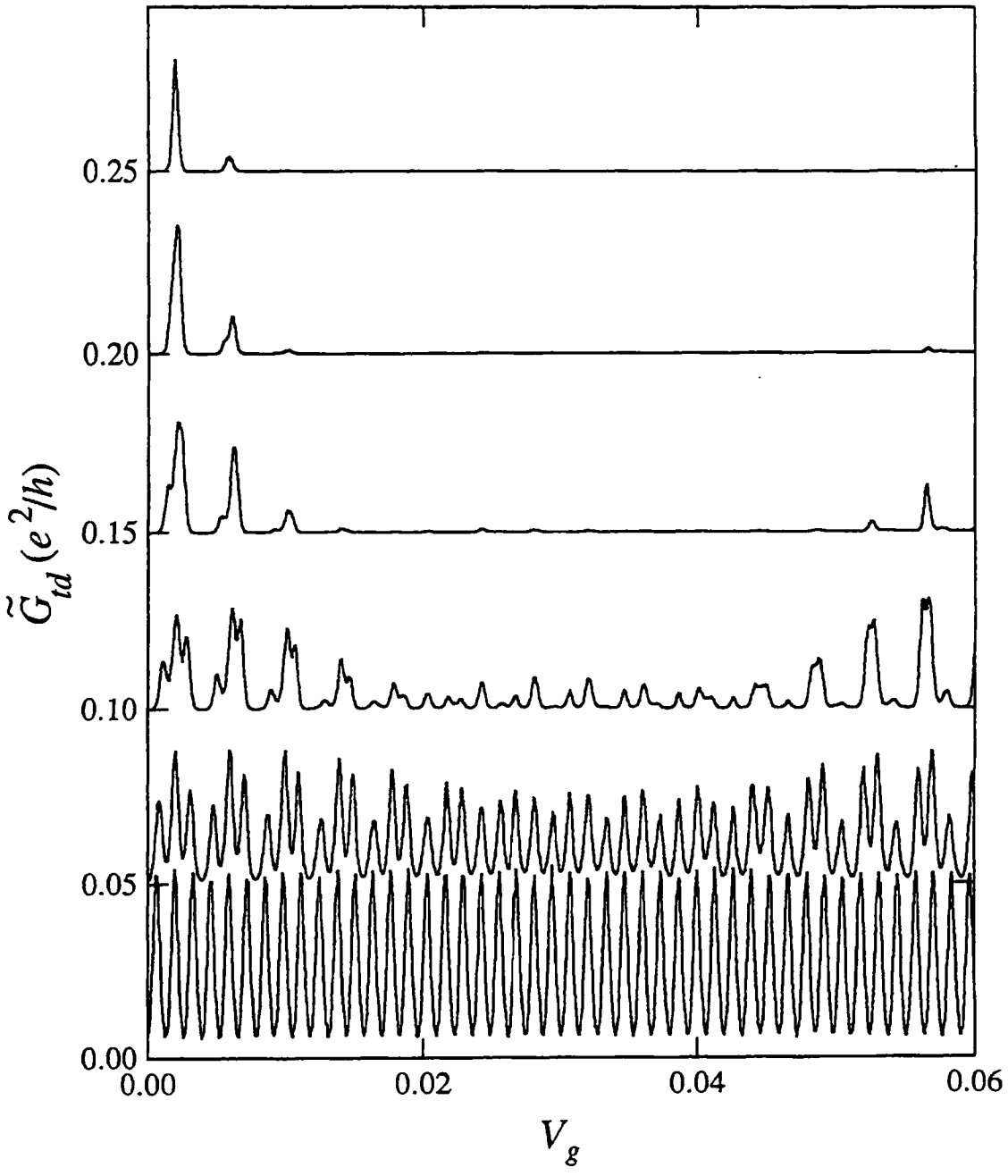


Fig. 6.22 Simulated triple-dot conductance $\tilde{G}_{td}$ vs. gate voltage V_g for increasing inter-dot capacitance C_{int}, with same gate capacitances C_2 = 38 aF, C_2 = 43 aF, and C_3 = 41 aF as for experimental data in Fig. 6.12. Gate capacitance mismatch leads to peak suppression and quasiperiodic beating. From top to bottom, C_{int} = 0 aF, 100 aF, 300 aF, 1 fF, 3 fF, and 10 fF. For all curves, C = 1 fF, T = 50 mK, and R = 1 MΩ. Curves are offset by 0.06 e^2/h.

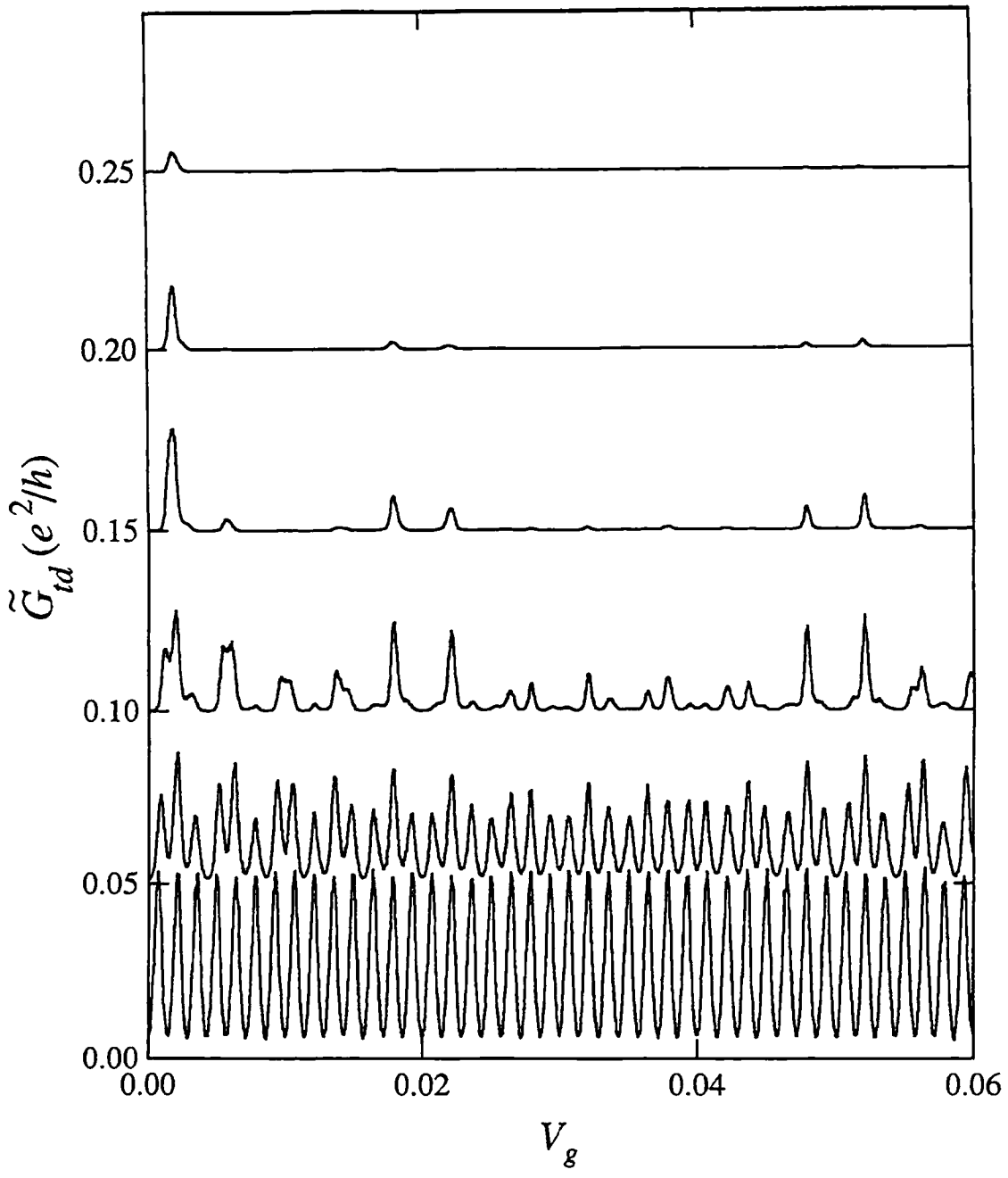


Fig. 6.23 Simulated triple-dot conductance $\tilde{G}_{td}$ vs. gate voltage V_g for increasing inter-dot capacitance C_{int}, with same gate capacitances C_1 = 41 aF, C_2 = 32 aF, and C_3 = 39 aF as for experimental data in Fig. 6.13. Gate capacitance mismatch leads to peak suppression and quasiperiodic beating. From top to bottom, C_{int} = 0 aF, 100 aF, 300 aF, 1 fF, 3 fF, and 10 fF. For all curves, C = 1 fF, T = 50 mK, and R = 1 MΩ. Curves are offset by 0.06 e^2/h.

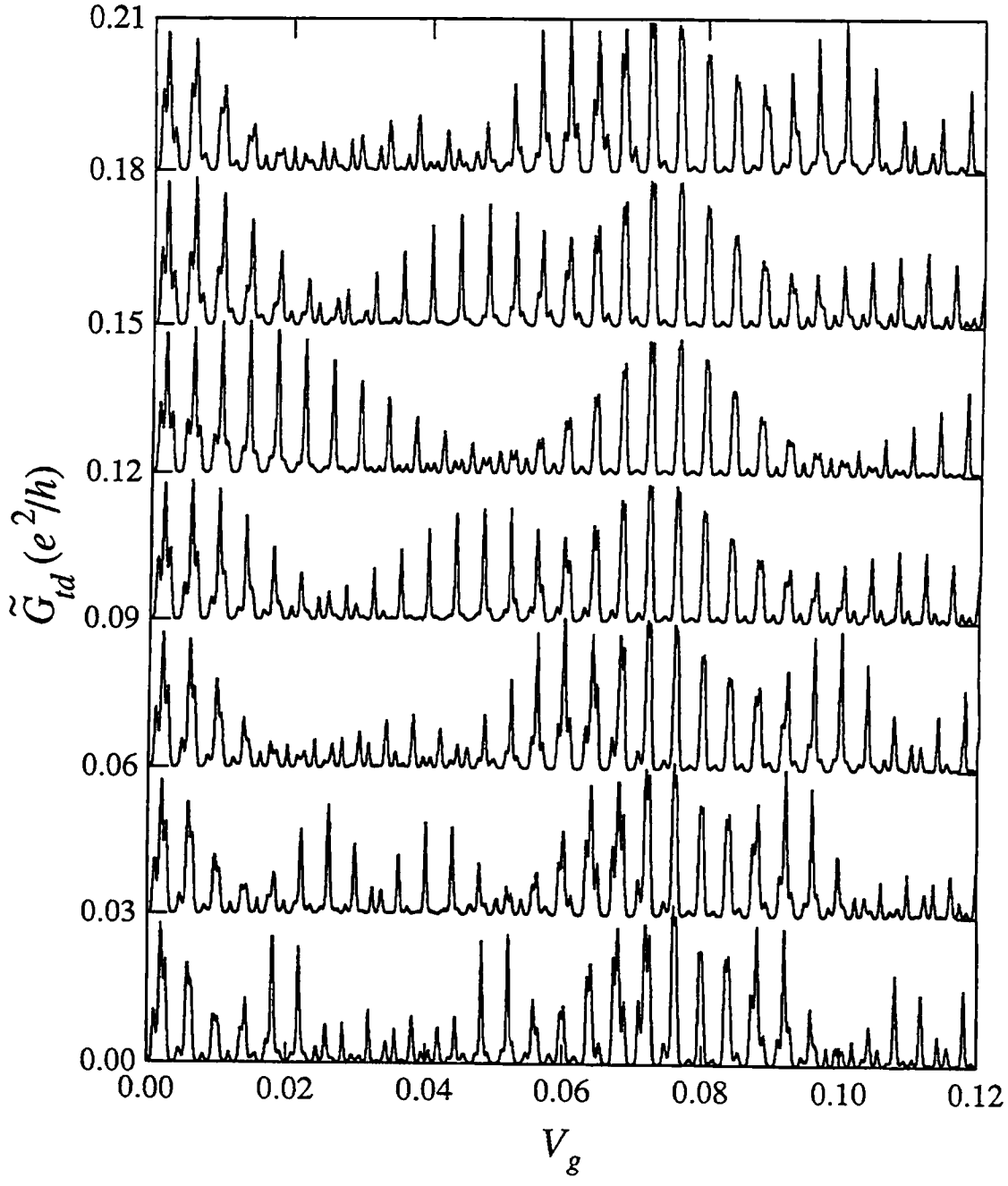


Fig. 6.24 Simulated triple-dot conductance $\tilde{G}_{td}$ vs. gate voltage V_g for increasing gate capacitance C_2 with gate capacitances C_1 = 41 aF and C_3 = 39 aF. Small changes in C_2 dramatically affect conductance. From top to bottom, C_2 increases from 36 aF to 48 aF in increments of 2 aF. For all curves, C = 1 fF, C_{int} = 1.5 fF, T = 50 mK, and R = 1 MΩ.

6.22 and 6.23, which show the simulated triple-dot conductance $\tilde{G}_{td}$ using the experimentally determined gate capacitances. The figures show that gate capacitance mismatches of 10% can suppress most of the conductance peaks when inter-dot coupling is weak. As for double dots, increasing the inter-dot coupling lifts the stochastic Coulomb blockade. The structure in peak height and spacing can be quite complicated for triple dots, as illustrated by the conductance curves for $C_{int} = 1000$ aF in Figs. 6.22 and 6.23 (fourth curve from top of both figures). More examples of this structure are shown in Fig. 6.24, which shows the triple-dot conductance $\tilde{G}_{td}$ as C_2 is increased from 36 aF to 48 aF for the case $C_1 = 41$ aF, $C_3 = 39$ aF, $C_{int} = 1.5$ fF, and $C = 1$ fF. The structure in these curves is qualitatively similar to the experimental examples of triple-dot peak structure shown *in* Fig. 6.14. The figure shows how small changes in gate capacitance dramatically affect the triple-dot conductance, a phenomenon often neglected in theories of coupled dots but of great significance in experiments and applications.

6.7 SUMMARY

This chapter has presented tunneling measurements of the conductance of double and triple quantum dot arrays. The measurements, performed in a He dilution refrigerator with no applied magnetic field, show how inter-dot coupling leads to a variety of new phenomena not observed in single dots. The most salient of these is conductance peak splitting controlled by a gate voltage; peaks split in two for double dots and in three for triple dots. The separation between split peaks is approximately proportional to the tunnel barrier conductance and experimentally determines the dot interaction energy due to inter-dot tunneling and inter-dot charging. Gate capacitance mismatch gives rise to additional

quasiperiodic structure in the conductance vs. gate voltage and leads to conductance peak suppression through the stochastic Coulomb blockade.

Computer simulations based on a phenomenological capacitive charging model show excellent qualitative agreement with experiment. The main features of the experimental data—peak splitting, quasiperiodic structure in peak height and spacing, and stochastic Coulomb blockade—appear also in the simulations. The simulations point out the important role of gate capacitance mismatch, often neglected in coupled dot theories but unavoidable in experiments. Another notable feature of the capacitive charging model is its ability, unlike Hubbard models, to reproduce the experimentally observed saturation of peak splitting.

APPENDIX 6A: SIMULATING DOUBLE AND TRIPLE DOTS

This appendix describes a computer algorithm for the capacitive charging model for coupled quantum dots. The algorithm uses changes in the electrostatic energy of the circuits in Fig. 6.15 to compute tunneling probabilities for each tunnel junction. These probabilities are the basis of a Monte Carlo process that simulates current flow, from which the conductances $\tilde{G}_{dd}$ and $\tilde{G}_{td}$ are calculated.

The electrostatic energies F_M for double ($M = 2$) and triple ($M = 3$) dots are the sum of the electrostatic charging energy of each capacitor and the work done by each battery. To calculate these terms, it is helpful to introduce the capacitor charges q_j shown in Fig. 6.25(a) and (b) for double and triple dots. The double-dot energy F_2 is

$$F_2 = \frac{q_1^2}{2C_1} + \frac{q_2^2}{2C_2} + \frac{q_3^2 + q_4^2}{2C} + \frac{q_5^2}{2C_{int}} + V_g(q_1 + q_2) + \frac{1}{2}V(q_L - q_R). \quad (6A.1)$$

The triple-dot energy F_3 is

$$F_3 = \frac{q_1^2}{2C_1} + \frac{q_2^2}{2C_2} + \frac{q_3^2}{2C_3} + \frac{q_4^2 + q_5^2 + q_6^2}{2C} + \frac{q_7^2 + q_8^2}{2C_{int}}$$

$$+ V_g(q_1 + q_2 + q_3) + \frac{1}{2}V(q_L - q_R). \quad (6A.2)$$

The first four terms of Eq. (6A.1) and the first five terms of Eq. (6A.2) are the charging energy of individual capacitors. The last two terms of each equation are the work done by

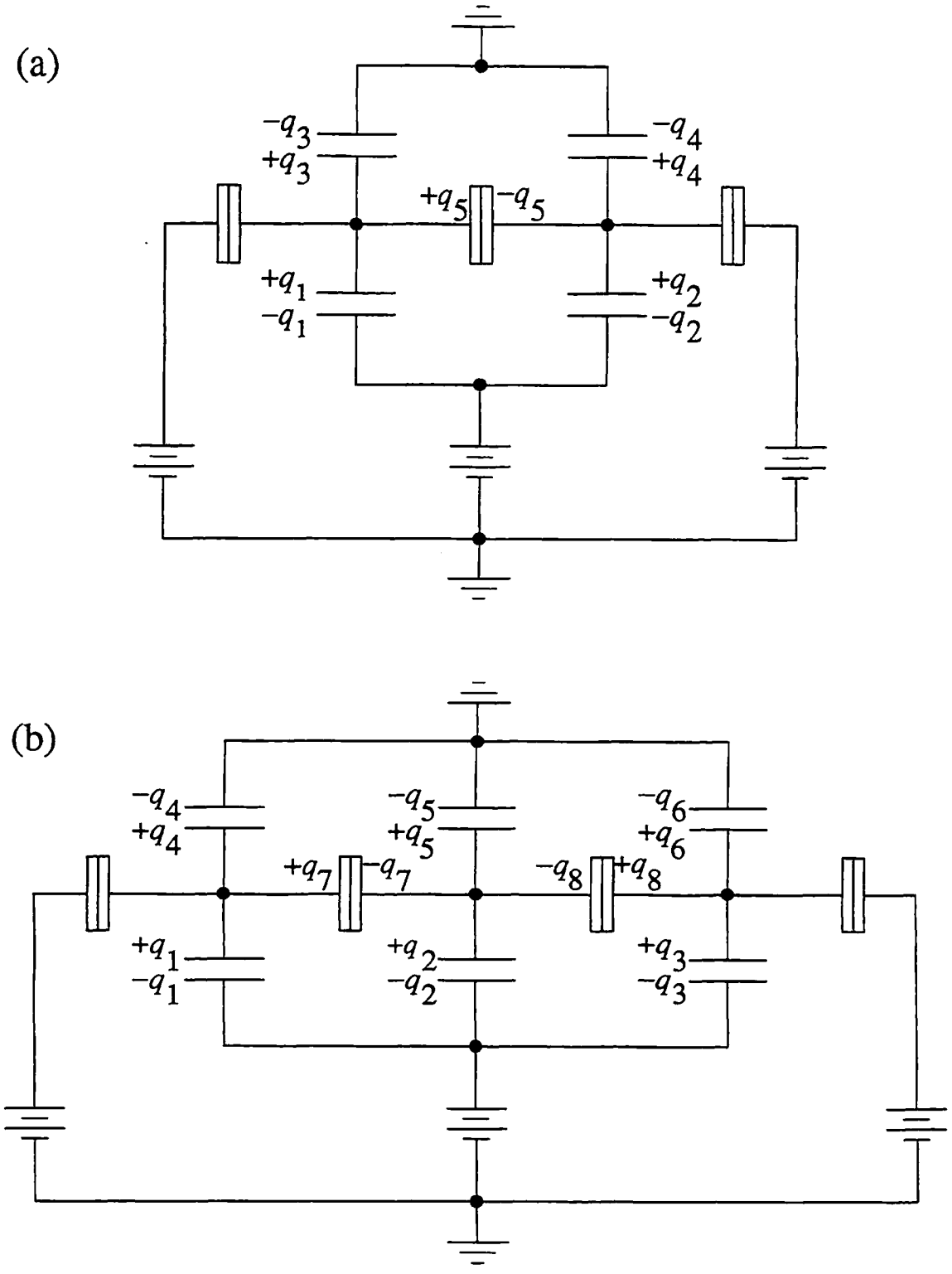


Fig. 6.25 Capacitor charges q_j for (a) double and (b) triple dots in capacitive charging model. Charges are used to calculate electrostatic energies in Eqs. (6A.1) and (6A.2).

the batteries providing the gate voltage V_g and the dc bias V; q_L and q_R are the total charge transferred into the arrays from the left and right leads, respectively.

The energies (6A.1) and (6A.2) can be rewritten in terms of the numbers N_i of electrons on each dot by applying the same procedure used in Sec. 4.4 for single dots, which is equivalent to the method of Lagrange multipliers. Consider first the double dot shown in Fig. 6.15(a), which has the energy F_2. Because the numbers N_1 and N_2 of electrons on the two dots must be integers, the capacitor charges q_i, $i = 1,\ldots,5$, are constrained by

$$q_1 + q_3 + q_5 = -eN_1 \tag{6A.3}$$

$$q_2 + q_4 - q_5 = -eN_2. \tag{6A.4}$$

Inserting these into Eq. (6A.1) to eliminate q_1 and q_2 yields

$$F_2 = \frac{(q_3 + q_5 + eN_1)^2}{2C_1} + \frac{(q_4 - q_5 + eN_2)^2}{2C_2} + \frac{q_3^2 + q_4^2}{2C} + \frac{q_5^2}{2C_{int}}$$

$$- V_g(q_3 + q_4 + eN_1 + eN_2) + \frac{1}{2}V(q_L - q_R)\,. \tag{6A.5}$$

The equilibrium capacitor charges q_3, q_4, and q_5 are found by solving $\partial F_2/\partial q_i = 0$ for $i = 3, 4, 5$ [Ruzin *et al*, 1992]. The result is

$$q_3 = -eC\left[(C + C_{int} + C_2)N_1 + C_{int}N_2\right]/D_2 \tag{6A.6}$$

$$q_4 = -eC\left[(C + C_{int} + C_1)N_2 + C_{int}N_1\right]/D_2 \tag{6A.7}$$

$$q_5 = -eC_{int}\left[(C + C_2)N_1 - (C + C_1)N_2\right]/D_2 \tag{6A.8}$$

$$D_2 = C^2 + CC_1 + CC_2 + C_1C_2 + C_{int}(2C + C_1 + C_2). \tag{6A.9}$$

The triple-dot electrostatic energy F_3 is calculated similarly. The result is

$$F_3 = \frac{(q_4 + q_7 + eN_1)^2}{2C_1} + \frac{(q_5 - q_7 - q_8 + eN_2)^2}{2C_2} + \frac{(q_6 + q_8 + eN_3)^2}{2C_3}$$

$$+ \frac{q_4^2 + q_5^2 + q_6^2}{2C} + \frac{q_7^2 + q_8^2}{2C_{int}}$$

$$- V_g(q_4 + q_5 + q_6 + eN_1 + eN_2 + eN_3) + \frac{1}{2}V(q_L - q_R). \tag{6A.10}$$

The triple-dot capacitor charges are

$$q_4 = -eC\left[\left(C_2C_3 + C_2C_{int} + 2C_3C_{int} + C_{int}^2 + C_2C + C_3C + 3C_{int}C + C^2\right)N_1\right.$$

$$\left. + \left(C_3C_{int} + C_{int}^2\right)N_2 + C_{int}^2N_3\right]/D_3 \tag{6A.11}$$

$$q_5 = -eC\left[\left(C_1C_3 + C_1C_{int} + C_3C_{int} + C_{int}^2 + C_1C + C_3C + C_{int}C + C^2\right)N_2\right.$$

$$\left. + \left(C_3C_{int} + C_{int}^2 + C_{int}C\right)N_1 + \left(C_1C_{int} + C_{int}^2 + C_{int}C\right)N_3\right]/D_3 \tag{6A.12}$$

$$q_6 = -eC\Big[\big(C_1C_2 + C_2C_{int} + 2C_1C_{int} + C_{int}^2 + C_1C + C_2C + 3C_{int}C + C^2\big)N_3$$

$$+ \big(C_1C_{int} + C_{int}^2\big)N_2 + C_{int}^2N_1\Big]\Big/D_3 \qquad (6A.13)$$

$$q_7 = -eC_{int}\Big[\big(C_2C_3 + C_2C_{int} + C_3C_{int} + C_2C + C_3C + 2C_{int}C + C^2\big)N_1$$

$$- \big(C_1C_3 + C_1C_{int} + C_1C + C_3C + C_{int}C + C^2\big)N_2$$

$$- \big(C_1C_{int} + C_{int}C\big)N_3\Big]\Big/D_3 \qquad (6A.14)$$

$$q_8 = -eC_{int}\Big[\big(C_1C_2 + C_1C_{int} + C_2C_{int} + C_1C + C_2C + 2C_{int}C + C^2\big)N_3$$

$$- \big(C_1C_3 + C_1C_{int} + C_1C + C_3C + C_{int}C + C^2\big)N_2$$

$$- \big(C_3C_{int} + C_{int}C\big)N_1\Big]\Big/D_3 \qquad (6A.15)$$

$$D_3 = C_1C_2C_3 + C_{int}(C_1C_2 + 2C_1C_3 + C_2C_3) + \big(C_{int}^2 + C^2\big)(C_1 + C_2 + C_3)$$

$$+ C(C_1C_2 + C_1C_3 + C_2C_3) + C_{int}C(3C_1 + 2C_2 + 3C_3 + 3C_{int} + 4C) + C^3. \qquad (6A.16)$$

The electrostatic energies F_2 and F_3 determine the rates at which electrons pass through tunnel junctions. Because electrons can move from left to right or from right to left through each junction, there are six possible tunneling processes for double dots, eight for triple dots, and in general $2(M+1)$ for M-dot arrays. Each process occurs at a rate Γ_k,

$k = 1,\ldots,2(M+1)$, that depends on the accompanying change ΔF_M in energy according to [Bakhvalov *et al.*, 1989]

$$\Gamma_k = -\frac{1}{e^2 R_{int}} \frac{\Delta F_M}{1-\exp\left(\Delta F_M / k_B T\right)}. \tag{6A.17}$$

In Eq. (6A.17), R_{int} is the tunnel junction resistance, assumed to be the same for each junction, and T is the temperature. For $T = 0$, the rate Γ_k is zero when $\Delta F_M > 0$ and is proportional to ΔF_M when $\Delta F_M < 0$; for $T > 0$, there is a small rate for processes that increase the energy. This procedure treats tunneling as sequential [Büttiker, 1988] and ignores higher-order co-tunneling events.

Equations (6A.5), (6A.10), and (6A.17) are the essence of the capacitive charging model. Simulations of the model proceed as follows [Bakhvalov *et al.*, 1989]. The first step is to find the set $\{N_i\}$, $i = 1,\ldots,M$ of numbers of electrons on each dot that minimizes the appropriate energy F_M, Eq. (6A.5) or (6A.10). This step is performed once at the beginning of a calculation. The next step is to find the rates Γ_k, $k = 1,\ldots,2(M+1)$, of all possible tunneling events. For example, the rate for tunneling from dot 1 to dot 2 is found by decrementing N_1 by one, incrementing N_2 by one, and inserting the energy difference between this configuration and the previous one into Eq. (6A.17). The rates Γ_k determine the probability $P(t)$ for remaining in the lowest energy configuration,

$$P(t) = \exp\left[\left(t - t_0\right) \sum_{k=1}^{2M+1} \Gamma_k\right], \tag{6A.18}$$

where t_0 is the time at which the previous tunneling event occurred. For large enough t—for example, for t such that $P(t) = 1/2$—it is likely that some tunneling event occurs.

The next step is to determine which of the $2(M+1)$ possible events occurs by a roll of the dice: the interval between 0 and 1 is broken up into $2(M+1)$ subintervals, each with size proportional to one of the rates Γ_k; a random number with uniform distribution over the interval between 0 and 1 is generated; and the subinterval in which the random number falls determines the tunneling event. The values $\{N_i\}$ are changed to take account of this event, and the procedure begins again with the calculation of tunneling rates for the new configuration. This process is repeated for a large number of tunneling events (usually about 5×10^4), after which the simulated conductance $\tilde{G}$ is found from $\tilde{G} = q/tV$, where $q = q_L - q_R$ is the net charge transferred across the array, t the total elapsed time, and V the voltage bias.

CONCLUSION

This thesis takes a bumpy ride from lofty visions of quantum neurocomputers in the Introduction to more mundane details of replica-symmetric calculations in Chapters 2 and 3 and dilution refrigerator experiments in Chapters 5 and 6. Along the way, some fascinating frontiers of physics are encountered. Chapters 2 and 3 show how statistical mechanics and nonlinear dynamics can be combined to analyze attractors in disordered dynamical systems. The experiments of Chapter 6 raise the intriguing possibility of building custom artificial atoms, molecules, and crystals to probe aspects of band formation not experimentally accessible in real materials. While these lines of research are important in their own right, it is also important to ask how they relate back to the lofty visions that inspired them. In other words, how likely is it that analog neural networks or quantum dot arrays will make the transition out of physics laboratories and into widespread use?

This question is easier to answer for neural networks. Already dozens of products apply neural networks to such varied tasks as electrocardiograph and pap-smear analysis, optical character recognition, and financial analysis [Hammerstrom, 1993]. In addition, a large number of firms routinely use neural networks to control complex manufacturing processes, and the armed forces use them for target recognition and flight control. While most of these are software applications, meaning that they run on conventional computers, CMOS neural network chips are also becoming commercially available. Clearly, for neural networks the transition to from laboratory to widespread use has already begun.

It is unlikely, however, that neural networks will ever supplant conventional von Neumann computers. Conventional computers do indeed perform many tasks poorly compared to biological neural networks; but for straightforward numerical computation

they will probably always reign supreme. Most likely the two architectures will achieve a kind of harmony, with neural networks performing analog-to-digital conversion, data compression, pattern recognition, and other front-end processing for conventional computers. The discrete-time, parallel, analog neural networks investigated in this thesis are well-suited for such integration with digital machines.

The future of single-electron nanoelectronics is murkier. Despite recent reports of room-temperature single-electron charging devices [Bock and Hartnagel, 1993; Yano *et al.*, 1993], the day when single-electron circuits can be manufactured and operated cheaply and reliably is probably far off. The difficulties arising from nanofabrication and cryogenic operation are likely to become less formidable with technological advances. On the other hand, the extreme sensitivity of single-electron circuits to radiation and charge noise may place inherent limits on their usefulness.

As with neural networks and conventional computers, nanoelectronic devices are likely to complement silicon VLSI devices rather than to replace them. Silicon is probably here to stay: for a large number of computing applications it may always be the most economic alternative. Single-electron devices may appear first in specialized applications—ultra high-density memory, or perhaps even neural networks—before gaining wider acceptance. Most likely these devices will not be formed using the split-gate technique in GaAs/AlGaAs heterostructures, which is well-suited for experiments but cumbersome for larger circuits. Instead, they may be fabricated from Al/AlO_x or perhaps one of the more speculative technologies alluded to in the Introduction. Either way, device interactions like those studied in this thesis will be crucial to their operation.

APPENDIX A

ASSOCIATIVE MEMORY STORAGE CAPACITIES

The following FORTRAN program calculates the storage capacity α_c of a k-winner competitive associative memory with $Q = 4$ neurons and $k = 2$ winners per cluster. The program output appears in the phase diagrams of Chapter 2 as the boundary between the recall and spin-glass regions. The k-winner program is presented here because it is more general than the winner-take-all program, for which the competitive biases can be calculated explicitly.

A binary search is used to find α_c. A trial value of α is chosen, and the saddle-point equations (2.21) through (2.23) are solved self-consistently. If the overlap order parameter $m \equiv 0$ then $\alpha_c < \alpha$, otherwise $\alpha_c > \alpha$. A new trial value of α is chosen based on this outcome, and the saddle-point equations are solved again. The process is repeated until α_c has been bracketed to the desired accuracy.

This procedure is quite complex to implement. The saddle-point equations are solved iteratively using Newton's method. Both the equations and their Jacobian, which must also be computed for Newton's method, contain integrals of the form given in Eq. (2.25) that must be computed numerically at each iteration (there are 12 integrals in all for $Q = 4$). The integrands, in turn, contain variables x_a defined implicitly by Eqs. (2.24), and some even contain derivatives with respect to these variables. The integrands must therefore also be solved self-consistently every time they are evaluated during integration.

```
      PROGRAM critical_capacity
*     finds critical capacity of analog competitive associative memory
*     with Q = 4 and k = 2 as a function of analog gain;
*     uses Newton's method to solve for order parameters;
*     uses vector integration routine d01eaf for integrals, and c05pbf
*     to find integrand zeroes
*
*     variable definitions:
*
*     NQ        number of neurons per cluster
*     nk        number of winners per cluster
*     alpha     storage fraction or (number of patterns)/(number of clusters)
*     gamma     gain parameter in neuron transfer function
*     em        order parameter, Eq. (2.21)
*     q1        order parameter, Eq. (2.22)
*     c         order parameter, Eq. (2.23)
*     r1        order parameter, Eq. (2.26)
*     rtilde    order parameter, Eq. (2.26)
*     dfdop     Jacobian of saddle-point equations
*     dop       change in order parameters in one step of Newton's method

      IMPLICIT DOUBLEPRECISION(a-h,o-z)
      IMPLICIT INTEGER(i-n)
      CHARACTER fn_file*30
      PARAMETER (NQ = 4,NGMAX = 10,IUN = 15)
*     parameters for c05pbf and f04atf
      PARAMETER (NOP = 3,IAA = NOP)
      COMMON alpha,gamma,gnorm,nk
      COMMON /order_parameters/ em,c,q1,r1,rtilde
*     arrays for Newton's method
      DIMENSION fvec1(1:NOP),fjac1(1:NOP,1:NOP)
*     arrays for f04atf
      DIMENSION dop(1:NOP),aa(1:NOP,1:NOP),wks1(1:NOP),wks2(1:NOP)
*     arrays for input section
      DIMENSION g(1:NGMAX),amin(1:NGMAX),amax(1:NGMAX)

**************************INPUT*SECTION*****************************

*     Number of winning neurons per cluster:
      nk = 2

*     Number of values of gain:
      ng = 5

*     Gain values:
      g(1) = 5.0D00
      g(2) = 1.0D01
      g(3) = 1.5D01
      g(4) = 2.0D01
      g(5) = 1.5D01
```

```
*   Minimum and maximum alpha for each gain:

      amin(1) = 0.D00
      amin(2) = 0.D00
      amin(3) = 0.D00
      amin(4) = 0.D00
      amin(5) = 0.D00

      amax(1) = 0.5D00
      amax(2) = 0.5D00
      amax(3) = 0.5D00
      amax(4) = 0.5D00
      amax(5) = 0.5D00

*   Number of values of alpha for each gain:
      na = 4

*   Initial values of order parameters:
      em_init = 1.D00
      c_init = 2.D-01
      q1_init = 1.D00/dble(NQ - 1)

*   Other paramters:
      nt = 8
      em_crit = 5.D-01
      step = 1.D00

*   file name: critcap_newt_kwin.NQ.nk.(hi or lo).datX
      ftn_file = 'critcap_newt_kwin.4.2.dat4'

**********END*OF*INPUT*SECTION**********

      rt_tpi = dsqrt(8.D00*datan(1.D00))
      gnorm = 1.D00
      DO i = 1,NQ
         gnorm = gnorm*rt_tpi
      ENDDO

      OPEN(IUN,FILE = ftn_file,STATUS = 'new')
      WRITE(IUN,'(4I5,3F16.8)') nq,ng,na,nt,em_init,c_init,q1_init
      WRITE(IUN,'(A1)') ' '
      CLOSE(IUN)

      DO ig = 1,ng
         gamma = g(ig)
         alphamin = amin(ig)
         alphamax = amax(ig)
         em = em_init
         c = c_init
         q1 = q1_init
         em_prev = em
         c_prev = c
         q1_prev = q1
```

```
      DO ia = 1,na
*       initialize alpha and order parameters
        alpha = 5.D-01*(alphamin + alphamax)
        IF (ia.gt.1) THEN
          em = em_prev
          c = c_prev
          q1 = q1_prev
        ENDIF
        r1 = q1/(1.D00 - c)/(1.D00 - c)
        rtilde = 1.D00/(1.D00 - c)
        iflag = 0
        fsum = 1.D10
        dsum = 1.D10
        DO it = 1,nt
*         evaluate fvec1, the negative of functions to be zeroed,
*         and fjac1, the derivative of functions to be zeroed
          CALL fcn1(fvec1,fjac1)
*         find corrections to order parameters using f04atf to solve
*         the equation fjac*dop = fvec1 for dop
          ifail = 0
           CALL f04atf(fjac1,NOP,fvec1,NOP,dop,aa,IAA,wks1,wks2,ifail)
*         take one step of Newton's method
           CALL newt(fvec1,dop,fsum,dsum,step,it,iflag)
            OPEN(IUN,FILE = ftn_file,STATUS = 'old',ACCESS = 'append')
             WRITE(IUN,'(I5,9F12.8)') it,alpha,gamma,em,c,q1,r1,rtilde,fsum,dsum
          CLOSE(IUN)
          IF (iflag.ne.0) GOTO 100
        ENDDO
100     CONTINUE
         OPEN(IUN,FILE = ftn_file,STATUS = 'old',ACCESS = 'append')
         WRITE(IUN,'(A1)') ' '
        CLOSE(IUN)
        IF ((iflag.eq.1).and.(em.gt.em_crit)) THEN
*         critical storage capacity is greater than alpha
          alphamin = alpha
          em_prev = em
          c_prev = c
          q1_prev = q1
        ELSE
*         critical storage capacity is less than alpha
          alphamax = alpha
        ENDIF
      ENDDO
    ENDDO

9999 END
```

```
      DOUBLEPRECISION FUNCTION f(x)
*     k-winner transfer function
      IMPLICIT DOUBLEPRECISION(a-h,o-z)
      IMPLICIT INTEGER(i-n)
      PARAMETER (NQ = 4)
      COMMON alpha,gamma,gnorm,nk

      fnorm = 1.D00/dble(nk*(NQ - nk))
      egx = dexp(-gamma*x)
      fcn = 1.D00/(1.D00 + egx)
      f = fnorm*(dble(NQ)*fcn - dble(nk))

999   END


      DOUBLEPRECISION FUNCTION fprime(x)
*     derivative of k-winner transfer function
      IMPLICIT DOUBLEPRECISION(a-h,o-z)
      IMPLICIT INTEGER(i-n)
      PARAMETER (NQ = 4)
      COMMON alpha,gamma,gnorm,nk

      fnorm = 1.D00/dble(nk*(NQ - nk))
      egx = dexp(-gamma*x)
      fcn = gamma*egx/(1.D00 + egx)/(1.D00 + egx)
      fprime = fnorm*dble(NQ)*fcn

999   END


      DOUBLEPRECISION FUNCTION ff(b)
*     called by c05agf to calculate bias b
      IMPLICIT DOUBLEPRECISION(a-h,o-z)
      IMPLICIT INTEGER(i-n)
      PARAMETER (NQ = 4)
      COMMON alpha,gamma,gnorm,nk
      COMMON /arguments/ arg
      DIMENSION arg(1:NQ)

      sum = 0.D00
      DO iq = 1,NQ
         sum = sum + f(arg(iq) + b)
      ENDDO
      ff = sum

999   END
```

```
      SUBROUTINE  newt(fvec1,dop,fsum,dsum,step,it,iflag)
*     implements one step of Newton's method, checks for convergence
      IMPLICIT DOUBLEPRECISION(a-h,o-z)
      IMPLICIT INTEGER(i-n)
      PARAMETER (NOP = 3)
      PARAMETER (FTOL = 8.D-05,DTOL = 8.D05)
      PARAMETER (FACT1 = 1.D03,FACT2 = 1.5D01)
      COMMON alpha,gamma,gnorm,nk
      COMMON /order_parameters/ em,c,q1,r1,rtilde
      DIMENSION fvec1(1:NOP),dop(1:NOP)

      em = em + step*dop(1)
      c = c + step*dop(2)
      q1 = q1 + step*dop(3)
      r1 = q1/(1.D00 - c)/(1.D00 - c)
      rtilde = 1.D00/(1.D00 - c)

      fsumprev = fsum
      dsumprev = dsum
      fsum = 0.D00
      dsum = 0.D00

      DO iop = 1,NOP
         fsum = fsum + dabs(fvec1(iop))
         dsum = dsum + step*dabs(dop(iop))
      ENDDO

*     set iflag = 1 if Newton's method converges
      IF ((fsum.lt.FTOL).and.(fsumprev.lt.FTOL)) iflag = 1
      IF ((dsum.lt.DTOL).and.(dsumprev.lt.DTOL)) iflag = 1

*     set iflag = -1 if Newton's method diverges (alpha > alphacrit)
      IF ((em.lt.0.D00).or.(c.lt.0.D00).or.(q1.lt.0.D00))                iflag = -1
      IF ((fsum.gt.FACT1*FTOL).and.(fsum.gt.FACT2*fsumprev))   iflag = -1
      IF ((dsum.gt.FACT1*DTOL).and.(dsum.gt.FACT2*dsumprev)   iflag = -1

      RETURN
999   END


      SUBROUTINE  fcn1(fvec1,fjac1)
*     calculates saddle point equations and their derivatives
      IMPLICIT DOUBLEPRECISION(a-h,o-z)
      IMPLICIT INTEGER(i-n)
      EXTERNAL int
      COMMON alpha,gamma,gnorm,nk
      COMMON /order_parameters/ em,c,q1,r1,rtilde
      PARAMETER (NQ = 4,NOP = 3,NINT = 12)
      PARAMETER (IRCLS = 2**NQ + 2*NQ*NQ + 2*NQ + 1,MAXCLS = 200000,
     :   LENWRK = 6*NQ + 9*NINT + (NQ + NINT + 2)*(1 + MAXCLS/IRCLS),
     :   ABSREQ = 0.D00,RELREQ = 1.D-06,CUTOFF = 7.D00)
      DIMENSION fvec1(1:NOP),fjac1(1:NOP,1:NOP)
```

```
*   arrays for d01eaf
    DIMENSION a(1:NQ),b(1:NQ),wrkstr(1:LENWRK),
   :   finest(1:NINT),absest(1:NINT)

    DO iq = 1,NQ
       a(iq) = -CUTOFF
       b(iq) = CUTOFF
    ENDDO
    ifail = 1
    mincls = 1000
    u = dble(nk*(NQ - nk))/dble(NQ*(NQ - 1))
    v = dsqrt(u/alpha/r1)

*   do integrals
    CALL d01eaf(NQ,a,b,mincls,MAXCLS,NINT,int,ABSREQ,RELREQ,
   :   LENWRK,wrkstr,finest,absest,ifail)

*   update fvec1, the NEGATIVE of saddle point equations
    fvec1(1) = -em + finest(1)
    fvec1(2) = -c + v*finest(2)
    fvec1(3) = -q1 + u*finest(3)

*   update fjac1, the Jacobian of saddle point equations
    fjac1(1,1) = 1.D00 - finest(4)
    fjac1(1,2) = -finest(5)
    fjac1(1,3) = -finest(6)
    fjac1(2,1) = -v*finest(7)
    fjac1(2,2) = 1.D00 + v*(rtilde*finest(2) - finest(8))
    fjac1(2,3) = v*(5.D-01*rtilde*rtilde/r1*finest(2) - finest(9))
    fjac1(3,1) = -2.D00*u*finest(10)
    fjac1(3,2) = -2.D00*u*finest(11)
    fjac1(3,3) = 1.D00 - 2.D00*u*finest(12)

      RETURN
999   END


      SUBROUTINE  int(ndim,z,nfun,f)
*   called by d01eaf to evaluate integrals over z that appear
*   in saddle point equations and their derivatives
    IMPLICIT DOUBLEPRECISION(a-h,o-z)
    IMPLICIT INTEGER(i-n)
    EXTERNAL fcn2,ff
    COMMON alpha,gamma,gnorm,nk
    COMMON /order_parameters/ em,c,q1,r1,rtilde
    COMMON /slow_noise/ zz
    COMMON /arguments/ arg
    PARAMETER (NQ = 4,NINT = 12,NOP = 3)
*   parameters for c05pbf
    PARAMETER (NE2 = NQ,LDFJAC2 = NQ,XTOL = 1.D-06,LWA2 = 100)
*   parameter for f04abf
    PARAMETER (IAA = NQ)
*   parameters for c05agf
```

```
      PARAMETER (EPS = 1.D-10,ETA = 0.D00)
      DIMENSION f(1:NINT),z(1:NQ),zz(1:NQ),arg(1:NQ),iarg(1:NQ)
*     arrays for chain rule
      DIMENSION dfdx(1:NQ,1:NQ),dfdop(1:NQ,1:NOP),dxdop(1:NQ,1:NOP)
*     arrays for c05pbf
      DIMENSION x(1:NQ),fvec2(1:NQ),fjac2(1:NQ,1:NQ),wa2(1:LWA2)
*     arrays for f04abf
      DIMENSION wkspce(1:NQ),aa(1:IAA,1:NOP)

      u = dble(nk*(NQ - nk))/dble(NQ*(NQ - 1))
      v = dsqrt(u/alpha/r1)
      coeffu = alpha*u*(rtilde - 1.D00)
      coeffv = alpha*v*r1

*     zz = z; zz is a dummy variable that can be passed in common block
      DO iq = 1,NQ
        zz(iq) = z(iq)
      ENDDO

*     set initial conditions for zero-finding routine: do this by
*     evaluating approximate arguments, ordering them, and setting
*     x(iq) = 1/nm for the nm largest arguments, where nm > = nk
*     evaluate arguments
      DO iq = 1,NQ
        x(iq) = -1.D00/dble(NQ - nk)
        arg(iq) = alpha*v*r1*z(iq)
        IF (iq.le.nk) arg(iq) = arg(iq) + em
      ENDDO
*     order arguments; iarg(jq) is the jqth largest argument
      DO iq = 1,NQ
        iarg(iq) = iq
      ENDDO
      DO iq = 1,NQ - 1
        DO jq = 1,NQ - 1
          IF (arg(iarg(jq)).lt.arg(iarg(jq + 1))) THEN
            itemp = iarg(jq)
            iarg(jq) = iarg(jq + 1)
            iarg(jq + 1) = itemp
          ENDIF
        ENDDO
      ENDDO
*     find number nm of winning neurons
      nm = nk
      DO iq = nk + 1,NQ
        IF (arg(iarg(iq)).eq.arg(iarg(iq - 1))) THEN
          nm = nm + 1
        ELSE
          GOTO 100
        ENDIF
      ENDDO
```

```
100  CONTINUE
*    set outputs of winning neurons
     DO iq = 1,nm
        x(iarg(iq)) = dble(NQ – nm)/dble(nm*(NQ – nk))
     ENDDO

*    evaluate x, the vector containing averaged values of analog
*    neuron states
     IF ((gamma*(arg(iarg(nm)) – arg(iarg(nm + 1)))).lt.4.D01) THEN
        ifail = 1
        CALL c05pbf(fcn2,NE2,x,fvec2,fjac2,LDFJAC2,XTOL,wa2,LWA2,ifail)
     ENDIF

*    evaluate arguments of transfer function
     argsum = 0.D00
     DO iq = 1,NQ
        arg(iq) = coeffv*zz(iq) + coeffu*x(iq)
        IF (iq.le.nk) arg(iq) = arg(iq) + em
        argsum = argsum + arg(iq)
     ENDDO

*    calculate Lagrange multiplier b
     ifail = 0
     b = –argsum/dble(NQ)
     bstep = 1.D00
     CALL c05agf(b,bstep,EPS,ETA,ff,blb,bub,ifail)

*    calculate dfdx and dfdop
     fprimesum = 0.D00
     emfprimesum = 0.D00
     xfprimesum = 0.D00
     zfprimesum = 0.D00
     DO iq = 1,NQ
        fprimesum = fprimesum + fprime(arg(iq) + b)
        IF (iq.eq.nk) emfprimesum = fprimesum
        xfprimesum = xfprimesum + x(iq)*fprime(arg(iq) + b)
        zfprimesum = zfprimesum + z(iq)*fprime(arg(iq) + b)
     ENDDO
     DO iq = 1,NQ
        DO jq = 1,NQ
           dfdx(iq,jq) = coeffu*fprime(arg(iq) + b)*fprime(arg(jq) + b)/fprimesum
           IF (iq.eq.jq)
     :        dfdx(iq,jq) = dfdx(iq,jq) + 1.D00 – coeffu*fprime(arg(iq) + b)
        ENDDO
*       these are really the NEGATIVE of dfdop
        dfdop(iq,1) = –fprime(arg(iq) + b)*emfprimesum/fprimesum
        IF (iq.le.nk) dfdop(iq,1) = dfdop(iq,1) + fprime(arg(iq) + b)
        dfdop(iq,2) = fprime(arg(iq) + b)
     :     *(alpha*u*rtilde*rtilde*(x(iq) – xfprimesum/fprimesum)
     :        + alpha*v*r1*rtilde*(z(iq) – zfprimesum/fprimesum))
        dfdop(iq,3) = fprime(arg(iq) + b)
     :     *5.D–01*alpha*v*rtilde*rtilde*(z(iq) – zfprimesum/fprimesum)
     ENDDO
```

```
*    calculate dxdop from the equation dfdx*dxdop = –dfdop
     ifail = 1
      CALL f04abf(dfdx,NQ,dfdop,NQ,NQ,NOP,dxdop,NQ,wkspce,aa,IAA,ifail)

     xsum = 0.D00
     zzsum = 0.D00
     zxsum = 0.D00
     xxsum = 0.D00
     dxdmsum = 0.D00
     dxdcsum = 0.D00
     dxdqsum = 0.D00
     zdxdmsum = 0.D00
     zdxdcsum = 0.D00
     zdxdqsum = 0.D00
     xdxdmsum = 0.D00
     xdxdcsum = 0.D00
     xdxdqsum = 0.D00
     DO iq = 1,NQ
        IF (iq.le.nk) THEN
           xsum = xsum + x(iq)
           dxdmsum = dxdmsum + dxdop(iq,1)
           dxdcsum = dxdcsum + dxdop(iq,2)
           dxdqsum = dxdqsum + dxdop(iq,3)
        ENDIF
        zzsum = zzsum + z(iq)*z(iq)
        zxsum = zxsum + z(iq)*x(iq)
        xxsum = xxsum + x(iq)*x(iq)
        zdxdmsum = zdxdmsum + z(iq)*dxdop(iq,1)
        zdxdcsum = zdxdcsum + z(iq)*dxdop(iq,2)
        zdxdqsum = zdxdqsum + z(iq)*dxdop(iq,3)
        xdxdmsum = xdxdmsum + x(iq)*dxdop(iq,1)
        xdxdcsum = xdxdcsum + x(iq)*dxdop(iq,2)
        xdxdqsum = xdxdqsum + x(iq)*dxdop(iq,3)
     ENDDO

     gaussian = dexp(–5.D–01*zzsum)/gnorm
*    these are integrands of averages over 'slow noise'
     f(1) = gaussian*xsum
     f(2) = gaussian*zxsum
     f(3) = gaussian*xxsum
     f(4) = gaussian*dxdmsum
     f(5) = gaussian*dxdcsum
     f(6) = gaussian*dxdqsum
     f(7) = gaussian*zdxdmsum
     f(8) = gaussian*zdxdcsum
     f(9) = gaussian*zdxdqsum
     f(10) = gaussian*xdxdmsum
     f(11) = gaussian*xdxdcsum
     f(12) = gaussian*xdxdqsum

        RETURN
999  END
```

```
      SUBROUTINE  fcn2(ne2,x,fvec2,fjac2,ldfjac2,iflag)
*     called by c05pbf to calculate x appearing in integrands; see Eqs. (2.24), (2.25)
      IMPLICIT DOUBLEPRECISION(a-h,o-z)
      IMPLICIT INTEGER(i-n)
      EXTERNAL ff
      COMMON alpha,gamma,gnorm,nk
      COMMON /order_parameters/ em,c,q1,r1,rtilde
      COMMON /slow_noise/ zz
      COMMON /arguments/ arg
      PARAMETER (NQ = 4,EPS = 1.D-10,ETA = 0.D00)
      DIMENSION x(1:NQ),zz(1:NQ),fvec2(1:NQ),fjac2(1:NQ,1:NQ),
     :   arg(1:NQ)

      u = dble(nk*(NQ - nk))/dble(NQ*(NQ - 1))
      v = dsqrt(u/alpha/r1)
      coeffu = alpha*u*(rtilde - 1.D00)
      coeffv = alpha*v*r1

*     evaluate arguments of transfer function
      argsum = 0.D00
      DO iq = 1,NQ
         arg(iq) = coeffv*zz(iq) + coeffu*x(iq)
         IF (iq.le.nk) arg(iq) = arg(iq) + em
         argsum = argsum + arg(iq)
      ENDDO

*     calculate Lagrange multiplier b
      ifail = 0
      b = -argsum/dble(NQ)
      bstep = 1.D00
      CALL c05agf(b,bstep,EPS,ETA,ff,blb,bub,ifail)
      IF (iflag.eq.1) THEN
*        update fvec2
         DO iq = 1,NQ
            fvec2(iq) = x(iq) - f(arg(iq) + b)
         ENDDO
      ELSE
*        update fjac2
         fprimesum = 0.D00
         DO iq = 1,NQ
            fprimesum = fprimesum + fprime(arg(iq) + b)
         ENDDO
         DO iq = 1,NQ
            DO jq = 1,NQ
               fjac2(iq,jq) = coeffu*fprime(arg(iq) + b)*fprime(arg(jq) + b)/fprimesum
               IF (iq.eq.jq)
     :            fjac2(iq,jq) = fjac2(iq,jq) + 1.D00 - coeffu*fprime(arg(iq) + b)
            ENDDO
         ENDDO
      ENDIF

      RETURN
999   END
```

APPENDIX B

CONDUCTANCE OF COUPLED QUANTUM DOTS

The following FORTRAN program implements the classical charging model described in Chapter 6 for calculating the conductance $\tilde{G}_{td}$ of a triple-dot array. The program uses a Markov process to calculate $\tilde{G}_{td}$ for a number of different inter-dot capacitances C_{int} and writes the results in a file.

```
      PROGRAM triple_dot
*
*  simulates transport through triple-dot Coulomb blockade device
*  using Markov process; for details see
*
*                N.S. Bakhvalov et al., Sov. Phys. JETP 68, 581 (1989)
*
*  variable definitions:
*
*  m(i)          net number of electrons transferred into lead i
*  n(i), xn(i)   number of electrons on dot i
*  c(i)          capacitance between gate and dot i
*  cint          capacitance between dots (same for all adjacent dots)
*  cgnd          capacitance to ground (same for all dots)
*  g             conductance ratio:
*
*                      conductance of point contacts between dots
*                     ---------------------------------------------
*                     conductance of point contacts between dots and leads
*
*  v             source-drain bias voltage
*  vg            gate voltage
*  tau           kT
*
      IMPLICIT DOUBLEPRECISION(a-h,o-z)
      IMPLICIT INTEGER(i-n)
      CHARACTER*35 filename
      PARAMETER (NDOT = 3,NCMAX = 11,IMTEST = 5000)
```

```
      COMMON c,cint,cgnd,v,vg,g,tau,den
      DIMENSION c(1:NDOT),m(1:2),n(1:NDOT),gtotvec(0:NCMAX),
     :   cintvec(1:NCMAX)

**************************INPUT*SECTION*****************************

*     Gate-dot capacitances (aF):
      c(1) = 3.80D01
      c(2) = 4.30D01
      c(3) = 4.10D01

*     Total capacitance (aF):
      cgnd = 1.D03

*     Number of dot-dot capacitances:
      ncint = 7

*     Dot-dot capacitances (aF):
      cintvec(1) = 0.D00
      cintvec(2) = 3.D01
      cintvec(3) = 1.D02
      cintvec(4) = 3.D02
      cintvec(5) = 1.D03
      cintvec(6) = 3.D03
      cintvec(7) = 1.D04

*     Conductance ratio (inner/outer point contacts):
      g = 1.D00

*     Source-drain voltage (μV):
      v = 1.D01

*     Gate voltage (min,max,number) (V):
      vgmin = 0.D00
      vgmax = 2.D-02
      nvg = 400

*     Temperature (K):
      tau = 5.00D-02

*     Markov iterations per conductance data point:
      nm = 30000

*     Filename, unit (filename parameters: c1,c2,c3,cgnd,tau):
       filename = 'td.38.43.41.1000.50.dat1'
      iun = 17

************************END*OF*INPUT*SECTION**********************

      OPEN(iun,FILE = filename,STATUS = 'new')
       WRITE(iun,'(7F16.6)'),c(1),c(2),c(3),cgnd,g,v,tau
      WRITE(iun,'(11F16.6)'),(cintvec(icint),icint = 1,ncint)
```

```
      WRITE(iun,'(3I8)'),ncint,nvg,nm
      CLOSE(iun)

*     initialize random number generator
      CALL g05ccf

*     rescale temperature, source-drain voltage: 1 K = 8.63D–05 eV
      tau = 8.63D–05*(1.D01/1.6D00)*tau
      v = 1.D–06*v

      DO ivg = 0,nvg
         vg = vgmin + dble(ivg)/dble(nvg)*(vgmax – vgmin)
         DO icint = 1,ncint
            cint = cintvec(icint)
            CALL initialize(t,m,n,itflag)
            DO im = 1,nm
               CALL markov(t,m,n,itflag)
               IF (im.eq.IMTEST) THEN
                  CALL conductance(t,m,n,itflag,gtot)
                  IF (gtot.eq.0.D00) GOTO 100
               ENDIF
               IF (itflag.eq.1) GOTO 100
            ENDDO
100         CONTINUE
            CALL conductance(t,m,n,itflag,gtot)
            gtotvec(icint) = gtot
         ENDDO
*        write to data file
         OPEN(iun,FILE = filename,STATUS = 'old',ACCESS = 'append')
         WRITE(iun,'(11F20.10)'),vg,(gtotvec(icint),icint = 1,ncint)
         CLOSE(iun)
      ENDDO

9999  END


      DOUBLEPRECISION FUNCTION f(m,n)
*     triple-dot charging energy; see Appendix to Chapter 6
      IMPLICIT DOUBLEPRECISION(a–h,o–z)
      IMPLICIT INTEGER(i–n)
*     (CONST = attofarads/electron charge = 1.D–18/1.6D–19)
      PARAMETER (NDOT = 3,CONST = 1.D01/1.6D00)
      COMMON c,cint,cgnd,v,vg,g,tau,den
      DIMENSION c(1:NDOT),m(1:2),n(1:NDOT),xn(1:NDOT)

      xn(1) = dble(n(1))
      xn(2) = dble(n(2))
      xn(3) = dble(n(3))
      q4 = –cgnd*(
     :    (c(2)*c(3) + c(2)*cint + 2.D00*c(3)*cint + cint*cint
     :    + c(2)*cgnd + c(3)*cgnd + 3.D00*cint*cgnd + cgnd*cgnd)*xn(1)
     :    + (c(3)*cint + cint*cint + cint*cgnd)*xn(2)
```

```
     :    + cint*cint*xn(3)
     :    )/den
      q5 = -cgnd*(
     :   (c(3)*cint + cint*cint + cint*cgnd)*xn(1)
     :    + (c(1)*c(3) + c(1)*cint + c(3)*cint + cint*cint
     :    + c(1)*cgnd + c(3)*cgnd + 2.D00*cint*cgnd + cgnd*cgnd)*xn(2)
     :    + (c(1)*cint + cint*cint + cint*cgnd)*xn(3)
     :    )/den
      q6 = -cgnd*(
     :    cint*cint*xn(1)
     :    + (c(1)*cint + cint*cint + cint*cgnd)*xn(2)
     :    + (c(1)*c(2) + 2.D00*c(1)*cint + c(2)*cint + cint*cint
     :    + c(1)*cgnd + c(2)*cgnd + 3.D00*cint*cgnd + cgnd*cgnd)*xn(3)
     :    )/den
      q7 = -cint*(
     :    (c(2)*c(3) + c(2)*cint + c(3)*cint + c(2)*cgnd + c(3)*cgnd
     :    + 2.D00*cint*cgnd + cgnd*cgnd)*xn(1)
     :    - (c(1)*c(3) + c(1)*cint + c(1)*cgnd + c(3)*cgnd
     :    + cint*cgnd + cgnd*cgnd)*xn(2)
     :    - (c(1)*cint + cint*cgnd)*xn(3)
     :    )/den
      q8 = cint*(
     :    (c(3)*cint + cint*cgnd)*xn(1)
     :    + (c(1)*c(3) + c(3)*cint + c(1)*cgnd + c(3)*cgnd
     :    + cint*cgnd + cgnd*cgnd)*xn(2)
     :    - (c(1)*c(2) + c(1)*cint + c(2)*cint + c(1)*cgnd + c(2)*cgnd
     :    + 2.D00*cint*cgnd + cgnd*cgnd)*xn(3)
     :    )/den

      f = (q4 + q7 + xn(1))*(q4 + q7 + xn(1))/2.D00/c(1)
     :    + (q5 - q7 - q8 + xn(2))*(q5 - q7 - q8 + xn(2))/2.D00/c(2)
     :    + (q6 + q8 + xn(3))*(q6 + q8 + xn(3))/2.D00/c(3)
     :    - vg*CONST*(q4 + q5 + q6 + xn(1) + xn(2) + xn(3))
     :    + 5.D-01*v*CONST*dble(m(1) - m(2))
      IF (cgnd.ne.0.D00) THEN
         f = f + (q4*q4 + q5*q5 + q6*q6)/2.D00/cgnd
      ENDIF
      IF (cint.ne.0.D00) THEN
         f = f + (q7*q7 + q8*q8)/2.D00/cint
      ENDIF

999   END


      DOUBLEPRECISION FUNCTION phi(x)
*     boltzmann probability for calculating tunneling rates
      IMPLICIT DOUBLEPRECISION(a-h,o-z)
      IMPLICIT INTEGER(i-n)
      PARAMETER (NDOT = 3,ARGLIM = 5.D01)
      COMMON c,cint,cgnd,v,vg,g,tau,den
      DIMENSION c(1:NDOT)
```

```
      arg = x/tau
      IF (dabs(arg).gt.ARGLIM) THEN
         phi = dmax1( -x,0.D00)
      ELSEIF (arg.eq.0.D00) THEN
         phi = tau
      ELSE
         phi = -x/(1 - dexp(arg))
      ENDIF

999   END


      SUBROUTINE initialize(t,m,n,itflag)
*     initializes parameters; calculates denominator used in charging
*     energy; finds integer values of n that minimize charging energy
      IMPLICIT DOUBLEPRECISION(a-h,o-z)
      IMPLICIT INTEGER(i-n)
      PARAMETER (NDOT = 3,NRES = 8,IBOUND = 1)
      PARAMETER (LIW = NDOT + 2,LW = NDOT*(NDOT - 1)/2 + 12*NDOT)
      PARAMETER (CONST = 1.D01/1.6D00)
      COMMON c,cint,cgnd,v,vg,g,tau,den
      DIMENSION c(1:NDOT),m(1:2),n(1:NDOT),xn(1:NDOT)
*     arrays for e04jaf; usually not used
***DIMENSION bl(1:NDOT),bu(1:NDOT),iw(1:LIW),w(1:LW)

*     set t = 0,m = 0,itflag = 0
      t = 0.D00
      DO i = 1,NDOT
         m(i) = 0
      ENDDO
      itflag = 0

*     calculate denominator used in charging energy
      den = cint*cint*(c(1) + c(2) + c(3))
     :    + cint*(c(1)*c(2) + c(2)*c(3) + 2.D00*c(1)*c(3))
     :    + c(1)*c(2)*c(3)
     :    + cgnd*cgnd*cgnd
     :    + cgnd*cgnd*(c(1) + c(2) + c(3) + 4.D00*cint)
     :    + cgnd*(c(1)*c(2) + c(1)*c(3) + c(2)*c(3)
     :       + 3.D00*c(1)*cint + 2.D00*c(2)*cint
     :       + 3.D00*c(3)*cint + 3.D00*cint*cint)


*     calculate initial estimates of numbers xn of electrons on each dot

      DO idot = 1,NDOT
         xn(idot) = CONST*vg*c(idot)
      ENDDO

*     OPTIONAL: call e04jaf to refine initial values of xn that minimize charging
*     energy; in practice this call is unnecessary as initial estimates are quite good
***ifail = 1
```

```
***CALL e04jaf(NDOT,IBOUND,bl,bu,xn,f,iw,LIW,w,LW,ifail)

   DO idot = 1,NDOT
      n(idot) = idnint(xn(idot))
   ENDDO

     RETURN
999  END


     SUBROUTINE  conductance(t,m,n,itflag,gtot)
*    calculates current and conductance assuming 1 MΩ resistance
*    of tunnel junctions 1 and 3; conductance is in units of e^2/h
     IMPLICIT DOUBLEPRECISION(a–h,o–z)
     IMPLICIT INTEGER(i–n)
     PARAMETER (NDOT = 3)
     COMMON c,cint,cgnd,v,vg,g,tau,den
     DIMENSION c(1:NDOT),m(1:2),n(1:NDOT)

     IF (itflag.eq.1) THEN
        gtot = 0.D00
     ELSE
        xi = 1.D–06*dble(m(2) – m(1))/t
        gtot = dmax1(2.5812D04*xi/v,0.D00)
     ENDIF

     RETURN
999  END


     SUBROUTINE  markov(t,m,n,itflag)
*    for given m and n, calculates the probability of all
*    2*(NDOT + 1) tunneling transitions; inter-dot tunneling
*    probabilities are scaled by conductance ratio g
     IMPLICIT DOUBLEPRECISION(a–h,o–z)
     IMPLICIT INTEGER(i–n)
     PARAMETER (NDOT = 3,TMAX = 1.D40)
     COMMON c,cint,cgnd,v,vg,g,tau,den
     DIMENSION c(1:NDOT),m(1:2),mm(1:2),n(1:NDOT),nn(1:NDOT),
    :  p(1:2*(NDOT + 1))

     f0 = f(m,n)

*    tunneling from left lead to dot1
     mm(1) = m(1) – 1
     mm(2) = m(2)
     nn(1) = n(1) + 1
     nn(2) = n(2)
     nn(3) = n(3)
     p(1) = phi(f(mm,nn) – f0)

*    tunneling from dot1 to left lead
     mm(1) = m(1) + 1
```

```
      mm(2) = m(2)
      nn(1) = n(1) - 1
      nn(2) = n(2)
      nn(3) = n(3)
      p(2) = phi(f(mm,nn) - f0)

*     tunneling from dot1 to dot2
      mm(1) = m(1)
      mm(2) = m(2)
      nn(1) = n(1) - 1
      nn(2) = n(2) + 1
      nn(3) = n(3)
      p(3) = g*phi(f(mm,nn) - f0)

*     tunneling from dot2 to dot1
      mm(1) = m(1)
      mm(2) = m(2)
      nn(1) = n(1) + 1
      nn(2) = n(2) - 1
      nn(3) = n(3)
      p(4) = g*phi(f(mm,nn) - f0)

*     tunneling from dot2 to dot3
      mm(1) = m(1)
      mm(2) = m(2)
      nn(1) = n(1)
      nn(2) = n(2) - 1
      nn(3) = n(3) + 1
      p(5) = g*phi(f(mm,nn) - f0)

*     tunneling from dot3 to dot2
      mm(1) = m(1)
      mm(2) = m(2)
      nn(1) = n(1)
      nn(2) = n(2) + 1
      nn(3) = n(3) - 1
      p(6) = g*phi(f(mm,nn) - f0)

*     tunneling from dot3 to right lead
      mm(1) = m(1)
      mm(2) = m(2) + 1
      nn(1) = n(1)
      nn(2) = n(2)
      nn(3) = n(3) - 1
      p(7) = phi(f(mm,nn) - f0)

*     tunneling from right lead to dot3
      mm(1) = m(1)
      mm(2) = m(2) - 1
      nn(1) = n(1)
      nn(2) = n(2)
      nn(3) = n(3) + 1
```

```
      p(8) = phi(f(mm,nn) – f0)

*     increment time (see Chapter 6 and Eq. 18 of Bakhvalov et al.)
      psum = 0.D00
      DO ip = 1,2*(NDOT + 1)
         psum = psum + p(ip)
      ENDDO
      dt = –dlog(5.D–01)/psum
      t = t + dt
      IF (t.gt.TMAX) itflag = 1

*     normalize probabilities
      DO ip = 1,2*(NDOT + 1)
         p(ip) = p(ip)/psum
      ENDDO

*     generate a random number and choose a
*     tunneling event

      xx = g05caf(xx)

*     tunneling from left lead to dot1
      psum = p(1)
      IF (xx.lt.psum) THEN
         m(1) = m(1) – 1
         n(1) = n(1) + 1
         GOTO 100
      ENDIF

*     tunneling from dot1 to left lead
      psum = psum + p(2)
      IF (xx.lt.psum) THEN
         m(1) = m(1) + 1
         n(1) = n(1) – 1
         GOTO 100
      ENDIF

*     tunneling from dot1 to dot2
      psum = psum + p(3)
      IF (xx.lt.psum) THEN
         n(1) = n(1) – 1
         n(2) = n(2) + 1
         GOTO 100
      ENDIF

*     tunneling from dot2 to dot1
      psum = psum + p(4)
      IF (xx.lt.psum) THEN
         n(1) = n(1) + 1
         n(2) = n(2) – 1
         GOTO 100
      ENDIF
```

```
*    tunneling from dot2 to dot3
     psum = psum + p(5)
     IF (xx.lt.psum) THEN
        n(2) = n(2) - 1
        n(3) = n(3) + 1
        GOTO 100
     ENDIF

*    tunneling from dot3 to dot2
     psum = psum + p(6)
     IF (xx.lt.psum) THEN
        n(2) = n(2) + 1
        n(3) = n(3) - 1
        GOTO 100
     ENDIF

*    tunneling from dot3 to right lead
     psum = psum + p(7)
     IF (xx.lt.psum) THEN
        m(2) = m(2) + 1
        n(3) = n(3) - 1
        GOTO 100
     ENDIF

*    tunneling from right lead to dot3
     m(2) = m(2) - 1
     n(3) = n(3) + 1

100  CONTINUE
     RETURN
999  END
```

REFERENCES

Adler, D. (1968). "Insulating and metallic states in transition metal oxides," in *Solid State Physics, Vol. 21*, ed. F. Seitz, D. Turnbull, and H. Ehrenreich (Academic Press, New York), 1.

Alspector, J., B. Gupta, and R. B. Allen (1989). "Performance of a stochastic learning microchip," in *Proc. Conf. Neural Information Processing Systems, Denver, CO 1987*, ed. D. S. Touretsky (Morgan Kaufmann, San Mateo), 748.

Alspector, J., J. W. Gannett, S. Haber, M. B. Parker, and R. Chu (1991). "A VLSI-efficient technique for generating multiple uncorrelated noise sources and its application to stochastic neural networks," *IEEE Trans. Circuits Sys.* **38**, 109.

Altshuler, B. L., P. A. Lee, and R. A. Webb, eds. (1991). *Mesoscopic Phenomena in Solids* (North-Holland Elsevier Science Publishers B. V., Amsterdam).

Amari, S.-I. (1972). "Learning patterns and pattern sequences by self-organizing nets of threshold elements," *IEEE Trans. Comput.* **C-11**, 1197.

Amit, D. J. (1989). *Modeling Brain Function: The World of Attractor Neural Networks* (Cambridge University Press, Cambridge).

Amit, D. J., H. Gutfreund, and H. Sompolinsky (1985a). "Spin-glass models of neural networks," *Phys. Rev. A* **32**, 1007.

Amit, D. J., H. Gutfreund, and H. Sompolinsky (1985b). "Storing infinite numbers of patterns in a spin-glass model of neural networks," *Phys. Rev. Lett.* **55**, 1530.

Amit, D. J., H. Gutfreund, and H. Sompolinsky (1987). "Statistical mechanics of neural networks near saturation," *Ann. Phys. NY* **173**, 30.

Amit, D. J., and A. Treves (1989). "Associative memory neural network with low temporal spiking rates," *Proc. Natl. Acad. Sci.* **86**, 7871.

Anderson, D. Z. (1990). "Competitive and cooperative dynamics in nonlinear optical circuits," in *An Introduction to Neural and Electronic Networks*, ed. S. F. Zornetzer, J. L. Davis, and C. Lau (Academic Press, New York), 349.

Anderson, J. A. (1968). "A memory storage model utilizing spatial correlation functions," *Kybernetik* **5**, 113.

Andreou, A. G., K. A. Boahen, P. O. Pouliquen, A. Pavasovic, R. E. Jenkins, and K. Strohbehn (1991). "Current-mode subthreshold MOS circuits for analog VLSI neural systems," *IEEE Trans. Neural Networks* **2**, 205.

Ashcroft, N. W., and N. D. Mermin (1976). *Solid State Physics* (Saunders College, Philadelphia), 689.

Ashoori, R. C., H. L. Stormer, J. S. Weiner, L. N. Pfeiffer, K. W. Baldwin, and K. W. West (1993). "*N*-electron ground state energies of a quantum dot in magnetic field," *Phys. Rev. Lett.* **71**, 613.

Averin, D. V., and K. K. Likharev (1991). "Single electronics: a correlated transfer of single electrons and Cooper pairs in systems of small tunnel junctions," in *Mesoscopic Phenomena in Solids*, ed. B. L. Altshuler, P. A. Lee, and R. A. Webb (North-Holland Elsevier Science Publishers B. V., Amsterdam), 173.

Averin, D. V., and K. K. Likharev (1992). "Possible applications of the single charge tunneling," in *Single Charge Tunneling*, ed. H. Grabert and M. H. Devoret (Plenum Press, New York), 311.

Averin, D. V., A. N. Korotkov, and K. K. Likharev (1991). "Theory of single-electron charging of quantum wells and dots," *Phys. Rev. B* **44**, 6199.

Bakhvalov, N. S., G. S. Kazacha, K. K. Likharev, and S. I. Serdyukova (1989). "Single-electron solitons in one-dimensional tunnel structures," *Sov. Phys. JETP* **68**, 581.

Basharan, G., Y. Fu, and P. W. Anderson (1986). "On the statistical mechanics of the traveling salesman problem," *J. Stat. Phys.* **45**, 1.

Baskey, J. H. (1994). "Transport and capacitance measurements of electron multilayers in wide parabolic quantum wells," Ph. D. thesis, Harvard University.

Bate, R. L (1988). "The quantum-effect device: tomorrow's transistor?" *Scientific American*, March 1988, 96.

Bauer, H.-U., and T. Geisel (1990). "Nonlinear dynamics of feedback multilayer perceptrons," *Phys. Rev. A* **42**, 2401.

Beenakker, C. W. J. (1991). "Theory of Coulomb-blockade oscillations in the conductance of a quantum dot," *Phys. Rev. B* **44**, 1646.

Beenakker, C. W. J., and H. van Houten (1991). "Quantum transport in semiconductor nanostructures," in *Solid State Physics, Vol. 44*, ed. H. Ehrenreich and D. Turnbull (Academic Press, San Diego), 1.

Bennett, A. (1989). "An exactly solvable model for competitive networks," *J. Phys. A* **22**, 2047.

Berry, M. J. (1994). "Mesoscopic transport and quantum chaos in ballistic quantum billiards," Ph. D. thesis, Harvard University.

Bilbro, G., and W. Snyder (1989). "Range image restoration using mean field annealing," in *Advances in Neural Information Processing Systems*, ed. D. S. Touretzky (Kaufmann, San Mateo), 594.

Binder, K., and A. P. Young (1986). "Spin glasses: experimental facts, theoretical concepts, and open questions," *Rev. Mod. Phys.* **58**, 801.

Birge, R. R. (1994). "Protein-based three-dimensional memory," *American Scientist* **82**, 348.

Blake, A., and A. Zisserman (1987). *Visual Reconstruction.* MIT Press, Cambridge.

Bock, K., and H. L. Hartnagel (1993). "Proposal for the concept of ultradense integrated memories based on Coulomb-blockade at room temperature," *Electr. Lett.* **29**, 2228.

Bohr, H., J. Bohr, S. Brunak, R. Coterill, H. Fredholm, B. Lautrop, and S. Petersen (1990). "A novel approach to the prediction of the three-dimensional structures of protein backbones by neural networks," *FEBS Lett.* **261**, 43.

Bollé, D., and P. Dupont (1990). "On Potts-glass neural networks with biased patterns," in *Statistical Mechanics of Neural Networks,* ed. L. Garrido (Springer Verlag, New York), 365.

Bollé, D., P. Dupont, and J. Huyghebaert (1992a). "Thermodynamic properties of the Q-state Potts-glass neural network," *Phys. Rev. A* **45**, 4194.

Bollé, D., P. Dupont, and J. Huyghebaert (1992b). "On the phase diagram of the Q-state Potts-glass neural network," *Physica A* **185**, 363.

Bollé, D., P. Dupont, and J. van Mourik (1991). "Stability properties of Potts neural networks with biased patterns and low loading," *J. Phys. A* **24**, 1065.

Bollé, D., P. Dupont, and B. Vinck (1992). "On the overlap dynamics of multi-state neural networks with a finite number of patterns," *J. Phys. A* **25**, 2859.

Bollé, D., and F. Mallezie (1989). "Image evolution in Potts-glass neural networks," *J. Phys. A* **22**, 4409.

Bradley, D. (1993). "Will future computers be all wet?", *Science* **259**, 890.

Bray, A. J., and M. A. Moore (1979). "Evidence for massless modes in the 'solvable model' of a spin glass," *J. Phys. C* **12**, L441.

Bray, A. J., and M. A. Moore (1980). "Metastable states in spin glasses," *J. Phys. C* **13**, L469.

Bray, A. J., and M. A. Moore (1981). "Metastable states in spin glasses with short-ranged interactions," *J. Phys. C* **14**, 1313.

Bray, A. J., H. Sompolinsky, and C. Yu (1986). "On the 'naive' mean-field equations for spin glasses," *J. Phys. C* **19**, 6389.

Bressloff, P. C., and J. G. Taylor (1992). "Temporal sequence storage capacity of time-summating neural networks," *J. Phys. A* **25**, 833.

Bridle, J. S., and S. J. Cox (1991). "RecNorm: simultaneous normalization and classification applied to speech recognition," in *Advances in Neural Information Processing Systems 3*, ed. R. P. Lippmann, J. E. Moody, and D. S. Touretsky (Morgan Kaufmann, San Mateo), 234.

Brout, R. (1959). "Statistical mechanical theory of a random ferromagnetic system," *Phys. Rev.* **115**, 824.

Bruce, A. D., E. J. Gardner, and D. J. Wallace (1987). "Dynamics and statistical mechanics of the Hopfield model," *J. Phys. A* **20**, 2909.

Bruus, H., and A. D. Stone (1994). "Quantum chaos in a deformable billiard: applications to quantum dots," preprint.

Bryant, G. W. (1993). "Electrons in coupled vertical quantum dots: interdot tunneling and Coulomb correlation," *Phys. Rev. B* **48**, 8024.

Buhmann, J. (1989). "Oscillations and low firing rates in associative memory neural networks," *Phys. Rev. A* **40**, 4145.

Buot, F. A. (1993). "Mesoscopic physics and nanoelectronics: nanoscience and nanotechnology," *Phys. Rep.* **234**, 73.

Burgess, N., and M. A. Moore (1989). "Cost distributions in large combinatorial optimisation problems," *J. Phys. A* **22**, 4599.

Büttiker, M. (1986). "Four-terminal phase-coherent conductance," *Phys. Rev. Lett.* **57**, 1761.

Büttiker, M. (1988). "Coherent and sequential tunneling in series barriers," *IBM J. Res. Develop.* **32**, 63.

Büttiker, M., Y. Imry, R. Landauer, and S. Pinhas (1985). "Generalized many-channel conductance formula with application to small rings," *Phys. Rev. B* **31**, 6027.

Caianiello, E. R. (1961). "Outline of a theory of thought and thinking machines," *J. Theor. Biol.* **1**, 204.

Capasso, F., S. Sen, F. Beltram, and A. Y. Cho (1989). "Resonant tunneling diodes and their applications," in *Submicron Integrated Circuits*, ed. R. K. Watts (Wiley-Interscience, Chichester, UK), 204.

Castaño, E., G. Kirczenow, and S. E. Ulloa (1990). "Nonlinear transport in ballistic quantum chains," *Phys. Rev. B* **42**, 3753.

Chandrasekhar, V., Z. Ovadyahu, and R. A. Webb (1991). "Single-electron charging effects in insulating wires," *Phys. Rev. Lett.* **67**, 2862.

Cleland, A. N., D. Esteve, C. Urbina, and M. H. Devoret (1992). "Very low noise photodetector based on the single electron transistor," *Appl. Phys. Lett.* **61**, 2820.

Cleland, A. N., J. M. Schmidt, and J. Clarke (1992). "Influence of the environment on the Coulomb blockade in submicrometer normal-metal tunnel junctions," *Phys. Rev. B* **45**, 2950.

Cohen, M. A., and S. Grossberg (1983). "Absolute stability of global pattern formation and parallel memory storage by competitive neural networks," *IEEE Trans.* **SMC-13**, 815.

Cole, B. C. (1993). "Scaling the limits: can ICs keep packing in transistors?" *Electronic Engineering Times*, 15 February, 1.

Coolen, A. C. C. (1990). "Ising-spin neural networks with spatial structure," in *Statistical Mechanics of Neural Networks,* ed. L. Garrido (Springer Verlag, New York), 381.

Cragg, B. G., and H. N. V. Temperley (1954). "The organization of neurones: a cooperative analogy," *Electroenceph. Clin. Neuro.* **6**, 85.

Crisanti, A., D. J. Amit, and H. Gutfreund (1986). "Saturation level of the Hopfield model for neural networks," *Europhys. Lett.* **2**, 337.

Crisanti, A., and H. Sompolinsky (1987). "Dynamics of spin systems with randomly asymmetric bonds: Langevin dynamics and a spherical model," *Phys. Rev. A* **36**, 4922.

Datta, S., and M. J. McLennan (1990). "Quantum transport in ultrasmall electronic devices," *Rep. Prog. Phys.* **53**, 1003.

De Dominicis, C. M. Gabay, T. Garel, and H. Orland (1980). "White and weighted averages over solutions of Thouless Anderson Palmer equations for the Sherrington Kirkpatrick spin glass," *J. Physique* **41**, 923.

Dehaene, S., J. P. Changeux, and J. P. Nadal (1987). "Neural networks that learn temporal sequences by selection," *Proc. Nat. Acad. Sci. USA* **84**, 2727.

Diederich, S., and M. Opper, (1987). "Learning of correlated patterns in spin-glass networks by local learning rules," *Phys. Rev. Lett.* **58**, 949.

Domany, E., W. Kinzel, and R. Meir (1989). "Layered neural networks," *J. Phys. A* **22**, 2081.

Domany, E., J. L. van Hemmen, and K. Schulten, eds. (1991). *Models of Neural Networks* (Springer Verlag, New York).

Dowling, J. E. (1987). *The Retina: An Approachable Part of the Brain* (Harvard University Press, Cambridge, Mass.).

Dresselhaus, P. D., L. Ji, S. Han, J. E. Lukens, and K. K. Likharev (1994). "Measurement of single electron lifetimes in a multijunction trap," *Phys. Rev. Lett.* **72**, 3226.

Drexler, K. E. (1992). *Nanosystems: Molecular Machinery, Manufacturing, and Computation* (John Wiley &Sons, New York).

Durbin, R., and D. Willshaw (1987). "An analogue approach to the travelling salesman problem using an elastic net method," *Nature* **326**, 689.

The Economist (1993). "Pent-up demand," 13 November, 100.

Feldman, J. A., and D. H. Ballard (1982). "Connectionist models and their properties," *Cog. Sci.* **6**, 205.

Ferrari, P. A., S. Martinez, and P. Picco (1992). "A lower bound for the memory capacity in the Potts-Hopfield model," *J. Stat. Phys.* **66**, 1643.

Fischer, K. H., and J. A. Hertz (1991). *Spin Glasses* (Cambridge University Press, Cambridge).

Fisher, W. A., R. J. Fujimoto, and R. C. Smithson (1991). "A programmable neural network processor," *IEEE Trans. Neural Networks* **2**, 222.

Fogelman Soulie, F., C. Mejia, E. Goles, and S. Martinez (1989). "Energy functions in neural networks with continuous local functions," *Complex Sys.* **3**, 269.

Fontanari, J. F., and R. Köberle (1987). "Information storage and retrieval in synchronous neural networks," *Phys. Rev. A* **36**, 2475.

Fontanari, J. F., and R. Köberle (1988a). "Information processing in synchronous neural networks," *J. Physique* **49**, 13.

Fontanari, J. F., and R. Köberle (1988b). "Neural networks with transparent memory," *J. Phys. A* **21**, L259.

Forrest, S. (1990). "Emergent computation: self-organizing, collective, and cooperative phenomena in natural and artificial computing networks," *Physica D* **42**, 1.

Foxman, E. B., P. L. McEuen, U. Meirav, N. S. Wingreen, Y. Meir, P. A. Belk, N. R. Belk, and M. A. Kastner (1993). "Effects of quantum levels on transport through a Coulomb island," *Phys. Rev. B* **47**, 10020.

Fu, Y., and P. W. Anderson (1986). "Application of statistical mechanics to NP-complete problems in combinatorial optimization," *J. Phys A* **19**, 1605.

Fukai, T. (1990). "Metastable states of neural networks incorporating the Dale hypothesis," *J. Phys. A* **23**, 249.

Fukai, T., and M. Shiino (1990). "Large suppression of spurious states in neural networks of nonlinear analog neurons," *Phys. Rev. A* **42**, 7459.

Fulton, T. A., and G. J. Dolan (1987). "Observation of single-electron charging in small tunnel junctions," *Phys. Rev. Lett.* **59**, 109.

Fukushima, K. (1980). "Neocognitron: a self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position," *Biol. Cyber.* **36**, 193.

Fukushima, K. (1988). "Neocognitron: a hierarchical neural network capable of visual pattern recognition," *Neural Networks* **1**, 199.

Gardner, E. J. (1986). "Structure of metastable states in the Hopfield model," *J. Phys. A* **19**, L1047.

Gardner, E. J. (1988). "The space of interactions in neural network models," *J. Phys. A* **21**, 257.

Garrido, L., ed. (1990). *Statistical Mechanics of Neural Networks* (Springer Verlag, New York).

Geiger, D., and A. Yuille (1991). "A common framework for image segmentation," *Int. J. Computer Vision* **6**, 227.

Geman, S. (1980). "A limit theorem for the norm of random matrices," *Ann. Prob.* **8**, 252.

Geman, S., and D. Geman (1984). *IEEE Trans. Pattern Analysis and Machine Intelligence* **6**, 721.

Giaever, I., and H. R. Zeller (1968). "Superconductivity of small tin particles measured by tunneling," *Phys. Rev. Lett.* **20**, 1504.

Glattli, D. C., C. Pasquier, U. Meirav, F. I. B. Williams, Y. Jin, and B. Etienne (1991). "Co-tunneling of the charge through a 2-D electron island," *Z. Phys. B – Condensed Matter* **85**, 375.

Glazman, L. I., and V. Chandrasekhar (1992). "Coulomb blockade oscillations in a double-dot system," *Europhys. Lett.* **19**, 623.

Glazman, L. I., and R. I. Shekhter (1989). "Coulomb oscillations of the conductance in a laterally confined heterostructure," *J. Phys.: Condens. Matter* **1**, 5811.

Golden, R. M. (1986). "The 'brain-state-in-a-box' neural model is a gradient descent algorithm," *J. Math. Psych.* **30**, 73.

Goles, E., and G. Y. Vichniac (1986). "Lyapunov functions for parallel neural networks," in *Neural Networks for Computing, AIP Conf. Proc. 151*, ed. J. S. Denker (American Institute of Physics, New York), 165.

Goles-Chacc, E., F. Fogelman-Soulie, and D. Pellegrin (1985). "Decreasing energy functions as a tool for studying threshold networks," *Disc. Appl. Math.* **12**, 261.

Golomb, D., N. Rubin, and H. Sompolinsky (1990). "Willshaw model: associative memory with sparse coding and low firing rates," *Phys. Rev. A* **41**, 1843.

Grabert, H., and M. H. Devoret, eds. (1992). *Single Charge Tunneling* (Plenum Press, New York).

Graf, H. P., L. D. Jackel, R. E. Howard, B. Straughn, J.S. Denker, W. Hubbard, D. M. Tennant, and D.Schwartz (1986). "VLSI implementation of a neural network memory with several hundreds of neurons," in *Neural Networks for Computing, AIP Conf. Proc. 151*, ed. J. S. Denker (American Institute of Physics, New York), 182.

Gross, D. J., I. Kanter, and H. Sompolinsky (1984). "Mean-field theory of the Potts glass," *Phys. Rev. Lett.* **55**, 304.

Grossberg, S. (1967). "Nonlinear difference-differential equations in prediction and learning theory," *Proc. Natl. Acad. Sci. USA* **58**, 1329.

Grossberg, S. (1968). "Some nonlinear networks capable of learning a spatial pattern of arbitrary complexity," *Proc. Natl. Acad. Sci. USA* **59**, 368.

Grossberg, S. (1987). "Competitive learning: from interactive activation to adaptive resonance," *Cog. Sci.* **11**, 23.

Gruner, G., and A. Zawadowski (1974). "Magnetic impurities in non-magnetic materials," *Rep Prog. Phys.* **37**, 1497.

Gutfreund, H., J. D. Reger, and A. P. Young (1988). "The nature of attractors in an asymmetric spin glass with deterministic dynamics," *J. Phys. A* **21**, 2775.

Guyon, I., L. Personnaz, J. P. Nadal, and G. Dreyfus (1988). "Storage and retrieval of complex sequences in neural networks," *Phys. Rev. A* **38**, 6365.

Haanappel, E. G., and D. van der Marel (1989). "Conductance oscillations in two-dimensional Sharvin point contacts," *Phys. Rev. B* **39**, 5484.

Hammerstrom, D. (1993). "Neural networks at work," IEEE Spectrum, June, 26.

Harris, J. J., J. A. Pals, and R. Woltjer (1989). "Electronic transport in low-dimensional structures," *Rep. Prog. Phys.* **52**, 1217.

Hassoun, M. H., ed. (1993). *Associative Neural Memories: Theory and Implementation* (New York, Oxford University Press).

Haug, R. J., J. M. Hong, and K. Y. Lee (1992). "Electron transport through one quantum dot and through a string of quantum dots," *Surf. Sci.* **263**, 415.

Hebb, D. O. (1949). *The Organization of Behavior* (Wiley, New York).

Hentschel, H. G. E., and A. Fine (1989). "Statistical mechanics of stereoscopic vision," *Phys. Rev. A* **40**, 3983.

Hertz, J. A., A. S. Krogh, and R. G. Palmer (1991). *Introduction to the Theory of Neural Computation* (Addison-Wesley, Reading, Mass.).

Herz, A. V. M. (1991). "Global analysis of parallel analog networks with retarded feedback," *Phys. Rev. A* **44**, 1415.

Herz, A. V. M., Z. Li, and J. L. van Hemmen (1991). "Statistical mechanics of temporal association in neural networks with transmission delays," *Phys. Rev. Lett.* **66**, 1370.

Holler, M., S. Tam, H. Castro, and R. Benson (1989). "An electrically trainable artificial neural network (ETANN) with 10240 'floating gate' synapses," in *Proc. Int. Joint Conf. Neural Networks, Washington DC,* 191.

Hopfield, J. J. (1982). "Neural networks and physical systems with emergent collective computational abilities," *Proc. Nat. Acad. Sci. USA* **79**, 2554.

Hopfield, J. J. (1984). "Neurons with graded response have collective computational properties like those of two-state neurons," *Proc. Nat. Acad. Sci. USA* **81**, 3088.

Hopfield, J. J., and D. W. Tank (1985). "'Neural' computation of decisions in optimization problems," *Biol. Cybern.* **52**, 141.

Hopfield, J. J., and D. W. Tank (1986). "Computing with neural circuits: a model," *Science* **233**, 625.

Hopkins, P. F. (1990). "Electron transport in wide parabolic gallium arsenide/aluminum gallium arsenide wells," Ph. D. thesis, Harvard University.

Hwang, S. W., D. C. Tsui, and M. Shayegan (1994). "Charge transport in a low-disorder, low-density one-dimensional electron system," *Phys. Rev. B* **49**, 16441.

Ingold, G.-L., and Y. V. Nazarov (1992). "Charge Tunneling Rates in Ultrasmall Junctions," in *Single Charge Tunneling*, ed. H. Grabert and M. H. Devoret (Plenum Press, New York), 21.

Jalabert, R. A., A. D. Stone, and Y. Alhassid (1992). "Statistical theory of Coulomb blockade oscillations: quantum chaos in quantum dots," *Phys. Rev. Lett.* **68**, 3468.

Johnson, A. T., L. P. Kouwenhoven, W. de Jong, N. C. van der Vaart, and C. J. P. M. Harmans (1992). "Zero-dimensional states and single electron charging in quantum dots," *Phys. Rev. Lett.* **69**, 1592.

Johnson, L. G., and S. M. S. Jalaleddine (1991). "MOS implementation of winner-take-all network with application to content-addressable memory," *Electronics Lett.* **27**, 957.

Johnson, N. F., and M. C. Payne (1993). "Microscopic theory of periodic conductance oscillations in narrow channels," *Phys. Rev. Lett.* **70**, 1513.

Johnson, R. C. (1993). "Mead's μ musings: VLSI guru sees 0.1 micron coming," *Electronic Engineering Times*, 5 April, 1.

Kanter, I. (1988). "Potts-glass models of neural networks," *Phys. Rev. A* **37**, 2739.

Kanter, I., and H. Sompolinsky (1987). "Associative recall of memory without errors," *Phys. Rev. A* **35**, 380.

Kastner, M. (1992). "The single-electron transistor," *Rev. Mod. Phys.* **64**, 849.

Kepler, T. B. (1991). "Domains of attraction and the density of static metastable states in single-pattern iterated neural networks," *J. Phys. A* **24**, 1083.

Keyes, R. M. (1991). "Limits and challenges in electronics," *Contemporary Phys.* **32**, 403.

Keyes, R. M. (1992). "The future of solid-state electronics," *Physics Today*, August, 42.

Keyes, R. M. (1993). "The future of the transistor," *Scientific American*, June, 70.

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

Kleinfeld, D. (1986). "Sequential state generation by model neural networks," *Proc. Natl. Acad. Sci. USA* **83**, 9469.

Klimeck, G., G. Chen, and S. Datta (1994). "Conductance spectroscopy in coupled quantum dots," *Phys. Rev. B.* **50**, 2316.

Koch, C., J. Marroquin, and A. Yuille (1986). "Analog 'neuronal' networks in early vision," *Proc. Nat. Acad. Sci. USA* **83**, 4263.

Kohring, G. A. (1990a). "Performance enhancement of Willshaw type networks through the use of limit cycles," *J. Physique* **51**, 2387.

Kohring, G. A. (1990b). "A high-precision study of the Hopfield model in the phase of broken replica symmetry," *J. Stat. Phys.* **59**, 1077.

Köhler, H., S. Diederich, W. Kinzel, and M. Opper (1990). "Learning algorithm for a neural network with binary synapses," *Z. Phys. B – Condensed Matter* **78**, 333.

Kohonen, T. (1974). "An adaptive associative memory principle," *IEEE Trans. Comput.* **C-23**, 444.

Kohonen, T. (1988). *Self-Organization and Associative Memory* (Springer-Verlag, Berlin).

Kouwenhoven, L. P., F. W. J. Hekking, B. J. van Wees, and C. J. P. M. Harmans (1990). "Transport through a finite one-dimensional crystal," *Phys. Rev. Lett.* **65**, 361.

Kouwenhoven, L. P., N. C. van der Vaart, A. T. Johnson, W. Kool, C. J. P. M. Harmans, J. G. Williamson, A. A. M. Staring, and C. T. Foxon (1991). "Single electron charging effects in semiconductor quantum dots," *Z. Phys. B – Condensed Matter* **85**, 367.

Kouwenhoven, L. P., S. Jauhar, K. McCormick, D. Dixon, P. L. McEuen, Y. V. Nazarov, N. C. van der Vaart, and C. T. Foxon (1994). "Photon-assisted tunneling through a quantum dot," preprint.

Kouwenhoven, L. P., A. T. Johnson, N. C. van der Vaart, D. J. Maas, and C. J. P. M. Harmans (1992). "Quantized current in a quantum dot turnstile," *Surf. Sci.* **263**, 405.

Kruglyak, L., and W. Bialek (1993). "Statistical mechanics for a network of spiking neurons," *Neural Comp.* **5**, 21.

Kühn, R. (1990). "Statistical mechanics for networks of analog neurons," in *Statistical Mechanics of Neural Networks*, ed. L. Garrido (Springer Verlag, New York), 19.

Kühn, R., S. Bös, and J. L. van Hemmen (1991). "Statistical mechanics for networks of graded-response neurons," *Phys. Rev. A* **43**, 2084.

Kulik, I. O., and R. I. Shekhter (1975). "Kinetic phenomena and charge discreteness effects in granulated media," *Sov. Phys. JETP* **41**, 308.

Lafarge, P., H. Pothier, E. R. Williams, D. Esteve, C. Urbina, and M. H. Devoret (1991). "Direct observation of macroscopic charge quantization," *Z. Phys. B – Condensed Matter* **85**, 327.

Lage, E. J. S., and J. M. Nunes da Silva (1984). "Mean field theory of Potts spin glass," *J. Phys. C* **17**, L593.

Landauer, R. (1957). "Spatial variation of currents and fields due to localized scatterers in metallic conduction," *IBM J. Res. Dev.* **1**, 223.

Landauer, R. (1988). "Spatial variation of currents and fields due to localized scatterers in metallic conduction," *IBM J. Res. Dev.* **32**, 306.

Landauer, R. (1989). "Nanostructure physics: fashion or depth?", in *Nanostructure Physics and Fabrication*, eds. M. A. Reed and W. R. Kirk (Academic Press, San Diego), 17.

Langton, C. G. (1990). "Computation at the edge of chaos: phase transitions and emergent computation," *Physica D* **42**, 12.

Laughton, M. J., J. R. Barker, J. A. Nixon, and J. H. Davies (1991). "Modal analysis of transport through quantum point contacts using realistic potentials," *Phys. Rev. B* **44**, 1150.

Le Cun, Y., I. Kanter, and S. A. Solla (1991). "Eigenvalues of covariance matrices: application to neural-network learning," *Phys. Rev. Lett.* **66**, 2396.

Lent, C. S., P. D. Tougaw, and W. Porod (1993). "Bistable saturation in coupled quantum dots for quantum cellular automata," *Appl. Phys. Lett.* **62**, 714.

Likharev, K. K. (1988). "Correlated discrete transfer of single electrons in ultrasmall tunnel junctions," *IBM J. Res. Develop.* **32**, 144.

Lippmann, R. P. (1987). "An introduction to computing with neural nets," *IEEE Acoustics, Speech, and Signal Processing* **4**, 4.

Little, W. A. (1974). "The existence of persistent states in the brain," *Math Biosci.* **19**, 101.

Lloyd, S. (1993). "A potentially realizable quantum computer," *Science* **261**, 1569.

Madelung, O. (1981). *Introduction to Solid State Theory* (Springer-Verlag, Berlin), 129.

Maher, M. A. C., S. P. Deweerth, M. A. Mahowald, and C. A. Mead (1989). "Implementing neural architectures using analog VLSI circuits," *IEEE Trans. Circuits Syst.* **36**, 643.

Mar, D. J. (1994). "Cryogenic field-effect transistors for the study of semiconductor nanostructures," Ph. D. thesis, Harvard University.

Marcus, C. M., A. J. Rimberg, R. M. Westervelt, P. F. Hopkins, and A. C. Gossard (1992). "Conductance fluctuations and chaotic scattering in ballistic microstructures," *Phys. Rev. Lett* **69**, 506.

Marcus, C. M., and R. M. Westervelt (1989a). "Stability of analog neural networks with delay," *Phys. Rev. A* **39**, 347.

Marcus, C. M., and R. M. Westervelt (1989b). "Dynamics of iterated-map neural networks," *Phys. Rev. A* **40**, 501.

Marcus, C. M., and R. M. Westervelt (1990). "Stability and convergence of analog neural networks with multiple-time-step parallel dynamics," *Phys. Rev. A* **42**, 2410.

Marcus, C. M., F. R. Waugh, and R. M. Westervelt (1990). "Associative memory in an analog iterated-map neural network," *Phys. Rev. A* **41**, 3355.

Marcus, C. M., F. R. Waugh, and R. M. Westervelt (1991). "Connection topology and dynamics in lateral-inhibition networks," in *Advances in Neural Information Processing Systems*, Vol. 3 (Morgan Kaufman, San Mateo), 98.

Martinis, J. M., M. H. Devoret, and J. Clarke (1987). "Experimental tests for the quantum behavior of a macroscopic degree of freedom: the phase difference across a Josephson junction," *Phys. Rev. B* **35**, 4682.

Martinis, J. M., and M. Nahum (1993). "Effect of environmental noise on the accuracy of Coulomb-blockade devices," *Phys. Rev. B* **48**, 18316.

Martinis, J. M., M. Nahum, and H. D. Jensen (1994). "Metrological accuracy of the electron pump," *Phys. Rev. Lett.* **72**, 904.

McCulloch, W., and W. Pitts (1943). "A logical calculus of ideas immanent in nervous activity," *Bull. Math. Biophys.* **5**, 115. Reprinted in *Brain Theory, Reprint Volume, Advanced Series in Neuroscience, Vol. 1*, ed. G. L. Shaw and G. Palm (World Scientific, Singapore, 1988).

McEuen, P. L., E. B. Foxman, U. Meirav, M. A. Kastner, Y. Meir, and N. S. Wingreen (1991). "Transport spectroscopy of a Coulomb island in the quantum Hall regime," *Phys. Rev. Lett.* **66**, 1926.

McEuen, P. L., E. B. Foxman, J. Kinaret, U. Meirav, and M. A. Kastner (1992). "Self-consistent addition spectrum of a Coulomb island in the quantum Hall regime," *Phys. Rev. B* **45**, 11419.

Mead, C. A. (1989). *Analog VLSI and Neural Systems* (Addison-Wesley, Reading, Mass.).

Meir, Y., N. S. Wingreen, and P. A. Lee (1991). "Transport through a strongly interacting electron system: theory of periodic conductance oscillations," *Phys. Rev. Lett* **66**, 3048.

Meir, Y., N. S. Wingreen, and P. A. Lee (1993). "Low-temperature transport through a quantum dot: the Anderson model out of equilibrium," *Phys. Rev. Lett.* **70**, 2601.

Meirav, U., P. L. McEuen, M. A. Kastner, E. B. Foxman, A. Kumar, and S. J. Wind (1991). "Conductance oscillations and transport spectroscopy of a quantum dot," *Z. Phys. B – Condensed Matter* **85**, 357.

Meirav, U., M. A. Kastner, and S. J. Wind (1990). "Single-electron charging and periodic conductance resonances in GaAs nanostructures," *Phys. Rev. Lett.* **65**, 771.

Mertens, S. (1991). "An extremely diluted asymmetric network with graded response neurons," *J. Phys. A* **24**, 337.

Metropolis, N., A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller (1953). "Equation of state calculations by fast computing machines," *J. Chem. Phys.* **21**, 1087.

Meurer, B., D. Heitmann, and K. Ploog (1992). "Single-electron charging of quantum-dot atoms," *Phys. Rev. Lett.* **68**, 1371.

Mezard, M., G. Parisi, and M. A Virasoro (1987). *Spin Glass Theory and Beyond* (World Scientific Publishing, Singapore).

Mori, Y., P. Davis, and N. Shigetoshi (1989). "Pattern retrieval in an asymmetric neural network with embedded limit cycles," *J. Phys. A* **22**, L525.

Murray, A. F. (1991). "Silicon implementations of neural networks," *IEE Proc. F* **138**, 3.

Nadal, J.-P., and A. Rau (1991). "Storage capacity of a Potts-perceptron," *J. Physique* **1**, 1109.

Nakata, S. (1993). "Observation of Coulomb-blockade oscillations by the back gate with subattofarad mutual capacitance," *Phys. Rev. B* **47**, 1679.

Nemoto, K., and H. Takayama (1985). "TAP free energy structure of SK spin glasses," *J. Phys. C* **18**, L529.

Nishimura, K., K. Nemoto, and H. Takayama (1990). "Metastable states of the naive mean-field model for spin glasses at finite temperatures," *J. Phys. A* **23**, 5915.

Nixon, J. A., J. H. Davies, and H. U. Baranger (1991). "Breakdown of quantized conductance in point contacts calculated using realistic potentials," *Phys. Rev. B* **43**, 12638.

Noest, A. J. (1989). "Domains in neural networks with restricted-range interactions," *Phys. Rev. Lett.* **63**, 1739.

Noest, A. J. (1990). "Semi-local signal processing in the visual system," in *Statistical Mechanics of Neural Networks,* ed. L. Garrido (Springer Verlag, New York), 303.

Nomoto, K., R. Ugajin, T. Suzuki, and I. Hase (1993). "Novel logic device using coupled quantum dots," *Electronics Lett.* **29**, 1380.

Oesterhelt, D., C. Brauchle, and N. Hampp (1991). "Bacteriorhopsin: a biological material for intformation processing," *Quarterly Rev. Biophys.* **24**, 425.

O'Neill, M. C. (1991). "Training back-propagation neural networks to define and detect DNA-binding sites," *Nucleic Acids Res.* **19**, 313.

Pankove, J., C. Radehaus, and K. Wagner (1990). "Winner-take-all neural net with memory," *Electronics Lett.* **26**, 349.

Parisi, G. (1986). "Asymmetric neural networks and the process of learning," *J. Phys. A* **19**, L675.

Pasquier, C., D. C. Glatti, U. Meirav, and F. I. B. Williams (1992). "Coulomb blockade of tunneling in a 2D electron gas," *Surf. Sci.* **263**, 419.

Perfetti, R. (1990). "'Winner-take-all' circuit for neurocomputing applications," *IEE Proc.* **137**, 353.

Personnaz, L., I. Guyon, and G. Dreyfus (1985). "Information storage and retrieval in spin-glass like neural networks," *J. Physique Lett.* **46**, L359.

Personnaz, L., I. Guyon, and G. Dreyfus (1986a). "Collective computational properties of neural networks: new learning mechanisms," *Phys. Rev. A* **34**, 4217.

Personnaz, L., I. Guyon, G. Dreyfus, and G. Toulouse (1986b). "A biologically constrained learning mechanism in networks of formal neurons," *J. Stat. Phys.* **43**, 411.

Peterson, C., and B. Söderberg (1989). "A new method for mapping optimization problems onto neural networks," *Int. J. Neural Sys.* **1**, 3.

Rieke, F., D. Warland, and W. Bialek (1993). "Coding efficiency and information rates for sensory neurons," *Europhys. Lett.* **22**, 151.

Rimberg, A. J. (1992). "Magnetotransport in uniform and modulated electron gases in wide parabolic quantum wells," Ph. D. thesis, Harvard University.

Rose, K., E. Gurewitz, and G. C. Fox (1990). "Statistical mechanics and phase transitions in clustering," *Phys. Rev. Lett.* **65**, 845.

Rubner, J., and K. Schulten (1990). "Development of feature detectors by self-organization—a network model," *Biol. Cyber.* **62**, 193.

Rumelhart, D. E., and D. Zipser (1986). "Feature discovery by competitive learning," in *Parallel Distributed Processing, Vol. 1*, ed. J. A. Feldman, P. J. Hayes, and D. E. Rumelhart (MIT Press, Cambridge, Mass.), 151.

Ruzin, I. M., V. Chandrasekhar, E. I. Levin, and L. I. Glazman (1992). "Stochastic Coulomb blockade in a double-dot system," *Phys. Rev. B* **45**, 13469.

Sakamoto, T., S. W. Hwang, F. Nihey, Y. Nakamura, and K. Nakamura (1994). *Jpn. J. Appl. Phys.* **33**, 4876.

Schmutz, M., and W. Banzhaf (1992). "Robust competitive networks," *Phys. Rev. A* **45**, 4132.

Scott-Thomas, J. H. F., S. B. Field, M. A. Kastner, H. I Smith, and D. A. Antoniadis (1989). "Conductance oscillations periodic in the density of a one-dimensional electron gas," *Phys. Rev. Lett.* **62**, 583.

Sherrington, D. and S. Kirkpatrick (1975). "Solvable model of a spin glass," *Phys. Rev. Lett.* **35**, 1792.

Shiino, M., and T. Fukai (1990). "Replica-symmetric theory of the nonlinear analogue neural networks," *J. Phys. A* **23**, L1009.

Shim, G. M., D. Kim, and M. Y. Choi (1992). "Potts-glass model of layered feedforward neural networks," *Phys. Rev. A* **45**, 1238.

Simic, P. D. (1990). "Statistical mechanics as the underlying theory of 'elastic' and 'neural' optimisations," *Network* **1**, 89.

Simic, P. D. (1991). "Constrained nets for graph matching and other quadratic assignment problems," *Neural Computation* **3**, 268.

Sompolinsky, H., D. Golomb, and D. Kleinfeld (1991). "Cooperative dynamics in visual processing," *Phys. Rev. A* **43**, 6990.

Sompolinsky, H., and I. Kanter (1986). "Temporal association in asymmetric neural networks," *Phys. Rev. Lett.* **57**, 2861.

Soukoulis, C. M., K. Levin, and G. S. Grest (1982). "Reversibility and irreversibility in spin glasses: the free-energy surface," *Phys. Rev. Lett.* **48**, 1756.

Soukoulis, C. M., K. Levin, and G. S. Grest (1983). "Irreversibility and metastability in spin glasses. I. Ising model," *Phys. Rev. B* **28**, 1495.

Stafford, C. A., and S. Das Sarma (1994). "Collective Coulomb blockade in an array of quantum dots: a Mott-Hubbard approach," *Phys. Rev. Lett.* **72**, 3590.

Staring, A. A. M., H. van Houten, and C. W. J. Beenakker (1992). "Coulomb-blockade oscillations in disordered quantum wires," *Phys. Rev. B* **45**, 9222.

Staring, A. A. M., J. G. Williamson, H. van Houten, and C. W. J. Beenakker (1991). "Coulomb-blockade oscillations in a quantum dot," *Physica B* **175**, 226.

Stopa, M. (1993). "Coulomb oscillation amplitudes and semiconductor quantum-dot self-consistent level structure," *Phys. Rev. B* **48**, 18340.

Szafer, A., and A. D. Stone (1989). "Theory of quantum conduction through a constriction," *Phys. Rev. Lett.* **62**, 300.

Sze, S. M. (1981). *The Physics of Semiconductor Devices* (Wiley, New York).

Takayama, H., and K. Nemoto (1990). "Spin glass properties of a class of mean-field models," *J. Phys.: Condens. Matter* **2**, 1997.

Tanaka, F., and S. F.Edwards (1980). "Analytic theory of the ground state properties of a spin glass: I. Ising spin glass," *J. Phys. F* **10**, 2769.

Thouless, D. J., P. W. Anderson, and R. G. Palmer (1977). "Solution of 'solvable model of a spin glass,'" *Phil. Mag.* **35**, 593.

Timp, G. (1992). "Is atomically precise lithography necessary for nanoelectronics?", in *Physics of Nanostructures*, ed. J. H. Davies and A. R. Long (Institute of Physics Publishing, Philadelphia), 101.

Tougaw, P. D., C. S. Lent, and W. Porod (1993). "Bistable saturation in coupled quantum-dot cells," *J. Appl. Phys.* **74**, 3558.

Treves, A. (1990). "Threshold-linear formal neurons in auto-associative nets," *J. Phys. A* **23**, 2631.

Treves, A., and D. J. Amit (1988). "Metastable states in asymmetrically diluted Hopfield networks," *J. Phys. A* **21**, 3155.

Treves, A., and D. J. Amit (1989). "Low firing rates: an effective Hamiltonian for excitatory neurons," *J. Phys. A* **22**, 2205.

Tucker, J. R. (1992). "Complementary digital logic based on the 'Coulomb blockade,'" *J. Appl. Phys.* **72**, 4399.

van Bentum, P. J. M., R. T. M. Smokers, and H. van Kempen (1988). "Incremental charging of single small particles," *Phys. Rev. Lett.* **60**, 2543.

van den Bout, D. E. (1990). "Graph partitioning using annealed neural networks," *IEEE Trans. Neural Networks* **1**, 192.

van Houten, H., C. W. J. Beenakker, and A. A. M. Staring (1992). "Coulomb-blockade oscillations in semiconductor nanostructures," in *Single Charge Tunneling*, ed. H. Grabert and M. H. Devoret (Plenum Press, New York), 167.

van Wees, B. J., H. van Houten, C. W. J. Beenakker, J. G. Williamson, L. P. Kouwenhoven, D. van der Marel, and C. T. Foxon (1988). "Quantized conductance of point contacts in a two-dimensional electron gas," *Phys. Rev. Lett.* **60**, 848.

van Wees, B. J. (1990). "Quantum ballistic electron transport and conductance quantization in a constricted two-dimensional electron gas," in *Proc. 3rd Int. Symp. Foundations of Quantum Mechanics, Tokyo, 1989*, ed. S. Kobayashi, H. Ezawa, Y. Murayama, and S. Nomura (Phys. Soc. Japan, Tokyo), 212.

Vogt, H., and A. Zippelius (1992). "Invariant recognition in Potts glass neural networks," *J. Phys. A* 25, 2209.

Wan, Y., P. Phillips, and Q. Li (1994). "Suppression of the Kondo effect in quantum dots by even-odd asymmetry," preprint.

Wang, L., J. K. Zhang, and A. R. Bishop (1994). "Microscopic theory for conductance oscillations of electron tunneling through a quantum dot," *Phys. Rev. Lett.* **73**, 585.

Washburn, S., and R. A. Webb (1992). "Quantum transport in small disordered samples from the diffusive to the ballistic regime," *Rep. Prog. Phys.* **55**, 1311.

Waugh, F. R., C. M. Marcus, and R. M. Westervelt (1990). "Fixed-point attractors in analog neural computation," *Phys. Rev. Lett.* **64**, 1986.

Waugh, F. R., C. M. Marcus, and R. M. Westervelt (1991). "Reducing neuron gain to eliminate fixed-point attractors in an analog associative memory," *Phys. Rev. A* **43**, 3131.

Waugh, F. R., and R. M. Westervelt (1993a). "Analog neural networks with local competition. I. Dynamics and stability," *Phys. Rev. E* **47**, 4524.

Waugh, F. R., and R. M. Westervelt (1993b). "Analog neural networks with local competition. II. Application to associative memory," *Phys. Rev. E* **47**, 4537.

Waugh, F. R., M. J. Berry, D. J. Mar, R. M. Westervelt, K. C. Campman, and A. C. Gossard (1994). "Single-electron charging in double and triple quantum dots with tuneable coupling," preprint.

Weis, J., R. J. Haug, K. von Klitzing, and K. Ploog (1992). "Transport spectroscopy of a confined electron system under a gate tip," *Phys. Rev. B* **46**, 12837.

Wolfram, S. (1984). "Universality and complexity in cellular automata," *Physica D* **10**, 1.

Wu, F. Y. (1982). "The Potts model," *Rev. Mod. Phys.* **54**, 235.

Yang, W., and A. Chiang (1990). "A full fill-factor CCD imager with integrated signal processors," in *Digest of Technical Papers, 1990 IEEE International Solid-State Circuits Conference, San Francisco* (IEEE, New York), 218.

Yano, K., T. Ishii, T. Hashimoto, T. Kobayashi, F. Murai, and K. Seki (1993). "A room-temperature single-electron memory device using fine-grain polycrystalline silicon," preprint.

Zang, J., and J. L. Birman (1993). "Theory of coherent transport thorugh a strongly disordered system: resonant tunneling in the one-dimensional tight-binding model," *Phys. Rev. B* **47**, 10654.