1

Simulation of large networks: technique and progress

William W. Lytton, Mark Stewart and Michael Hines

abstract

Computer models of epilepsy and seizures require simulation of large networks in order to produce emergent phenomenology, but also require inclusion of details of cellular and synaptic physiology to retain connection with pharmacological effectors for therapeutic intervention. The former constraint suggests the use of simplifying approaches: integrate-and-fire models or mean-field approximations. The latter constraint pushes one towards multi-compartment models. Compromise is required. We have developed highly-simplified event-driven complex artificial-cell models that utilize a series of rules to emulate critical underlying phenomena likely important for seizure generation and patterning. These include depolarization blockade, voltage-dependent NMDA activation, afterhyperpolarization and several others. Using these units, we can readily run simulations of 1-2 million units on a single processor. Here, we present results of data-mining 138,240 simulations of a 3,500-unit network. Parameter explorations can then relate single-unit dynamical "structure" to network function. We find that the networks are prone to latch-up, reminiscent of a seizure tonic phase. This can alternate with a slow repetitive clonic-like activity. Parameter dependence of epileptiform features was typically complex. For example, massive initial network response increased with greater unit excitability only in the context of particular AMPA and NMDA strength settings. This consequence of network complexity fits with the multi-site activity of endogenous neuroeffectors and may help explain why "dirty" drugs, those acting at multiple sites, might be particularly effective for seizure control.

goals of computer modeling for clinical disease

We use the fruits of computer simulation every day when we consult a weather report. Although weather reporting accuracy remains low for periods of greater than 10 days from the present, the improvement over weather reporting of 50 years ago is striking. The big difference is size: both the size of the data set available for initial conditions and the size of the computers for running the massive simulations that are now used. In the next 50 years we will likely see comparable progress in our ability to:

1. predict seizures
2. understand the genesis of seizures based on alterations in the underlying neurobiological substrate
3. intervene in directed ways to prevent a single seizure or to alter the substrate and prevent seizures from occurring.

Unlike weather simulation, where the underlying factors involved (temperature, wind, humidity) are well characterized and where the interactions are fully understood, the nervous system remains full of under- or uncharacterized neurotransmitters, receptors, second and third messengers and electrical interactions. Also unlike weather, many locations where we would like to measure chemical concentrations or currents are inaccessible. For these reasons, accurate simulation will require growth not only in simulation but also in measuring capabilities.

The hope and expectation is that accurate simulation will lead to rational therapeutics, whether prophylactic or acute, whether pharmacological, electrical or surgical. Acute electrical intervention prior to a seizure is becoming a readily foreseeable scenario with the increasing use of implanted electrodes for other brain diseases. It seems likely that practical efforts in this direction will result from direct analysis of brain signals. Here, modeling will assist by permitting rapid

assessment of many possible stimulation protocols. With regard to prophylaxis, genetic analysis and molecular biology can identify an underlying substrate that predisposes to seizures but cannot trace the dynamical path whereby an abnormality results in a seizure. Characterization of this dynamical pathway, the chain of causality, can identify critical loci where interventions are most practical.

Clinically useful seizure simulations require that connection be made both with clinical observables – the seizure and its electroencephalographic manifestation – and with therapeutic approaches – whether pharmacological or surgical. For this reason, we and others have largely focused on detailed compartmental models which provide explicit modeling of dendritic regions, of voltage-sensitive ion-channel populations and of various synaptic receptor subtypes. These receptors and channels are the targets of pharmacotherapeutic manipulations and can be comparably manipulated in these models. However, simplified alternatives to compartmental models are proving useful for investigation as well.

DETAILED VERSUS SIMPLIFYING MODELING

There is a tension between the desire to model as accurately as possible and the desire to simplify, both due to practical limitations of computer size and due to the need to isolate the critical factors and associations that lead to understanding. In the neural modeling literature this tension arises where the study of dynamical systems meets biology.

The most common approach to mathematical modeling of neural systems has involved explicit modeling of individual neurons. However, there is also a long tradition of mean field approaches, utilizing the insights of statistical mechanics. Here, the individual neurons are assumed to be indistinguishable particles whose interactions produce bulk properties, just as the interactions of molecules in a liquid produce the bulk properties familiar from thermodynamics (Wilson and Cowan, 1972). These approaches have lately been applied to epilepsy as well (Wilson et al., 2006).

Moving up toward slightly greater complexity, many followers of the simplifying ethic utilize leaky integrate-and-fire models. These models forego all biological details except for a threshold and a leaky membrane. When the threshold is reached, the cell discharges. The simulated membrane allows inputs to summate while the leakiness permits membrane potential gradually to return to rest after a subthreshold input. At the synapse level, these simulations may utilize instantaneous or brief depolarizing or hyperpolarizing synapses to drive the cell.

N. Brunel and colleagues have been pioneers in studying the dynamics of large networks of leaky integrate-and-fire cells, also using mean field methods to enhance understanding of observed activity patterns (Brunel, 2000; Brunel and Wang, 2003). They identified several different characteristic patterns of firing: asynchronous irregular, asynchronous regular, synchronous irregular and synchronous regular, demonstrating the parameter associations for each of these regimes. In general, firing patterns depended on the strength of network driving and the strength of excitatory connections within the network. They demonstrated that low inhibition states gave fast firing with some coordination and that higher levels of inhibition produced irregular firing that, nonetheless, coordinated in global population oscillations that would produce a high amplitude field potential.

Leaping from these highly simplified models to the other end of the complexity spectrum brings us to the models of Traub and associates (Traub and Wong, 1982; Traub et al., 2005). They pioneered simulations of large networks with detailed compartmental models of the individual neurons. Such compartment models included many details of ion channels, dendritic integration and synaptic dynamics that are omitted from the leaky integrate-and-fire cell models. Compartment models are built by linking up multiple Hodgkin-Huxley parallel-conductance models in trees that reflect the morphology of a real neuron, often one that has been traced microscopically (Figure 1.1). The Hodgkin-Huxley model is comprised of a four state-variable system representing voltage, sodium channel activation, sodium channel inactivation and potassium channel activation. Each additional type of sodium, potassium or calcium channel or synapse in a compartment generally adds from one to three additional state-variables. A somatic compartment (representing the soma or cell body of a neuron) might easily have 10 such channels, while other compartments may have three to four channels. A fully traced neuron may be fairly accurately represented using between 100 and 1000 compartments. Therefore, a full single neuron model with 500 compartments may have 5000–10 000 state-variables corresponding to 5000–10 000 linked ordinary differential equations (ODEs). Although computers have gotten much faster over the past few decades, an accurate simulation of such a model on a desktop computer might still require several minutes to an hour for one second of simulated time, depending on the pattern of activity being observed. Networks corresponding to substantial lumps of brain tissue require the simulation of between one thousand and one million neurons. In many cases, we may also want to do parameter explorations over several hundred to several thousand simulations. Thus, we face the age-old problem: unbounded desire constrained by limited resources.

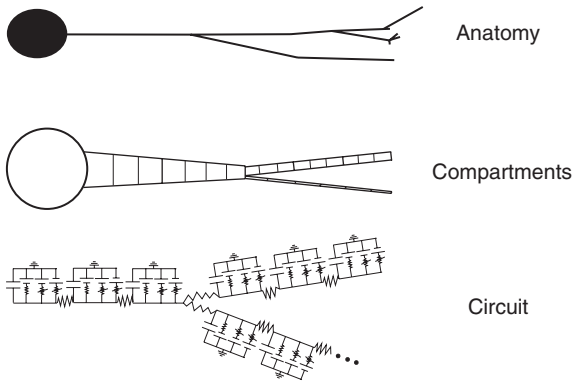

FIGURE 1.1 How to make a compartment neuron.

Compromises must be made. In epilepsy modeling, we need large networks, but we wish to retain connection with membrane physiology. A common approach is to compromise on the number of compartments, sacrificing some of the details of intraneuronal voltage dynamics by reducing representations of each cell from several hundred compartments to between 5 and 20 compartments (Traub and Wong, 1982; Bush and Sejnowski, 1991). A more extreme compromise is to collapse the cell down to only one or two compartments (Pinsky and Rinzel, 1994; Mainen and Sejnowski, 1996).

NETWORK SIMULATION TECHNIQUE: CONTINUITY AND EVENTS

The heart of realistic neural simulation is the integrator. A major effort in simulator technology has been to optimize integrators to obtain maximal speed without loss of accuracy (Hines, 1989). However, as network simulation has become more important, simulators have been developed that do no integration but instead focus on *events* (Morrison et al., 2005).

Events in the nervous system, as in the rest of nature, occur continuously in time and therefore could be solved using the basic ODE solution methodology typically used for many types of computer simulation applications. However, a short-cut is usually taken: the axon is regarded as a signal delivery system that is not integrated in detail, foregoing the type of axon simulation that was famously done in the original Hodgkin and Huxley work on the squid axon. This classical simulation took many days due to the fact that a 'computer' was a woman working a primitive adding machine. (A roomful of these women made up a parallel supercomputer.) Indeed, axon simulation can still take days of supercomputer time if simulated at the level of detail currently available from electron microscopy and single-channel kinetics. Instead of bothering with all of this, the axon is typically reduced to one parameter: a delay. This delay represents the time for action potential transmission from soma to the presynaptic bouton. In some simulations, an additional parameter is used representing the probability of transmission failure (Rhodes, 2006). Synaptic weight is a synaptic property and therefore not an axon parameter.

This short-cut makes neuronal network simulation unusual among the panoply of computer simulation applications. Because nature is continuous in time and space, many computer simulations, like weather, are continuous in time and space. Neuronal network simulations are not spatially continuous since the elemental units of simulation are cells. More importantly, neuronal network simulations are not fully temporally continuous since intermittently occurring spikes trigger events in other neurons. However, the simulation of the individual neuron is a continuous simulation governed by ODE solution.

Hence, neuronal network simulation is a hybrid of multiple continuous simulations interrupted by synaptic events. The approach required for simulating the events, called event-driven simulation, is widely used in industry. Event-driven simulation is used, for example, in internet modeling, where one asks how a packet, following many possible paths through routers, reaches a distant computer. The need to combine event-driven simulation with continuous dynamical (ODE) simulation suggests a variety of approaches, some of which are described below.

COMPARTMENT CELL TECHNIQUES: VARIABLE STEP METHODS

We have augmented the NEURON simulation system to include a variety of techniques that improve simulation efficiency and allow larger networks to be run in less time. Simulation of the individual neuron, as in other continuous domains, requires the numerical solution of linked ordinary differential equations. These differential equations are of a general form

familiar from kinematics and electricity: e.g., derivative of voltage (change with respect to time) equals some function of voltage, time and other state variables. Groups of linked ODEs typically cannot be solved analytically (i.e., expressed as a function of time). Instead, they are solved approximately on a computer by using small time steps that approximate the infinitely small time step assumed by calculus.

Traditional numerical simulation techniques, some preceding the invention of the computer by centuries, project forward a simulation by a series of fixed time steps. We have been exploiting the CVODES variable time step methods developed by Hindmarsh and collaborators (Cohen and Hindmarsh, 1994; Hindmarsh and Serban, 2002). These allow the time steps to change according to the needs of a simulation. In a neural simulation, this typically means that a time step can increase up to hundreds of milliseconds when a neuron is not firing but drop down into the sub-millisecond range to simulate properly an action potential.

A fixed time-step method suffers two disadvantages: one practical and the other theoretical. The practical issue is speed. The fixed-step method will plod through a simulation at a fixed rate, irrespective of whether the network is spiking furiously or is largely quiet. The variable-step method adapts to the level of activity. Although this can give the variable-step method an advantage during periods of quiet, it may be a significant disadvantage in a network that has a lot of firing. Therefore, the choice of variable-step method or fixed-step method for a particular simulation remains an empirical one.

The theoretical disadvantage of a fixed time-step method is particularly relevant for the case of epilepsy simulation, where issues of synchrony are paramount. A fixed-step imposes an artifactual synchrony on the network by virtue of the fact that all spikes and synaptic activations within the time step must occur at precisely the step boundary. Although this reduction in jitter is slight, it has been shown to alter population activity once it is magnified by network connections (Hansel et al., 1998). This artifactual time locking means that time steps as small as one nanosecond (about one one-millionth the duration of spike and four hundred-thousandth the length of a typical simulation time step) may be required to model accurately spike delivery times when using a fixed-step method (Shelley and Tao, 2001).

Unfortunately, network simulations with variable time-step methods rapidly run into a related problem. Synchrony means that some number of neurons are firing at almost exactly the same time. This produces high event-time densities which results in extremely small time steps. This spike-density crisis occurs with high interconnectivity between synchronous neurons and can lead to time steps being driven down to and below one nanosecond, resulting in performance worse than that of the worst-case fixed-step method noted above. The essential problem is that the integrator is too fastidious, dealing with every spike interval as an individual integration problem, where a fixed-step method would simply aggregate all nearby spikes as occurring in a single moment.

We have considered different approaches to handle this high spike-time density problem. As intimated above, the variable-step method is of the type predictor-corrector, as contrasted with the continuous forward progress of a linear method. Using a predictor-corrector algorithm, the integrator tries out something and then ensures that it is accurate enough before proceeding forward. Although the actual answer and hence the actual error is not known, the error at each step can be approximated by noting gradients of solution convergence with respect to time-step shrinkage. Thus, the voltages and conductances are being assessed at each time step in the continuous neuron integration. However, the issue in an event-driven simulation is not accuracy in voltage but accuracy in time. It would be desirable to produce an error estimate for spike time, but it is not clear how this could be done. Failing this it would, nonetheless, be possible to cut off further accuracy-seeking once the integrator starts making changes that only alter spike times by one microsecond or less, forcing all spikes onto a resolution boundary. We have not yet done investigations in this realm, but have noted that such a minimum interval would permit additional speed-ups through the use of a bin queue for constant-time insertion and removal of events.

The other approach to handling high spike-time density is to break up the problem into multiple integrations. Instead of having one *global* integrator imposing a single global variable time step on the entire problem, we use multiple *local* integrators each of which has its own local variable time step (Lytton and Hines, 2005). This reduces the problem by reducing the density for each integrator, though it does not eliminate it entirely. Additionally, a major advantage to using local variable time step is that each neuron then integrates itself forward at its own rate (Figure 1.2). Active neurons are integrated actively while quiescent neurons do not use any central processing unit (CPU) time. This can produce significant speed-ups in a large network with many interacting subpopulations. On the other hand, busy networks with neurons firing at high rates will likely run faster by stepping through with a consistent fixed time step. Again, the determination of speed-up for any particular simulation is an empirical one.

The major stumbling block in simulation of networks with this method is to ensure that neurons do not run so far ahead that they cannot be influenced by an event. Quiescent neurons can be integrated well forward in time but software must be able to back-track if an event comes in earlier. Fortunately, the CVODES integrator used in NEURON makes it easy

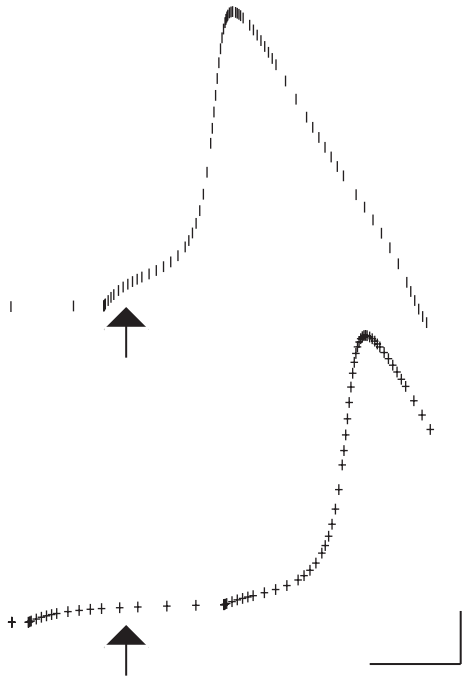

FIGURE 1.2 Local variable time-step method produces different integration time points for different neurons in a network. As the neuron in upper trace passes through threshold (arrow), time-steps are very small (time points close together) near this unstable equilibrium. At the same time the neuron in lower trace is subthreshold and shows large time-steps. Note that the peak of the action potential also requires frequent integration. Scale: 1 ms, 20 mV.

to interpolate to any point bounded by the current integration time step. The integration thus lurches forward, integrators back-tracking to pick up spikes as needed.

COMPARTMENT CELL TECHNIQUES: SUPERCOMPUTING APPROACHES

Early supercomputers, such as the Cray series, were large CPUs with a lot of memory which could stream in vectors of numbers for simultaneous processing. Modern supercomputers are made up of multiple CPUs, each of which is comparable to a CPU in a laptop which produce speed-ups not by doing more or by spinning faster but by working together. These are therefore considered parallel supercomputers (vector parallelism was used in the earlier supercomputers). We have begun to study expanded versions of a cortical epilepsy model on parallel supercomputers (Figure 1.3; Bush et al., 1999). Thus far, we have used this simulation primarily to improve our supercomputer algorithms, as will now be discussed.

Consideration of the hybrid nature, continuous and event-driven, of neuronal network simulations immediately suggests that they will be amenable to running on modern parallel supercomputers (Hereld et al., 2005; Migliore et al., 2006; Sperelakis and Ramasamy, 2006). Subpopulations of neurons are run on individual processors of the supercomputer and exchange information at intervals. As with the local variable time-step method, care must be taken lest one subpopulation runs too far ahead. This is handled by noting the minimum synaptic delay in the network. This minimum delay represents the maximum interval during which integration can move forward without possibly requiring external information – a spike at the beginning of this interval which must be delivered and consequently integrated at the beginning of the next interval. Within each individual processor, integration of the subpopulation can be performed using either a fixed-or variable-step method.

Speed-up with parallel supercomputers depends on several factors. First, of course, is the number of processors. The size of a ‘supercomputer’ can range from the eight CPUs that can be inexpensively put together in the laboratory to the 2048 CPU and larger behemoths available at national supercomputer centers. Next is the nature of the individual CPUs themselves. There is heterogeneity in CPU speed among machines. Additionally, variation in CPU speed within a single machine can cause unexpected run-time alterations when running across CPUs of different specifications. We found the size of CPU cache to be a critical factor determining run-times. Cache is the onboard memory of the CPU which allows

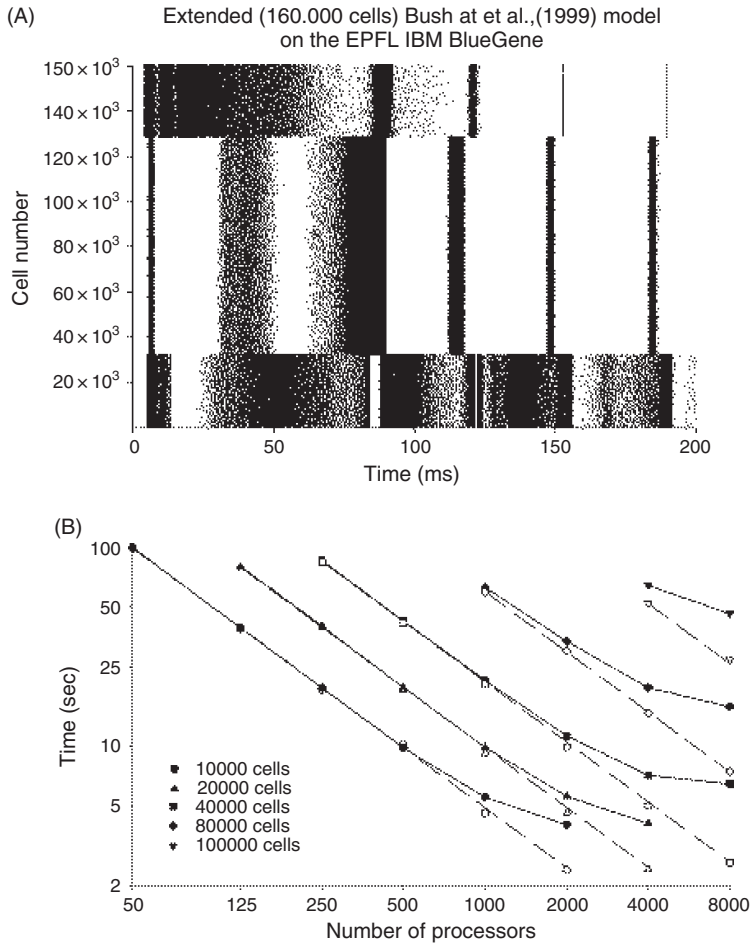

FIGURE 1.3 Performance of extended Bush et al. simulation (Bush et al., 1999) on parallel supercomputers. (A) Raster plot from a 200 ms 160 000 cell simulation shows development of recurrent synchronous activity most clearly expressed in the regular-spiking excitatory subpopulation (center group). Top group are inhibitory cells and bottom excitatory intrinsically bursting cells. (B) CPU time (open symbols) and total runtime (closed symbols) compared to linear speed-up (dashed lines) for 200 ms simulation. (Reproduced from Bush et al. (1999), Figure 5, with kind permission of Springer Science and Business Media, Copyright Springer Press).

code and data to be accessed at speeds up to fivefold faster than from main memory (RAM). A simulation that can fit each CPU's apportionment into cache will enjoy a substantial speed-up.

Intrinsic CPU factors are one factor determining simulation speed-up. Another potential factor is organization of the processors – how well can the CPU network run the neuron network? These factors include hardware – how are the CPUs physically connected to one another – and software – how is the network split up into the subpopulations that are assigned to the various CPUs. Although hardware design cannot be readily altered physically, some machines may have software switches that determine the effective connectivity being utilized at a given time. A related design choice occurs when choosing the underlying software protocol used to send information:

1. data can be sent specifically from one CPU to another
2. data can be broadcast from each CPU to all CPUs
3. data can be gathered from all CPUs and then rebroadcast to all CPUs.

In practice, the third option is easiest to implement and appears to be efficient despite the larger amount of data being passed around. However, in any of these cases, there is an advantage to minimizing the size of data-structures carrying the spike-time information among CPUs.

Other factors that involve the layout of subpopulations across CPUs may also play a large role in simulation efficiency. First and most important is load balancing. *A priori*, one does not want to put 10 neurons on one CPU and 1000 neurons on another, but such an apparent imbalance may be desirable in a case where 10 large pyramidal neurons are being run

with 100 active dendritic compartments together in a simulation with 1000 small point neuron interneurons. Similarly, one can envision scenarios where one population of cells fires frequently, require more CPU time and would benefit from being distributed at a lower density. Thus far we have not experimented with these design issues but have simply handed out the neurons to CPUs using a round-robin algorithm, as one uses when dealing cards. Load balance statistics are readily gathered from the supercomputer after a run by looking at wait times for individual CPUs – periods when a CPU had finished its allotted simulation slice and had to wait to ascertain whether there were any inputs (spikes) from neurons in other CPUs. Despite the simple algorithm, load balance has not been a problem until we get down to a few cells per CPU.

Consideration of wait times also leads to another organizational consideration for distributing subpopulations on CPUs. Ideally, cells that wire together should ride together on a CPU so as to minimize off-CPU communications. In addition to the presumably static wiring (assuming here no sprouting during simulation), one would also want to consider here again the dynamics of the simulation. Cells that are firing at high rate will have much more to say and will want to have their follower cells nearby in order to communicate quickly. Of course, any improvements in subpopulation organization will only be valuable if they are supported by the choices made for communication data-structures and inter-CPU data distribution.

INTEGRATE-AND-FIRE REVISITED: COMPLEX ARTIFICIAL CELLS

This juxtaposition of event-driven and continuous simulation has led several groups to consider circumstances that would allow the time-intensive numerical integration to be jettisoned entirely (Watts, 1994; Mattia and Del Giudice, 2000; Delorme and Thorpe, 2003; Makino, 2003; Rudolph and Destexhe, 2006). The simplest way to do this is to conceive of the neuron as a simple delay mechanism, receiving an event and then reliably producing another event after a certain period of time (Pytte et al., 1991). This retains the influence of the network graph on activity propagation but discards the dynamics of the neuron, including the phenomena of inhibition and of temporal integration. To get these critical phenomena back, we have returned to the integrate-and-fire neuron, revising it to allow it to be processed in an event-driven framework instead of solving it numerically. The resulting model neurons are called artificial cells in NEURON and provide the speed-ups one might expect by going from time steps of 25 μs to time steps that largely represent the interval between activity in the network (Carnevale and Hines, 2006).

As noted at the beginning of the chapter, integrate-and-fire models are limited in their usefulness in epileptology or elsewhere in neurobiology since they lack the critical details that permit us to connect the dots from the microworld of genetics and pharmacology (anticonvulsants) to the observables of electrophysiology (EEG) and bulk physiology (fMRI, behavioral alterations and seizures). In order to make these connections, we needed to develop more complex models that are still far simpler and far faster than full compartment models. Approaches to this have included moving from the one-dimensional (one state variable) leaky integrate-and-fire cell model to a two-dimensional model (Izhikevich, 2003, 2004), utilizing a map based model (Rulkov et al., 2004), simplifying within the context of a compartmental model (MacGregor, 1987; Destexhe, 1997), or devising semi-analytic expressions for simulation with integration (Rudolph and Destexhe, 2006).

To bridge simultaneously the performance gap from event-driven to continuous models and the credibility gap between integrate-and-fire neurons and compartmental neurons, we have been developing complex artificial cells that include the details needed to make connection with pharmacology (Lytton and Stewart, 2005, 2006). We do this by progressively adding rules to the three basic rules of a leaky integrate-and-fire neuron:

1. add up the inputs (integrate)
2. slowly relax back to rest in the absence of inputs (leak)
3. trigger an event when threshold is reached (fire).

Typically, standard integrate-and-fire models will also have a fourth rule that will reset the membrane potential to rest after a spike. Each of these rules is controlled by one or more parameters:

1. a synaptic time constant determining the time over which temporal summation will take place (technically a synaptic rather than a cellular parameter but lumped in with the rest of the neuron for convenience)
2. a membrane time constant which also influences temporal integration by determining how long the membrane will stay depolarized or hyperpolarized after an input
3. a threshold voltage above which spiking occurs.

The integrate-and-fire model has a single state variable representing membrane voltage. By contrast, a compartmental model may have 100s of state variables, representing voltage at different locations throughout the dendritic tree, as well as

the conductance states of different ion channels in the soma and dendrites. (It is worth reminding the reader at this point of the difference between parameters and state variables. Parameters are constant values that determine the simulation: e.g. the capacitance or the length of the apical dendrite. States are the dependent variables of ordinary differential equations that vary during the course of the simulation: e.g. voltage.) The far greater complexity of integration for compartmental models is due to the large number of state variables and to the fact that these state variables are linked. The differential equations for most of the state variables include membrane voltage so that any alteration in voltage will change these other state variables which will change voltage which will change more state variables in never-ending loops of positive and negative feedback.

In addition to having more parameters than a leaky integrate-and-fire cell model, the complex artificial cell has more state variables. However, these state variables are not linked to one another, making them far easier to integrate and making it possible to skip numerical integration entirely by updating them as needed using the analytic solution for the single ODE in the context of an event-driven simulation. To create the complex artificial cell model, rules were added to the leaky integrate-and-fire cell model in order to include features thought to be important for neural system dynamics in general and for epilepsy in particular. One of these features is depolarization blockade: a cell will not spike if driven excessively. This feature can be added by including a simple additional rule: a threshold above which spiking does not occur: the cell will fire only within a window between the spike and block thresholds. Clearly, a depolarization blockade threshold is only a very crude approximation to the dynamics of populations of sodium channels leading to depolarization blockade in real neurons. It should here be re-emphasized that the relation of these complex artificial cell models to compartmental models can be taken as that between a sketch and a photograph: reminiscent but without correspondence in detail.

State variables for complex artificial cells can be classified as intrinsic or synaptic. Both types of state variables are handled similarly. In order to avoid linked differential equations, state variables are combined linearly to produce a membrane potential. Membrane potential in this model is therefore a derived value rather than a primary state variable. Four synaptic state variables are maintained in each complex artificial cell: corresponding to excitatory AMPA and NMDA inputs and inhibitory $GABA_A$ and $GABA_B$ inputs. (The chemical acronyms AMPA (alpha-amino-3-hydroxy-5-methly-4-isoxazole propionic acid), NMDA (N-methyl-D-asparta), GABA (gamma-aminobutynic acid) are used as a short-hand to refer to receptors with sensitivity to the corresponding compounds.)

The response to an individual synaptic input is a jump in the corresponding state variable. For example, in the case of $GABA_A$, weight parameter W_{GABAA} determines the size of a hyperpolarizing step in state variable V_{GABAA}. The step is scaled so as to provide a saturation comparable to that which occurs due to the reversal potential with the more realistic conductance change in a compartmental model: $V^{step}_{GABAA} = W_{GABAA} \cdot \frac{V - E_{GABAA}}{E_{GABAA}}$ (E_{GABAA} in the denominator is a normalization; W_{GABAA} is unitless). Following the step, V_{GABAA} decays with time constant τ_{GABAA}. The complex artificial cell utilizes more complex rules for NMDA and $GABA_B$. The NMDA mechanism uses standard Mg^{++} unblocking equation (Jahr and Stevens, 1990). The $GABA_B$ mechanism employs a non-linear augmentation with size of presynaptic burst (Destexhe and Sejnowski, 1995).

In addition to the synaptically driven state variables, there is one intrinsic hyperpolarizing state variable (AHP). AHP primarily produces a post-spike afterhyperpolarization (I_{AHP}, SK channel effect), although it also is used to augment the refractory period (I_C, BK channel effect). By altering the distance between current voltage and threshold, the AHP state variable produces the same effect as one would get with a varying threshold. Additional intrinsic rules include the aforementioned depolarization blockade and refractory period, post-spike synaptic depression, a bursting mechanism and jitter.

Since the complex artificial cell has no integrator overhead, network simulations depend only on the amount of spiking in the model. At high spiking activity, events may occur with a frequency that is comparable to the time step of a fixed-step integrated model. However, typically only a subset of the neurons, representing only a few equations, need to be updated so that the simulations are still relatively fast.

In a compartment model, physiologically-observable dynamics such as adaptation are dependent on multiple voltage-sensitive ion channels. Each of these voltage-sensitive ion channels has its own parameterization (Borg-Graham, 1991). Changes in channel-level parameters will alter not only the activity aspect of interest but also other neuron-level dynamics. Altering activity in a compartmental model requires manipulations that are two steps removed from the phenomenon of interest. In the complex artificial cell model, a selected neuron dynamic can be altered directly without interfering with other phenomena. For example, the voltage at which depolarization blockade occurs is controlled by a single parameter rather than being a consequence of the density balance for sodium and potassium channels as well as time constants of the state variables determining the dynamics of each.

The complex artificial cell model provides a framework that can incorporate other rules as needed in particular systems where they are thought to be important. For example, some neurons have prolonged bursts with characteristic firing patterns

(Destexhe et al., 1996). These patterns can be incorporated into the rule based either by constructing an analytically calculable dynamical rule or by providing a cut-and-paste spike-form. Similarly, additional rules can be incorporated by providing waveforms copied from electrophysiological records. More flexible rules can be arrived at by emulating the behavior of the more complex compartmental models. In this case, complex artificial cell units can be run together with compartmental simulations in NEURON with use of a fitting algorithm controlling complex artificial cell unit parameters.

A useful additional rule-set would incorporate input/output relations from dendritic inputs. There is considerable debate as to whether dendrites simply provide reach, with all inputs being handled linearly, or provide substantial signal processing. In the latter case, it is possible that the dendrites make the cell into the equivalent of an entire neural network of simplified units (Poirazi et al., 2003). It would be possible to represent dendritic fields as individual complex artificial cell units with separate specialized rule base. A simpler approach would be to take the dendritic transform as a mapping that can be replicated using a multidimensional table look-up.

EPILEPSY SIMULATION WITH COMPLEX ARTIFICIAL CELLS

We previously referred to the complex artificial cell model as a 'sketch' of a neuron. Similarly, network models built with complex artificial cells serve as a network sketch-pad, enabling us rapidly to try many architectures and parameters. Activity sketches that look promising can then be re-evaluated with more complex models, both full compartment models and hybrid networks, comprised of both compartment and complex artificial cell neurons. Hybrid networks use the local time-step method so as to allow the compartment models to be individually integrated while the event-driven complex artificial cells take up no integrator time and jump from event to event as before.

The highly coordinated firing that underlies both normal activity and seizures is a generic property of model networks. Excitatory connections spread firing across a network. The activation can then be coordinated into repetitive activity by the modulating or sculpting effects of inhibition (Lytton and Sejnowski, 1991). Other networks will show incoherent firing with minimal correlations between spikes of individual neurons. We explore parameter changes to see how these effect particular patterns at the network level.

We designed a simple network sketch to illustrate an organizational principle that applies to both slice and *in vivo* epileptogenesis: a subset of cells or a separate brain area provides activation, provoking activity in a set of follower cells that express a seizure (Dudek et al., 1999; Meeren et al., 2002; Van Drongelen et al., 2003; Dominguez et al., 2005). We make the distinction between drivers (provokers) and expressors of epileptic activity to use this organizational principle without specifying whether we are considering local or remote activation. The network was therefore made up of three cell populations:

1. an excitatory population which was made spontaneously active in order to play the role of a driver of activity ('drivers', $N_D = 100$ for small network or $N_D = 1000$ for large network)
2. a larger excitatory population which then expresses epileptiform activity ('expressors', $N_E = 1000$ or $N_E = 2000$)
3. an inhibitory population (inhibitors, $N_I = 250$ or $N_I = 500$).

In addition to ongoing activation from the drivers, the networks are typically activated by a global 'shock' which activates about half of the cellular elements simultaneously (Figure 1.4). The expressor cells are the major contributors to the simulated field potential. The drivers (bottom row) have an average firing rate of 2 Hz. Despite random inputs, structure is created in the field due to the tendency of random coincident input to produce brief activation chains that involve a large and changing proportion of the expressors.

Evaluation of simulations showed that excessive excitatory connectivity could easily lead to a *latch-up* (tonic) condition where expressor cells all fire at high rates, at or near the maximum rate allowed by the refractory period. Figure 1.5A shows a network that immediately ramps up to maximal firing in the expressors (E) cells. This activity eventually gives way to a brief period of intermittent (clonic) activity and then returns to tonic activity.

We explored the conditions that would terminate or trigger the latch-up condition. In Figure 1.5B, a slow-building AHP provided a hyperpolarizing influence that terminated the latch-up condition. Figure 1.5C demonstrates that a slow-building depolarization, here caused by NMDA, can lead to latch-up. The apparent dynamic duality of tonic triggering and tonic turn-off hides a distinction with regard to event timing. The gradual build-up in the intrinsic AHP state-variable reliably terminated tonic activity after an interval that depended on the time constant and strength (density) parameters for AHP in the individual cells. By contrast, the timing of the triggering of the tonic state by NMDA activation was far more variable, depending not only on the NMDA parameters (synaptic strength and time constant) but, more importantly, on the coincidence of random inputs that simultaneously pushed a critical number of cells above threshold. This distinction

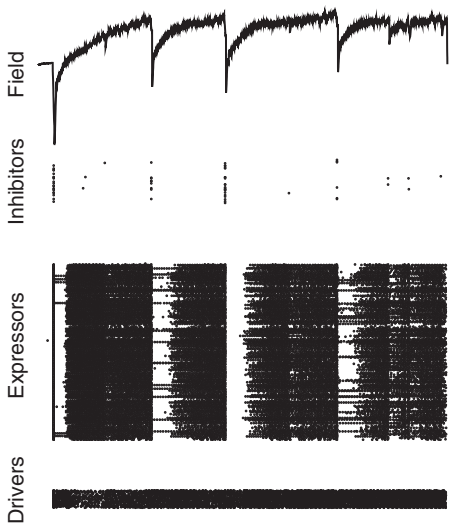

FIGURE 1.4 Raster plots showing neuron action potentials for three classes of cells during one second of epileptiform activity in a sample network. Simulated field primarily reflects activity in the expressor cells. Note simultaneous activation of multiple cells coincident with first large field potential corresponding to shock stimulation near beginning of trace.

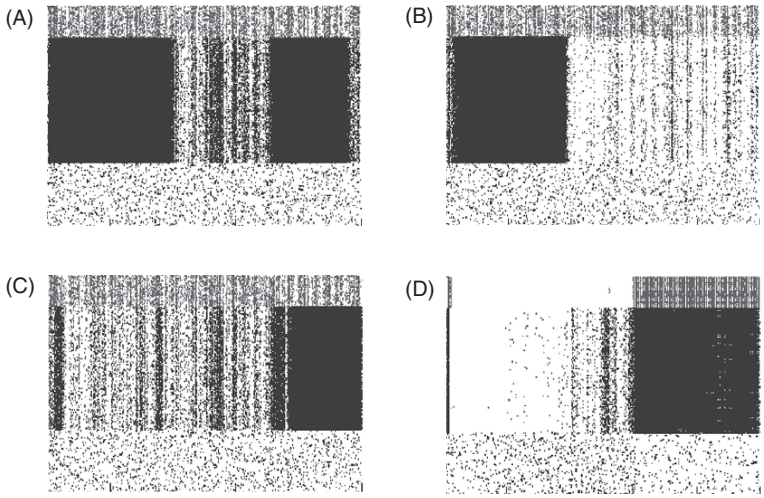

FIGURE 1.5 Raster plots of 1 s of 3500 cell network simulations. (See text for details.)

between a ramped versus triggered transition did not strictly follow the difference between intrinsic and synaptic parameters, however. Figure 1.5D shows a ramped transition to tonic activity based on the wearing off of synaptic activation associated with a $GABA_B$ mechanism. Although $GABA_B$ is a powerful inhibitory mechanism that tends to silence the network, in this case the silencing would be the triggered mechanism while the latch-up would be predictable ramped release response.

SIMULATION DATA-MINING

Scientific data-mining is a process of seeking patterns among the enormous amounts of data generated by modern scientific techniques. Data-mining infrastructure originally developed in business, where accounting and legal information collected over decades was found to be a valuable resource, a figurative gold-mine of data. In the scientific realm, similar techniques have been developed and adapted over the past decade as massive information veins associated with the genomes of human, fly, mouse and others have come on-line. The complexity of our simulations with the mass production of thousands or millions of simulations for further study has led us to develop a data-mining resource within the simulator environment.

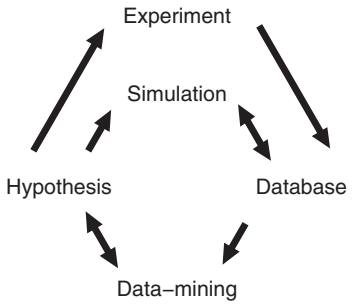

FIGURE 1.6 Simulation and experiment produce massive amounts of data to be stored, managed and analyzed.

We have been using our data-mining system to analyze data from both electrophysiology and simulation, to organize simulation parameters and to relate parameters to simulation results. The large amounts of data involved makes data-mining an explicit endeavor that lies at the juncture of simulation, experiment and hypothesis generation (Lytton, 2006). Figure 1.6 illustrates the use of simulation as a partner to experiment. Data flows out of experiment and simulation into data-bases from where it is mined to generate hypotheses that can be tested by further experiment or additional simulation. Simulation not only generates data but is also a consumer of data (two-headed arrow at right): a massive parameter complex must be managed and refined by repeated simulation. These parameters are stored in tables that relate to the tables storing simulation output. The need for data-mining implies the luxury of having so much data that it will never be fully explored. In this setting, a new hypothesis will, in some cases, be first assessed by simply looking back at the data already obtained (two-headed arrow at left) in order to determine whether an experiment or simulation already performed provides additional supporting data for that hypothesis.

We have used our data-mining software to evaluate 138 240 parameter sets for the network described above. We evaluated 12 synaptic parameters and two intrinsic parameters (threshold and AHP strength). Various parameter combinations produce a wide variety of responses as indicated by field potential (Figure 1.7). Due to storage limitations, we are only able to save field potentials and not raster plots or single cell spike trains. Several patterns of activity are apparent: spike-and-wave, low-amplitude irregular activity, high-amplitude irregular spiking, and oscillatory responses.

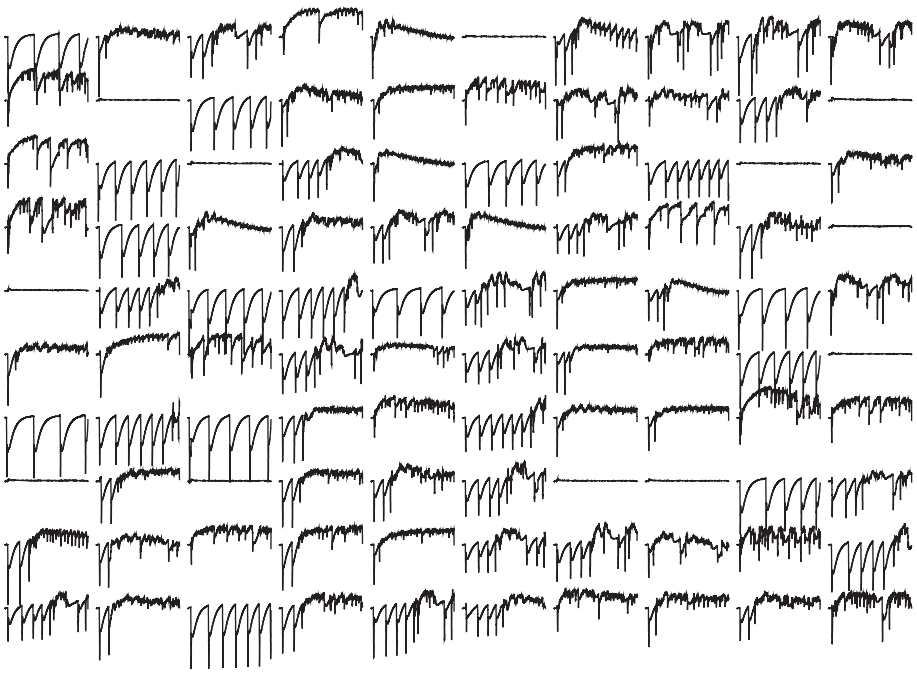

FIGURE 1.7 100 simulated electrographic field traces from 138 240 simulation parameter exploration.

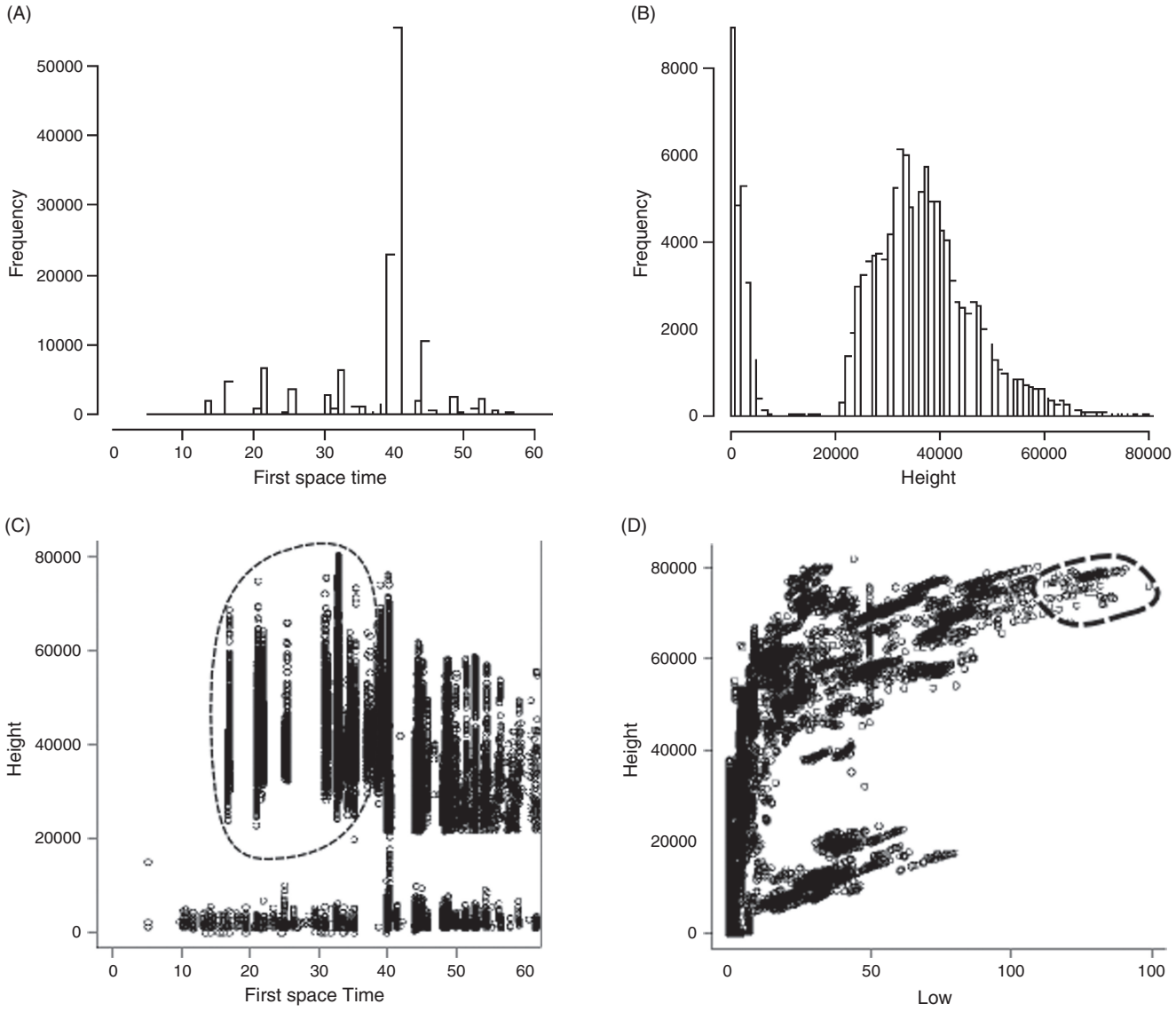

FIGURE 1.8 Statistics of early spikes.

We have begun our exploration of these simulations by looking at simulations that are 'spontaneously' active, meaning that they begin spiking due to the introduction of small random excitatory postsynaptic potentials prior to the large 'shock' stimulation at 40 ms. We do this by identifying population spike times in the simulated field, measuring such characteristics as size, width and sharpness of the population spike as well as occurrence time for over 8 million individual population spikes. In this simulation set, 23% of traces showed a first spike time prior to the shock-stimulation at 40 ms (Figure 1.8). We identified a bimodal distribution of average spike heights, with 83% of traces showing an average spike height of greater than 20000 (arbitrary units; Figure 1.8B). Combining these measures demonstrated that 19% of traces had an early spike time and large average spike size (dashed outline in Figure 1.8C). Looking at height and width for individual first spikes for these simulations (rather than per-trace averages) shows several apparent clusters. We looked more closely at the outlying cluster (dashed outline in Figure 1.8D).

The associated set of 110 simulations displayed large repetitive spikes, distinct from the spike-and-wave pattern seen in other traces (Figure 1.9). Some of these persisted through the full second of simulation, while others gave way to low amplitude activity. Parameter analysis of this simulation set demonstrated that these simulations were produced by a combination of low threshold (high neuronal excitability) with high inhibition. Surprisingly, this pattern is associated with relatively low strength of excitatory interconnectivity.

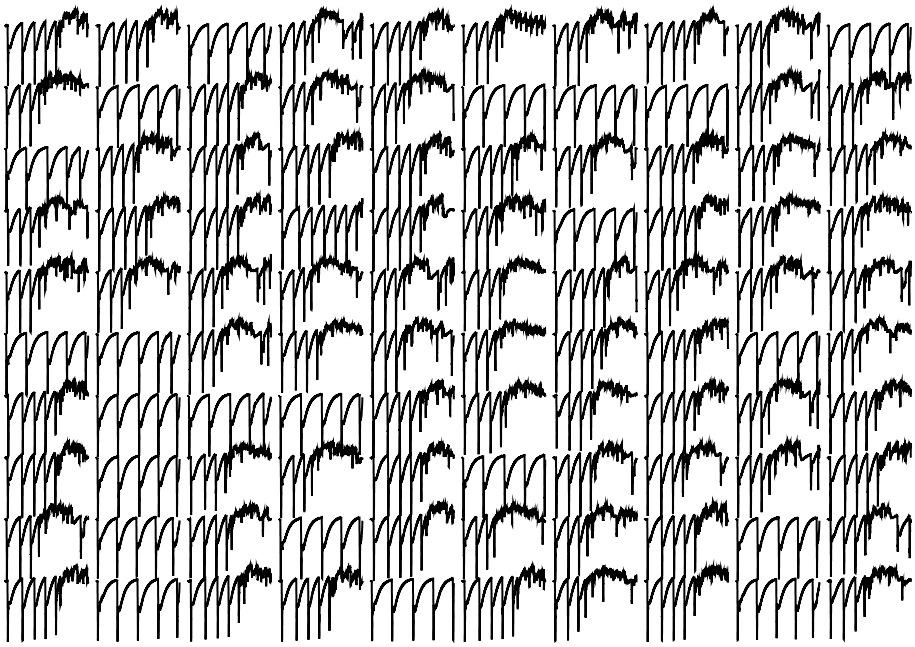

FIGURE 1.9 100 traces with large wide early spikes taken from circled area of Figure 1.8D.

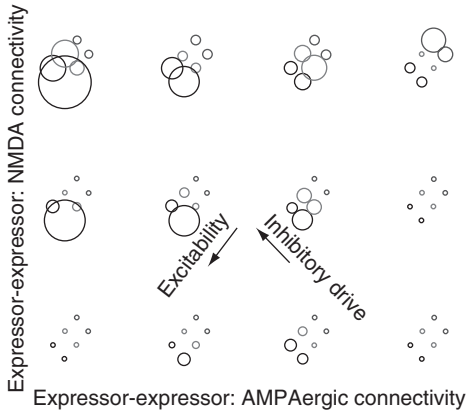

FIGURE 1.10 Parameter dependence of early spikes.

This observation prompted further analysis of the interplay of excitatory interconnectivity and cell excitability producing early large spikes. Figure 1.10 shows the number of early large spikes (size of circle) based on four parameters: excitatory-excitatory AMPAergic connectivity (x-axis), excitatory-excitatory NMDAergic connectivity (y-axis), threshold (first oblique axis; higher threshold produces lower excitability) and inhibitory drive (second oblique axis). Large early spike incidence (large circles) is generally associated with high neuronal excitability but with low AMPAergic connectivity. However, in the setting of high AMPAergic connectivity, increased excitability has the opposite effect, reducing the incidence of large early spikes. Another paradoxical effect is seen with respect to inhibition: large early spikes are associated with high inhibition (not shown) but with low drive onto the inhibitory cells (second oblique axis).

RATIONAL PHARMACOTHERAPEUTICS

The foregoing data-mining exercise represents one approach to the goal of rational pharmacotherapeutics: identifying underlying parameters that promote particular network dynamics so as to suggest targets for interventions that would

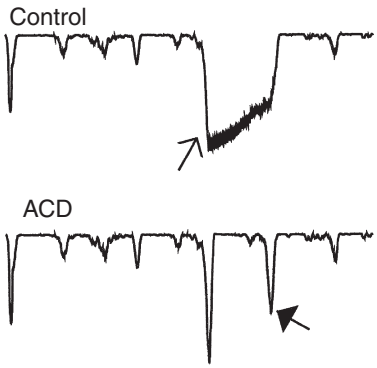

FIGURE 1.11 Effect of burst-suppressing anticonvulsant simulation on tonic activity. ACD: anti-convulsant drug.

convert those dynamics to non-pathological forms. Another approach involves using simulation to map known neuron-level pharmacological effects onto the network level. In general, these neuron-level effects will be expected to reduce identifiable epileptiform characteristics at the network level. On the other hand, our experience with parameter exploration suggests that there will be some parameter domains (subspaces) for which a single parameter alteration will show a paradoxical effect, as demonstrated in Figure 1.10. Thus, it should not be surprising that anticonvulsants can sometimes exacerbate seizures under particular pathological conditions or in some dose ranges.

Several of the variety of current and past anticonvulsants have been found to affect voltage-dependent sodium channels so as to reduce bursting in excitatory cells. This includes older anticonvulsants, such as phenytoin and carbamazepine, as well as newer agents such as lamotrigine (Rogawski and Loscher, 2004a, 2004b). We therefore have looked at the effect of reduction of burst duration on network dynamics.

Above we discussed the prevalence of latch-up tonic activity in simulations with strong excitatory-excitatory connectivity (see Figure 1.5). We looked at the effect of reducing burst size on this latch-up condition. Our preliminary data suggest that this pharmacological effect tends to reduce latch-up with little effect on other aspects of neural dynamics. In Figure 1.11, the latch-up period is eliminated (open arrow) without other alterations of the timing of population spike activity, except for the addition of a spike simultaneous with the termination of latch-up in the control condition (closed arrow). This additional spike in the ACD condition represents coincident synaptic activity. We hypothesize that the same coincident activity that produced a population spike in the ACD condition served conversely to terminate the latch-up condition in the control simulation.

CONCLUSIONS

Epilepsy, like bad weather, is an ideal pathological condition to explore with simulation technologies. It is primarily a dynamical disease, meaning that it is the expression of enormous numbers of feedback loops, negative and positive, at both short and long intervals. Such dynamics are quite naturally expressed in mathematical form using differential equations. Like the atmosphere, the brain is huge and is subject to innumerable changing influences, external and internal, which are difficult and sometimes impossible to pin down. Nonetheless, more information is coming in all the time from experiments. Simulation technology is advancing. In the next few years we should begin to see the fruits of this convergence with implications for seizure prediction, seizure treatment and epilepsy prophylaxis.

REFERENCES

Borg-Graham, L.J. (1991). Modeling the non-linear conductances of excitable membranes. In: Cellular and molecular neurobiology: a practical approach (H. Wheal, and J. Chad, ed) pp. 247–275. Oxford, New York.

Brunel, N. (2000). Dynamics of networks of randomly connected excitatory and inhibitory spiking neurons. J Physiol (Paris) 94:445–463.

Brunel, N. and Wang, X.J. (2003). What determines the frequency of fast network oscillations with irregular neural discharges? I synaptic dynamics and excitation-inhibition balance. J Neurophysiol 90:415–430.

Bush, P.C. and Sejnowski, T.J. (1991). Simulations of a reconstructed cerebellar Purkinje cell based on simplified channel kinetics. Neural Comput 3:321–332.
Bush, P.C. Prince, D.A. and Miller, K.D. (1999). Increased pyramidal excitability and NMDA conductance can explain posttraumatic epileptogenesis without disinhibition: a model. J Neurophysiol 82:1748–1758.
Carnevale, N.T. and Hines, M.L. (2006). The NEURON Book. Oxford, New York.
Cohen, S.D. and Hindmarsh, A.C. (1994). Cvode user guide. Technical report. Lawrence Livermore National Laboratory, Livermore.
Delorme, A. and Thorpe, S.J. (2003). Spikenet: an event-driven simulation package for modelling large networks of spiking neurons. Netw Comput Neural Syst 14:613–627.
Destexhe, A. (1997). Conductance-based integrate-and-fire models. Neural Comput 9:503–514.
Destexhe, A. Contreras, D. Steriade, M. Sejnowski, T.J. and Huguenard, J.R. (1996). In vivo, in vitro and computational analysis of dendritic calcium currents in thalamic reticular neurons. J Neurosci 16:169–185.
Destexhe, A. and Sejnowski, T.J. (1995). G-protein activation kinetics and spillover of gamma-aminobutyric acid may account for differences between inhibitory responses in the hippocampus and thalamus. Proc Nat Acad Sci USA 92:9515–9519.
Dominguez, L. Wennberg, R.A., Gaetz, W., Cheyne, D. and Velazquez, J.L. (2005). Enhanced synchrony in epileptiform activity? Local versus distant phase synchronization in generalized seizures. J Neurosci 25:8077–8084.
Dudek, F.E. Patrylo, P.R. and Wuarin, J.P. (1999). Mechanisms of neuronal synchronization during epileptiform activity. Adv Neurol 79:699–708.
Hansel, D., Mato, G., Meunier, C. and Neltner, L. (1998). On numerical simulations of integrate-and-fire neural networks. Neural Comput 10:467–483.
Hereld, M., Stevens, R.L., Teller, J. and van Drongelen, W. (2005). Large neural simulations on large parallel computers. Int J Bioelectromagn 7:44–46.
Hindmarsh, A.C. and Serban, R. (2002). User documentation for cvodes, an ode solver with sensitivity analysis capabilities. Technical report. Lawrence Livermore National Laboratory, Livermore.
Hines, M.L. (1989). A program for simulation of nerve equations with branching geometries. Int J Biomed Comput 24:55–68.
Izhikevich, E.M. (2003). Simple model of spiking neurons. IEEE Trans Neural Netwo 14:1569–1572.
Izhikevich, E.M. (2004). Which model to use for cortical spiking neurons. IEEE Trans Neural Netwo 15:1063–1070.
Jahr, C.E. and Stevens, C.F. (1990). A quantitative description of NMDA receptor-channel kinetic behavior. J Neurosci 10:1830–1837.
Lytton, W.W. (2006). Neural query system: data-mining from within the neuron simulator. Neuroinformatics 4:163–176.
Lytton, W.W. and Hines, M. (2005). Independent variable timestep integration of individual neurons for network simulations. Neural Comput 17:903–921.
Lytton, W.W. and Sejnowski, T.J. (1991). Inhibitory interneurons may help synchronize oscillations in cortical pyramidal neurons. J Neurophysiol 66:1059–1079.
Lytton, W.W. and Stewart, M. (2005). A rule-based firing model for neural networks. Int J Bioelectromagn 7:47–50.
Lytton, W.W. and Stewart, M. (2006). Rule-based firing for network simulations. Neurocomputing 69:1160–1164.
MacGregor, R.J. (1987). Neural and Brain Modeling. Academic Press, Inc., San Diego.
Mainen, Z.F. and Sejnowski, T.J. (1996). Influence of dendritic structure on firing pattern in model neocortical neurons. Nature 382:363–366.
Makino, T. (2003). A discrete-event neural network simulator for general neuron models. Neural Comput Appl 11:210–223.
Mattia, M. and Del Giudice, P. (2000). Efficient event-driven simulation of large networks of spiking neurons and dynamical synapses. Neural Comput 12:2305–2329.
Meeren, H.K., Pijn, J.P., Luijtelaar, E.L., Coenen, A.M. and Silva, F.H. (2002). Cortical focus drives widespread corticothalamic networks during spontaneous absence seizures in rats. J Neurosci 22:1480–1495.
Migliore, M., Cannia, C., Lytton, W.W. and Hines, M.L. (2006). Parallel network simulations with neuron. J Comput Neurosci 6:119–129.
Morrison, A., Mehring, C., Geisel, T., Aertsen, A.D. and Diesmann, M. (2005). Advancing the boundaries of high-connectivity network simulation with distributed computing. Neural Comput 17:1776–1801.
Pinsky, P. and Rinzel, J. (1994). Intrinsic and network rhythomogenesis in a reduced Traub model for CA3 neurons. J Comput Neurosci 1:39–60.
Poirazi, P., Brannon, T. and Mel, B.W. (2003). Arithmetic of subthreshold synaptic summation in a model cal pyramidal cell. Neuron 37:977–987.
Pytte, E., Grinstein, G. and Traub, R.D. (1991). Cellular automaton models of the CA3 region of the hippocampus. Network 2:149–167.
Rhodes, P.A. (2006). The role of gap junctions in the context of a simulated thalamocortical column. Soc Neurosci Abstr 132.13.
Rogawski, M.A. and Loscher, W. (2004a). The neurobiology of antiepileptic drugs. Nat Rev Neurosci 5:553–564.
Rogawski, M.A. and Loscher, W. (2004b). The neurobiology of antiepileptic drugs for the treatment of nonepileptic conditions. Nat Med 10:685–692.
Rudolph, M. and Destexhe, A. (2006). Analytical integrate-and-fire neuron models with conductance-based dynamics for event-driven simulation strategies. Neural Comput 18:2146–2210.
Rulkov, N.F., Timofeev, I. and Bazhenov, M. (2004). Oscillations in large-scale cortical networks: map-based model. J Comput Neurosci 17:203–223.
Shelley, M.J. and Tao, L. (2001). Efficient and accurate time-stepping schemes for integrate-and-fire neuronal networks. J Comput Neurosci 11:111–119.
Sperelakis, N. and Ramasamy, L. (2006). Propagation velocity profile in a cross-section of a cardiac muscle bundle from pspice simulation. Theor Biol Med Model 3:29.
Traub, R.D. and Wong, R.K.S. (1982). Cellular mechanism of neuronal synchronization in epilepsy. Science 216:745–747.
Traub, R.D., Contreras, D. and Whittington, M.A. (2005). Combined experimental/simulation studies of cellular and network mechanisms of epileptogenesis in vitro and in vivo. J Clin Neurophysiol 22:330–342.
Van Drongelen, W., Koch, H., Marcuccilli, C., Pena, F. and Ramirez, J.M. (2003). Synchrony levels during evoked seizure-like bursts in mouse neocortical slices. J Neurophysiol 90:1571–1580.
Watts, L. (1994). Event-driven simulation of networks of spiking neurons. In: Advances in neural information processing systems, vol 6 (J.D. Cowan, G. Tesauro, and J. Alspector, eds.) pp. 927–934. Morgan Kaufmann Publishers.
Wilson, H.R. and Cowan, J.D. (1972). Excitatory and inhibitory interactions in localized populations of model neurons. Biophys J 12:1–24.
Wilson, M.T., Sleigh, J.W., Steyn-Ross, D.A. and Steyn-Ross, M.L. (2006). General anesthetic-induced seizures can be explained by a mean-field model of cortical dynamics. Anesthesiology 104:588–593.