Extended dynamic clamp: controlling up to four neurons using a single desktop computer and interface

R.D. Pinto a,*, R.C. Elson a, A. Szücs a, M.I. Rabinovich a, A.I. Selverston a, H.D.I. Abarbanel b

a Institute for Nonlinear Science, University of California, San Diego, 9500 Gilman Dr. # 0402, La Jolla, CA 92093-0402, USA
b Department of Physics and Marine Physical Laboratory, Scripps Institution of Oceanography, University of California, San Diego, La Jolla, CA 92093-0402, USA

Received 21 February 2001; received in revised form 18 April 2001; accepted 18 April 2001

Abstract

The dynamic clamp protocol allows an experimenter to simulate the presence of membrane conductances in, and synaptic connections between, biological neurons. Existing protocols and commercial ADC/DAC boards provide ready control in and between 2 neurons. Control at > 2 sites is desirable when studying neural circuits with serial or ring connectivity. Here, we describe how to extend dynamic clamp control to four neurons and their associated synaptic interactions, using a single IBM-compatible PC, an ADC/DAC interface with two analog outputs, and an additional demultiplexing circuit. A specific C++ program, DYNCLAMP4, implements these procedures in a Windows environment, allowing one to change parameters while the dynamic clamp is running. Computational efficiency is increased by varying the duration of the input–output cycle. The program simulates ≤ 8 Hodgkin–Huxley-type conductances and ≤ 18 (chemical and/or electrical) synapses in ≤ 4 neurons and runs at a minimum update rate of 5 kHz on a 450 MHz CPU. (Increased speed is possible in a two-neuron version that does not need auxiliary circuitry). Using identified neurons of the crustacean stomatogastric ganglion, we illustrate on-line parameter modification and the construction of three-member synaptic rings. © 2001 Elsevier Science B.V. All rights reserved.

Keywords: Dynamic clamp; Artificial synapses; Simulation; Hodgkin–Huxley; Conductance; Neural circuits; Synaptic rings

1. Introduction

The dynamic current clamp protocol is a powerful, computer-based method that allows an experimentalist to inject voltage- and time-dependent currents into a neuron, so as to simulate the presence of specific ionic conductances in the neuronal membrane (Robinson and Kawai, 1993; Sharp et al., 1993a,b). The computer monitors the membrane potential of the neuron under study by intracellular recording and calculates the current to be injected according to equations supplied by the experimenter. The technique has two main applications. First, one can study the contribution of particular voltage-dependent, ligand-gated, or leak conductances to the electrophysiology of individual neurons (Sharp et al., 1993a,b; Gramoll et al., 1994; Harris-Warrick et al., 1995; Ma and Koester, 1996; Turrigiano et al., 1996; Hughes et al., 1998, 1999; Bartos et al., 1999; Kinard et al., 1999). Second, one can simulate the action of postsynaptic currents. The simulated synaptic input may be driven by patterns of voltage signals defined by the experimenter (Reyes et al., 1996; Ulrich and Huguenard, 1996, 1997; Bartos et al., 1999; Jaeger and Bower, 1999), or by the membrane potential recorded in a second cell (Sharp et al., 1993a,b, 1996; Wilders et al., 1996). In the latter case, the dynamic clamp inserts a simulated synaptic connection between the two neurons; in this way one can begin to build up artificial synaptic circuits with neurons among which natural synapses are blocked or non-existent.

Implementation of the dynamic clamp requires hardware, an ADC/DAC interface so that the computer can acquire voltage signals and generate current commands; and software programs that calculate the current equa-
neurons, thereby allowing the experimenter to disconnect a neuron from its connections without altering either cellular or synaptic properties within artificial circuits constructed by inserting simulated conductances. In principle, this protocol may be extended by the action of neuromodulators (Harris-Warrick and Marder, 1991; Harris-Warrick et al., 1992a; Katz et al., 1994). In this context, the dynamic clamp is an important experimental tool, because it allows one to observe the functional consequences of manipulating either biological neurons (Sharp et al., 1993a,b, 1996). However, this method is constrained by (a) the need to complete the cycle in a timely manner while computing multiple equations, and (b) the need to generate output signals to control multiple cells. The main methodological advances that we describe here are concerned with surmounting these two constraints.

2. Methods

2.1. Dynamic clamp cycle

Implementation of the basic dynamic current clamp protocol involves repetition of a three-stage cycle: 1) Acquisition. A computer reads the neuronal membrane potential, recorded and amplified by a conventional intracellular electrometer, via an analog to digital converter (ADC). 2) Computation. Using the sampled membrane potential together with mathematical models of membrane and/or synaptic conductances, the computer calculates the current, which would be generated by the chosen combination of conductances if they were actually present in the neuron (at the site of voltage recording). 3) Command. The computer commands the injection of the calculated current by generating an output voltage signal through a digital to analog converter (DAC). The appropriately scaled voltage is applied to the voltage-to-current circuit of the electrometer, thus commanding the injection of current into the neuron.

Repetition of this cycle ensures that the current injected into each neuron varies dynamically with the recorded membrane potential, according to the set of simulated conductances. In principle, this protocol may be performed in parallel for as many neurons as the experimenter can record. In practice, the implementation is constrained by (a) the need to complete the cycle in a timely manner while computing multiple equations, and (b) the need to generate output signals to control multiple cells. The main methodological advances that we describe here are concerned with surmounting these two constraints.
2.2. Computing the dynamic clamp cycle in an asynchronous mode

For computational efficiency, we implement the dynamic clamp cycle in an asynchronous mode, such that the time taken to execute each cycle depends on the number and the complexity of the conductances being simulated. At each cycle, therefore, one reads not only the membrane potential, but also the time interval between this update and the last. The asynchronous mode is also advantageous in a single-processor multitask environment such as Windows NT or Linux, in which one never knows exactly when the CPU will briefly stop running the application to execute system tasks.

The DYNCLAMP4 application itself is written in C++ and runs in an IBM-compatible computer with a Pentium III microprocessor operating Windows NT 4.0. (In Windows NT, the CPU works in a protected mode, so that a software driver (Salaj, 2000) is needed to enable access to the input/output (I/O) ports of the ADC/DAC board). The program is also compatible with Windows 95, 98 and Windows 2000 operating systems, although its performance on these platforms was not fully tested. An additional reason for choosing the C++ language was to provide easier implementation in Linux or Real Time Linux, as well. We have not yet ported the program to a Linux platform; however, the provision of the source code allows individual Linux users to make their own implementations.

2.3. Obtaining up to four analog command voltages from the two analog outputs of a conventional ADC/DAC board

Most ADC/DAC boards offer only two analog outputs. To obtain up to four analog command signals from such an interface, we employed a multiplex–demultiplex procedure. First, a software routine multiplexes the (≤4) output signals of the dynamic clamp program, generating two analog voltages at the DAC output ports of the interface. Second, a hardware circuit is used to demultiplex the two DAC signals, reconstituting the original command signals as (≤4) analog voltages. For DYNCLAMP4 we configured the software and hardware circuit to work with the commonly used Digidata 1200A interface from Axon Instruments, but the general methodology applies to any comparable ADC/DAC board.

A simple electronic circuit can implement the analog demultiplex. This is not available commercially, but can be easily constructed by the user from the given description (Fig. 1). Up to four parallel, sample-and-hold integrated circuits (SMP04, Analog Devices) can be

---

**Fig. 1.** Schematic diagram of the analog demultiplex circuit. The signals CH0 and CH1 correspond to the analog outputs of the ADC/DAC board, D0-D3 are the first four digital outputs of the ADC/DAC board, and I0-I3 are the outputs of the analog demultiplex that contains the encoded current to be injected into the neurons.
used to generate up to four analog output voltages (I0-I3), suitably amplified for use as command signals for current injection by the microelectrode electrometers. The two, multiplexed voltage signals produced at the DAC board outputs (CH1 and CH2) are fed to the four sample-and-hold circuits through switches controlled by monostable TTL circuits (U3 and U4). The TTL circuits are, in turn, commanded by pulses from four digital outputs of the ADC/DAC board (D0-D3). At the onset of a command pulse, a given sample-and-hold circuit is charged to the voltage which it sees, at that instant, in the DAC signal to which it has access. After an initial charging time of 15 µs, the sample-and-hold circuit sustains the sampled voltage until the onset of the next command pulse (Fig. 2).

2.4. Control panel and multiprocessing

We implemented all controls — of program and parameters — in a single window, giving the user exact information at a glance. All possible combinations of single chemical (12) and electrical synapses (6) between \(\leq 4\) cells are allowed. However, because space is constrained on a single screen, we limit the total number of Hodgkin–Huxley conductances that can be added to the set of neurons to eight. Nevertheless, the software itself is not constrained; one can easily change the code to add to a second window more conductances and synapses (to simulate biphasic synapses, for example).

To allow an experimenter to change particular parameter settings without terminating the dynamic clamp protocol, one can structure the program as in Fig. 3. When first invoked, the program resets the ADC/DAC board, and initializes all parameter values to default values while setting all synapses and conductances to off. The two main routines of the program are the dynamic clamp routine (initiated by the command, ‘start/update’), and a routine that resets and keeps the outputs of the analog demultiplex at null voltage (‘demux reset’; initiated by commanding ‘stop/reset’) (Fig. 3). The routines are implemented as C++ threads (i.e. sub-processes that once started are able to run in parallel to the main program until they finish by themselves or are stopped by a specific command). The threads are executed at higher priority at the same time as the program (at a lower priority) checks for changes in the control panel. Changes in parameters can be entered on the control panel while the dynamic clamp is still running, then implemented by commanding ‘start/update’.

2.5. Mathematical models of synapses and membrane conductances

The equations simulating synaptic operation and membrane conductances generally follow the formulations used by Sharp et al. (1993a,b) and Nadim et al. (1999). In the DYNCLAMP programs, voltage-dependent membrane conductances are modeled using direct formulations for (in)activation variables and their time constants, rather than expressions for rate variables, to permit easier simulation of currents measured in voltage-clamp experiments. A simple modification of the model of electrical coupling allows for rectification. The only detailed description provided here is for the chemical synapse simulations used in the experiments illustrated in Section 3.

A chemical synaptic connection between two neurons uses a first-order kinetics model of transmitter release (Sharp et al., 1993a,b; Destexhe et al., 1994). To add the option of ‘depression’ of release, we include an inactivation term in the current equation (cf. Nadim et al., 1999):

\[
I_C = g_{\text{syn}}S(t)h(t)(V_{\text{rev}} - V_{\text{post}}(t)),
\]

where \(I_C\) is the postsynaptic current, \(g_{\text{syn}}\) is the maximal synaptic conductance; \(V_{\text{rev}}\) is the synaptic reversal potential, and \(V_{\text{post}}\) is the instantaneous postsynaptic membrane potential. Transmitter release is modeled by the instantaneous activation term, \(S(t)\), and the inactivation term, \(h(t)\). \(S(t)\) is given by the differential equations...
(1 - S\(\infty\)(V\(\text{prev}\)))\(\frac{dS(t)}{dt}\) = \((S\(\infty\)(V\(\text{prev}\)) - S(t))\),

where

\(S\(\infty\)(V\(\text{prev}\)) = \tanh\left(\frac{V\(\text{prev}\) - V_{\text{thres}}}{V_{\text{slope}}}\right)\) when \(V\(\text{prev}\) > V_{\text{thres}}\),

otherwise \(S\(\infty\)(V\(\text{prev}\)) = 0;\)

and \(V\)\(\text{prev}\) is the membrane potentials of the presynaptic neuron; \(S\)\(\infty\) is the steady state synaptic activation; \(\tau_S\) is the synaptic characteristic time constant; \(V_{\text{thres}}\) is the synaptic threshold voltage; \(V_{\text{slope}}\) is the synaptic slope voltage.

The inactivation term, \(h(t)\), is modeled by

\[\tau_h(V\(\text{prev}\)) \frac{dh(t)}{dt} = h_{\text{prev}}(V\(\text{prev}\)) - h(t),\]

where \(h_{\text{prev}}\) is the steady state synaptic inactivation and \(\tau_h\) is its time constant. These variables are governed by Hodgkin–Huxley-type functions (cf. Hodgkin and Huxley, 1952; Nadim et al., 1999). For synapses without inactivation the program keeps the term \(h(t) = 1\).

At each clamp cycle we have

\[S(t) = S_{\text{prev}} + (S_{\text{prev}} - S_{\infty})\exp\left[\frac{-\Delta t}{(S_{\infty} - 1)\tau_S}\right],\]

where \(S_{\text{prev}}\) and \(S_{\text{prev}}\) are the values of \(S\) and \(S_{\infty}\) calculated in the earlier cycle and \(\Delta t\) is the time interval since the last update.

2.6. Experimental techniques

Adult spiny lobsters, \(\text{Panulirus interruptus}\), were caught locally and kept in running seawater until use. The stomatogastric nervous system (Harris-Warrick et al., 1992b), consisting of the stomatogastric ganglion (STG) and anterior (commissural and esophageal) ganglia and their connecting and motor nerves, was removed from the foregut (Mulloney and Selverston, 1974) and pinned out in a silicone elastomer (Sylgard)-lined dish, filled with normal saline (in mM: 479 NaCl, 13 KCl, 14 CaCl\(_2\), 6 MgSO\(_4\), 4 Na\(_2\)SO\(_4\), 5 HEPES, and 5 TES; pH 7.4). The STG was separately superfused by a continuous flow of chilled saline (14–17°C; temperature was kept within 1°C during each recording session), using a Vaseline chamber to which drugs were added as needed. The ganglion was desheathed and the somata of pyloric neurons impaled by microelectrodes (filled with 3 M KCl; resistance 10–20 MΩ). Neurons were identified by their characteristic phase of bursting and by correlation of spikes with impulses monitored simultaneously from output motor nerves. Conventional electrometers (Neuroprobe 1600 current-clamp amplifiers) were used both to amplify the voltage signals from the cellular membrane as well as to insert current signals into the cells.

In the intact STG, the neurons of the pyloric central pattern generator (CPG) are connected to each other by a known set of chemical and electrical synapses (Miller, 1987; Harris-Warrick et al., 1992b). Two methods were used to disconnect this natural circuit so as to yield pyloric neurons in ‘synaptic isolation’. (1) Pharmacological blockade of chemical synapses. We used up to 10 mM picrotoxin (PTX) to block glutamatergic inhibition and 0.6–2 mM tetraethylammonium (TEA) to block slower cholinergic inhibition (Eisen and Marder, 1982; Marder and Eisen, 1984). (2) Photoinactivation of electrically-coupled neurons (Miller and Selverston, 1979; Selverston and Miller, 1980). As long as the STG continued to receive descending modulatory inputs from anterior ganglia, synaptically-isolated pyloric neurons could continue to produce slow oscillations of membrane potential with bursts of spikes (Bal et al., 1988).
3. Results

3.1. Update rate capability of the extended dynamic clamp

The minimum update rate of the dynamic clamp cycle was measured with all 18 synapses and eight Hodgkin–Huxley conductances activated. Under these conditions the processor needs to calculate the effect of all conductances in the currents for each cell, and the performance will be the slowest possible. The minimum update rate, for a Pentium III 450 MHz computer running only the dynamic clamp application under Windows NT 4.0, measured directly from one of the digital gating pulses (D0-D3, see Fig. 2), was 5 kHz (the longest period between two successive updates of the same current channel was 200 µs). With all the conductances and synapses inactivated (no calculations need to be done) the maximum update rate was 5.1 kHz, which indicates the number of synapses and conductances can be increased even more since the limiting factor for the speed are the ADC/DAC operations.

To illustrate the use of these extended dynamic clamp methods, we present two examples of constructing, and modifying the operation of, artificial synaptic circuits involving pyloric neurons of the lobster STG.

3.2. Changing parameters while the dynamic clamp is running

The first example illustrates an on-line change of dynamic clamp parameters, in this case the maximal postsynaptic conductances (effectively, synaptic strength) of reciprocal inhibitory connections between a pair of neurons (Fig. 5). In this experiment, photoablation and pharmacological blockade had been used to remove or greatly weaken natural synaptic interactions within the pyloric CPG, such that the LP and PD neurons were spiking and bursting independently of each other. As shown in the top panel of Fig. 5, with DYNCLAMP4 off, the ongoing activity of both neurons consisted of irregular bursts of action potentials (spikes) riding on slower oscillations of membrane potential. (Spikes appear attenuated because they are initiated at some distance from the recording site in the inexcitable soma). At the time marked by the dashed line, the dynamic clamp was turned on, inserting reciprocal inhibitory synapses between LP and PD (see Fig. 4C, test circuit 1). The neurons began to burst in antiphase and also more regularly. The inhibitory postsynaptic currents (IPSCs) generated by the dynamic clamp are shown in the current monitor traces for the two neurons (Fig. 5, test circuit 1). When either neuron generated a burst, it evoked a compound IPSC in the other. This IPSC comprised both spike-evoked and graded components (because the synaptic threshold voltage was set more negative than the spike threshold). During a burst in the presynaptic neuron, the two components of the IPSC underwent ‘depression’ determined by the voltage-dependence and kinetics of an inactivation term included in the presynaptic ‘release’ function (see Section 2.5).

The bottom panel of Fig. 5 shows activity some minutes later in the same experiment. The dynamic clamp was still on, but the maximal postsynaptic conductance ($g_{syn}$) of both connections had been reduced to lower values than those used in the top panel. LP and PD still showed a tendency to burst in antiphase, but the activity was less well coordinated and more irregular. At the dashed line, both maximal conductances were reset to larger values while the clamp was still on.
and running. The adjustment occurred as the PD neuron was depolarized and firing, and an immediate increase is apparent in the IPSC and postsynaptic voltage response in LP.

3.3. Constructing a three-member synaptic ring circuit

The second example shows the construction of a three-member synaptic ring circuit. The ring circuit was inspired by a simplification of the pyloric CPG (Miller, 1987) which preserves the strongest inhibitory synapses but lumps together in three subsets those neurons which are electrically coupled or otherwise co-active (because their mutual inhibition is relatively weak) (Fig. 4B). Due to their intrinsic properties and the asymmetric connectivity, the three groups normally burst rhythmically in the sequence: AB/PD, LP/IC, PY/VD (Miller, 1987).

In the experiment illustrated in Fig. 6, we attempted to re-constitute a portion of this synaptic ring after some neurons (AB and VD) had been photoinactivated, and remaining chemical synapses had been blocked. Before turning on the dynamic clamp, the PD, LP and PY neurons were generating rhythmic bursts (Fig. 6, left panel, DYNCLAMP4 off). The activity in PD and LP was uncoordinated; that in LP and PY showed some coordination due to a weak, rectifying electrical connection (cf. Fig. 4A). (The intensified bursting was probably a cellular membrane effect of the higher concentration of TEA used). Subsequently, we turned on the dynamic clamp, inserting five simulated inhibitory connections in a connectivity pattern mimicking that of the pyloric CPG (test circuit 2, Fig. 4D). When connected in this synaptic ring, the bursting of the three neurons became synchronized in a three-phase pattern with an approximate 120° phase shift; the burst sequence of these neurons was PD, LP, PY, as in the intact circuit (Fig. 6, right panel, DYNCLAMP4 on). The two phases of synaptic input received by LP and PY are apparent in their current traces. The IPSC
4. Discussion

Since its effective introduction by Sharp et al. (1993a,b), the dynamic clamp protocol has been a valuable tool for studying the functional roles of specific membrane and synaptic properties in the electrophysiology of neurons in many preparations (Marder, 1998). For research upon circuits of neurons, however, the technique has been limited to studying synaptic pairs (Sharp et al., 1996; Wilders et al., 1996). Our present work extends dynamic clamp capabilities in a number of ways, with the main advance occurring in the number of neurons controlled, and thus the complexity of synaptic circuit which can be manipulated or constructed.

The methods we describe extend dynamic clamp control to ≤4 neurons and their interconnections by adding a simple analog demultiplex circuit to a conventional ADC/DAC interface. Earlier commercial and custom-designed applications have been limited to controlling a maximum of two neurons, doubtless because of the corresponding limitation of analog output channels in most ADC/DAC boards. Additionally, our approach uses nested C++ subroutines to maintain real-time dynamic clamp control while the experimenter is free to enter new parameters in a Windows-type graphical interface; activating an update command causes the operative parameter values to change to the new settings while the clamp continues to run. This means that the effects of changing maximal conductances, voltage-dependencies, kinetics, etc. can be tested without the possible disruptive effects of stopping and restarting dynamic clamp control. We found that the cycle of acquiring voltage signals, computing equations and commanding current outputs was accelerated by running C++ subroutines, as opposed to using virtual instruments within Windows. Speed was further increased by varying the duration of the input–output cycle. This way of implementing the dynamic clamp combines the speed of customized programs with the ease of Windows oversight.

Exerting control at more than two sites allows manipulation, or construction, of more complex circuits than the pairs of neurons earlier studied. The dynamic clamp technique can now be applied to synaptic chains or rings of neurons. Synaptic rings are a common ‘building-block’ (Getting, 1989a) in the functional architecture of CPGs. Rings of three neurons (or small subsets of neurons) contribute to the rhythmical operation in certain states of the pyloric and gastric CPGs of
the lobster STG (Elson and Selverston, 1992; Miller, 1987) and the swimming CPGs of the mollusks, Tritonia (Getting, 1989b) and Clione (Arshavsky et al., 1998). A three-membered ring architecture has been used in modeling pattern generation for simple walking movements of a single leg (Chiel et al., 1999). For modeling the coordination of different legs in quadrupedal vertebrates, some investigators employ a synaptic ring of four ‘oscillators’ (Canavier et al., 1997, 1999; Collins and Richmond, 1994). With extended dynamic clamp capability, one can modify membrane or synaptic properties in and among rings of four neurons within an existing synaptic circuit, or construct three- or four-member chains or rings de novo using simulated synapses. Targeted experimental manipulations of several neurons and/or synapses allow one to explore the dynamical function of these more complex networks. It also becomes possible to insert synapses between variable numbers of biological and electronic neurons (Pinto et al., 2000b; Szücs et al., 2000). We note that the beginnings of a similar approach were reported by Le Masson et al. (1995). However, these authors did not publish their methods in a form that could be readily implemented by other researchers.

The four-neuron program that we have described, DYNCAMP4, and a faster two-neuron version (DYNCMP 2, not requiring auxiliary circuitry) can be downloaded from the internet (Pinto, 2000) or obtained directly from the authors.

Acknowledgements

R.D. Pinto was supported by the Brazilian agency Fundação de Amparo à Pesquisa do Estado de São Paulo-FAPESP, under proc. 98/15124-5. Partial support for this work came from the US Department of Energy, Office of Science, under grants DE-FG03-90ER14138 and DE-FG03-96ER14592, and the US Office of Naval Research under grant N00014-00-1-0181. R.D. Pinto also acknowledges encouragement and discussion from José C. Sartorelli, Ramón Huerta, Pablo Varona, Mauro Copelli, Gregg Stiesberg, and Paco Rodríguez.

References


Nadim F, Manor Y, Kopell N, Marder E. Synaptic depression creates a switch that controls the frequency of an oscillatory circuit. Proc Natl Acad Sci USA 1999;96:8206–11.


Pinto RD. The program DYNCLAMP4, including source code and an operation manual, can be free downloaded from: http://inls.ucsd.edu/~rpinto/dynclamp.html. There is also a version for only two neurons, DYNCLAMP2, that do not need building extra electronic circuitry and can be found in the same web site, 2000.

Pinto RD, Elson RC, Szucs A, Selverston AI, Rabinovich MI, Abarbanel HDI. DYNCLAMP4: Inserting simulated synapses and ionic conductances in up to four neurons. Soc Neurosci Abstr 2000a;26:748.5.