High-speed dynamic-clamp interface

Yang Yang,1 Timothy Adowski,2 Bina Ramamurthy,2 Andreas Neef,3 and Matthew A. Xu-Friedman1

1Department of Biological Sciences, University at Buffalo, State University of New York, Buffalo, New York; 2Department of Computer Science and Engineering, University at Buffalo, State University of New York, Buffalo, New York; and 3Max Planck Institute for Dynamics and Self-Organization, Göttingen, Germany

Submitted 22 July 2014; accepted in final form 27 January 2015

Yang Y, Adowski T, Ramamurthy B, Neef A, Xu-Friedman M A. High-speed dynamic-clamp interface. J Neurophysiol 113: 2713–2720, 2015. First published January 28, 2015; doi:10.1152/jn.00543.2014.—The dynamic-clamp technique is highly useful for mimicking synaptic or voltage-gated conductances. However, its use remains rare in part because there are few systems, and they can be expensive and difficult for less-experienced programmers to implement. Furthermore, some conductances (such as sodium channels) can be quite rapid or may have complex voltage sensitivity, so high speeds are necessary. To address these issues, we have developed a new interface that uses a common personal computer platform with National Instruments data acquisition and WaveMetrics IGOR to provide a simple user interface. This dynamic clamp implements leak and linear synaptic conductances as well as a voltage-dependent synaptic conductance and kinetic channel conductances based on Hodgkin-Huxley or Markov models. The speed of the system can be assayed using a testing mode, and currently speeds of >100 kHz (10 μs per cycle) are achievable with short latency and little jitter.

WaveMetrics IGOR; Markov model

METHODS AND RESULTS

Overview of System Design

The dynamic clamp is implemented as a master-slave setup (Fig. 1A). The master is responsible for collecting and storing cellular responses and specifies conductance amplitude and time course \([G(t)]\) to be carried out by the dynamic clamp. This requires the familiar equipment normally required for voltage- or current-clamp recordings, i.e., computer, data acquisition system, and amplifier. The slave is responsible for monitoring the requested conductances at any given moment and calculating the appropriate currents. The slave setup measures the membrane potential \([V_m(t)]\) and compares against the reversal potential \((V_{rev})\) of each specified conductance to calculate the current output: \(I_{DC} = G(t) (V_m - V_{rev})\). The dynamic clamp also allows kinetic channel models, such as Hodgkin-Huxley-type models, where the conductance is effectively modeled in the slave setup. Once the user sets the dynamic clamp in motion on the slave setup, the master setup becomes the focus for data acquisition. This architecture allows the experimenter to test rapidly many different conductance parameters, which can be quite powerful.

This dynamic clamp is implemented using National Instruments boards. We have used M Series (6221 and 6229) and X Series (6361) boards using older Peripheral Component Interconnect (PCI) and faster PCI Express (PCIe) architecture. The board is controlled using an external operation (XOP) for WaveMetrics IGOR called “mafDC.” IGOR helps simplify the user interaction with the mafDC XOP to specify all of the gain settings for interacting with the amplifier and the parameters and voltage dependence for the various conductances (Fig. 1B). We have used a Dell OptiPlex 745 computer running Windows XP on the slave setup. Faster computers are available all of the time, which could provide additional speed improvements, particularly for computationally intensive components of the dynamic clamp. National Instruments drivers for Apple Macintosh computers are unfortunately inadequate to control a dynamic clamp.

General System Operation

The master setup functions like a familiar electrophysiology rig and is the primary setup that the experimenter interacts with. The experimenter typically executes multiple sweeps of data collection, specifying the conductances upfront, similar to a sweep of current injection (Fig. 1). When a data collection sweep is executed, the conductances are sent to the slave setup, and the resulting activity is recorded by the data acquisition system on the master and analyzed in the normal fashion.
The slave setup works quite differently. To achieve the fastest speeds, no data are saved on the slave. Rather, it constantly monitors the voltage of the cell and the various conductances. Usually the dynamic clamp is set going on the slave setup once at the beginning of the experiment and not interacted with again. Some conductances are controlled by the master setup through an analog-to-digital (A/D) channel on the slave setup (e.g., synaptic conductances), whereas others are tracked autonomously by the slave setup (e.g., channel conductances). Each conductance, $G(t)$, has its own $V_{rev}$. The amplifier informs the dynamic clamp of $V_m$ through one A/D channel, and the dynamic clamp calculates the current passed through that conductance:

$$I(t) = \frac{G(t)(V_m - V_{rev})}{V_{rev}} + G_2(V_m - V_{rev2}) + \ldots$$

The total current ($I_{DC}$) is sent to the command input on the amplifier (Fig. 1).

It is useful to think of the dynamic clamp on the slave setup as a cycle. On each cycle, all of the relevant A/D channels are read by the dynamic clamp, the kinetic conductances (if any) are updated, and the current is calculated and sent out of the D/A of the slave setup to the amplifier. A fast cycle time is ideal to make sure the channel states are up to date, and the current is appropriate to the momentary driving force. The calculations associated with dynamic clamp usually take very little time (see Test modes). The primary constraint is the speed of data acquisition, especially the number of A/D channels the slave setup must read. The mafDC XOP has a testing mode for measuring cycle time to make sure it meets the requirements of the experimenter.

The liquid junction potential is an issue for many recording configurations (Maertz 2008). If the amplifier pipette offset is adjusted to report the true membrane potential (online correction), then no special steps are needed for accurate dynamic clamp. If the pipette offset is not adjusted at the time of recording (offline correction), the value of the reversal potential for each conductance should be offset by the measured or estimated liquid junction potential before starting the experiment.

We recommend using two-electrode dynamic clamp if it is practical, one channel to read voltage and the other to pass current. This configuration prevents voltage transients arising from pipette capacitance, such as when large currents are applied to a model cell of nominal capacitance 30 pF and input resistance 100 MΩ (Fig. 2). Such transients are not passed to the cell but can interfere with measurements of membrane potential and analysis of spike timing. It is also possible to use single-electrode dynamic clamp, in which case bridge balance must also be used. Compensation of pipette capacitance is important for accurate clamp using both single- and dual-electrode configurations.

**Features**

The mafDC implementation includes a number of components. Any or all of them can be implemented simultaneously. The features are illustrated below, applying the dynamic clamp to a model cell of nominal capacitance 30 pF and input...
It is also possible to use the dynamic clamp for negative conductances such as to make a leaky cell more tight. For example, the model cell in Fig. 3B had nominal capacitance 30 pF and input resistance 20 MΩ (50 nS), so current injection caused a small change in voltage (Fig. 3B, middle). We applied a leak conductance of -40 nS with \( V_{\text{rev}} = +15 \) mV to generate a cell with 10-nS conductance and \( V_{\text{MC}} = (50 \text{ nS} \cdot 0 \text{ mV} + -40 \text{ nS} \cdot 15 \text{ mV})/(50 - 40 \text{ nS}) = -60 \) mV. The \( \tau \) of the model cell slowed from 0.7 to 3.6 ms, consistent with the fivefold increase in input resistance. The dynamic-clamp XOP was initiated on the slave setup using the IGOR command:

\[
\text{mafDC LEAK} = (-40, +15), \text{PULS} = \{2\},
\]

where the pulse parameter was as described above, and parameters dictated the conductance \(-40 \text{ nS} \) and \( V_{\text{rev}} (+15 \text{ mV})\).

**Linear synaptic conductances.** To mimic a synaptic conductance, the master computer sends out the time-varying synaptic conductance through a D/A channel to an A/D channel on the slave setup (Fig. 1, “G”). This is useful for representing AMPA or GABA receptor-mediated conductances. An example is shown in Fig. 4. The AMPA conductance itself is specified by the master setup, in this case based on voltage-clamp measurements of AMPA-mediated excitatory postsynaptic currents (EPSCs; Fig. 4, top black traces; Yang and Xu-Friedman 2009). The AMPA conductance was passed from a D/A on the master setup to the slave setup and applied by the dynamic clamp (Fig. 4, bottom black traces). The conductance specified by the master setup can be simple pulses (Fig. 4A) or complex trains of activity (Fig. 4B). The dynamic-clamp XOP was initiated on the slave setup using the IGOR command:

\[
\text{mafDC AMPA} = \{1,0\}, \text{LEAK} = \{8, -75\},
\]

where the leak parameters were as described above, and the AMPA parameters identified the A/D channel for the slave setup to read the instantaneous AMPA conducance in nanosiemens (channel 1) and the AMPA reversal potential (0 mV).

The XOP can also mimic inhibitory conductances using the GABA feature. This is useful for studying the interaction of

```plaintext
J Neurophysiol • doi:10.1152/jn.00543.2014 • www.jn.org
```
Excitatory and inhibitory linear conductances. The GABA conductance is specified just like the AMPA conductance, being sent out a D/A on the master setup to the slave setup. In the example of Fig. 4, the GABA was mimicked as spontaneously active, which visibly hyperpolarized the model cell during single AMPA conductances (Fig. 4A, blue trace) and interacted in a more complex way with the train of AMPA conductances (Fig. 4B, blue trace). The degree of excitation and inhibition depends on the reversal potential specified by the user. For the GABA trials, the dynamic-clamp XOP was initiated on the slave setup using the IGOR command:

```
mafDC AMPA={1,0},NMDA={2,18,-11},
LEAK={8,-75},
```

where the leak and AMPA were specified as above, and the NMDA parameters specified the A/D channel for the slave setup to read the instantaneous NMDA conductance in nanosiemens (channel 2), the half-activation voltage \( V_{\text{half}} = -11 \text{ mV} \), and the activation rate \( V_{\text{rate}} = 18 \text{ mV} \). These values were based on recordings of NMDA receptor-mediated EPSCs in bushy cells of the anteroventral cochlear nucleus (AVCN; Pliss et al. 2009). The most appropriate activation parameters may depend on the preparation.

**Hodgkin-Huxley-like sodium conductance.** The sodium conductance follows the form described by Hodgkin and Huxley (1952). The XOP tracks the state of three activation \( m \) gates and one inactivation \( h \) gate. These gates have voltage-dependent opening and closing rates, conventionally named \( \alpha_m, \alpha_h, \beta_m, \beta_h \). These voltage-dependent functions are specified by the user within IGOR as “waves.” These are then passed to the XOP as parameters, and the XOP takes care of updating the state of the gates and calculating the overall conductance. The user may specify the maximal sodium conductance either as a constant value or as a variable conductance determined by an A/D channel. The fixed option is the simplest way to create a constant sodium conductance. The unfixed option allows an experimenter to add different amounts of sodium conductance by changing the D/A output from the master setup (e.g., from 1,000 to 0 nS). The XOP updates the channel model using simple, explicit Euler integration according to the duration of the previous cycle.

An example is shown in Fig. 5. With no sodium conductance, a 20-nS AMPA conductance caused depolarization of the model cell (Fig. 5A, black trace). Then, the dynamic clamp was run with a sodium conductance with \( \alpha_n, \beta_n \), and \( \beta_h \) based on the model in Rothman and Manis (2003), which led to a spike (Fig. 5A, red trace). The dynamic-clamp XOP was initiated on the slave setup using the IGOR command:

```
mafDC/Na=1 AMPA={1,0},LEAK={8,-75},
NaV={1000,55,ah,bb,am,bm},
```

where the leak and AMPA were specified as above, and the NaV parameters specify the maximal sodium conductance (represented as \( S_{Na} \) in Hodgkin and Huxley (1952)), \( V_{\text{rev}} \) for sodium, and IGOR waves containing the voltage-dependent rate constants for \( \alpha_n, \beta_n, \alpha_m, \beta_m \) respectively. The command also included the option /Na=1 to indicate that the maximal sodium conductance was a fixed value (1,000 nS). When that flag is turned off (“/Na=0”), the XOP reads the maximal sodium conductance from an A/D channel on the slave setup, which allows experimenters to change the size of the sodium current flexibly during an experiment without starting and stopping the dynamic clamp.

---

*Fig. 4. Linear and nonlinear synaptic conductances implemented by dynamic clamp.* A 20-nS AMPA (black traces), a 1-nS GABA (blue traces), and a 2-nS NMDA (red traces) conductance were based on synaptic conductances recorded in voltage-clamp experiments and delivered to the model cell using 2-electrode dynamic clamp. The dynamic clamp also specified a leak conductance as in Fig. 3. The cellular responses to AMPA alone (black), AMPA plus GABA (blue), or AMPA plus NMDA (red) are shown in the bottom traces in response to single stimuli (A) or a 200-Hz train of stimuli (B). The NMDA conductances are designed on the master setup and passed to the slave setup through separate D/A and A/D channels from the AMPA conductances. The effects of applying both AMPA and NMDA conductances are shown in the bottom red traces. The dynamic clamp determines the degree of unblock of the NMDA conductance according to a Boltzmann relationship:

\[
I(t) = (V_{\text{rev}} - V_m) G_{\text{NMDA}}(t)/(1 + \exp((V_{\text{half}} - V_m)/V_{\text{rate}})).
\]

The dynamic-clamp XOP was initiated on the slave setup using the IGOR command:

```
mafDC AMPA={1,0},NMDA={2,18,-11},
LEAK={8,-75},
```

where the leak and AMPA were specified as above, and the NMDA parameters specified the A/D channel for the slave setup to read the instantaneous NMDA conductance in nanosiemens (channel 2), the half-activation voltage \( V_{\text{half}} = -11 \text{ mV} \), and the activation rate \( V_{\text{rate}} = 18 \text{ mV} \). These values were based on recordings of NMDA receptor-mediated EPSCs in bushy cells of the anteroventral cochlear nucleus (AVCN; Pliss et al. 2009). The most appropriate activation parameters may depend on the preparation.

**Hodgkin-Huxley-like sodium conductance.** The sodium conductance follows the form described by Hodgkin and Huxley (1952). The XOP tracks the state of three activation \( m \) gates and one inactivation \( h \) gate. These gates have voltage-dependent opening and closing rates, conventionally named \( \alpha_m, \alpha_h, \beta_m, \beta_h \). These voltage-dependent functions are specified by the user within IGOR as “waves.” These are then passed to the XOP as parameters, and the XOP takes care of updating the state of the gates and calculating the overall conductance. The user may specify the maximal sodium conductance either as a constant value or as a variable conductance determined by an A/D channel. The fixed option is the simplest way to create a constant sodium conductance. The unfixed option allows an experimenter to add different amounts of sodium conductance by changing the D/A output from the master setup (e.g., from 1,000 to 0 nS). The XOP updates the channel model using simple, explicit Euler integration according to the duration of the previous cycle.

An example is shown in Fig. 5. With no sodium conductance, a 20-nS AMPA conductance caused depolarization of the model cell (Fig. 5A, black trace). Then, the dynamic clamp was run with a sodium conductance with \( \alpha_n, \beta_n \), based on the model in Rothman and Manis (2003), which led to a spike (Fig. 5A, red trace). The dynamic-clamp XOP was initiated on the slave setup using the IGOR command:

```
mafDC/Na=1 AMPA={1,0},LEAK={8,-75},
NaV={1000,55,ah,bb,am,bm},
```

where the leak and AMPA were specified as above, and the NaV parameters specify the maximal sodium conductance (represented as \( S_{Na} \) in Hodgkin and Huxley (1952)), \( V_{\text{rev}} \) for sodium, and IGOR waves containing the voltage-dependent rate constants for \( \alpha_n, \beta_n, \alpha_m, \beta_m \) respectively. The command also included the option /Na=1 to indicate that the maximal sodium conductance was a fixed value (1,000 nS). When that flag is turned off (“/Na=0”), the XOP reads the maximal sodium conductance from an A/D channel on the slave setup, which allows experimenters to change the size of the sodium current flexibly during an experiment without starting and stopping the dynamic clamp.
To verify accuracy, current pulses of various amplitudes were applied to the model cell using the dynamic clamp with a 1,000-nS sodium conductance (Fig. 5B, red traces). These waveforms were compared against a pure computer model that mimicked the experimental conditions as closely as possible. The model was run using Euler integration with a step size of 1 μs (Fig. 5B, black traces). There was generally good agreement between the two. It was particularly clear in this set of comparisons that small changes in resting potential or pipette capacitance compensation could influence the timing or presence of the action potential.

**Markov conductance.** The mafDC XOP also allows a highly general approach for implementing a voltage-dependent channel using a Markov model. A Markov model of a channel consists of a series of interconnected channel states as well as the transitions between the different states. These transitions may be voltage-dependent (as in the Hodgkin-Huxley conductance) or voltage-independent. These transitions are specified as parameters by the user. The user also specifies which of the states are conducting and whether they are subconductance states. During each cycle, the dynamic clamp updates the proportion of channels in each state according to the membrane potential and the transition waves specified by the user.

One issue in the development of the Markov conductance was how to update the channel states from one cycle to the next. In pure computer modeling, one often must set step sizes as small as 1 μs for model accuracy and stability. However, the dynamic clamp must run in real time, and the size of the time step (i.e., the cycle time) depends primarily on how many A/D channels are being read (see Test modes), which is constrained by the experimental conditions. The cycle time could be as large as 20–50 μs, which could lead to inaccuracy or instability depending on the method used to update the Markov model.

To evaluate different integration methods, we used pure computer modeling of a representative Markov channel. We used the 12-state sodium channel model of Kuo and Bean (1994) and applied a step change in voltage from −80 to 0 mV. The results of different models are shown in Fig. 6 with the open probability ($P_O$) displayed unsmoothed to facilitate distinguishing between models of different time steps. When we used a 4th-order Runge-Kutta integration method with a step size of 1 μs, it yielded a smooth increase in $P_O$ during the 1st 0.1 ms followed by inactivation (Fig. 6, smooth black trace). However, when the step size was increased to 10 μs, the model showed considerable instability (Fig. 6, asterisk). Similar behavior was observed with simple Euler integration and a 4th-order Adams-Bashforth (data not shown). Another explicit integration method, exponential Euler, showed improved stability but was highly inaccurate at step sizes relevant for the dynamic clamp (Fig. 6, red traces). Thus none of the explicit approaches appeared usable.

We therefore considered implicit Euler integration. This method was very stable and reasonably accurate even for large step sizes (Fig. 6, blue traces). Based on this, we implemented implicit Euler integration in the mafDC XOP. One concern is that this method requires solving a system of linear equations during each cycle, which could slow cycle time significantly due to the extra computational overhead and might compromise accuracy. To test this, we used the Hodgkin-Huxley model as transformed into a Markov model (see Johnston and Wu 1995). This Markov model version consists of eight states for which transition rates are given by the same $\alpha$, $\beta_m$, and $\beta_h$ functions. When the dynamic clamp was running the Markov model, the model cell generated spikes that looked very similar to those generated by the pure computer model (Fig. 5B, blue and black traces). This suggests that the computer model chosen utilizes the correct transition rates but does not capture the full complexity of the Hodgkin-Huxley channels.

Fig. 6. Effects of different integration methods on accuracy for Markov models of ionic conductances. All traces are from a pure computer model, testing different time steps for model integration. The different time steps are distinguished by the durations of the horizontal steps (1, 10, 20, and 50 μs). The Markov model was the 12-state sodium channel model of Kuo and Bean (1994). Top trace: voltage step for the model. Bottom traces: 4th-order Runge-Kutta (RK4; black), exponential Euler (red), and implicit Euler (blue). The implicit Euler had better stability than Runge-Kutta for time steps >10 μs (asterisk) and better accuracy than exponential Euler for time steps as large as 50 μs. $P_O$, open probability.

Fig. 5. Sodium conductance implemented by dynamic clamp using Hodgkin-Huxley and Markov models. A: Hodgkin-Huxley sodium conductance. A 20-nS AMPA conductance was applied with (red trace) and without (black trace) a sodium conductance of 1,000 nS. The model cell also had a leak conductance as specified by the user. Top trace: voltage step for the model. Bottom traces: 4th-order Runge-Kutta (RK4; black), exponential Euler (red), and implicit Euler (blue). The mafDC XOP also allows a highly general approach for implementing a voltage-dependent channel using a Markov model. A Markov model of a channel consists of a series of interconnected channel states as well as the transitions between the different states. These transitions may be voltage-dependent (as in the Hodgkin-Huxley conductance) or voltage-independent. These transitions are specified as parameters by the user. The user also specifies which of the states are conducting and whether they are subconductance states. During each cycle, the dynamic clamp updates the proportion of channels in each state according to the membrane potential and the transition waves specified by the user.

One issue in the development of the Markov conductance was how to update the channel states from one cycle to the next. In pure computer modeling, one often must set step sizes as small as 1 μs for model accuracy and stability. However, the dynamic clamp must run in real time, and the size of the time step (i.e., the cycle time) depends primarily on how many A/D channels are being read (see Test modes), which is constrained by the experimental conditions. The cycle time could be as large as 20–50 μs, which could lead to inaccuracy or instability depending on the method used to update the Markov model.

To evaluate different integration methods, we used pure computer modeling of a representative Markov channel. We used the 12-state sodium channel model of Kuo and Bean (1994) and applied a step change in voltage from −80 to 0 mV. The results of different models are shown in Fig. 6 with the open probability ($P_O$) displayed unsmoothed to facilitate distinguishing between models of different time steps. When we used a 4th-order Runge-Kutta integration method with a step size of 1 μs, it yielded a smooth increase in $P_O$ during the 1st 0.1 ms followed by inactivation (Fig. 6, smooth black trace). However, when the step size was increased to 10 μs, the model showed considerable instability (Fig. 6, asterisk). Similar behavior was observed with simple Euler integration and a 4th-order Adams-Bashforth (data not shown). Another explicit integration method, exponential Euler, showed improved stability but was highly inaccurate at step sizes relevant for the dynamic clamp (Fig. 6, red traces). Thus none of the explicit approaches appeared usable.

We therefore considered implicit Euler integration. This method was very stable and reasonably accurate even for large step sizes (Fig. 6, blue traces). Based on this, we implemented implicit Euler integration in the mafDC XOP. One concern is that this method requires solving a system of linear equations during each cycle, which could slow cycle time significantly due to the extra computational overhead and might compromise accuracy. To test this, we used the Hodgkin-Huxley model as transformed into a Markov model (see Johnston and Wu 1995). This Markov model version consists of eight states for which transition rates are given by the same $\alpha$, $\beta_m$, and $\beta_h$ functions. When the dynamic clamp was running the Markov model, the model cell generated spikes that looked very similar to those generated by the pure computer model (Fig. 5B, blue and black traces). This suggests that the computer model chosen utilizes the correct transition rates but does not capture the full complexity of the Hodgkin-Huxley channels.

Fig. 5. Sodium conductance implemented by dynamic clamp using Hodgkin-Huxley and Markov models. A: Hodgkin-Huxley sodium conductance. A 20-nS AMPA conductance was applied with (red trace) and without (black trace) a sodium conductance of 1,000 nS. The model cell also had a leak conductance as specified by the user. Top trace: voltage step for the model. Bottom traces: 4th-order Runge-Kutta (RK4; black), exponential Euler (red), and implicit Euler (blue). The mafDC XOP also allows a highly general approach for implementing a voltage-dependent channel using a Markov model. A Markov model of a channel consists of a series of interconnected channel states as well as the transitions between the different states. These transitions may be voltage-dependent (as in the Hodgkin-Huxley conductance) or voltage-independent. These transitions are specified as parameters by the user. The user also specifies which of the states are conducting and whether they are subconductance states. During each cycle, the dynamic clamp updates the proportion of channels in each state according to the membrane potential and the transition waves specified by the user.

One issue in the development of the Markov conductance was how to update the channel states from one cycle to the next. In pure computer modeling, one often must set step sizes as small as 1 μs for model accuracy and stability. However, the dynamic clamp must run in real time, and the size of the time step (i.e., the cycle time) depends primarily on how many A/D channels are being read (see Test modes), which is constrained by the experimental conditions. The cycle time could be as large as 20–50 μs, which could lead to inaccuracy or instability depending on the method used to update the Markov model.

To evaluate different integration methods, we used pure computer modeling of a representative Markov channel. We used the 12-state sodium channel model of Kuo and Bean (1994) and applied a step change in voltage from −80 to 0 mV. The results of different models are shown in Fig. 6 with the open probability ($P_O$) displayed unsmoothed to facilitate distinguishing between models of different time steps. When we used a 4th-order Runge-Kutta integration method with a step size of 1 μs, it yielded a smooth increase in $P_O$ during the 1st 0.1 ms followed by inactivation (Fig. 6, smooth black trace). However, when the step size was increased to 10 μs, the model showed considerable instability (Fig. 6, asterisk). Similar behavior was observed with simple Euler integration and a 4th-order Adams-Bashforth (data not shown). Another explicit integration method, exponential Euler, showed improved stability but was highly inaccurate at step sizes relevant for the dynamic clamp (Fig. 6, red traces). Thus none of the explicit approaches appeared usable.

We therefore considered implicit Euler integration. This method was very stable and reasonably accurate even for large step sizes (Fig. 6, blue traces). Based on this, we implemented implicit Euler integration in the mafDC XOP. One concern is that this method requires solving a system of linear equations during each cycle, which could slow cycle time significantly due to the extra computational overhead and might compromise accuracy. To test this, we used the Hodgkin-Huxley model as transformed into a Markov model (see Johnston and Wu 1995). This Markov model version consists of eight states for which transition rates are given by the same $\alpha$, $\beta_m$, and $\beta_h$ functions. When the dynamic clamp was running the Markov model, the model cell generated spikes that looked very similar to those generated by the pure computer model (Fig. 5B, blue and black traces). This suggests that the computer model chosen utilizes the correct transition rates but does not capture the full complexity of the Hodgkin-Huxley channels.
putational overhead of the Markov model does not greatly interfere with its accuracy. The dynamic-clamp XOP was initiated on the slave setup using the IGOR command:

```plaintext
mafDC/M1fx=1 LEAK={8,-72},PULS={2},
Mrk1={1000,55,HHTransitions,HHStates},
```

where the pulse and leak parameters were specified as above, and the Mrk1 parameters specified the peak channel conductance (1,000 nS), the reversal potential (55 mV), the transition rates between states (as a matrix of waves specifying the voltage dependence), and a wave specifying the conductance of each state. The /M1fx flag indicates that the peak conductance was constant in this example. A second Markov channel can be added using the “Mrk2” keyword and “/M2fx” flag.

**Test modes.** The speed of the dynamic clamp is an important issue (Bettencourt et al. 2008). To evaluate the speed of the dynamic clamp, we included a test mode. In test mode, the dynamic clamp performs all of the requested A/D reads and calculates the current to be passed, but instead of passing the current, it toggles the output between two arbitrary values. In other words, each time the output current switches from negative to positive or positive to negative, it signals that one cycle of the dynamic clamp completed. An oscilloscope on the master setup can be used to measure the cycle time under real experimental conditions.

Examples are shown in Fig. 7A. In the first example, the dynamic-clamp XOP was initiated on the slave setup using the IGOR command:

```plaintext
mafDC /TEST=700,
```

where the test flag indicated the dynamic clamp was to be run in test mode, oscillating between the arbitrary values of 700 and ~700. Because no conductances were specified, the dynamic clamp only read membrane potential and wrote the current, so the cycle time would represent the fastest the dynamic clamp could go. The cycle time for our slave setup (Dell OptiPlex 745, 2.13-GHz Intel Core 2 6400 CPU, Windows XP Service Pack 3, PCI-6361e) was 6.6 s. We measured the impact of each dynamic-clamp component on cycle time for our system setup. The durations are shown in Fig. 7B. The most important consideration is the number of A/D channels to be read. The membrane potential, current pulse, and synaptic conductances each take 6–10 μs. Therefore, the most important consideration for a fast cycle time is reducing the number of conductances to be read. The actual calculations associated with the dynamic clamp were extremely fast. The leak and Hodgkin-Huxley sodium conductances took very little time, reaching cycle rates of 150 and 143 kHz. The Markov conductance took surprisingly little time, considering the computational overhead. The 8-state Markov model of the sodium conductance took ~4 μs (total cycle time 10.6 μs, or 94 kHz), and the 12-state ~7 μs (total cycle time 13.6 μs, or 74 kHz). These

![Fig. 7. Using the test functions to measure speed, latency, and jitter of the dynamic clamp. For these measurements, the dynamic clamp was run on a Dell OptiPlex 745 running Windows XP with a National Instruments Peripheral Component Interconnect Express (PCIe) 6361 data acquisition system. Test results are expected to be highly system-specific.](http://jn.physiology.org/)

---

**References:**

Bettencourt et al. (2008)
computations are likely to go much faster on a more up-to-date computer.

Another issue is the latency before the dynamic clamp reacts to a change in membrane potential or specified conductance amplitude and the standard deviation of this latency (“jitter”). To measure these quantities, there is an “echo” test mode in which the dynamic clamp performs all of the requested A/D reads and calculates the current to be passed, but instead of passing the current, the D/A output simply echoes the A/D channel carrying the membrane potential. Echo test mode is invoked by using a 0 for the test value:

mafDC /TEST=0.

To use echo mode, the master setup sends any signal, such as a square pulse, to the membrane potential input on the slave setup, and the echo from the dynamic clamp is recorded (Fig. 7C). (The A/D conversion on the master setup itself contributes no detectable latency or jitter.) We measured the mean and variance of the latency between pulse and echo with different dynamic-clamp features. The added latency was ~15 μs for each component that required reading another A/D channel and for purely computational components was less (Fig. 7D). The variance of the echo timing also increased with additional components. For multiple components, latency and variance combine additively. As the square root of variance, jitters of different components do not simply add but remained <10 μs in most situations (error bars in Fig. 7D). Thus, similar to the cycle time, the latency and variance of the echo depended on the number of A/D channels being read.

The effects of latency and jitter will depend on the experimental context. Test mode in this dynamic clamp allows experimenters to measure latency, variance, and jitter to determine whether it will be an issue for their specific experimental conditions.

Dynamic Clamp in Real Neurons

We confirmed the operation of the dynamic clamp using real neurons, bushy cells in the AVCN. These were prepared as described previously and recorded in slices using two-electrode dynamic clamp (Yang and Xu-Friedman 2009). These procedures were approved by the Institutional Animal Care and Use Committee at the University at Buffalo. Sagittal slices (150 μm) of the AVCN were cut from postnatal day 15 (P15) CBA/CaJ mice. Recordings were made at 34°C in external solution containing the following (in mM): 125 NaCl, 26 NaHCO3, 2.5 KCl, 1.25 Na2PO4, 20 glucose, 1.5 CaCl2, 1 MgCl2, 4 Na-t-lactate, 2 Na-pyruvate, 0.4 Na-t-ascorbate, bubbled with 95% O2-5% CO2. Recordings were carried out using an Axon Instruments MultiClamp 700B with two 3- to 4-MΩ pipettes containing (in mM): 130 KMeSO3, 10 NaCl, 2 MgCl2, 0.16 CaCl2, 0.5 EGTA, 10 HEPES, 4 Na2ATP, 0.4 NaGTP, and 14 Tris-CrPhos (pH 7.2, 302 mosmol/kgH2O).

The AMPA conductance (Fig. 8, top trace) was delivered by the master setup to the slave setup and applied by the dynamic clamp. It was based on voltage-clamp measurements of AMPA-mediated EPSCs (Yang and Xu-Friedman 2009). The bushy cell responded with excitatory postsynaptic potentials (EPSPs) and spikes (Fig. 8, asterisks, bottom trace). This configuration largely avoids series resistance and reduces pipette capacitance errors compared with the single-electrode patch. The AMPA conductance was passed from a D/A on the master setup to the slave setup and applied by the dynamic clamp (Fig. 4, bottom black traces).

DISCUSSION

The dynamic-clamp system described here is relatively easy to implement, using commonly available hardware that is not overly expensive. The user interface through WaveMetrics IGOR is not difficult to learn, and indeed the IGOR environment has already been adopted by quite a few laboratories for collecting and analyzing electrophysiology data. The functionality of the synaptic and channel conductances is very useful for a range of experiments. The Markov conductance is particularly powerful, and the implicit Euler integration provides exceptional stability and accuracy. Furthermore, the test feature makes the system quality easy to assess. We find that this system is extremely fast, even on older hardware, and interruption from the computer operating system is not a significant issue. Thus this dynamic-clamp tool likely will be useful both to new and existing users of dynamic clamp.

Limitations and Future Directions

The mafDC XOP is generally capable of imitating many types of synaptic and intrinsic conductances. However, there are certain major classes of channels that are not currently implemented. One important set of channels are those that are gated by both voltage and ligand, such as calcium-activated potassium channels, sodium-activated potassium channels, and cyclic nucleotide-activated channels. Even if their gating were simplified to purely ligand-dependent, it would be difficult to implement. For example, calcium-activated potassium channels are remarkable for the burst behavior they produce in neurons. Calcium levels rise during periods of activity, which activates KCa channels, which then repolarize the membrane.
and cause calcium levels to decline. The dynamic-clamp interface would need to read both membrane potential and calcium concentration to drive such behavior, which it is not able to do at present. Given the importance of these channels to neuronal function, it would be desirable to implement this in future versions. Furthermore, this dynamic clamp cannot directly control calcium concentration, although there are approaches for that (Idoux and Mertz 2011).

A second major issue is that the speed of the interface decreases with additional conductance amplitudes that need to be read (Fig. 7). This could be a problem if there are multiple types of synaptic receptors being mimicked (e.g., AMPA, GABA, and NMDA). This may possibly be resolved by taking a different approach to the dynamic-clamp architecture. Currently, the dynamic clamp is centralized on a single slave setup, so all of the A/D reads must be done serially. A different approach would be to do the conductance calculations in parallel on separate interfaces, each responsible for a single conductance. This would require reads per cycle at a maximum for each interface. This is analogous to how the nervous system actually functions, where each channel acts independently from the others. This does not seem practical for the system described in Fig. 1. However, there are devices, such as the Arduino Due, that can integrate A/D and computation in a highly compact and inexpensive form. We are currently experimenting with this platform to provide a more rapid parallel interface.

Another issue for dynamic clamp are interruptions generated by the operating system. Operating systems such as Windows may occasionally occupy processor time, thereby lengthening cycle time. In our experience, this is not a major issue. We can visualize such interruptions using the test feature in the dynamic clamp: computer-and-experiment results. J Neurosci Methods 169: 282–289, 2008.


ACKNOWLEDGMENTS

We thank L. S. Milescu for sharing insights into improving cycle time on the dynamic clamp in QuB (Milescu et al. 2008), R. Manning for advice on implementing the implicit Euler integration method, and T. Ngodup, H. Yang, and X. Zhuang for helpful comments on the manuscript.

GRANTS

This work was supported by Grant 1208131 from the National Science Foundation.

DISCLOSURES

No conflicts of interest, financial or otherwise, are declared by the author(s).

AUTHOR CONTRIBUTIONS


REFERENCES


