9

Interacting with systems in general

Concept. To design and build a generalised concept of interacting with stochastic processes of any kind. The mathematical formalism and software that we introduce here will serve as a common language and interface for any simulation studies into manipulating real world phenomena, and should enable the learning of control algorithms. We will call this software 'dexetera', since it originates as an extension to the stochadex. For the mathematically-inclined, this chapter will cover how dexetera is structured in theory by developing some useful extensions to the stochadex formalism and illustrating with some simple examples. For the programmers, the public Git repository for the code described in this chapter can be found here: https://github.com/umbralcalc/dexetera.

9.1 Formalising general interactions

Rewrite this section to remove the existence of parametric actions, rewrite the code in the stochadex to match, and make this section more about the kinds of actions that can be performed on the basic stochadex simulations.

Let's start by considering how we might adapt the mathematical formalism we have been using so far to be able to take actions which can manipulate the state at each timestep. Using the mathematical notation that we inherited from the stochadex, we may extend the formula for updating the state history matrix $X' \rightarrow X$ to include two layers of possible interactions which are facilitated by a new vector-valued 'parametric action' function G_{t} and a new vector-valued 'state action' function H_{t}. In doing so we shall be defining the domain of an acting entity in the stochastic process environment — which we shall hereafter refer to as simply the 'agent'.

During a timestep over which actions are performed by the agent, the stochadex state update formula can be extended to look like this system of equations

$$Z^i_{\mathsf{t}+1} = G^i_{\mathsf{t}+1}(Z_{\mathsf{t}}, \mathcal{A}_{\mathsf{t}+1}) \tag{9.1}$$

$$X^i_{\mathsf{t}+1} = H^i_{\mathsf{t}+1}[F_{\mathsf{t}}(X', Z_{\mathsf{t}+1}, \mathsf{t}), \mathcal{A}_{\mathsf{t}+1}]\,, \tag{9.2}$$

where we have also introduced the concept of the 'actions' performed $\mathcal{A}_{\mathsf{t}+1}$ on the system; some vector of parameters which define what actions are taken at timestep $\mathsf{t}+1$.

In Eqs. (9.1) and (9.2), notice that we have replaced the constant vector of parameters z (as in the stochadex formalism) for a time-dependent vector Z_{t} of parameters that can be updated by G_{t} at any (but not necessarily every) timestep. From the perspective of the whole matrix X update step; G_{t} and H_{t} combine to become technically equivalent to applying this formal composition of functions

$$X^i_{\mathsf{t}+1} = H^i_{\mathsf{t}+1}(F_{\mathsf{t}}(X', G^i_{\mathsf{t}+1}(Z_{\mathsf{t}}, \mathcal{A}_{\mathsf{t}+1}), \mathsf{t}), \mathcal{A}_{\mathsf{t}+1}) = \mathcal{F}^i_{\mathsf{t}+1}(X', Z_{\mathsf{t}}, \mathcal{A}_{\mathsf{t}+1}, \mathsf{t}) \,, \tag{9.3}$$

which looks like an absolute mess! However, it illustrates how $\mathcal{F}$, which refers to a modified version of the F function, contains our actions but no other new parameters need be specified; only function operations. Hence, while we have provided two distinct ways one might encode actions to manipulate a stochastic phenomenon, we shall often just refer to them together as 'taking an action' $\mathcal{A}_{\mathsf{t}+1}$ at timestep $\mathsf{t}+1$ — because the parameters which define either type of action at this timestep should all be stored within $\mathcal{A}_{\mathsf{t}+1}$ anyway. It is, however, important not to forget the full mathematical formulation when performing calculations.

The code for the new iteration formula given by Eq. (9.3), which includes taking actions in the same timestep, would look something like Fig. 9.1.

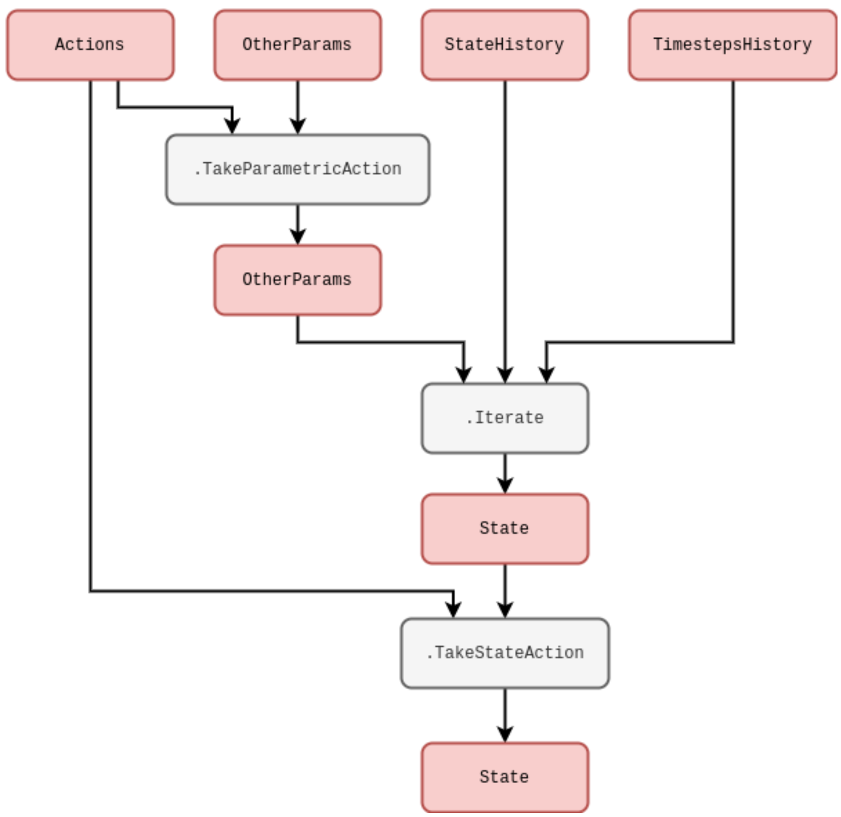

Figure 9.1: Code schematic of Eq. (9.3).

Bibliography