7

Markov Chain Approximations: Path and Control Delayed

7.0 Outline of the Chapter

This chapter adapts the Markov chain approximation methods that were introduced in Chapter 6 to the problem with delays. The approximating chains are constructed almost exactly as they are for the no-delay case, except that the transition probabilities must take the delays into account. Various numerical approximations are developed. They are reasonable and well motivated. But in view of the fact that rather little is known about either approximation or numerical methods for delay equations, the algorithms are to be viewed as a first step and will hopefully encourage additional work. When constructing an algorithm, there are two large issues of concern, and both must be kept in mind. One is numerical feasibility. The other concerns the proof of convergence, as the approximating parameter goes to zero.

Because the basic state space of the problem with delays is infinite-dimensional, one must work with approximations. One can devise "Markov chain like" approximations that converge to the original model and for which the optimal value functions converge to that for the original model. Alternatively, one can first approximate the original model, say along the lines done in Chapter 4, so that the resulting problem is finite-dimensional. Then approximate the result for numerical purposes. Both approaches are taken in this chapter, although the latter one is more realistic, as the memory requirements are much less. As seen in Chapter 4, the suggested finite-dimensional approximations are often quite good. Further approximations are developed in Chapters 8 and 9 when the path or control values are delayed, and they will often be advantageous.

The validity of an approximation to the original model depends on the relative insensitivity of the values and controls to the quantities that are being approximated, whether it is the path, path and delay, control, and so forth. The greater the sensitivity, the finer the approximation needs to be. This is a particularly difficult problem for the delay model, as the behavior can be quite sensitive to the delay, and little is known about this in general.

H.J. Kushner, *Numerical Methods for Controlled Stochastic Delay Systems*,
doi: 10.1007/978-0-8176-4621-9_7,

The proofs of convergence in [58] are purely probabilistic, being based on weak convergence methods. The idea is to interpolate the chain to a continuous-time process in a suitable manner, show that the Bellman equation for the interpolation is the same as for the chain, and then show that the interpolated processes converge to an optimal diffusion as the approximating parameter goes to zero. The approach is parallel to this for the problem with delays, and we try to arrange the development with an eye to using the powerful methods and results of [58] to the extent possible, so as to simplify the proof of convergence.

Section 1 introduces the unapproximated model and the main assumptions. As for the nondelay case, the main assumption is local consistency. This condition is the same as that for the nondelay problem, with the appropriate delay-dependent drift and diffusion terms used. The state of the problem, as needed for the numerical procedure, consists of a segment of the path (over the delay interval) and of the control path as well (if the control is also delayed). The only change in the local consistency condition is the use of the "memory segment" arguments in the drift and diffusion functions. As in Chapter 6, the local consistency condition says no more than that the conditional mean change (resp, covariance) in the state of the approximating chain is proportional to the drift (resp, covariance) of the original diffusion process, modulo small errors. It need not hold everywhere (see, e.g., [58, Section 5.5]). Transition probabilities for the approximating chain are readily obtained from the formulas that are used for the nondelay case in [58].

For pedagogical purposes, in much of the development, we divide the discussion into a part where only the path is delayed in the dynamics and a part where both the control and path are delayed, for which the algorithms are much more complicated The delay system analogs of all of the cost functions covered in [58] can be dealt with. But for simplicity of exposition, most of the discussion is confined to the discounted case, with boundary reflection. If the process is stopped on reaching a boundary then, with the model of Section 3.1 and the cost function (3.4.1), all of the approximation methods and convergence results will hold, and the necessary theorems are stated. Section 1 concludes with the discussion of the continuous time interpolations. These interpolations, which are used for the convergence proofs only and not for the numerical algorithms, are a little more complicated than those used for the no-delay case in Chapter 6, owing to the need to represent the "memory segment" argument in a way that can be used in the development of efficient approximation methods.

In Section 2, some particular Markov chain approximations are introduced, with the aim of efficiency in the use of memory. The implicit approximation method of Chapter 5 has some advantages in dealing with the memory problem, and this is discussed in Section 3. Section 4 deals with various details concerning the relation between the implicit approximation procedure and the model with randomly varying delays in Subsection 4.2.3. Keep in mind that these randomly varying delays are not a feature of the original model, but appear in the numerical approximation as a consequence of the use of the implicit approximation procedure to simplify the memory problem. One could treat the case where the original model has randomly time varying delays as well, as noted in the comments below (3.1.8), but at the expense of increased memory requirements.

Chapter 8 continues the development of the ideas in this chapter and contains the proofs of the convergence theorems.

7.1 The Model and Local Consistency

The approach to numerical approximation is analogous to what was done for the no-delay case in Chapter 6. The main new issues concern accounting for the fact that $b(\cdot), \sigma(\cdot)$, and $k(\cdot)$ depend on the "memory" segments of the solution path and/or the control, whichever is delayed in the dynamics. We will construct an "approximating" controlled finite-state process $\{\xi_n^h, n \geq 0\}$ and interpolation intervals $\{\Delta t_n^h, n \geq 0\}$ in much the same way as was done in Chapter 6. This approximating process will serve as the basis of the numerical procedure. It will be seen that these processes are constructed as easily as they are for the no-delay problem in [58]. Although $\{\xi_n^h\}$ itself is not a Markov chain due to the memory, one can embed it into a finite-state Markov chain. It is the Bellman equation for the embedded chain that needs to be solved to get the optimal cost. Indeed a main concern are representations for such Markov chains that are efficient from the point of view of computation. In this section, a generic approximation will be constructed. Although it often requires too much memory to be of practical use, it will provide the foundation for the alternative and more practical approximations in Section 3 and in the next chapter.

7.1.1 The Models

The model is the controlled reflected diffusion of Section 3.2. Assumptions (A3.2.1) and (A3.2.2) on the constraint set G are always used. Other conditions will be given when needed. Rewriting the equations for convenience, when both the path and control are delayed and in terms of ordinary controls, the model is (3.2.3):

$$x(t) = x(0) + \int_0^t b(\bar{x}(s), \bar{u}(s))ds + \int_0^t \sigma(\bar{x}(s))dw(s) + z(t), \tag{1.1}$$

where the conditions (A3.1.2) and (A3.1.3) hold. For notational simplicity, we suppose that, if both the path and control are delayed, then the maximum delay is the same for both. The case where they are not the same is a simple and obvious modification. In relaxed control notation, (1.1) is

$$x(t) = x(0) + \int_0^t \bar{b}(\bar{x}(s), \bar{r}(t))ds + \int_0^t \sigma(\bar{x}(s))dw(s) + z(t), \tag{1.2}$$

where, as in (3.1.6),

$$\bar{b}(\bar{x}(t), \bar{r}(t)) = \int_{-\bar{\theta}}^0 \int_U b(\bar{x}(t), \alpha, \theta) r'(d\alpha, t+\theta)\mu_c(d\theta), \tag{1.3}$$

and

$$\int_0^t \bar{b}(\bar{x}(s), \bar{r}(s))ds = \int_{-\bar{\theta}}^0 \left[\int_0^t ds \int_U b(\bar{x}(s), \alpha, \theta) r'(d\alpha, s+\theta)\right] \mu_c(d\theta).$$

The discounted cost function (3.4.4) is

$$W(\hat{x}, \hat{r}, r) = E^r_{\hat{x},\hat{r}} \int_0^\infty ds \int_{-\bar{\theta}}^0 \int_U e^{-\beta t} \left[k(\bar{x}(t), \alpha, \theta) r'(d\alpha, t+\theta)\mu_c(d\theta)dt + q'dy(t)\right], \tag{1.4}$$

where $\hat{x}$ and $\hat{r}$ denote the initial memory segments of the path and control, resp. The existence of an optimal control was shown in Theorem 3.5.1.

If the path only is delayed, then we drop the control memory segment term, and the model specializes to

$$x(t) = x(0) + \int_0^t ds \int_U b(\bar{x}(s), \alpha) r'(d\alpha, s) + \int_0^t \sigma(\bar{x}(s))dw(s) + z(t), \tag{1.5}$$

$$W(\hat{x}, r) = E^r_{\hat{x}} \int_0^\infty \int_{-\bar{\theta}}^0 \int_U e^{-\beta t} \left[k(\bar{x}(t), \alpha) r'(d\alpha, t)dt + q'dy(t)\right]. \tag{1.6}$$

As usual, if the process stops on hitting the boundary, then drop (A3.1.2) and (A3.1.3) and add (A3.4.1) and (A3.4.2).

7.1.2 Delay in Path Only: Local Consistency and Interpolations

The approximating chain ξ^h_n takes values in the set S_h, and the definitions of $S_h, G_h = S_h \cap G$ and ∂G^+_h from the beginning of Section 6.2 are used. As for the no-delay problem, the key requirement that is placed on the approximating chain is that it satisfy a local consistency condition analogous to (6.2.1). The dynamics of (1.5) at time t involve the memory segment $\bar{x}(t)$ of the path on the delay interval $[t - \bar{\theta}, t]$. An analogous dependence must hold for the dynamics of the ξ^h_n process. The definition of the memory segment of the approximating chain will depend on the particular continuous-time interpolation of the ξ^h_n values that is used, and several useful forms will be developed in the sequel and in the next chapter. For simplicity, we will start by using an analog of the explicit approximation procedure of Sections 6.2–6.4. This will not usually yield the best form of the memory segment, but it provides a convenient introduction to the overall approximation method. Suppose that $\xi_n^h, \Delta t_n^h$ are available (these will be constructed below) and, as in Section 6.2, define the interpolated time $t_n^h = \sum_{i=0}^{n-1} \Delta t_i^h$. The process $\xi^h(\cdot)$ is defined to be the piecewise-constant continuous-time interpolation of $\{\xi_n^h\}$ with intervals $\{\Delta t_n^h\}$, as in (6.3.1). Recall the discussion below (6.3.1) concerning the interpolation at the reflecting states. In particular, if ξ_n^h is a reflecting state, then $\xi^h(t_n^h) = \xi_{n+1}^h$, which is the state that the reflecting state ξ_n^h is instantaneously sent to.

Path memory segments. Define the segment $\bar{\xi}_n^h$ of the path $\xi^h(\cdot)$ by

$$\bar{\xi}_n^h(\theta) = \xi^h(t_n^h + \theta) \quad \text{for } \theta \in [-\bar{\theta}, 0), \quad \text{and} \quad \bar{\xi}_n^h(0) = \xi_n^h. \tag{1.7a}$$

This is the segment of the interpolated path on $[t_n^h - \bar{\theta}, t_n^h)$ with the value ξ_n^h at $\theta = 0$. If $\xi_n^h \in G_h$, then $\xi_n^h(\theta) = \xi^h(t_n^h + \theta)$ for all $\theta \in [-\bar{\theta}, 0]$. Define the process $\bar{\xi}^h(t)$ by

$$\bar{\xi}^h(t) = \bar{\xi}_n^h, \qquad \text{for } t_n^h \leq t < t_{n+1}^h. \tag{1.7b}$$

Let $\hat{\xi}$ denote the canonical value of $\bar{\xi}_n^h$.

To construct the dynamics of the approximating chain, we will need to define a path memory segment that plays the role of $\bar{x}(t)$. There is a great deal of flexibility in the way that this approximation is constructed from the $\{\xi_n^h\}$. The choice influences the computational complexity, and we return to this issue in subsequent sections. Until further notice, we use $\bar{\xi}_n^h$. This choice is not always suitable for numerical purposes, and will later be modified in various ways to simplify the numerical computations. The exact form of the approximation is not important at this point.

The initial condition $\bar{x}(0) = \{x(t) : -\bar{\theta} \leq t \leq 0\}$ for (1.5) is an arbitrary continuous function. This will have to be approximated for numerical convenience. Until further notice, we simply assume that we use a sequence $\bar{\xi}_0^h \in D(G_h; [-\bar{\theta}, 0])$, that is piecewise-constant and that converges to $\bar{x}(0)$ uniformly on $[-\bar{\theta}, 0]$ as $h \to 0$.

Local consistency in G_h. For numerical purposes it is often useful to approximate the set U. Thus, as in Section 6.2, let U^h be a sequence of compact sets that converges to U as $h \to 0$ in the sense that the closed convex hull of $(b(x, U^h), k(x, U^h))$ converges to the closed convex hull of $(b(x, U), k(x, U))$ as $h \to 0$. Each U^h might contain only a finite set of points.

Let $\xi_n^h \in G_h$. Analogously to the no-delay case in Section 6.2, the chain and intervals are assumed to satisfy the following local consistency properties. Let u_n^h (with values in U^h) denote the control applied at time n. The distribution of ξ_{n+1}^h, given the initial data and $\{\xi_i^h, u_i^h, i \leq n\}$, will depend only on the current path memory segment $\bar{\xi}_n^h$ and current control u_n^h and not otherwise on n, analogously to the case in Chapter 6. Recall the definition $\Delta\xi_n^h = \xi_{n+1}^h - \xi_n^h$ and that of the martingale difference β_n^h in (6.2.3), and let $E_{\hat{\xi},n}^{h,\alpha}$ denote the expectation given all data to time n, with $u_n^h = \alpha$ and $\bar{\xi}_n^h = \hat{\xi}$. Analogously to the definition for the no-delay case in (6.2.1), local consistency is said to hold if there is a function $\Delta t^h(\cdot)$ such that, for $\hat{\xi}(0) = \xi_n^h \in G_h$,[1]

$$
\begin{aligned}
&E_{\hat{\xi},n}^{h,\alpha} \Delta\xi_n^h = b_h(\hat{\xi},\alpha)\Delta t^h(\hat{\xi},\alpha) = b(\hat{\xi},\alpha)\Delta t^h(\hat{\xi},\alpha) + o(\Delta t^h(\hat{\xi},\alpha)),\\
&E_{\hat{\xi},n}^{h,\alpha} \beta_n^h[\beta_n^h]' = a_h(\hat{\xi})\Delta t^h(\hat{\xi},\alpha) = a(\hat{\xi})\Delta t^h(\hat{\xi},\alpha) + o(\Delta t^h(\hat{\xi},\alpha)),\\
&a(\hat{\xi}) = \sigma(\hat{\xi})\sigma'(\hat{\xi}),\\
&\sup_{n,\omega} |\xi_{n+1}^h - \xi_n^h| \xrightarrow{h} 0, \quad \sup_{\hat{\xi},\alpha} \Delta t^h(\hat{\xi},\alpha) \xrightarrow{h} 0.
\end{aligned}
\tag{1.8}
$$

The reflecting boundary is treated the same as in Section 6.2. If ξ_n^h is a reflecting state, then it is sent to a state in G_h, with no control applied. The mean of $\xi_{n+1}^h - \xi_n^h$, conditioned on the data to time n, is a reflection direction at the point ξ_n^h. In particular, (6.2.2) holds. Define $\Delta t_n^h = \Delta t^h(\bar{\xi}_n^h, u_n^h)$.

We have the analog of (6.2.4)

$$
\xi_{n+1}^h = \xi_n^h + \Delta t_n^h b(\bar{\xi}_n^h, u_n^h) + \beta_n^h + \Delta z_n^h + o(\Delta t_n^h). \tag{1.9}
$$

Constructing the transition probabilities. For simplicity in the development, we will suppose that S_h is a regular h-grid. Hence the points in G_h are h units apart in each direction. This is done only to simplify the notation. Any of the state spaces G_h that are allowed in [58] can be used here. In particular, the state space approximation parameter can depend on the coordinate direction. The simple example of the construction in Section 6.4 and, indeed, any of the methods in [58] for obtaining the transition probabilities and interpolation intervals for the no-delay case can be readily adapted to the delay case.

For the no-delay problem and $x \in G_h$, all of the methods in [58] for generating the controlled transition probabilities $p^h(x,\tilde{x}|\alpha)$ when the grid spacing was uniform in each coordinate direction gave results that depended only on the grid spacings, the "next state" $\tilde{x}$, and on the drift and covariance functions $b(x,\alpha)$ and $a(x) = \sigma(x)\sigma'(x)$, resp. They did not depend on the state and control values in any other way. In addition the transition probability for the chains in [58] for the no-delay case could be written as a ratio in the following way. There are functions $N^h(\cdot)$ and $D^h(\cdot)$ such that for $x \in G_h$,

$$
\begin{aligned}
P\{\xi_1^h = \tilde{x}|\xi_0^h = x, u_0^h = \alpha\} = p^h(x,\tilde{x}|\alpha) = \frac{N^h(b(x,\alpha),a(x),\tilde{x})}{D^h(b(x,\alpha),a(x))},\\
\Delta t^h(x,\alpha) = T^h(b(x,\alpha),a(x)) = \frac{h^2}{D^h(b(x,\alpha),a(x))}.
\end{aligned}
\tag{1.10}
$$

[1] (1.8) defines $b_h(\cdot)$ and $a_h(\cdot)$.

The particular forms of $N^h(\cdot)$ and $D^h(\cdot)$ depend on the actual approximation method.[2] The function $D^h(\cdot)$ is simply a normalization, so that the sum of the probabilities over $\tilde{x}$ is unity. The transition probability from a state x to a state $\tilde{x}$ must be a function of $b(\cdot)$, $a(\cdot)$, and $\tilde{x}$, only, because that is the only information that is available. Hence, the representation (1.10) is unrestrictive. The values of $N^h(\cdot)$ and $D^h(\cdot)$ for the two examples of construction in Section 6.4 are obvious from (6.4.5) or (6.4.7).

For the delay case, we can use the identical forms. For any of the approximation methods in [58] or elsewhere for getting the $N^h(\cdot), D^h(\cdot)$ in (1.10) that yield locally consistency in the sense of (6.2.1), for $\hat{\xi}(0) \in G_h$ we can use the forms

$$\begin{aligned} p^h(\hat{\xi}, \tilde{x}|\alpha) = P\{\xi_1^h = \tilde{x}|\bar{\xi}_0^h = \hat{\xi}, u_0^h = \alpha\} = \frac{N^h(b(\hat{\xi},\alpha), a(\hat{\xi}), \tilde{x})}{D^h(b(\hat{\xi},\alpha), a(\hat{\xi}))}, \\ \Delta t^h(\hat{\xi},\alpha) = \frac{h^2}{D^h(b(\hat{\xi},\alpha), a(\hat{\xi}))}. \end{aligned} \tag{1.11}$$

In particular,

$$p^h(\bar{\xi}_n^h, \tilde{x}|u_n^h) = P\{\xi_{n+1}^h = \tilde{x}|\bar{\xi}_n^h, u_n^h\} = \frac{N^h(b(\bar{\xi}_n^h, u_n^h), a(\bar{\xi}_n^h), \tilde{x})}{D^h(b(\bar{\xi}_n^h, u_n^h), a(\bar{\xi}_n^h))}. \tag{1.12}$$

With the use of (1.11), local consistency in the sense of (6.2.1) implies the local consistency (1.8). It is only the dependence on $b(\cdot), a(\cdot), \tilde{x}$, and h that matters, no matter what the form of the memory segment $\hat{\xi}$. The above discussion is formalized by the following assumption. The assumption is not needed if local consistency is otherwise assured.

A1.1. *The transition probabilities and interpolation intervals are given in the form* (1.10) *with the delay dependencies incorporated* (*yielding* (1.11)), *where* (1.10) *is locally consistent for the no-delay case.*[3]

A discounted cost function. Let $E_{\hat{\xi}}^{h,u^h}$ denote the expectation given initial condition[4] $\hat{\xi} = \bar{\xi}_0^h$ and control sequence $u^h = \{u_n^h, 0 \le n < \infty\}$. Define Δz_n^h

[2] The form of $T^h(\cdot)$ in (1.10) supposes that S_h is a grid with the same spacing in each coordinate direction, so that h is real-valued. This is chosen for simplicity in the development. For more general forms of S_h, the functions $T^h(\cdot), N^h(\cdot)$ and $D^h(\cdot)$ might also depend on the current state x and the local spacing of the states. But whatever they are, they are functions of the drift and diffusion functions. See [58, Section 5.2]. With the delay-dependencies of these functions incorporated, the resulting transition probabilities and interpolation interval would yield the desired local consistency. All that is needed is local consistency.

[3] The form is usually $N(hb(x,\alpha), a(x), \tilde{x})/D(hb(x,\alpha), a(x))$ for some functions $N(\cdot), D(\cdot)$.

[4] The approximation $\hat{\xi}$ of the initial condition will depend on h in general.

and Δy_n^h as above (6.2.3). An approximation to the discounted cost function (3.4.3) for the chain is

$$
\begin{aligned}
W^h(\hat{\xi}, u^h) &= E_{\hat{\xi}}^{h,u^h} \sum_{n=0}^{\infty} e^{-\beta t_n^h} \left[k(\bar{\xi}_n^h, u_n^h) \Delta t^h(\bar{\xi}_n^h, u_n^h) + q' \Delta y_n^h \right], \\
V^h(\hat{\xi}) &= \inf_{u^h} W^h(\hat{\xi}, u^h).
\end{aligned}
\tag{1.13}
$$

By Lemma 6.3.1 (which is [58, Theorem 11.1.3]), the costs are well defined. Let $y^h(\cdot)$ denote the continuous-time interpolation of $\{\Delta y_n^h\}$ with intervals $\{\Delta t_n^h\}$.

A "Markov" continuous-time interpolation. One continuous-time interpolation, namely $\xi^h(\cdot)$, has already been defined. We will now define the analog of the interpolation $\psi^h(\cdot)$ that was defined in Subsection 6.3.2. Let the random variables $\{\nu_n\}$, the interval $\Delta \tau_n^h = \nu_n \Delta t_n^h$, and $\tau_n^h = \sum_{i=0}^{n-1} \Delta \tau_n^h$, be defined as in the first paragraph of Subsection 6.3.2. Then define $\psi^h(t)$ by (6.3.4) or (6.3.8), using the intervals $\Delta \tau_n^h$, all based the processes ξ_n^h and Δt_n^h of this chapter. Because the timescale of the $\psi^h(\cdot)$ uses the intervals $\Delta \tau_n^h$, and that of the memory segment $\bar{\xi}_n^h$ uses the intervals Δt_n^h, the dynamical equation for $\psi^h(\cdot)$ will be a little awkward. But keep in mind that this dynamical equation will be used only in the proofs of convergence and not for the numerical computations. As for the no-delay case, the chains ξ_n^h are used for the numerics, with whatever approximation to the path memory segment is used.

Recall the definitions of the interpolations $u_\tau^h(\cdot), B_\tau^h(\cdot), r_\tau^h(\cdot)$, and $z_\tau^h(\cdot)$, in Subsection 6.3.2, and the definition $d_\tau^h(s) = \max\{n : \tau_n^h \le s\}$ in (6.5.23). Define the function $q_\tau^h(s) = t_{d_\tau^h(s)}^h$. Given interpolated time s in the scale determined by the $\Delta \tau_n^h$, the function $d_\tau^h(s)$ is the index of the process ξ_n^h that gives $\psi^h(s)$ in the sense that we have $\psi^h(s) = \xi_{d_\tau^h(s)}^h = \xi^h(q_\tau^h(s))$. With this notation, the conditional drift rate of the process $\psi^h(\cdot)$ at time s is $b_h(\bar{\xi}^h(q_\tau^h(s)), u_\tau^h(s))$ ($b_h(\cdot)$ was defined in (1.8)). Decomposing the process $\psi^h(\cdot)$ into a compensator, martingale, and reflection term as in (6.3.8), and using relaxed control terminology, leads to the representation

$$
\psi^h(t) = \xi_0^h + \int_0^t \int_{U^h} b_h(\bar{\xi}^h(q_\tau^h(s)), \alpha) r_\tau^h(d\alpha\, ds) + B_\tau^h(t) + z_\tau^h(t), \tag{1.14}
$$

where $\xi_0^h = \bar{\xi}_0^h(0)$ and $B_\tau^h(\cdot)$ is a martingale with quadratic variation process

$$
\int_0^t a_h(\bar{\xi}^h(q_\tau^h(s)))ds.
$$

As noted below (6.3.8), there is a martingale $w^h(\cdot)$ with quadratic variation It and that converges weakly to a Wiener process such that

$$B_\tau^h(t) = \int_0^t \sigma(\bar{\xi}^h(q_\tau^h(s)))dw^h(s) + \epsilon^h(t)$$

where $\lim_{h\to 0} E \sup_{s\le t} |\epsilon^h(s)|^2 \to 0$ for each $t < \infty$

Modulo an asymptotically negligible error due to the "continuous time" approximation of the discount factor, the cost function (1.13) can be written as

$$W^h(\hat{\xi}, u^h) = E_{\hat{\xi}}^{h,u^h} \int_0^\infty \int_{U^h} e^{-\beta t} \left[k(\bar{\xi}^h(q_\tau^h(s)), \alpha) r_\tau^h(d\alpha\, ds) + q' dy_\tau^h(s) \right]. \tag{1.15}$$

The following theorem says that any method for solving the control problem for any locally consistent approximation will yield an approximation to the value for the original model (1.5). The proof is in Section 5 of the next chapter. The absorbing boundary case is dealt with in Theorem 1.3.

Theorem 1.1. *Let* $\xi_n^h, \Delta t_n^h$ *be locally consistent with the model* (1.5) *whose initial condition is* $\bar{x}(0)$*, a continuous function, and with cost function* (1.6) *and its approximation*

$$E_{\hat{\xi}}^{h,u^h} \sum_{n=0}^\infty e^{-\beta t_n^h} \left[k(\bar{\xi}_n^h, u_n^h) \Delta t_n^h + q' \Delta y_n^h \right] \tag{1.16}$$

being used. Let $\bar{\xi}_0^h \in D(G_h; [-\bar{\theta}, 0])$ *be any piecewise-constant sequence that converges to* $\bar{x}(0)$ *uniformly on* $[-\bar{\theta}, 0]$*. Assume* (A3.1.1), (A3.1.2), (A3.2.1)–(A3.2.3), *and* (A3.4.3)*. Then* $V^h(\bar{\xi}_0^h) \to V(\bar{x}(0))$ *as* $h \to 0$*.*

7.1.3 Delay in the Path and Control

Now consider the model (relaxed control form) (1.2), with cost (1.4), where both the path and control are delayed. As for the case where only the path is delayed, one constructs an approximating chain $\{\xi_n^h, n \ge 0\}$ and interpolation intervals $\{\Delta t_n^h, n \ge 0\}$. The initial data for (1.2) is $\bar{x}(0) = \{x(s), -\bar{\theta} \le s \le 0\} \in C(G; [-\bar{\theta}, 0])$ and $\bar{u}(0) = \{u(s), -\bar{\theta} \le s \le 0\} \in L_2(U; [-\bar{\theta}, 0])$, where the control segment is needed due to the delay in the control. The control memory segment for the approximating chain is slightly different. For the process (1.2), either the segment $\{u(s), s \in [-\bar{\theta}, 0]\}$ or the segment $\{u(s), s \in [-\bar{\theta}, 0)\}$ will do for the initial control data. But for the chain, the control at time 0, namely, u_0^h, which is used to get ξ_1^h, is to be determined at time 0, and should not be given as part of the initial data. This fact accounts for our use of the control segment on the half open $[-\bar{\theta}, 0)$ as the initial data. Let $\hat{u}$ denote the canonical value of the control memory segment on the half open interval. With α denoting the canonical value of the current value of the control, we can write terms such as $\bar{b}(\hat{\xi}, \hat{u}, \alpha)$ without ambiguity, depending on the memory segments and the current control value.

Definitions of the control memory segments. In the remainder of this section, we continue to use the full path memory segment $\bar{\xi}_n^h$ from the previous subsection. Given the initial control data $\bar{u}(0)$, we need to approximate it for use on the chain, and, analogously, obtain a control memory segment for each step of the chain. In this subsection, we will use a form for the control memory segment that is analogous to $\bar{\xi}_n^h$. It will usually be very costly in terms of the required memory, but serves as a useful introduction. Alternative, and more efficient, approximations will be discussed in the next chapter. The control memory segment will be denoted by $\bar{u}_n^h$, with canonical value $\hat{u}$, and is defined in terms of the continuous-time interpolation of the control process, as follows. Let u_n^h denote the control that is used on step n. An interpolation interval $\Delta t^h(\hat{\xi}, \hat{u}, \alpha)$ will be defined in the local consistency condition (1.23). Redefine $\Delta t_n^h = \Delta t^h(\bar{\xi}_n^h, \bar{u}_n^h, u_n^h)$ and $t_n^h = \sum_{i=0}^{n-1} \Delta t_i^h$, and define the interpolation $u^h(\cdot)$ of $\{u_n^h\}$ with intervals $\{\Delta t_n^h\}$. Then define the full control memory segment $\bar{u}_n^h = \{u^h(t_n^h + \theta), \theta \in [-\bar{\theta}, 0)\}$. It is the segment of $u^h(\cdot)$ on $[t_n^h - \bar{\theta}, t_n^h)$. Then $(\hat{u}, \alpha)$ denotes the canonical value of the control on a closed interval $[t - \bar{\theta}, t]$ for any t. Let $\bar{u}_0^h$ be any piecewise-constant function in $D(U^h; [-\bar{\theta}, 0))$ that converges to the function $\bar{u}(0)$ in the L_2-sense as $h \to 0$ and extend the definition of $u^h(\cdot)$ to $[-\bar{\theta}, \infty)$.

Summarizing, in this section the memory state at time n of the approximating chain and the associated dynamic program is $\bar{\xi}_n^h, \bar{u}_n^h$, the value of $\xi^h(\cdot)$ on the closed interval $[t_n^h - \bar{\theta}, t_n^h]$, together with the segment of $u^h(\cdot)$ on $[t_n^h - \bar{\theta}, t_n^h)$.

The distribution of ξ_{n+1}^h, given the initial data and $\{\xi_i^h, u_i^h, i \le n\}$, will depend only on the current memory segments $\bar{\xi}_n^h, \bar{u}_n^h$, and the current control u_n^h, and not on n otherwise. Let $E_{\hat{\xi},\hat{u},n}^{h,\alpha}$ denote the expectation given all data to step n, and that $\bar{\xi}_n^h = \hat{\xi}, \bar{u}_n^h = \hat{u}$, with control value α used at time n. Keep in mind that if control value α is used at step n for the chain, then it is used on $[t_n^h, t_{n+1}^h)$ for the interpolation $\xi^h(\cdot)$. Letting $r^h(\cdot)$ denote the relaxed control representation of $u^h(\cdot)$ and with the memory segment $\bar{u}_n^h$ being used, we can write the drift term as

$$\bar{b}(\bar{\xi}_n^h, \bar{u}_n^h, u_n^h) = \int_{-\bar{\theta}}^0 \int_{U^h} b(\bar{\xi}_n^h, \alpha, \theta) r^{h,\prime}(d\alpha, t_n^h + \theta) \mu_c(d\theta). \tag{1.17}$$

Example. Before proceeding with the general definition of local consistency when the control is delayed, which is essentially that used in Chapter 6 and in the previous subsection, let us consider a simple example. In (1.1), let

$$b(\bar{x}(t), \bar{u}(t)) = b_1(\bar{x}(t), u(t - \bar{\theta})) + b_0(\bar{x}(t), u(t)). \tag{1.18}$$

Then the measure $\mu_c(\cdot)$ is concentrated on the points $-\bar{\theta}$ and 0. The analog of the first line of (1.8) will be

$$E_{\bar{\xi}_n^h, \bar{u}_n^h, n}^{h, u_n^h} \Delta \xi_n^h = \left[b_1(\bar{\xi}_n^h, u(t_n^h - \bar{\theta})) + b_0(\bar{\xi}_n^h, u_n^h) \right] \Delta t_n^h + o(\Delta t_n^h). \tag{1.19}$$

Notation. Recall the definition of $\tilde{r}'(d\alpha, t, \theta)$ above (3.1.8) and in (4.4.2) and its role in the development of the approximating models in Chapter 4. Analogous definitions will be useful in the proofs of convergence in dealing with the various approximations to the piecewise constant control memory segments, as it will be the control memory segment at each t that is being approximated. For this purpose, define the relaxed control derivatives $\tilde{r}^{h,\prime}_{\tau}(d\alpha, t, \theta)$ and $\tilde{r}^{h,\prime}(d\alpha, t, \theta)$, for $t \in [0, \infty)$ and $\theta \in [-\bar{\theta}, 0]$, by

$$\begin{aligned} \tilde{r}^{h,\prime}_{\tau}(d\alpha, t, \theta) &= r^{h,\prime}(d\alpha, \tau^h_n + \theta), \quad \text{for} \;\; t \in [\tau^h_n, \tau^h_{n+1}), \\ \tilde{r}^{h,\prime}(d\alpha, t, \theta) &= r^{h,\prime}(d\alpha, t^h_n + \theta), \quad \text{for} \;\; t \in [t^h_n, t^h_{n+1}). \end{aligned} \tag{1.20}$$

Define the relaxed control derivative $\bar{r}^{h,\prime}_n$ with values $\bar{r}^{h,\prime}_n(d\alpha, \theta)$, for $\theta \in [-\bar{\theta}, 0)$, by

$$\bar{r}^{h,\prime}_n(d\alpha, \theta) = r^{h,\prime}(d\alpha, t^h_n + \theta). \tag{1.21}$$

The $\bar{r}^{h,\prime}_n$ is a representation of the control memory segment in terms of the derivative of its relaxed control representation, which we will find to be very useful. Using (1.20) and the fact that $\tilde{r}^{h,\prime}_{\tau}(d\alpha, s, \theta)$ is constant for $s \in [\tau^h_n, \tau^h_{n+1})$, we can write

$$\begin{aligned} \bar{b}(\bar{\xi}^h_n, \bar{u}^h_n, u^h_n)\Delta\tau^h_n &= \bar{b}(\bar{\xi}^h_n, \bar{r}^{h,\prime}_n, u^h_n)\Delta\tau^h_n \\ &= \int_{\tau^h_n}^{\tau^h_{n+1}} \int_{-\bar{\theta}}^{0} \int_{U^h} b(\bar{\xi}^h_n, \alpha, \theta)\tilde{r}^{h,\prime}_{\tau}(d\alpha, s, \theta)\mu_c(d\theta)ds. \end{aligned} \tag{1.22}$$

Equation (1.22) defines $\bar{b}(\bar{\xi}^h_n, \bar{r}^{h,\prime}_n, u^h_n)$, and we will use this notation when working in terms of relaxed controls.

The general definition of local consistency when the control is delayed. The local consistency condition for the chain is that there exists a function $\Delta t^h(\cdot)$ such that for $\hat{\xi} = \bar{\xi}^h_n$, with $\hat{\xi}(0) \in G_h$, and $\hat{u} = \bar{u}^h_n, \alpha = u^h_n$,

$$\begin{aligned} &E^{h,\alpha}_{\hat{\xi},\hat{u},n}\Delta\xi^h_n = \bar{b}_h(\hat{\xi}, \hat{u}, \alpha)\Delta t^h(\hat{\xi}, \hat{u}, \alpha) = \bar{b}(\hat{\xi}, \hat{u}, \alpha)\Delta t^h(\hat{\xi}, \hat{u}, \alpha) + o(\Delta t^h(\hat{\xi}, \hat{u}, \alpha)), \\ &E^{h,\alpha}_{\hat{\xi},\hat{u},n}\beta^h_n[\beta^h_n]' = a_h(\hat{\xi}, \hat{u}, \alpha)\Delta t^h(\hat{\xi}, \hat{u}, \alpha) = a(\hat{\xi})\Delta t^h(\hat{\xi}, \hat{u}, \alpha) + o(\Delta t^h(\hat{\xi}, \hat{u}, \alpha)), \\ &a(\hat{\xi}) = \sigma(\hat{\xi})\sigma'(\hat{\xi}), \\ &\sup_{n,\omega} |\xi^h_{n+1} - \xi^h_n| \xrightarrow{h} 0, \quad \sup_{\hat{\xi},\hat{u},\alpha} \Delta t^h(\hat{\xi}, \hat{u}, \alpha) \xrightarrow{h} 0. \end{aligned} \tag{1.23}$$

The relations in (1.23) define the functions $b_h(\cdot)$ and $a_h(\cdot)$. The reflecting boundary is treated exactly as it was when only the path was delayed, using transition probabilities satisfying (6.2.2).[5]

[5] Recall that $\Delta t^h_n = \Delta t^h(\bar{\xi}^h_n, \bar{u}^h_n, u^h_n)$ when the control is delayed.

The transition probabilities. The following analogs of (1.11) and (1.12) ensure the local consistency, if the same functions $N^h(\cdot)$ and $D^h(\cdot)$ are used. As for (1.11) and (1.12), it is only the dependence on $b(\cdot), a(\cdot), \tilde{x}$, and h that matters, no matter what the form of the memory segments $\hat{\xi}, \hat{u}$.

$$p^h(\hat{\xi},\hat{u},\tilde{x}|\alpha) = P\{\xi_1^h = \tilde{x}|\bar{\xi}_0^h = \hat{\xi}, \bar{u}_0^h = \hat{u}, u_0^h = \alpha\} = \frac{N^h(\bar{b}(\hat{\xi},\hat{u},\alpha), a(\hat{\xi}), \tilde{x})}{D^h(\bar{b}(\hat{\xi},\hat{u},\alpha), a(\hat{\xi}))},$$
$$\Delta t^h(\hat{\xi},\hat{u},\alpha) = T^h(\bar{b}(\hat{\xi},\hat{u},\alpha), a(\hat{\xi})) = \frac{h^2}{D^h(\bar{b}(\hat{\xi},\hat{u},\alpha), a(\hat{\xi}))}, \tag{1.24}$$

and

$$p^h(\bar{\xi}_n^h, \bar{u}_n^h, \tilde{x}|u_n^h) = P\{\xi_{n+1}^h = \tilde{x}|\bar{\xi}_n^h, \bar{u}_n^h, u_n^h\} = \frac{N^h(\bar{b}(\bar{\xi}_n^h, \bar{u}_n^h, u_n^h), a(\bar{\xi}_n^h), \tilde{x})}{D^h(\bar{b}(\bar{\xi}_n^h, \bar{u}_n^h, u_n^h), a(\bar{\xi}_n^h))}.$$

The notation $p^h(\bar{\xi}_n^h, \bar{r}_n^{h,\prime}, \tilde{x}|u_n^h)$ will also be used for $p^h(\bar{\xi}_n^h, \bar{u}_n^h, \tilde{x}|u_n^h)$. We formalize the above discussion as follows. The assumption is not needed if local consistency is otherwise ensured.

A1.2. *The transition probabilities and interpolation intervals are given in the form* (1.24), *where* (1.10) *is locally consistent for the nondelay case.*

Continuous-time interpolations. The continuous-time interpolations are defined as for the case where only the path is delayed, dealt with in the previous subsection. We will write out the expressions for the interpolation $\psi^h(\cdot)$ and the associated discounted cost that are analogous to (1.14) and (1.15). Extend the definition of $u_\tau^h(t)$ to the interval $[-\bar{\theta}, \infty)$ by letting it equal $\bar{u}_0^h(\theta)$ for $\theta \in [-\bar{\theta}, 0)$, and let $r_\tau^h(\cdot)$ denote the relaxed control representation of this extended $u_\tau^h(\cdot)$. Recalling the definition (1.20), for $\xi_0^h \in G_h$ the continuous time interpolation (1.14) is replaced by

$$\psi^h(t) = \xi_0^h + \int_{-\bar{\theta}}^0 \left[\int_0^t \int_{U^h} b_h(\bar{\xi}^h(q_\tau^h(s)), \alpha, \theta)\tilde{r}_\tau^{h,\prime}(d\alpha, s, \theta)ds\right]\mu_c(d\theta) \tag{1.25}$$
$$+B_\tau^h(t) + z_\tau^h(t).$$

Let $E_{\hat{\xi},\hat{u}}^{h,u^h}$ denote the expectation under initial data $\bar{\xi}_0^h = \hat{\xi}$ and control sequence $u^h = \{u_n^h, n \geq 0\}$, with initial control segment (on $[-\bar{\theta}, 0)$) being $\hat{u}$. The analog of the cost function (1.13) is

$$W^h(\hat{\xi},\hat{u},u^h) = E_{\hat{\xi},\hat{u}}^{h,u^h} \sum_{n=0}^{\infty} e^{-\beta t_n^h} \left[\bar{k}(\bar{\xi}_n^h, \bar{u}_n^h, u_n^h)\Delta t_n^h + q'\Delta y_n^h\right],$$
$$V^h(\hat{\xi},\hat{u}) = \inf_{u^h} W^h(\hat{\xi},\hat{u},u^h), \tag{1.26}$$

where $\bar{k}(\cdot)$ is defined analogously to $\bar{b}(\cdot)$ in (1.17).

In integral and relaxed control form, and modulo an asymptotically negligible error due to the approximation of the discount factor, (1.26) equals

$$\begin{aligned} &W^h(\hat{\xi}, \hat{u}, u^h) \\ &= E^{h,u^h}_{\hat{\xi},\hat{u}} \int_{-\bar{\theta}}^{0} \mu_c(d\theta) \left[\int_0^\infty dt \int_{U^h} e^{-\beta t} k(\bar{\xi}^h(q^h_\tau(t)), \alpha, \theta) \tilde{r}^{h,\prime}_\tau(d\alpha, t, \theta) \right] \\ &+ E^{h,u^h}_{\hat{\xi},\hat{u}} \int_0^\infty e^{-\beta t} q' dy^h_\tau(t). \end{aligned} \tag{1.27}$$

The following convergence theorem, whose proof is in Section 5 of the next chapter, says that any method for solving the control problem for any locally consistent approximation will yield an approximation to the value for the original model (1.1) or (1.2).

Theorem 1.2. *Let* $\xi^h_n, \Delta t^h_n$ *be locally consistent with* (1.1) *or* (1.2) *in the sense of* (1.23)*, with initial data* $\bar{x}(0)$*, a continuous function on* $[-\bar{\theta}, 0]$*, and* $\bar{u}(0) \in L_2(U; [-\bar{\theta}, 0))$*. The cost function for* (1.2) *is* (1.4) *and that for the approximating chain is* (1.26)*. Let* $\bar{\xi}^h_0 \in D(G_h; [-\bar{\theta}, 0])$ *be piecewise-constant, and converge to* $\bar{x}(0)$ *uniformly on* $[-\bar{\theta}, 0]$*. Let* $\bar{u}^h_0 \in D(U^h; [-\bar{\theta}, 0))$ *be piecewise-constant and converge to* $\bar{u}(0)$ *in the sense of* L_2*. Assume* (A3.1.2), (A3.1.3), *and* (A3.2.1)–(A3.2.3), (A3.4.3)*. Then* $V^h(\bar{\xi}^h_0, \bar{u}^h_0) \to V(\bar{x}(0), \bar{u}(0))$ *as* $h \to 0$*.*

7.1.4 Absorbing Boundaries and Other Cost Functions

The next theorem covers the case where the boundary is absorbing rather than reflecting. The proof will be discussed in Section 5 of the next chapter.

Theorem 1.3. *Assume the conditions of either Theorems* 1.1 *or* 1.2*, except those on the reflection directions. Use the cost function* (3.4.1) *if the control is not delayed and* (3.4.2) *if the control is delayed. Assume* (A3.4.1) *and* (A3.4.2)*. For the chain let* N^h_G *denote the first time that it leaves* G^0*, the interior of* G*, and use either the cost function*

$$W^h(\hat{\xi}, u^h) = E^{h,u^h}_{\hat{\xi}} \left[\sum_{n=0}^{N^h_G - 1} e^{-\beta t^h_n} k(\bar{\xi}^h_n, u^h_n) \Delta t^h(\bar{\xi}^h_n, u^h_n) + e^{-\beta N^h_G} g_0(\xi^h_{N^h_G}) \right], \tag{1.28}$$

or

$$\begin{aligned} &W^h(\hat{\xi}, \hat{u}, u^h) \\ &= E^{h,u^h}_{\hat{\xi},\hat{u}} \left[\sum_{n=0}^{N^h_G - 1} e^{-\beta t^h_n} \bar{k}(\bar{\xi}^h_n, \bar{u}^h_n, u^h_n) \Delta t^h(\bar{\xi}^h_n, \bar{u}^h_n, u^h_n) + e^{-\beta N^h_G} g_0(\xi^h_{N^h_G}) \right], \end{aligned} \tag{1.29}$$

according to the case. Then, according to the case, as $h \to 0$, $V^h(\bar{\xi}_0^h) \to V(\bar{x}(0))$ *or* $V^h(\bar{\xi}_0^h, \bar{u}_0^h) \to V(\bar{x}(0), \bar{u}(0))$.

Optimal stopping. Suppose that we have the option of stopping before G^0 is exited. Then replace N_G^h by the minimum of N_G^h and the stopping time. The theorem continues to hold. Similarly, Theorems 1.1 and 1.2 hold if we allow stopping with a continuous stopping cost. See the development of the optimal stopping problem in [58].

7.1.5 Approximations to the Memory Segments

In applications, keeping the full computed memory segments $\bar{\xi}_n^h, \bar{u}_n^h$ (or $\bar{r}_n^{h,\prime}$) might be too costly in terms of memory. Specific approximations based on truncations and discretizations will be discussed in the next chapter, and an approximation if only the path is delayed is discussed in Section 3. Considerable flexibility is possible in the modeling of the memory segments. It is preferable to use relaxed control notation for the control memory segments, and this will be done in terms of its derivative. So, following the notation for the control memory segment in (1.21), when the full memory segments[6] are used, let us rewrite the equation below (1.24):

$$p^h(\bar{\xi}_n^h, \bar{r}_n^{h,\prime}, \tilde{x} | u_n^h) = \frac{N^h(\bar{b}(\bar{\xi}_n^h, \bar{r}_n^{h,\prime}, u_n^h), a(\bar{\xi}_n^h), \tilde{x})}{D^h(\bar{b}(\bar{\xi}_n^h, \bar{r}_n^{h,\prime}, u_n^h), a(\bar{\xi}_n^h))}, \tag{1.30a}$$

where $\bar{b}(\bar{\xi}_n^h, \bar{r}_n^{h,\prime}, u_n^h)$ is defined in (1.22).

Approximations: Definitions. The approximations to the full memory segments $(\bar{\xi}_n^h, \bar{r}_n^{h,\prime})$ will be denoted by $(\bar{\xi}_{a,n}^{h,\kappa}, \bar{r}_{a,n}^{h,\kappa,\prime})$, where, for $\theta \in [-\bar{\theta}, 0)$, $\bar{r}_{a,n}^{h,\kappa,\prime}(\theta)$ is a probability measure on U^h, and $\bar{\xi}_{a,n}^{h,\kappa}(\theta)$ is G_h-valued for $\theta \in [-\bar{\theta}, 0)$, and $\bar{\xi}_{a,n}^{h,\kappa}(0)$ will have values either in G_h or in the set of reflecting states ∂G_h^+. The variable $\kappa \to 0$ is a parameter of the approximation. It will also be used to index the associated chain, control, interpolation interval, and so forth, and in the applications will generally take the values δ or (δ_0, δ), analogously to the parameters of the approximations used in Chapter 4. The subscript "a" denotes the type of memory segment approximation, analogously to the usage with the approximations in Chapter 4 (e.g., random, periodic, periodic-Erlang), and, unless noted otherwise, it will be used to index only the approximating memory segments and the relaxed control representation of the approximating control memory segment.

With these approximations used, the true transition probabilities are

[6] The full memory segments at iterate n are the interpolations (with intervals $\{\Delta t_n^h\}$) of the paths and control, resp., over the intervals $[t_n^h - \bar{\theta}, t_n^h]$ and $[t_n^h - \bar{\theta}, t_n^h)$, resp.

$$p^h(\bar{\xi}^{h,\kappa}_{a,n}, \bar{r}^{h,\kappa,\prime}_{a,n}, \tilde{x}|\alpha = u^{h,\kappa}_n) = \frac{N^h(\bar{b}(\bar{\xi}^{h,\kappa}_{a,n}, \bar{r}^{h,\kappa,\prime}_{a,n}, u^{h,\kappa}_n), a(\bar{\xi}^{h,\kappa}_{a,n}), \tilde{x})}{D^h(\bar{b}(\bar{\xi}^{h,\kappa}_{a,n}, \bar{r}^{h,\kappa,\prime}_{a,n}, u^{h,\kappa}_n), a(\bar{\xi}^{h,\kappa}_{a,n}))}, \tag{1.30b}$$

where $\bar{b}(\bar{\xi}^{h,\kappa}_{a,n}, \bar{r}^{h,\kappa,\prime}_{a,n}, u^{h,\kappa}_n)$ is defined by (1.22) with $(\bar{\xi}^{h,\kappa}_{a,n}, \bar{r}^{h,\kappa,\prime}_{a,n}, u^{h,\kappa}_n)$ replacing $(\bar{\xi}^h_n, \bar{r}^{h,\prime}_n, u^h_n)$. Define $\Delta t^{h,\kappa}_n = \Delta t^h(\bar{\xi}^{h,\kappa}_{a,n}, \bar{u}^{h,\kappa}_{a,n}, u^{h,\kappa}_n)$, $t^{h,\kappa}_n = \sum_{i=0}^{n-1} \Delta t^{h,\kappa}_i$, with analogous definitions for $\Delta\tau^{h,\kappa}_n$ and $\tau^{h,\kappa}_n$.

For whatever the type "a" of the approximation, let the relaxed control that is defined by the controls $\{u^{h,\kappa}_n\}$ with interpolation intervals $\{\Delta\tau^{h,\kappa}_n\}$ be denoted by $r^{h,\kappa}_\tau(\cdot)$, and let that defined by the interpolation with intervals $\{\Delta t^{h,\kappa}_n\}$ be denoted by $r^{h,\kappa}(\cdot)$. Define the following function of α, t and θ, where $\theta \in [-\bar{\theta}, 0)$:

$$\bar{r}^{h,\kappa,\prime}_n(d\alpha, \theta) = r^{h,\kappa,\prime}(d\alpha, t^{h,\kappa}_n + \theta). \tag{1.31a}$$

$\bar{r}^{h,\kappa,\prime}_n(\cdot)$ is the full memory segment defined by the actual realized control on the interval $[t^{h,\kappa}_n - \bar{\theta}, t^{h,\kappa}_n)$. Keep in mind that it is not necessarily equal to the approximating memory segment $\bar{r}^{h,\kappa,\prime}_{a,n}(\cdot)$, which is the one that is actually used in the dynamics and cost function at step n of the chain when the approximation type is "a."

The following functions of α, t and θ, where $\theta \in [-\bar{\theta}, 0]$, will be useful in analyzing the approximations and their convergence:

$$\left.\begin{aligned} \tilde{r}^{h,\kappa,\prime}_a(d\alpha, t, \theta) &= \bar{r}^{h,\kappa,\prime}_{a,n}(d\alpha, \theta), \quad \text{for } \theta \in [-\bar{\theta}, 0) \\ \tilde{r}^{h,\kappa,\prime}_a(d\alpha, t, 0) &= I_{\{u^{h,\kappa}_n \in d\alpha\}}, \end{aligned}\right\} \text{for } t \in [t^{h,\kappa}_n, t^{h,\kappa}_{n+1}),$$
$$\left.\begin{aligned} \tilde{r}^{h,\kappa,\prime}_{a,\tau}(d\alpha, t, \theta) &= \bar{r}^{h,\kappa,\prime}_{a,n}(d\alpha, \theta), \quad \text{for } \theta \in [-\bar{\theta}, 0) \\ \tilde{r}^{h,\kappa,\prime}_{a,\tau}(d\alpha, t, 0) &= I_{\{u^{h,\kappa}_n \in d\alpha\}}, \end{aligned}\right\} \text{for } t \in [\tau^{h,\kappa}_n, \tau^{h,\kappa}_{n+1}). \tag{1.31b}$$

We will always use the definitions:

$$\begin{aligned} &\bar{\xi}^{h,\kappa}_a(\cdot) \text{ is the interpolation of } \{\bar{\xi}^{h,\kappa}_{a,n}\}, \text{ with intervals } \{\Delta t^{h,\kappa}_n\}, \\ &\bar{\xi}^{h,\kappa}_n \text{ is the full memory path segment } \{\xi^{h,\kappa}(t^{h,\kappa}_n + \theta), \theta \in [-\bar{\theta}, 0]\}. \end{aligned} \tag{1.32}$$

The interpolated process $\psi^{h,\kappa}(\cdot)$ with the memory segment approximation. With the above definitions, we can write the analog of the interpolation (1.14) with the approximating memory segments used as

$$\begin{aligned} \psi^{h,\kappa}(t) = \xi^h_0 + \int_{-\bar{\theta}}^0 \left[\int_0^t \int_{U^h} b_h(\bar{\xi}^{h,\kappa}_a(q^{h,\kappa}_\tau(s)), \alpha, \theta)\tilde{r}^{h,\kappa,\prime}_{a,\tau}(d\alpha, s, \theta)\right] \mu_c(d\theta)ds \\ + B^{h,\kappa}_\tau(t) + z^{h,\kappa}_\tau(t), \end{aligned} \tag{1.33}$$

where the martingale $B^{h,\kappa}_\tau(\cdot)$ has quadratic variation process

$$\int_0^t a_h(\bar{\xi}_a^{h,\kappa}(q_\tau^{h,\kappa}(s)))ds.$$

General assumptions on the approximating memory segments and a convergence theorem. In subsequent sections and in Chapter 8, particular approximations will be proposed. But for maximum usefulness and simplicity of the proofs, it is convenient to state a convergence theorem in terms of some general properties. Suppose that

$$\lim_{h\to 0} \sup_{\text{control}} \sup_n E \sup_{-\bar{\theta}\le\theta\le 0} \left|\bar{\xi}_{a,n}^{h,\kappa}(\theta) - \bar{\xi}_n^{h,\kappa}(\theta)\right| = 0 \tag{1.34}$$

and (note that the upper limit of integration is $0-$)

$$\sup_{\text{control}} \sup_n E \left| \int_{-\bar{\theta}}^{0-} \int_{U^h} f(\alpha,\theta)\left[r^{h,\kappa,\prime}(d\alpha, t_n^h+\theta) - \bar{r}_{a,n}^{h,\kappa,\prime}(d\alpha,\theta)\right]\mu_c(d\theta)\right| \to 0 \tag{1.35}$$

for each bounded and continuous real-valued function $f(\cdot)$, as $h \to 0$ and $\kappa \to 0$. Then, the approximations and the full memory segments are close for small κ and h, and the drift rate at iterate n of the chain is approximated as follows:

Drift rate under the approximating memory segments =

$$\begin{aligned}&\int_{-\bar{\theta}}^{0-}\int_{U^h} b(\bar{\xi}_{a,n}^{h,\kappa},\alpha,\theta)\bar{r}_{a,n}^{h,\kappa,\prime}(d\alpha,\theta)\mu_c(d\theta) + b(\bar{\xi}_{a,n}^{h,\kappa}, u_n^{h,\kappa},0)\mu_c(\{0\})\\ &\quad\approx \int_{-\bar{\theta}}^{0-}\int_{U^h} b(\bar{\xi}_n^{h,\kappa},\alpha,\theta)\bar{r}_n^{h,\kappa,\prime}(d\alpha,\theta)\mu_c(d\theta) + b(\bar{\xi}_n^{h,\kappa}, u_n^{h,\kappa},0)\mu_c(\{0\}).\end{aligned} \tag{1.36}$$

Condition (1.35) is quite strong because it concerns the behavior at each iterate. Consider the following weaker condition, which allows us to consider averages of the differences between the full control memory segment and its approximations over a finite time interval. For bounded and continuous $f(\cdot)$, replace (1.35) by the assumption that

$$E\left|\int_t^{t+\Delta} ds \int_{-\bar{\theta}}^{0}\int_{U^h} f(s,\alpha,\theta)\left[r^{h,\kappa,\prime}(d\alpha,s+\theta) - \tilde{r}_a^{h,\kappa,\prime}(d\alpha,s,\theta)\right]\mu_c(d\theta)\right| \to 0 \tag{1.37}$$

as $h \to 0$, uniformly in the control and in t for each $\Delta > 0$. Using this, (1.34), and the timescale equivalences in Theorem 3.1 will allow us to asymptotically approximate the drift term in (1.33) by

$$\int_{-\bar{\theta}}^{0}\left[\int_0^t\int_{U^h} b_h(\bar{\xi}^{h,\kappa}(q_\tau^{h,\kappa}(s)),\alpha,\theta) r_\tau^{h,\kappa,\prime}(d\alpha,s+\theta)\right]\mu_c(d\theta)ds. \tag{1.38}$$

Let $V^{h,\kappa}(\hat{x},\hat{u})$ denote the optimal cost function for the model modified as above, using approximating memory segments $\bar{\xi}_{a,n}^{h,\kappa}$ and $\bar{r}_{a,n}^{h,\kappa,\prime}$. Then we have the following result.

Theorem 1.4. *Assume the conditions of Theorems* 1.1, 1.2, *or* 1.3, *but with the use of memory segment approximations* $\bar{\xi}^{h,\kappa}_{a,n}$ *and* $\bar{r}^{h,\kappa,\prime}_{a,n}$ *satisfying* (1.34) *and* (1.37), *resp. Then* $V^{h,\kappa}(\bar{\xi}^h_0, \bar{u}^h_0) \to V(\bar{x}(0), \bar{u}(0))$ *as* $h \to 0$ *and then* $\kappa \to 0$.

7.2 Computational Procedures

7.2.1 Delay in the Path Only: State Representations and the Bellman Equation

Theorem 1.1 gave sufficient conditions for a numerical approximation to the optimal control problem for system (1.5) and cost function (1.6) to converge to the optimal value as the approximation parameter h goes to zero. But it does not give any hint as to how the approximation might be constructed so that the numerical procedure is actually reasonable from a computational perspective. Suppose that the process ξ^h_n is locally consistent and the transition probabilities satisfy (1.11). Because the transition probabilities in (1.11) depend on $\hat{\xi}$, a key problem is that the state space must include the information that is needed to define $\hat{\xi}$, and this might require considerable memory. The effective use of dynamic programming methods requires that the system (i.e., the memory) state be embedded into a finite-state Markov chain. The size and structure of this chain determines the numerical feasibility of the algorithm, and this is the subject of the rest of this section. The next section and Chapter 8 show some advantages of the implicit approximation method as well as of methods motivated by it. Keep in mind that the reflection directions depend only on the reflecting point, as the reflection directions do not depend on delayed values and are not controlled.

A first and crude Markov chain representation. We will begin the discussion of representations and approximations of the path memory segment with a rather crude form. Until further notice, continue to use the interpolation $\xi^h(\cdot)$ defined above (1.7a). Let us start with the memory state at step n being $\bar{\xi}^h_n$, defined in (1.7b), which we recall is a piecewise-constant function with $\bar{\xi}^h_n(\theta) = \xi^h(t^h_n + \theta)$, for $\theta \in [-\bar{\theta}, 0]$. All of its values must be in G_h, except possibly the most recent one, $\bar{\xi}^h_n(0) = \xi^h_n$, which can take values in either G_h or ∂G^+_h.

The $\bar{\xi}^h_n$ can be represented in terms of a finite-state Markov process as follows. Let $\overline{\Delta}^h = \inf_{\alpha,\hat{\xi}} \Delta t^h(\hat{\xi}, \alpha)$, where $\alpha \in U^h$ and $\hat{\xi} \in D(G_h; [-\bar{\theta}, 0])$. Suppose (w.l.o.g.) that $\bar{\theta}/\overline{\Delta}^h = K^h$ is an integer. The interpolated time interval $[t^h_n - \bar{\theta}, t^h_n]$ is covered by at most K^h intervals of length $\bar{\Delta}^h$. The reflection states do not appear in the construction of $\bar{\xi}^h_n(\theta)$, for $\theta < 0$, but it is possible that $\bar{\xi}^h_n(0) \in \partial G^+_h$. Suppose that $\xi^h_n \in G_h$. Let $\xi^h_{n,i}, i > 0$, denote the ith non-reflection state before step n, and $\Delta t^h_{n,i}$ the associated interpolation interval. Then we can represent $\bar{\xi}^h_n$ in terms of $\{(\xi^h_{n,K^h}, \Delta t^h_{n,K^h}), \ldots, (\xi^h_{n,1}, \Delta t^h_{n,1}), \xi^h_n\}$.

If $\xi_n^h \notin G_h$, so that it is a reflecting state, then to compute the transition probability to the next state the values of the path before step n are not needed and the above vector is still a complete description of the needed memory.

This new representation clearly evolves as a $(2K^h + 1)$-dimensional controlled Markov chain, although it will usually be much too complicated to be of any practical use for computation. If the interpolation interval $\Delta t^h(\hat{\xi}, \alpha)$ is not constant, then the construction of the $\bar{\xi}_n^h$ requires that we keep a record of the values of both the $\xi_i^h, \Delta t_i^h$, for the indices i that contribute to $\bar{\xi}_n^h$. The use of constant interpolation intervals simplifies this problem. Consider the special case where $\Delta t^h(\hat{\xi}, \alpha)$ is a constant. This would be the case if $\sigma(\cdot)$ were a constant and an approximation analogous to that in the example in Section 6.4 were used. Then the vector $\{(\xi_{n,K^h}^h, \ldots, \xi_{n,1}^h, \xi_n^h\}$ evolves as a Markov process and $\bar{\xi}_n^h$ is a piecewise-constant and right-continuous (except possibly at $\theta = 0$) interpolation of these values, with $\bar{\xi}_n^h(0) = \xi_n^h$. We can identify $\bar{\xi}_n^h$ with this vector without ambiguity.

Transforming to a constant interpolation interval. If $\Delta t^h(\hat{\xi}, \alpha)$ is not constant, then (6.2.7) showed how to transform the transition probabilities to yield a chain with a constant interpolation interval for the no-delay case, and we now write the analogous equations for the delay case. Let $\bar{p}^h(\cdot)$ denote the transition probabilities for the constant interpolation interval case and use the form (1.11) for $p^h(\hat{\xi}, \tilde{x}|\alpha)$. Suppose (w.l.o.g.) that a state does not transit to itself in that $p^h(\hat{\xi}, \hat{\xi}(0)|\alpha) = 0$. To get the transition probabilities $\bar{p}^h(\cdot)$ for the delay case with the constant interpolation interval $\bar{\Delta}^h$, use the analog of (6.2.7):

$$\begin{aligned} \bar{p}^h(\hat{\xi}, \tilde{x}|\alpha) &= p^h(\hat{\xi}, \tilde{x}|\alpha)\left(1 - \bar{p}^h(\hat{\xi}, \hat{\xi}(0)|\alpha)\right) \\ \bar{p}^h(\hat{\xi}, \hat{\xi}(0)|\alpha) &= 1 - \frac{\bar{\Delta}^h}{\Delta t^h(\hat{\xi}, \alpha)}. \end{aligned} \tag{2.1}$$

A one-dimensional example with a constant interpolation interval. Let $\Delta t^h(\hat{\xi}, \alpha) = \bar{\Delta}^h$, so that the interpolation interval is constant. Detailed examination of the memory vector suggests various ways of simplifying the state space. To simplify the presentation, until further notice we let $x(t)$ be one-dimensional with $G = [0, B]$, where $B > 0$ is assumed to be an integral multiple of the approximation parameter h. We assume that nonreflection states move only to their nearest neighbors. Then $G_h = \{0, h, \ldots, B\}$ and the reflection states are $\{-h, B + h\}$.

For $\hat{\xi}(0) \in G_h$, the Bellman equation for the process defined by this chain with cost (1.13) can be written as

$$V^h(\hat{\xi}) = \inf_{\alpha \in U^h} \left[e^{-\beta \bar{\Delta}^h} \sum_{\pm} p^h(\hat{\xi}, \hat{\xi}(0) \pm h|\alpha) V^h(\hat{y}^{\pm}) + k(\hat{\xi}, \alpha)\bar{\Delta}^h \right]. \tag{2.2a}$$

The terms $\hat{y}^{\pm}$ denote the functions on $[-\bar{\theta}, 0]$ that represent the memory segment at the next step, where the state of the chain is $\xi_1^h = \hat{\xi}(0) \pm h$. The values are obtained as follows:

$$\hat{y}^{\pm}(\theta) = \hat{\xi}(\theta + \bar{\Delta}^h), \ -\bar{\theta} \le \theta < -\bar{\Delta}^h,$$

$$\hat{y}^{\pm}(\theta) = \hat{\xi}(0), \ -\hat{\Delta}^h \le \theta < 0, \quad \hat{y}^{\pm}(0) = \hat{\xi}(0) \pm h.$$

If $\hat{\xi}(0)$ is a reflecting state, then there is no shift and only the value $\hat{\xi}(0)$ changes. It becomes ξ_1^h. In particular, if $\hat{\xi}(0) = -h$, then $\Delta t^h(\hat{\xi}, \alpha) = 0$ and

$$V^h(\hat{\xi}) = V^h(\hat{\xi}^+) + q_1 h, \tag{2.2b}$$

where $\hat{\xi}^+(\theta)$ equals $\hat{\xi}(\theta)$, except at $\theta = 0$ where $\hat{\xi}^+(0) = 0$. If $\hat{\xi}(0) = B + h$, then $\Delta t^h(\hat{\xi}, \alpha) = 0$ and

$$V^h(\hat{\xi}) = V^h(\hat{\xi}^-) + q_2 h, \tag{2.2c}$$

where $\hat{\xi}^-(\theta) = \hat{\xi}(\theta)$, except for $\theta = 0$, where $\hat{\xi}^-(0) = B$. Owing to the contraction due to the discounting, there is a unique solution to (2.2).

More simply, as noted above we can represent $\bar{\xi}_n^h$ unambiguously as

$$\bar{\xi}_n^h = (\xi_{n,K^h}^h, \ldots, \xi_{n,1}^h, \xi_n^h).$$

If $\xi_n^h \in G_h$, then we can represent $\bar{\xi}_{n+1}^h$ unambiguously as

$$\bar{\xi}_{n+1}^h = (\xi_{n,K^h-1}^h, \ldots, \xi_{n,1}^h, \xi_n^h, \xi_{n+1}^h).$$

If $\xi_n^h = -h$, then we can represent $\bar{\xi}_{n+1}^h$ unambiguously as

$$\bar{\xi}_{n+1}^h = (\xi_{n,K^h}^h, \cdots, \xi_{n,1}^h, 0),$$

and analogously if $\xi_n^h = B+h$. With this representation, the maximum number of possible values can be very large, up to $(B/h + 1)^{K^h}(B/h + 3)$ where, typically, $K^h = O(1/h^2)$.

Simplifying the state representation by using differences. The representation that is used for the memory segment in the above one-dimensional example requires a state space of enormous size. This can be reduced by using the standard data compression method of using only the current ξ_n^h and the differences between successive values. This gives the representation

$$\bar{\xi}_n^h = (c_{n,K^h}^h, \cdots, c_{n,1}^h, \xi_n^h),$$

where

$$\begin{aligned} c_{n,1}^h &= \xi_{n,1}^h - \xi_n^h \\ c_{n,i}^h &= \xi_{n,i}^h - \xi_{n,i-1}^h, \ \text{for } 1 < i \le K^h. \end{aligned} \tag{2.3}$$

If the path can move only its nearest neighbors, then the $c_{n,i}^h$ take at most two values, and the number of values in the state space is reduced to $2^{K^h}(B/h+3)$. The two values and the reconstruction of the $\xi_{n,i}^h$ from them are easily determined by an iterative procedure. For example, if $\xi_n^h = -h$, then $\xi_{n,1}^h = 0$. If $\xi_n^h = 0$, then $\xi_{n,1}^h \in \{0, h\}$. If ξ_n^h is not a reflecting or boundary value then $\xi_{n,1}^h = \xi_{n-1}^h = \xi_n^h \pm h$. If $\xi_{n,i}^h = 0$, then $\xi_{n,i-1}^h \in \{0, h\}$. If $\xi_{n,i}^h$ is not a boundary value (it cannot be a reflecting state), then $\xi_{n,i-1}^h = \xi_{n,i}^h \pm h$, and so forth.

If $\Delta t^h(\hat{\xi}, \alpha)$ is not constant, so that we need to use (2.1) to transform the transition probabilities, we then have the possibility of transitions from a state to itself, since $\bar{p}^h(\hat{\xi}, \hat{\xi}(0)|\alpha)$ might not now be zero. Because $\xi_{n+1}^h - \xi_n^h \in \{-h, 0, h\}$, each of the $c_{n,i}^h$ can take as many as three values and we have at most $(B/h+3)3^{K^h}$ points in the state space. Theorem 1.1 holds. Keep in mind that the memory state at time $n+1$ must be computable from the memory state at time n and the new value ξ_{n+1}^h. The use of differences reduces the memory requirements, but at the price of increased computation. It would be worthwhile to evaluate other data coding and compression schemes, even those with a small loss of information.

The approaches in Section 3 and in the next chapter use fewer intervals to cover $[-\bar{\theta}, 0]$ and have the promise of being more efficient in terms of memory requirements as they use approximations to the path over interpolation intervals that are larger than $\bar{\Delta}^h$.

7.2.2 Delay in Both Path and Control

Now suppose that both the control and the path are delayed, with the maximum delay being $\bar{\theta}$ for each. The memory requirements can be greatly increased. In this subsection, we suppose that $\Delta t_n^h = \bar{\Delta}^h$, a constant, and give a representation of the memory segment of the control process that is an analog of the representation that was used for the path in the previous subsection. The general case will be dealt with in the next chapter.

For illustrative purposes, let us continue to work with a one-dimensional example and the notation of the previous subsection. Let $u_{n,i}^h$ denote the control action that was used in the ith no-reflection step before step n. Let $p^h(\hat{\xi}, \hat{u}; \tilde{x}|\alpha)$ denote the probability $P\{\xi_1^h = \tilde{x} | \bar{\xi}_0^h = \hat{\xi}, \bar{u}_0^h = \hat{u}, u_0^h = \alpha\}$. Analogously to what was done in the previous subsection, the memory variables can be embedded into a Markov process, with values at time n being

$$\left\{ (\xi_{n,K^h}^h, u_{n,K^h}^h), \ldots, (\xi_{n,1}^h, u_{n,1}^h), \xi_n^h \right\}.$$

The analog of (2.2a) with cost function (1.29) is, for $\hat{\xi}(0) \in G_h$,

$$V^h(\hat{\xi}, \hat{u}) = \inf_{\alpha \in U^h} \left[e^{-\beta \bar{\Delta}^h} \sum_{\pm} p^h(\hat{\xi}, \hat{u}; \hat{\xi}(0) \pm h|\alpha) V^h(\hat{y}^{\pm}, \hat{u}_\alpha) + \bar{k}(\hat{\xi}, \hat{u}, \alpha) \bar{\Delta}^h \right], \tag{2.4}$$

where $\hat{y}^{\pm}$ denotes the new "path memory sections" defined below (2.2a). The new "control memory segment" depends on the current choice of control, namely α. The interpolated form is $\hat{u}_\alpha$, defined by

$$\hat{u}_\alpha(\theta) = \hat{u}(\theta + \bar{\Delta}^h), \ -\bar{\theta} \le \theta < -\bar{\Delta}^h,$$
$$\hat{u}_\alpha(\theta) = \alpha, \ -\bar{\Delta}^h \le \theta < 0.$$

It can be unambiguously represented in the form

$$\hat{u}_\alpha = \left(u^h_{n,K^h-1}, \ldots, u^h_{n,1}, \alpha \right).$$

The reflecting states are treated as for the no-delay case. Because of the contraction due to the discounting, there is a unique solution to (2.4).

We can use the more efficient representation (2.3) for the path variable. However, the total memory requirements with this approach would be large, unless U^h itself can be approximated by only a few values. Suppose that $U = U^h$ consists of only the two points $\{0, 1\}$. Then the number of points needed to represent the control memory segment is 2^{K^h}, comparable to what was needed for the one-dimensional problem of the previous subsection where only the path was delayed. If only the control were delayed, then this crude representation for the control memory would be more acceptable.

The dynamics depend on delayed values of the control, but not the current value. In this case, $p^h(\hat{\xi}, \hat{u}; \tilde{x}|\alpha)$ does not depend on the current control choice α, and (2.4) simplifies to

$$V^h(\hat{\xi}, \hat{u}) = \inf_{\alpha \in U^h} \left[e^{-\beta \bar{\Delta}^h} \sum_{\pm} p^h(\hat{\xi}, \hat{u}; \hat{\xi}(0) \pm h) V^h(\hat{y}^{\pm}, \hat{u}_\alpha) + \bar{k}(\hat{\xi}, \hat{u}, \alpha) \bar{\Delta}^h \right]. \tag{2.5}$$

7.2.3 A Comment on Higher-Dimensional Problems

The discussion in the previous subsection concentrated on one-dimensional models. The representations of the memory all extend to higher-dimensional problems, but the required memory grows exponentially in the dimension. When the path only is delayed, there are representations that are analogous to (2.3). Consider a two-dimensional problem in a box $[0, B_1] \times [0, B_2]$, with the same path delay in each coordinate, no control delay, and discretization level h in each coordinate. The ξ^h_n in (2.3) is replaced by vector containing the current two-dimensional value of the chain. The difference $c_i = \xi^h_{n,i} - \xi^h_{n,i-1}$ is now a two-dimensional vector. The values can be computed iteratively, as for the one-dimensional case, but the somewhat boring details will not be presented here.

7.3 The Implicit Numerical Approximation: Path Delayed

The implicit method of constructing the approximating chain that was introduced in Section 6.5 can play an important role in reducing the memory requirements and state space size. It also serves as the basis of a variety of other useful approximations with memory requirements that are less than what was needed in Section 2.[7] The equations (6.5.6) provided a simple way of getting the transition probabilities and interpolation interval for the implicit approximation method directly from those for the explicit approximation method for the no-delay problem. The approach is the same for the problem with delays. In this section, we concentrate on the model where only the path is delayed. Further developments are in the next chapter.

Let $\delta > 0$ be the discretization interval for the time variable, with $h^2/\delta \to 0$ as $h \to 0, \delta \to 0$. As in Section 6.5, let $\xi_n^{h,\delta}$ denote the state process for the spatial component, $\phi_n^{h,\delta}$ that for the time variable, and define $\zeta_n^{h,\delta} = (\phi_n^{h,\delta}, \xi_n^{h,\delta})$. To get the transition probabilities, one starts with the delay form of (6.5.6), where the $p^h(\cdot)$ are defined as in Section 1.

7.3.1 Local Consistency and the Memory Segment

Transition probabilities. In this section, let $\bar{\xi}_{r,n}^{h,\delta}$ denote the path memory segment that is used at iterate n for the chain. It will replace the $\bar{\xi}_n^h$ that was used in Sections 1 and 2 and will be defined precisely in (3.8) after defining the transition probabilities and interpolations. As for the method of Sections 1 and 2, it is a function on $[-\bar{\theta}, 0]$ with the value at $\theta = 0$ being $\bar{\xi}_{r,n}^{h,\delta}(0) = \xi_n^{h,\delta}$. The canonical value of $\bar{\xi}_{r,n}^{h,\delta}$ is again denoted by $\hat{\xi}$. The subscript r is used owing to the relationship with the random delay approximation of (4.2.7). With the implicit approximation method, there are several possibilities for the interpolation that defines the memory segment, and the choice affects the computational complexity.

Let $p^{h,\delta}(\hat{\xi}, i\delta; \tilde{x}, i\delta|\alpha)$ denote the probability that $\xi_{n+1}^{h,\delta} = \tilde{x}$ and that $\phi_{n+1}^{h,\delta} = i\delta$, given all past data and $\bar{\xi}_{r,n}^{h,\delta} = \hat{\xi}$, $\phi_n^{h,\delta} = i\delta$, $u_n^{h,\delta} = \alpha$ (i.e., the time variable is not advancing). Let $p^{h,\delta}(\hat{\xi}, i\delta; \hat{\xi}(0), i\delta + \delta|\alpha)$ denote the probability that $\xi_{n+1}^{h,\delta} = \xi_n^{h,\delta}$ and $\phi_{n+1}^{h,\delta} = i\delta + \delta$, given all past data and the values $u_n^{h,\delta} = \alpha$, and $\bar{\xi}_{r,n}^{h,\delta} = \hat{\xi}$, $\phi_n^{h,\delta} = i\delta$, with $\hat{\xi}(0) = \xi_n^{h,\delta}$ (i.e., the time variable is advancing, and the spatial state does not change). These probabilities depend on the past only via the value of the current path memory segment $\hat{\xi}$.

[7] Since we do not know the rate of convergence as a function of the parameters of the various approximations, this assertion is not quantifiable at the present time, except by computations and simulations for selected problems.

Now, adapting the procedure that led to (6.5.6) to the delay case yields the transition probabilities and interpolation intervals $\Delta t^{h,\delta}(\hat{\xi},\alpha)$ for the $\zeta_n^{h,\delta} = (\phi_n^{h,\delta}, \xi_n^{h,\delta})$ process in terms of those for the ξ_n^h process as:

$$
\begin{aligned}
p^{h,\delta}\left(\hat{\xi}, i\delta; \tilde{x}, i\delta \middle| \alpha\right) &= p^h\left(\hat{\xi}, \tilde{x}\middle|\alpha\right)\left(1 - p^{h,\delta}\left(\hat{\xi}, i\delta; \hat{\xi}(0), i\delta+\delta\middle|\alpha\right)\right) \\
p^{h,\delta}\left(\hat{\xi}, i\delta; \hat{\xi}(0), i\delta+\delta\middle|\alpha\right) &= \frac{\Delta t^h(\hat{\xi},\alpha)}{\Delta t^h(\hat{\xi},\alpha)+\delta},
\end{aligned}
\tag{3.1}
$$

$$
\Delta t^{h,\delta}(\hat{\xi},\alpha) = \frac{\delta \Delta t^h(\hat{\xi},\alpha)}{\Delta t^h(\hat{\xi},\alpha)+\delta}. \tag{3.2}
$$

Redefine

$$
\Delta t_n^{h,\delta} = \Delta t^{h,\delta}(\bar{\xi}_n^h, u_n^h), \quad t_n^{h,\delta} = \sum_{i=0}^{n-1} \Delta t_i^{h,\delta}. \tag{3.3}
$$

An alternative form of the implicit process. An alternative construction allows both the spatial and time variable to change simultaneously. Then the transition probabilities for the spatial component is just (1.11), (1.12), the conditional probability that time advances at step n is just $\Delta t^h(\hat{\xi},\alpha)/\delta$, and the interpolation interval is $\Delta t^h(\hat{\xi},\alpha)$. This procedure is equivalent to reindexing the process determined by (3.1) by omitting the indices at which the time variable advances. The corresponding spatial path is that of the explicit procedure. This variation will be useful in Chapter 8.

Local consistency and dynamical representations. Define $\Delta\xi_n^{h,\delta} = \xi_{n+1}^{h,\delta} - \xi_n^{h,\delta}$ and the martingale differences

$$
\beta_n^{h,\delta} = \left[\Delta\xi_n^{h,\delta} - E_n^{h,\delta}\Delta\xi_n^{h,\delta}\right] I_{\{\xi_n^{h,\delta}\in G_h\}},
$$

$$
\beta_{0,n}^{h,\delta} = (\phi_{n+1}^{h,\delta} - \phi_n^{h,\delta}) - E_n^{h,\delta}(\phi_{n+1}^{h,\delta} - \phi_n^{h,\delta}),
$$

where $E_n^{h,\delta}$ is the expectation conditioned on the data to step n. Let $E_{\hat{\xi},n}^{h,\delta,\alpha}$ denote the expectation conditioned on the data to step n with $u_n^{h,\delta} = \alpha$ and $\bar{\xi}_{r,n}^{h,\delta} = \hat{\xi}$. Then, for $\xi_n^{h,\delta} \in G_h$, the definitions (3.1), (3.2), and (1.8) yield the analog of (6.5.7):

$$
\begin{aligned}
&E_{\hat{\xi},n}^{h,\delta,\alpha}\Delta\xi_n^{h,\delta} = b_h(\hat{\xi},\alpha)\Delta t^{h,\delta}(\hat{\xi},\alpha) = b(\hat{\xi},\alpha)\Delta t^{h,\delta}(\hat{\xi},\alpha) + o(\Delta t^{h,\delta}(\hat{\xi},\alpha)), \\
&E_{\hat{\xi},n}^{h,\delta,\alpha}\beta_n^{h,\delta}[\beta_n^{h,\delta}]' = a_h(\hat{\xi})\Delta t^{h,\delta}(\hat{\xi},\alpha) = a(\hat{\xi})\Delta t^{h,\delta}(\hat{\xi},\alpha) + o(\Delta t^{h,\delta}(\hat{\xi},\alpha)), \\
&E_{\hat{\xi},n}^{h,\delta,\alpha}\left[\phi_{n+1}^{h,\delta} - \phi_n^{h,\delta}\right] = \Delta t_n^{h,\delta}.
\end{aligned}
\tag{3.4}
$$

The reflecting states are dealt with as in Section 1. The use of the process $\zeta_n^{h,\delta}$ leads to some intriguing possibilities for efficient representation of the memory data for the delay problem. Note that either the spatial variable $\xi_n^{h,\delta}$ changes or the time variable $\phi_n^{h,\delta}$ advances at each iteration, but not both. There are several choices for the timescale of the continuous-time interpolations. We will start by using the $\Delta t_n^{h,\delta}$ defined in (3.3) as the interpolation intervals, and construct $\xi^{h,\delta}(\cdot)$. Then we will define an interpolation with which it will be convenient to define the memory segment $\bar{\xi}_{r,n}^{h,\delta}$.

Let $\Delta z_n^{h,\delta}$ denote the reflection term at step n, with components $\Delta y_{i,n}^{h,\delta}$. Recall the definition of the time $d^{h,\delta}(\cdot)$ given in (6.5.23). Let $\xi^{h,\delta}(\cdot)$ and $\phi^{h,\delta}(\cdot)$ denote the continuous-time interpolations of the $\{\xi_n^{h,\delta}\}$ and $\{\phi_n^{h,\delta}\}$, resp., with the intervals $\{\Delta t_n^{h,\delta}\}$ when the path memory segments $\{\bar{\xi}_{r,n}^{h,\delta}\}$ are used. We always define $\phi_0^{h,\delta} = 0$. Then we can write

$$\xi_{n+1}^{h,\delta} = \xi_n^{h,\delta} + b_h(\bar{\xi}_{r,n}^{h,\delta}, u_n^{h,\delta})\Delta t_n^{h,\delta} + \beta_n^{h,\delta} + \Delta z_n^{h,\delta}, \tag{3.5}$$

$$\xi^{h,\delta}(t) = \xi_0^{h,\delta} + \sum_{i=0}^{d^{h,\delta}(t)-1} b_h(\bar{\xi}_{r,i}^{h,\delta}, u_i^{h,\delta})\Delta t_i^{h,\delta} + \sum_{i=0}^{d^{h,\delta}(t)-1} \beta_i^{h,\delta} + \sum_{i=0}^{d^{h,\delta}(t)-1} \Delta z_i^{h,\delta}, \tag{3.6}$$

$$\phi_{n+1}^{h,\delta} = \phi_n^{h,\delta} + \Delta t_n^{h,\delta} + \beta_{0,n}^{h,\delta}. \tag{3.7}$$

Interpolations using $\phi^{h,\delta}(\cdot)$ as the timescale. Definition of the memory segment $\bar{\xi}_{r,n}^{h,\delta}$. In analogy to the definition (1.7a), define $\bar{\xi}_n^{h,\delta} = \{\xi^{h,\delta}(t_n^{h,\delta} + \theta), \theta \in [-\bar{\theta}, 0]\}$. If $\bar{\xi}_{r,n}^{h,\delta}$ were simply a segment of the interpolated process $\xi^{h,\delta}(\cdot)$, say $\bar{\xi}_n^{h,\delta}$, then the issues concerning the number of required values of the memory variable that arose in Section 2 would arise here in the same way, and there would be no advantage in the use of the implicit approximation procedure. Consider the alternative where the time variables $\phi_n^{h,\delta}$ determine interpolated time, in that real (i.e., interpolated) time advances (by an amount δ) only when the time variable is incremented and it does not advance otherwise. This will be an analog of the "random" Approximation 4 defined by (4.2.7).

To make this precise, consider $\xi_n^{h,\delta}$ at only the times that $\phi_n^{h,\delta}$ changes. Suppose that $\bar{\theta}/\delta = Q_\delta$ is an integer. Recall the definition (6.5.13) where $v_0^{h,\delta} = 0$, and, for $n > 0$,

$$v_n^{h,\delta} = \inf\{i > v_{n-1}^{h,\delta} : \phi_i^{h,\delta} - \phi_{i-1}^{h,\delta} = \delta\}.$$

The path memory segment denoted by $\bar{\xi}_{r,n}^{h,\delta}$ is defined to be the function on $[\bar{\theta}, 0]$, with the following values: For any l and n satisfying $v_l^{h,\delta} \leq n < v_{l+1}^{h,\delta}$, set

$$\bar{\xi}^{h,\delta}_{r,n}(0) = \xi^{h,\delta}_n,$$
$$\bar{\xi}^{h,\delta}_{r,n}(\theta) = \begin{cases} \xi^{h,\delta}_{v^{h,\delta}_l}, & \theta \in [-\delta, 0), \\ \vdots & \\ \xi^{h,\delta}_{v^{h,\delta}_{l-Q_\delta+1}}, & \theta \in [-\bar{\theta}, -\bar{\theta}+\delta). \end{cases} \tag{3.8}$$

Figure 3.1 illustrates the construction of $\xi^{h,\delta}(\cdot)$ and $\bar{\xi}^{h,\delta}_{r,n}(\cdot)$ for $\bar{\theta}/\delta = 3$ and $v^{h,\delta}_l \le n < v^{h,\delta}_{l+1}$, and where we define $\sigma^{h,\delta}_l = t^{h,\delta}_{v^{h,\delta}_l}$.

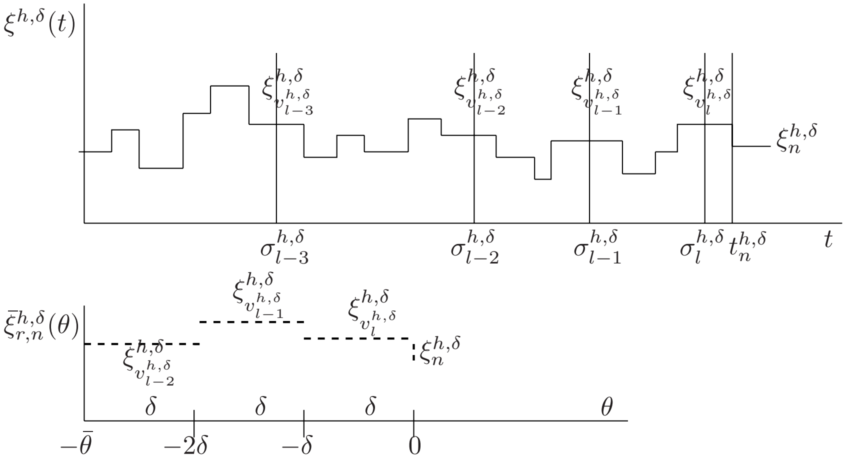

Figure 3.1. Illustration of $\bar{\xi}^{h,\delta}_{r,n}(\theta)$, for $v^{h,\delta}_l \le n < v^{h,\delta}_{l+1}$, $\bar{\theta}/\delta = 3$.

Recall that, in this section, $\hat{\xi}$ denotes the canonical value of the memory $\bar{\xi}^{h,\delta}_{r,n}$. It can be represented as the piecewise-constant right-continuous interpolation with interval δ of its values

$$\big(\hat{\xi}(-\bar{\theta}), \ldots, \hat{\xi}(-\delta), \hat{\xi}(0)\big)$$

with a discontinuity at $\theta = 0$, and we can unambiguously call the above set $\hat{\xi}$.

The possible transitions are as follows. If the time variable advances at the current step, then we have the shift

$$\big(\hat{\xi}(-\bar{\theta}), \ldots, \hat{\xi}(-\delta), \hat{\xi}(0)\big) \to \big(\hat{\xi}(-\bar{\theta}+\delta), \ldots, \hat{\xi}(-\delta), \hat{\xi}(0), \hat{\xi}(0)\big). \tag{3.9a}$$

This implies that $\hat{\xi}(0) \in G_h$, as otherwise there must be a reflection at the current step and the time variable could not advance. Let $\hat{\xi}(0) \in G_h$ and suppose that the time variable does not advance. Then

$$\hat{\xi} = \big(\hat{\xi}(-\bar{\theta}), \ldots, \hat{\xi}(-\delta), \hat{\xi}(0)\big) \to \big(\hat{\xi}(-\bar{\theta}), \ldots, \hat{\xi}(-\delta), \xi_1\big), \tag{3.9b}$$

where, conditioned on the time variable not advancing and the use of control value α, the probability that $\xi_1 = \tilde{x}$ is $p^h(\hat{\xi}, \tilde{x}|\alpha)$. Suppose that $\hat{\xi}(0) \notin G_h$, so that it is a reflecting point. Then

$$\hat{\xi} = \big(\hat{\xi}(-\bar{\theta}), \ldots, \hat{\xi}(-\delta), \hat{\xi}(0)\big) \rightarrow \big(\hat{\xi}(-\bar{\theta}), \ldots, \hat{\xi}(-\delta), \xi_1\big), \tag{3.9c}$$

where $\xi_1 \in G_h$ is the state that the reflecting state $\hat{\xi}(0)$ moves to, with the transition probabilities satisfying (6.2.2).

Size of the state space. For the one-dimensional problem discussed at the end of Section 2, the maximum size of the state space that is required with the use of $\bar{\xi}^{h,\delta}_{r,n}$ for the path memory segment is

$$(B/h+1)^{\bar{\theta}/\delta}(B/h+3) \tag{3.10}$$

compared with $(B/h+1)^{K^h}(B/h+3)$ there, where commonly $K^h = O(1/h^2)$. This saving is partly due to the fact that, for the implicit approximation procedure, the memory consists of the samples at iterates separated by many steps, and not the set of values or differences in the values for each of those individual steps. One could approximate the values of the $\hat{\xi}(-i\delta), i > 0$, further by discretizing to a coarser set of values. Further reductions in the size of the state space will be dealt with in the next chapter, where we discuss the advantages of using differences of the values in lieu of the values themselves, and also develop alternative constructions that are motivated by the implicit approximation procedure and are likely to be advantageous.

Note on the interpolation interval. An important additional point to note is that the implicit approximation procedure does not require the use of a constant interpolation time interval. It allows us to use the original time intervals $\Delta t^{h,\delta}_n \approx \Delta t^h_n$, and not the minimal value $\overline{\Delta}^h$. This is computationally advantageous when the values $\Delta t^h(\hat{\xi}, \alpha)$ vary a great deal, as for example when the upper bound on the control is large or when $a(\cdot)$ is not constant. In the example of Section 6.4, $\Delta t^h(x) = h^2/\sigma^2(x)$, and if $\sigma^2(\cdot)$ varies a great deal, the transformation to a constant interval might entail a considerable increase in the dimension of the memory segment $\bar{\xi}^h_n$ that was used in Section 2. The implicit approximation procedure does not have this disadvantage.

The effective maximum delay. The approximation procedure that we have just illustrated has replaced the true maximum delay by a random delay. The actual effective maximum delay for the example in the figure is $t^{h,\delta}_n - \sigma^{h,\delta}_{l-2}$. In general, for $\sigma^{h,\delta}_l \le t^{h,\delta}_n < \sigma^{h,\delta}_{l+1}$, the maximum delay is $t^{h,\delta}_n - \sigma^{h,\delta}_{l-Q_\delta+1}$. As $\delta \rightarrow 0$, the delays converge to their values for the original model (1.5). Let δ be fixed. It is shown in Theorem 4.1 that, as $h \rightarrow 0$, the interpolated times between increases in the time variable $\phi^{h,\delta}(\cdot)$ are exponentially distributed with mean δ. The interval between a random time and the most recent time before it that $\phi^{h,\delta}(\cdot)$ increased is also (asymptotically) exponentially distributed with mean δ, and these intervals are asymptotically mutually independent. Thus, as $h \to 0$, the maximum delay is the sum of Q_δ exponentially distributed and mutually independent random variables, each with mean δ. Hence it has an Erlang distribution of order Q_δ, and with total mean $\bar{\theta}$.

7.3.2 The Cost Function and Bellman Equation

With the use of the process $\zeta_n^{h,\delta}$, with $\bar{\xi}_{r,0}^{h,\delta} = \hat{\xi}$ and the control sequence $u^{h,\delta} = \{u_n^{h,\delta}, n < \infty\}$ used, an approximation to the discounted cost function (3.4.3) is

$$W^{h,\delta}(\hat{\xi}, u^{h,\delta}) = E_{\hat{\xi}}^{h,\delta,u^{h,\delta}} \sum_{n=0}^{\infty} e^{-\beta\phi_n^{h,\delta}} \left[k(\bar{\xi}_{r,n}^{h,\delta}, u_n^{h,\delta}) \delta I_{\{\phi_{n+1}^{h,\delta} \neq \phi_n^{h,\delta}\}} + q' \Delta y_n^{h,\delta} \right]. \tag{3.11}$$

By using the last line of (3.4) and taking a conditional expectation, the term $\delta I_{\{\phi_{n+1}^{h,\delta} \neq \phi_n^{h,\delta}\}}$ can be replaced by $\Delta t_n^{h,\delta}$. It will be shown in Theorem 3.2 that (3.11) is well defined and is asymptotically equal to

$$E_{\hat{\xi}}^{h,\delta,u^{h,\delta}} \sum_{n=0}^{\infty} e^{-\beta t_n^{h,\delta}} \left[k(\bar{\xi}_{r,n}^{h,\delta}, u_n^{h,\delta}) \Delta t_n^{h,\delta} + q' \Delta y_n^{h,\delta} \right]. \tag{3.12}$$

With the form (3.12), the effective canonical cost rate when the memory segment is $\hat{\xi}$ and control value α is used is just $k(\hat{\xi}, \alpha)$ times δ times the probability that the time variable advances, and the product is $k(\hat{\xi}, \alpha)\Delta t^{h,\delta}(\hat{\xi}, \alpha)$.

The Bellman equation. The Bellman equation can be based on either (3.11) or (3.12). They might yield different results but will be asymptotically equal by Theorem 3.2. For (3.11) and $\hat{\xi}(0) = \xi_0^{h,\delta} \in G_h$, the Bellman equation is[8]

$$\begin{aligned} V^{h,\delta}(\hat{\xi}) = \inf_{\alpha \in U^h} \Bigg[& \sum_{\tilde{x}} p^{h,\delta}\left(\hat{\xi}, \phi; \tilde{x}, \phi \middle| \alpha\right) V^{h,\delta}\left(\hat{\xi}(-\bar{\theta}), \ldots, \hat{\xi}(-\delta), \tilde{x}\right) \\ & + e^{-\beta\delta} p^{h,\delta}\left(\hat{\xi}, \phi; \hat{\xi}(0), \phi + \delta\right) V^{h,\delta}\left(\hat{\xi}(-\bar{\theta}+\delta), \ldots, \hat{\xi}(-\delta), \hat{\xi}(0), \hat{\xi}(0)\right) \\ & + k(\hat{\xi}, \alpha)\Delta t^{h,\delta}(\hat{\xi}, \alpha) \Bigg], \end{aligned} \tag{3.13}$$

where $V^{h,\delta}(\hat{\xi})$ is the optimal value. The analog for (3.12) can be written as (using a more succinct notation)

[8] The time variable ϕ does not appear in the state as the dynamical terms are time-independent.

$$V^{h,\delta}(\hat{\xi}) = \inf_{\alpha \in U^h} E_{\hat{\xi}}^{h,\delta,\alpha} \left[e^{-\beta \Delta t^{h,\delta}(\hat{\xi},\alpha)} V^{h,\delta}\left(\bar{\xi}_{r,1}^{h,\delta}\right) + k(\hat{\xi},\alpha)\Delta t^{h,\delta}(\hat{\xi},\alpha) \right], \tag{3.14}$$

where $\bar{\xi}_{r,1}^{h,\delta}$ is the successor memory segment to $\hat{\xi}$ under control value α. If $\hat{\xi}(0) \notin G_h$, then for either (3.11) or (3.12)

$$V^{h,\delta}(\hat{\xi}) = E_{\hat{\xi}}^{h,\delta,\alpha} \left[V^{h,\delta}(\bar{\xi}_{r,1}^{h,\delta}) + q' \Delta y_0^{h,\delta} \right], \tag{3.15}$$

where $\Delta y_0^{h,\delta}$ is the vector of the components of the reflection term from state $\hat{\xi}(0)$. These equations make it clear that the full state at iterate n is $\bar{\xi}_{r,n}^{h,\delta}$, namely, the current values of the spatial variable $\xi_n^{h,\delta}$, together with its value at the last $Q_\delta = \bar{\theta}/\delta$ times that the time variable advances.

7.3.3 The Use of Averaging in Constructing the Path Memory Approximation

One might be tempted to use an average of the path values over the intervals in lieu of the samples $\xi_{v_i^{h,\delta}}^{h,\delta}$ in (3.8). This can be done, but it entails a considerable increase in the memory requirements. One possibility is as follows. Let $v_l^{h,\delta} \le n < v_{l+1}^{h,\delta}$. Define

$$\xi_{av,l,n}^{h,\delta} = \frac{\sum_{i=v_l^{h,\delta}+1}^{n} \xi_i^{h,\delta} \Delta t_i^{h,\delta}}{\sum_{i=v_l^{h,\delta}+1}^{n} \Delta t_i^{h,\delta}}, \quad \xi_{av,l}^{h,\delta} = \frac{\sum_{i=v_l^{h,\delta}+1}^{v_{l+1}^{h,\delta}-1} \xi_i^{h,\delta} \Delta t_i^{h,\delta}}{\sum_{i=v_l^{h,\delta}+1}^{v_{l+1}^{h,\delta}-1} \Delta t_i^{h,\delta}}.$$

Then replace the $\xi_{v_l^{h,\delta}}^{h,\delta}$ in (3.8) by $\xi_{av,l,n}^{h,\delta}$, the path average over the interval $[v_l^{h,\delta}+1, n]$. Replace $\xi_{v_{l-i}^{h,\delta}}^{h,\delta}$ by $\xi_{av,l-i}^{h,\delta}$. The ratio can be computed recursively on each interval: At the beginning of the lth cycle, set $\xi_{av,l,v_l^{h,\delta}+1}^{h,\delta} = \xi_{v_l^{h,\delta}+1}^{h,\delta}$, and for $n > v_l^{h,\delta}+1$, use

$$\xi_{av,l,n+1}^{h,\delta} = \frac{\xi_{n+1}^{h,\delta} \Delta t_{n+1}^{h,\delta}}{\sum_{i=v_l^{h,\delta}+1}^{n+1} \Delta t_i^{h,\delta}} + \frac{\xi_{av,l,n}^{h,\delta}}{1 + \Delta t_{n+1}^{h,\delta} / \sum_{i=v_l^{h,\delta}+1}^{n} \Delta t_i^{h,\delta}}.$$

The computation is simpler if the interpolation interval is constant. In general, one needs to keep track of the running sums of the weighted path variables and the accumulated time, which introduces two new variables, one whose dimension is that of G. The set of such values will have to be discretized. For example, discretize the possible values and update the approximations by randomization if the new values fall between the allowable discrete points. The randomization method could be analogous to what is to be done in Section 8.4 for the periodic-Erlang approximation for the control variables. This will, in any case, yield a value that is a close to a convex combination of a subset of values within the interval, so it might be worth considering. Similar considerations apply to the approximations that are used for the path in Chapter 8, but the issue will not be pursued further.

A simpler procedure is to use a linear interpolation of the values in (3.8), which would not entail any increase in the required memory.

7.3.4 Timescales

The Interpolation $\psi^{h,\delta}(\cdot)$ and its timescale. The discrete-parameter process $\{\xi_n^{h,\delta}\}$ with memory segments $\{\bar{\xi}_{r,n}^{h,\delta}\}$ (or the variations discussed in the next chapter) are used for the numerical computations. The proofs of convergence in Section 8.5 will be based on a continuous-time process $\psi^{h,\delta}(\cdot)$ that is analogous to those defined by (1.14), (6.3.10), and (6.5.12), analogously to what was done in [58, Chapters 10 and 11]. Next, recalling the method of defining (1.14), let us define the interpolation $\psi_n^{h,\delta}(\cdot)$. Let $\nu_n, n < \infty$, be mutually independent and identically and exponentially distributed with unit mean (as above (6.3.4)), and independent of $\{\zeta_n^{h,\delta}, u_n^h\}$. Then set $\Delta\tau_n^{h,\delta} = \nu_n \Delta t_n^{h,\delta}$ and $\tau_n^{h,\delta} = \sum_{i=0}^{n-1} \Delta\tau_n^{h,\delta}$. Recall the definition of $d_\tau^{h,\delta}(s)$ from (6.5.23) and let $r_\tau^{h,\delta}(\cdot)$ denote the relaxed control representation of the interpolation (intervals $\Delta\tau_n^{h,\delta}$) of the control process. Analogously to what was done in getting (1.14), define the interpolation $\bar{\xi}_r^{h,\delta}(\cdot)$ (with intervals $\{\Delta t_n^{h,\delta}\}$) of the memory segment by $\bar{\xi}_r^{h,\delta}(s) = \bar{\xi}_{r,n}^{h,\delta}$, for $t_n^{h,\delta} \le s < t_{n+1}^{h,\delta}$, and set $q_\tau^{h,\delta}(s) = t_{d_\tau^{h,\delta}(s)}^{h,\delta}$. With these definitions, $\bar{\xi}_{r,d_\tau^{h,\delta}(s)}^{h,\delta} = \bar{\xi}_r^{h,\delta}(q_\tau^{h,\delta}(s))$. Let $\psi^{h,\delta}(\cdot)$ denote the interpolation of the sequence $\xi_n^{h,\delta}$ using the random intervals $\Delta\tau_n^{h,\delta}$. Then, analogously to (1.14),

$$\psi^{h,\delta}(t) = \xi_0^{h,\delta} + \int_0^t \int_{U^h} b_h(\bar{\xi}_{r,d_\tau^{h,\delta}(s)}^{h,\delta}, \alpha) r_\tau^{h,\delta}(d\alpha\, ds) + B_\tau^{h,\delta}(t) + z_\tau^{h,\delta}(t), \quad (3.16)$$

where the drift term can be written as

$$\int_0^t \int_{U^h} b_h(\bar{\xi}_r^{h,\delta}(q_\tau^{h,\delta}(s)), \alpha) r_\tau^{h,\delta}(d\alpha\, ds),$$

and the quadratic variation of the martingale $B_\tau^{h,\delta}(\cdot)$ is

$$\int_0^t a_h(\bar{\xi}_r^{h,\delta}(q_\tau^{h,\delta}(s)))ds.$$

Asymptotic equivalence of the timescales. It follows from the proof of Theorem 6.5.1 that the timescales used in the $\xi^{h,\delta}(\cdot)$ and the $\psi^{h,\delta}(\cdot)$ processes coincide asymptotically. That is, $q^{h,\delta}(s) - s \to 0$, $\phi^{h,\delta}(s) - s \to 0$ and $q_\tau^{h,\delta}(s) -$

$s \to 0$. The following theorem reasserts this result in the context of the current chapter.

Theorem 3.1. *Assume local consistency,* (A3.1.1), (A3.1.2), (A3.2.1), (A3.2.2) *and* (A3.4.3), *with system* (1.5) *and memory segment* (3.8). *Let* $\phi^{h,\delta}(\cdot)$ *denote the interpolation of the* $\phi_n^{h,\delta}$ *with the intervals* $\Delta t_n^{h,\delta}$, *and suppose that* h/δ *is bounded as* $h \to 0$ *and* $\delta \to 0$. *Then Theorem* 6.5.1 *holds and for each* $T < \infty$,

$$\lim_{h,\delta\to 0} \sup_{\hat{\xi},u^{h,\delta}} E_{\hat{\xi}}^{h,\delta,u^{h,\delta}} \sup_{-\bar{\theta}\le\theta\le 0, t\le T} \left|\psi^{h,\delta}(t+\theta) - \bar{\xi}^{h,\delta}_{d_\tau^{h,\delta}(t)}(\theta)\right| = 0. \tag{3.17}$$

If the memory segments $\bar{\xi}_n^h$ *are used, as in Section* 2, *then the index* δ *is redundant and we have*

$$\lim_{h\to 0} \sup_{\hat{\xi},u^h} E_{\hat{\xi}}^{h,u^h} \sup_{-\bar{\theta}\le\theta\le 0, t\le T} \left|\psi^h(t+\theta) - \bar{\xi}^h_{d_\tau^h(t)}(\theta)\right| = 0. \tag{3.18}$$

An alternative construction of the implicit procedure. Time and spatial variables changing simultaneously. Recall the comments on the alternative construction of an implicit procedure below (3.3), where we allowed the possibility that both the path and time variables change simultaneously. With the memory segment taking any of the forms that were discussed, the resulting processes and costs are asymptotically equivalent to those for the implicit procedure.

7.3.5 Convergence Theorems

The next theorem asserts that the cost functions (3.11) and (3.12) are well defined and asymptotically equal.

Theorem 3.2. *Assume local consistency,* (A3.1.1), (A3.1.2), (A3.2.1), (A3.2.2), *and* (A3.4.3), *and the model* (1.5) *with memory segment* (3.8). *Then* (3.11) *is asymptotically equal to* (3.12) *uniformly in the control and in the initial condition* $\hat{\xi}$, *where the function* $\hat{\xi}$ *is piecewise-constant, with intervals* δ *and with values in* G_h.

Proof. To show that the sum involving $k(\cdot)$ in (3.11) is well defined, first note that it can be bounded by a constant times the expectation of $\int_0^\infty e^{-\beta\phi^{h,\delta}(s)}ds$. By Theorem 3.1 or Theorem 6.5.1, for each $K > 0$ there is an $\epsilon_1 > 0$, which does not depend on the controls, initial conditions, or T, such that for small enough h, δ,

$$P\left\{\phi^{h,\delta}(T+K) - \phi^{h,\delta}(T) \ge \epsilon_1 \middle| \text{data to } T\right\} > \epsilon_1, \quad \text{w.p.1.}$$

Hence, for each $K > 0$ there is $\epsilon_2 > 0$, not depending on the controls, initial conditions, or T, such that for small enough h, δ,

$$E\left[e^{-\beta(\phi^{h,\delta}(T+K)-\phi^{h,\delta}(T))}\middle|\text{data to } T\right\} \leq e^{-\epsilon_2} \quad \text{w.p.1.}$$

This implies that the "tail" of the sum (3.11) can be neglected and we need only consider the sum $\sum_{i=0}^{N^{h,\delta}(t)}$ where $N^{h,\delta}(t) = \min\{n : t_n^{h,\delta} \geq t\}$ for arbitrary t. But, by Theorem 3.1 or Theorem 6.5.1, for such a sum the asymptotic values are the same if $\phi_i^{h,\delta}$ is replaced by $t_i^{h,\delta}$ for $i \leq N^{h,\delta}(t)$. Hence the terms involving $k(\cdot)$ in (3.11) and (3.12) are asymptotically equal. The above estimates and Lemma 6.3.1 yield the same result for the terms involving $\Delta y_n^{h,\delta}$.
■

The convergence theorem. As in Theorem 1.1, approximate the initial condition $\bar{x}(0)$ by $\bar{\xi}_0^{h,\delta}$ (in the sense of uniform convergence as $h \to 0, \delta \to 0$), and let it be constant on the Q_δ intervals $[-\bar{\theta}, -\bar{\theta}+\delta), \ldots, [-\delta, 0)$, with all values being in G_h. Because by Theorem 3.2 we can use (3.12) for the cost function when proving convergence, the proof of the next theorem is nearly identical to that of Theorem 1.1, which is to be given in Section 8.5.

Theorem 3.3. *Assume local consistency,* (A3.1.1), (A3.1.2), (A3.2.1)–(A3.2.3), *and* (A3.4.3), *with system* (1.5) *and cost function* (3.4.3). *The memory segment for the numerical approximation is* (3.8). *Let* $p^{h,\delta}(\cdot)$ *be derived via* (3.1)–(3.3) *from the transition probabilities* $p^h(\cdot)$ *that are locally consistent* (*in the sense of* (1.8)). *Let* $\bar{\xi}_0^{h,\delta}$ *approximate the continuous initial condition* $\hat{x}$ *as in Theorem* 1.1. *Let* h/δ *be bounded. With either* (3.11) *or* (3.12) *used,* $V^{h,\delta}(\bar{\xi}_0^{h,\delta}) \to V(\hat{x})$ *as* $h \to 0, \delta \to 0$. *The analogous result holds for the analog of the cost functional* (1.28) *if* (A3.4.1) *and* (A3.4.2) *are assumed and the conditions on the reflection directions are dropped.*

7.4 The Implicit Approximation Procedure and the Random Delay Model

The intervals between time advances, δ fixed. Consider the implicit approximation procedure of Section 3, with the value of δ fixed and only $h \to 0$. The following theorem shows that the sequence of times between shifts in the time variable converges to a sequence of i.i.d. random variables, each of which is exponentially distributed with mean δ. Define $\hat{\sigma}_l^{h,\delta} = \tau_{v_l^{h,\delta}}^{h,\delta}$, where $v_l^{h,\delta}$ was defined above (3.8). Recall the definition of $\sigma_l^{h,\delta} = t_{v_l^{h,\delta}}^{h,\delta}$ below (3.8).

In the theorem, we ignore the time-shift steps in the indexing. This does not change the distribution of the quantities of interest. The resulting path is that for the explicit procedure if the same controls are used.

Theorem 4.1. *Assume the model of Section* 3 *and that* $\Delta t^h(\hat{\xi}, \alpha) = O(h^2)$, *with the assumptions of Theorem* 3.3, *but with* δ *fixed. As* $h \to 0$, $\phi^{h,\delta}(\cdot)$ *converges to a Poisson process with rate* $1/\delta$ *and jump size* δ, *and this process on* $[t, \infty)$ *is independent of the other weak-sense limits on* $[0, t]$. *The conditional distribution of* $\sigma^{h,\delta}_{l+1} - \sigma^{h,\delta}_l$, *given the data to time* $\sigma^{h,\delta}_l$, *converges to an exponentially distributed random variable with mean* δ, *and the conditional mean value converges to* δ, *all uniformly in the data and* l. *Now let* $\delta = O(h)$ *and replace* $\sigma^{h,\delta}_{l+1} - \sigma^{h,\delta}_l$ *by* $[\sigma^{h,\delta}_{l+1} - \sigma^{h,\delta}_l]/\delta$. *Then the results of the previous sentence hold, but with mean unity. The analogous results hold if the* $\hat{\sigma}^{h,\delta}_l$ *are used in lieu of the* $\sigma^{h,\delta}_l$.

Proof. Fix $\delta > 0$. Let $R(\cdot)$ be a Poisson process with rate $1/\delta$ and jump size δ. Approximate $\phi^{h,\delta}(\cdot)$ as follows. For each n, if $R(\cdot)$ has multiple jumps on $[t^{h,\delta}_n, t^{h,\delta}_{n+1})$, then ignore any jump beyond the first, and assign the remaining jump (if any) to time $t^{h,\delta}_n$. The difference between this process and both $\phi^{h,\delta}(\cdot)$ and $R(\cdot)$ converges weakly to zero as $h \to 0$. This yields the first two assertions of the theorem. The assertion concerning the convergence of the conditional mean follows from this and the uniform integrability of $\{\sigma^{h,\delta}_{l+1} - \sigma^{h,\delta}_l; h, l, \delta\}$. If $\delta = O(h)$, then the result for the $[\sigma^{h,\delta}_{l+1} - \sigma^{h,\delta}_l]/\delta$ follows by a rescaling of time and amplitude. A similar argument is used if the $\hat{\sigma}^{h,\delta}_l$ are used in lieu of the $\sigma^{h,\delta}_l$. ■

Convergence to the random delay model if δ is fixed. If $\delta > 0$ is fixed and only $h \to 0$, then the limit is the optimal value for the Approximation 4 of Section 4.2. This assertion follows from Theorem 4.1 and the proof of Theorem 3.3 (see Section 8.5) and is stated in the following theorem.

Theorem 4.2. *Let the initial condition for* (1.1) *be* $\bar{x}(0)$, *assumed to be continuous and* G*-valued. Let the* G*-valued piecewise constant* $\bar{x}^\delta(0)$ (*intervals* δ) *converge to* $\bar{x}(0)$ *uniformly on* $[-\bar{\theta}, 0]$. *Assume the conditions of Theorem* 3.3, *but with the memory segment defined by* (4.2.7), *the random case. Hence the model is* (4.1.9b), *with* $\bar{x}_a = \bar{x}^\delta_r$, *for which we suppose that there is a weak-sense unique solution for each control and initial condition* $\bar{x}^\delta(0)$. *Let* $V^\delta(\hat{\xi})$ *denote the optimal cost for this model. Let* $\bar{\xi}^{h,\delta}_0$ *approximate* $\bar{x}^\delta(0)$ (*in the sense of uniform convergence as* $h \to 0$), *with values in* G_h, *and use* (3.1) *and* (3.2) *for the transition probabilities and interpolation intervals. Then, with* $\delta > 0$ *fixed,* $V^{h,\delta}(\bar{\xi}^{h,\delta}_0) \to V^\delta(\bar{x}^\delta(0))$. *As* $\delta \to 0$, $V^\delta(\bar{x}^\delta(0)) \to V(\bar{x}(0))$. *The same results hold for the analog of the cost function* (1.28) *for the implicit procedure.*

Consider the analog of Theorem 1.4 *for the implicit procedure. Then, under the analogs of its conditions for the implicit method and the path memory segment* $\bar{\xi}^{h,\delta}_{r,n}$ *used in lieu of* $\bar{\xi}^{h}_{n}$*, its conclusions hold.*

Comment. The implicit approximation algorithm illustrates one way of reducing the memory requirement over that needed for the procedure of Sections 1 or 2. In addition, one does not need the interval $\Delta t^h(\hat{\xi}, \alpha)$ to be constant, which is a considerable advantage when $\sigma(\cdot)$ is either small or is not a constant. The motivation for the implicit approximation procedure was the desire for a simpler representation of the path memory segment for the approximating process. However, the randomness of the effective delays with this procedure might be too large unless δ is small. Approximations that aim at compromises between the explicit procedure of Section 1 and the implicit approximation procedure will be discussed in the next chapter. Reliable numerical comparisons are still lacking, however.