Nonlinear state dynamics: computational methods and manufacturing application*

J. J. WESTMAN† and F. B. HANSON‡

Abstract: Stochastic optimal control problems are considered that are nonlinear in the state dynamics, but otherwise are an LQGP problem in the control, i.e. the dynamics are linear in the control vector and the costs are quadratic in the control. In addition the system is randomly perturbed by both continuous Gaussian (G) and discontinuous Poisson (P) noise. The approach to the solution is by way of computational stochastic dynamic programming using a new enhancement with a least squares equivalent LQGP problem in the state to accelerate the iterative convergence, without adding to the state space computational complexity since the LQGP coefficient equations are independent of the state. General Gauss statistics quadratures are developed to numerically handle Poisson jump integrals. The methods are illustrated for a multistage manufacturing system (MMS) with sufficient realism in an uncertain environment, together with implementation procedures needed to modify the formal general theory.

1. Introduction

The Linear–Quadratic–Gaussian–Poisson in control only (LQGP/U) problem has nonlinear state dynamics, linear control dynamics, and both Gaussian and Poisson random perturbations, which is governed by the stochastic differential equation (SDE)

$$d\mathbf{X}(t) = [\mathbf{F}_0(\mathbf{X}(t),t) + F_1(\mathbf{X}(t),t)\mathbf{U}(t)]dt + G_0(\mathbf{X}(t),t)d\mathbf{W}(t) + H_0(\mathbf{X}(t),t)d\mathbf{P}(t), \qquad (1)$$

representing a rather general Markov process in continuous time. Here, $\mathbf{X}(t)$ is the $m \times 1$ state vector in the state space $\mathcal{D}_x$, $\mathbf{U}(t)$ is the $n \times 1$ control vector in the control space $\mathcal{D}_u$, $d\mathbf{W}(t)$ is the $r \times 1$ Gaussian noise vector, and $d\mathbf{P}(t)$ is the $q \times 1$ space–time Poisson noise vector. The dynamic (subscript '0' denotes control–independence and subscript '1' denotes linear in control) coefficients are as follows: $\mathbf{F}_0(\mathbf{x},t)$ is the $m \times 1$ control–independent part of the drift, $F_1(\mathbf{x},t)$ is the $m \times n$ coefficient of the linear control term in the drift, $G_0(\mathbf{x},t)$ is the $m \times r$ amplitude of the Gaussian noise, $H_0(\mathbf{x},t)$ is the $m \times q$ jump amplitude of the Poisson noise. Coefficients of both the Gaussian and Poisson noise could include terms linear in the control, but these terms are omitted here for simplicity to avoid extra complications with the control in the noise term and for brevity. However, the Poisson noise term coefficient that is linear in the control has been treated by Westman and Hanson (1997). All coefficients in Eq. 1 are assumed to be continuously differentiable in their arguments. The state dependence

*Conditionally accepted for publication in the *International Journal of Control*, 1999

†Department of Mathematics, University of California, Box 951555, Los Angeles, CA 90095–1555, USA; email: jwestman@math.ucla.edu ; tel: +310–206–8094

‡Laboratory for Advanced Computing, Department of Mathematics, Statistics, and Computer Science; University of Illinois at Chicago, 851 Morgan St., M/C 249; Chicago, IL 60607–7045, USA; email: hanson@math.uic.edu; tel: +312–413–2142

is assumed to be generally nonlinear. The SDE (1) is interpreted in the sense of Itô (see Gihman and Skorohod (1972) or see Hanson 1996 for applicable exposition).

The LQGP/U problem advances the model of the LQGP problem (Wonham 1970, Westman and Hanson 1997) which is an extension of the LQG analysis (Bryson and Ho (1975), Lewis 1986, Dorato et al. 1995). In contrast to the LQGP problem, the LQGP/U problem does not exhibit a formal closed form solution in terms of a state–independent, nonlinear system of ordinary differential equations. Numerical methods for partial differential equations are necessary in order to solve the stochastic dynamic programming for the LQGP/U problem. Hence, the computational requirements for the LQGP/U problem, with a search over all the discretized state space, are much greater than that for LQGP problem.

The LQGP/U problem is a generalization to the *LQGP in control only* model used by Hanson (1991, 1996), and Naimipour and Hanson (1993), since here the Poisson processes are distributed over jump amplitudes. The model here is thus more robust and realistic over a wider range of stochastic control models. Hanson (1991, 1996) developed many high performance techniques to handle the large computational demands of the PDE of stochastic dynamic programming for Markov problems in continuous time. The Poisson terms of (1) introduce a discrete jump in the state, superimposed on the continuous, yet nonsmooth Gaussian noise contribution, currently called a *hybrid system*. The continuous Gaussian noise is useful for modelling random background fluctuations, while the Poisson terms are useful for modelling large or disastrous random fluctuations such as machine failure (see Westman and Hanson 1997). Poisson SDEs have also been used to model both natural disasters and bonanzas in population growth (see Hanson and Tuckwell 1997), as well as optimal control with both resource and price fluctuations (see Hanson and Ryan 1998). The Poisson process plays a central role in performance models of queuing systems (see Kleinrock 1975). All of the analysis here should go through for the more general martingales.

The quadratic or Q part of the LQGP/U refers to a quadratic performance criterion with respect to control costs only. The performance index is the cost measure that the plant manager seeks to optimize, such as the operation cost of machines in a manufacturing plant, or the cancer palliation manager seeking to reduce patients' tumor burdens and enhance the patients' overall well–being. The quadratic performance index considered here is in the time–to–go or cost–to–go form:

$$V(\mathbf{X}, \mathbf{U}, t) = \frac{1}{2}\mathbf{X}^\top(t_f) S_f \mathbf{X}(t_f) + \int_t^{t_f} C(\mathbf{X}(\tau), \mathbf{U}(\tau), \tau) d\tau, \tag{2}$$

where the time horizon is $(t, t_f]$, the instantaneous cost function is a quadratic function of control is

$$C(\mathbf{x}, \mathbf{u}, t) = C_0(\mathbf{x}, t) + \mathbf{C}_1^\top(\mathbf{x}, t)\mathbf{u} + \frac{1}{2}\mathbf{u}^\top C_2(\mathbf{x}, t)\mathbf{u}, \tag{3}$$

and S_f is the salvage or final cost matrix. This salvage cost is given by the quadratic form, $\mathbf{x}^\top S_f \mathbf{x}$. For minimization, the quadratic cost coefficient C_2 is assumed to be a symmetric, positive definite $n \times n$ array, while S_f is assumed to be symmetric, positive semi–definite $m \times m$ array. The linear cost coefficient $\mathbf{C}_1$ is an $n \times 1$ vector and C_0 is a scalar. All cost coefficients in Eq. 3 along with the salvage cost are assumed to be continuously differentiable in their arguments. The quadratic performance index (2)–(3) was selected to be the most general form for the LQGP/U problem, such that Eqs. (1)–(3) comprise the stochastic optimal control problem. Note that time discounting can be included in the quadratic cost model Eq. (3) since $C(\mathbf{x},t)$ has time dependence. For example, the performance can include the comparison to potentially better opportunity costs through investments, upon introducing a discount factor. Also, linear, singular control costs can be properly approximated in the *cheap control* limit in which $||C_2|| \longrightarrow 0^+$ in some norm. The optimal, expected performance or value, $v(\mathbf{x},t)$, is defined as

$$v(\mathbf{x},t) \equiv \underset{\mathbf{u}(t,t_f]}{\mathrm{Min}} \left[\underset{\{\mathbf{P},\mathbf{W}\}(t,t_f]}{\mathrm{E}} [V(\mathbf{X},\mathbf{U},t) \mid \mathbf{X}(t)=\mathbf{x}, \mathbf{U}(t)=\mathbf{u}] \right]. \tag{4}$$

The restrictions on the state and control are that they belong to the admissible classes for the state and control, respectively. The final condition on the optimal, expected value is determined by the salvage costs:

$$v(\mathbf{x},t_f) = \frac{1}{2}\mathbf{x}^\top S_f \mathbf{x} \quad \text{for} \quad \mathbf{x} \in \mathcal{D}_x. \tag{5}$$

In this section, some properties of the Poisson and Gaussian stochastic noise are derived, because they are useful for deciding the kinds of noise that would be appropriate for a continuous time application and their representations. These include the conditional infinitesimal expectations and covariances, (see Feller 1966:345), but also the jumps in the state due to the Poisson noise. The Gaussian white noise term, $d\mathbf{W}(t)$, consists of r independent, standard Wiener processes $dW_i(t)$, for $i=1$ to r. It is assumed that the Gaussian components have zero infinitesimal expectation and diagonal covariance,

$$\mathrm{E}[d\mathbf{W}(t)] = \mathbf{0}_{r\times 1} \qquad \text{and} \qquad \mathrm{Cov}[d\mathbf{W}(t), d\mathbf{W}^\top(t)] = I_r dt. \tag{6}$$

For cross–correlated noise, see Stengel (1986). The space–time Poisson white noise terms, $d\mathbf{P}(t) = [dP_i(t)]_{q\times 1}$, consist of q independent differentials of space–time Poisson processes each. They are related to Poisson random measure, $\mathcal{P}_l(dz_l, dt)$, formulation (see Gihman and Skorohod 1972):

$$dP_l(t) = \int_{\mathcal{Z}_l} z_l \mathcal{P}_l(dz_l, dt), \tag{7}$$

where z_l is the Poisson jump amplitude random variable or the mark of the $dP_l(t)$ Poisson process for $l=1$ to q. These have expectation:

$$\mathrm{E}[d\mathbf{P}(t)] = \Lambda(t)dt \int_{\mathcal{Z}} \mathbf{z}\phi(\mathbf{z},t)d\mathbf{z} = \Lambda(t)dt \left[\int_{\mathcal{Z}_i} z_i\phi_i dz_i\right]_{q\times 1} \equiv \Lambda(t)\overline{\mathbf{Z}}(t)dt, \tag{8}$$

where $\Lambda(t)$ is the diagonal matrix representation of the Poisson rates $\lambda_l(t)$ for $l=1$ to q, $\overline{\mathbf{Z}}(t)$ is the mean of the jump amplitude mark vector and $\phi_l(z_l,t)$ is the density of the lth amplitude mark component. Assuming component–wise independence, $d\mathbf{P}(t)$ has covariance given by

$$\mathrm{Cov}[d\mathbf{P}(t), d\mathbf{P}^\top(t)] = \Lambda(t)dt \int_{\mathcal{Z}} (\mathbf{z}-\overline{\mathbf{Z}})(\mathbf{z}-\overline{\mathbf{Z}})^\top \phi(\mathbf{z},t)d\mathbf{z} \equiv \Lambda(t)\sigma(t)dt = [\lambda_i(t)\sigma_{i,j}(t)\delta_{i,j}]_{q\times q}dt, \tag{9}$$

with $\sigma(t) = [\sigma_{i,j}\delta_{i,j}]_{q\times q}$ denoting the diagonalized covariance of the amplitude mark distribution for $d\mathbf{P}(t)$. If the mark vector density does not exist as a continuous density, then $\phi(\mathbf{z},t)d\mathbf{z}$ could be easily replaced by generalized functions or by the corresponding distribution $\Phi(d\mathbf{z},t)$. Also, note that the mark vector is not assumed to have a zero mean, i.e. $\overline{\mathbf{Z}}(t) \neq \mathbf{0}_{q\times 1}$, permitting some additional modelling complexity. It is further assumed that all of the individual components of the Gaussian and Poisson noise terms are independent, implying $\mathrm{Covar}[d\mathbf{W}(t), d\mathbf{P}^\top(t)] = 0_{r\times q}$. The jth jump of the lth space–time Poisson process at time $t_{l,j}$ with amplitude $z_{l,j}$ causes the following jump from $t_{l,j}^-$ to $t_{l,j}^+$ in the state:

$$[\mathbf{X}](t_{l,j}) = [[H_0(\mathbf{X}(t_{l,j}^-), t_{l,j})]_{i,l} \cdot z_{l,j}]_{m\times 1}. \tag{10}$$

for each $l = 1$ to q and all j. The above statistical properties of the stochastic processes, $d\mathbf{W}$ and $d\mathbf{P}$, imply that the conditional infinitesimal expectation of the state is

$$\mathrm{E}[d\mathbf{X}(t) \mid \mathbf{X}(t) = \mathbf{x}, \mathbf{U}(t) = \mathbf{u}] = [F_0(\mathbf{x},t) + F_1(\mathbf{x},t)\mathbf{u} + H_0(\mathbf{x},t)\Lambda(t)\overline{\mathbf{Z}}(t)]dt \tag{11}$$

and the conditional infinitesimal covariance,

$$\mathrm{Cov}[d\mathbf{X}(t), d\mathbf{X}^\top(t) \mid \mathbf{X}(t) = \mathbf{x}, \mathbf{U}(t) = \mathbf{u}] = \left[(G_0G_0^\top)(t) + H_0(\mathbf{x},t)(\Lambda\sigma)(t)H_0^\top(\mathbf{x},t)\right] dt. \tag{12}$$

The conditional infinitesimal moments (11) and (12), in conjunction with the discrete jumps in the state (10) are fundamental for modelling applications. Note that the conditional infinitesimal mean (11) is determined by the nonlinear plant drift and the nonzero mean of the jump distributed Poisson noise, whereas the conditional infinitesimal covariance (12) is determined exclusively from covariance of the Gaussian and Poisson noise processes compounded by their respective amplitude functions, G_0 and H_0.

The stochastic dynamic programming formulation follows from an application of the principle of optimality to the optimal expected performance value (4) with (2)–(3), and the general Itô chain rule for Markov stochastic processes in continuous time (see Gihman and Skorohod 1972, Hanson 1996 for a more tractable derivation). This chain rule differs from the chain rule of calculus in that it must account for the lack of continuity or smoothness in the stochastic noise, and to a lesser extent on the randomness of the noise. The form of the chain rule needed here is

$$\begin{aligned} dv(\mathbf{X}(t),t) &= v(\mathbf{X}(t+dt), t+dt) - v(\mathbf{X}(t),t) \\ &= \left(\frac{\partial v}{\partial t} + (\mathbf{F}_0 + F_1\mathbf{U}(t))^\top \nabla_x[v] + \frac{1}{2}(G_0G_0^\top) : \nabla_x[\nabla_x^\top[v]]\right)(\mathbf{X}(t),t)dt \\ &+ \left((G_0 d\mathbf{W}(t))^\top \nabla_x[v]\right)(\mathbf{X}(t),t) \\ &+ \int_{\mathcal{Z}} [v(\mathbf{X}(t) + H_0(\mathbf{X}(t),t)\mathbf{z}, t) - v(\mathbf{X}(t),t)]\, \mathcal{P}(d\mathbf{z}, dt), \end{aligned} \tag{13}$$

where the double dot product is defined by $A : B = \sum_i \sum_j A_{i,j}B_{j,i} = \mathrm{Trace}[AB^\top]$. The last term of (13) represents the total contributions of the jumps in the state from the Poisson processes. Using the chain rule, along with the principle of optimality over the time horizon, yields the partial differential equation of stochastic dynamic programming:

$$0 = \frac{\partial v}{\partial t}(\mathbf{x},t) + \operatorname*{Min}_{\mathbf{u}\in\mathcal{D}_\mathbf{u}} \Big[(\mathbf{F}_0(\mathbf{x},t) + F_1(\mathbf{x},t)\mathbf{u})^\top \nabla_x[v](\mathbf{x},t)$$

$$
\begin{aligned}
&+ \quad \frac{1}{2}\left(\left(G_0G_0^\top\right) : \nabla_x[\nabla_x^\top[v]]\right)(\mathbf{x},t) + C_0(\mathbf{x},t) + \mathbf{C}_1^\top(\mathbf{x},t)\mathbf{u} + \frac{1}{2}\mathbf{u}^\top C_2(\mathbf{x},t)\mathbf{u} \qquad (14)\\
&+ \quad \sum_{l=1}^{q} \lambda_l \int_{\mathcal{Z}_l} [v(\mathbf{x}+\mathbf{H}_{0,l}(\mathbf{x},t)z_l,t) - v(\mathbf{x},t)]\,\phi_l(z_l,t)dz_l \Big], \quad \mathbf{x} \in \mathcal{D}_x,
\end{aligned}
$$

where the lth component of the vector jump coefficient is $\mathbf{H}_{0,l}(\mathbf{x},t) \equiv [H_{0,i,l}(\mathbf{x},t)]_{m\times 1}$ for $l = 1$ to q. The backward functional partial differential equation (PDE) (14) is known as the Hamilton–Jacobi–Bellman (HJB) equation and is augmented by the final condition Eq. (5). The argument of the minimum function of Eq. (14) is called the Hamiltonian of the system. Upon separating the control independent and dependent parts, the HJB equation (14) for the LQGP/U problem can be rewritten as follows:

$$
\begin{aligned}
0 \quad &= \quad \frac{\partial v}{\partial t}(\mathbf{x},t) + C_0(\mathbf{x},t) + \left(\mathbf{F}_0{}^\top \nabla_x[v]\right)(\mathbf{x},t) + \frac{1}{2}\left((G_0G_0^\top) : \nabla_x[\nabla_x^\top[v]]\right)(\mathbf{x},t)\\
&+ \quad \sum_{l=1}^{q} \lambda_l \int_{\mathcal{Z}_l} [v(\mathbf{x}+\mathbf{H}_{0,l}(\mathbf{x},t)z_l,t) - v(\mathbf{x},t)]\,\phi_l(z_l,t)dz_l + \mathcal{S}^*(\mathbf{x},t), \quad \mathbf{x} \in \mathcal{D}_x, \qquad (15)
\end{aligned}
$$

where the control minimization terms have been collected in the control switching term,

$$
\mathcal{S}^*(\mathbf{x},t) \quad \equiv \quad \operatorname*{Min}_{\mathbf{u}\in\mathcal{D}_\mathbf{u}} \left[\mathbf{C}_1^\top(\mathbf{x},t)\mathbf{u} + (F_1(\mathbf{x},t)\mathbf{u})^\top \nabla_x[v](\mathbf{x},t) + \frac{1}{2}\mathbf{u}^\top C_2(\mathbf{x},t)\mathbf{u}\right] \qquad (16)
$$

In absence of control constraints, the argument of minimum of the control switching term (16) is known as the regular control vector, $\mathbf{u}_{\text{reg}}(\mathbf{x},t)$, which by standard calculus optimization is

$$
\mathbf{u}_{\text{reg}}(\mathbf{x},t) = -\left(C_2^{-1}\left(\mathbf{C}_1 + F_1^\top \nabla_x[v]\right)\right)(\mathbf{x},t). \qquad (17)
$$

In our prior paper, Westman and Hanson (1997), the regular control had a much more complex form, because it included a Poisson noise term with a coefficient that was linear in the control vector. This results in the need to invert a matrix that is the combination of the quadratic cost coefficient, C_2, and the Poisson coefficient term, making the problem much more difficult. In the case of component–wise or hypercube control constraints, the control domain $\mathcal{D}_u$ is specified by

$$
U_{\min,i}(t) \le u_i \le U_{\max,i}(t), \quad \text{for} \quad i = 1 \quad \text{to} \quad n, \qquad (18)
$$

and the stochastic optimal control vector $\mathbf{u}^*(\mathbf{x},t)$ are given in terms of components:

$$
u_i^*(\mathbf{x},t) = \left\{ \begin{array}{cc} U_{\min,i}(t), & u_{\text{reg},i}(\mathbf{x},t) < U_{\min,i}(t) \\ u_{\text{reg},i}(t), & U_{\min,i}(t) \le u_{\text{reg},i}(\mathbf{x},t) \le U_{\max,i}(t) \\ U_{\max,i}(t), & U_{\max,i}(t) < u_{\text{reg},i}(\mathbf{x},t) \end{array} \right\}, \quad \text{for} \quad i = 1 \quad \text{to} \quad n. \qquad (19)
$$

Unlike standard PDEs, solving the HJB equation (15)–(16) requires finding the optimal value function $v(\mathbf{x},t)$ simultaneously with the optimal control vector $\mathbf{u}^*(\mathbf{x}$,t).

The paper is arranged as follows. In Section 2., the formal general theory of our predictor–corrector Crank–Nicolson method for the numerical approximation of the Hamilton–Jacobi–Bellman equation is described, with new modifications for an accelerating least squares LQGP approximation and for treating multidimensional distributed Poisson jump integrals. In Section 3., a computational example for a multistage manufacturing system (MMS) is implemented as an illustration of how to modify the formal general theory to accommodate degenerate coefficients.

2. Numerical considerations for HJB equation

The backward, functional PDE of stochastic dynamic programming (15) subject to the final condition (5) is far too complex to consider exact solutions, so that numerical approximations are necessary. In addition, the HJB equation is not like standard PDEs due to its functional parts, since the minimization embodied in the switching term (16) is needed to obtain the optimal control along with the optimal value, and the global dependence of the Poisson jump integral term is unlike the local dependence of standard partial derivatives. This global functional dependence makes Poisson noise terms far more complex than the local second order partial derivatives due to Gaussian noise.

A summary of the second order central finite difference discretization scheme in the state and backward time, using a compact subscript notation is given below:

$$\begin{gathered}
\mathbf{x} \to \mathbf{X_j} = [X_{i,j_i}]_{m\times 1} = [X_{i,0} + j_i \cdot \mathrm{DX}_i]_{m\times 1} \;\; \text{for} \;\; \mathbf{j} = [j_i]_{m\times 1},\; j_i = 0 \;\; \text{to} \;\; \mathrm{NX}_i,\; i = 1 \;\; \text{to} \;\; m; \\
t \to T_k = t_f - k * DT \;\; \text{for} \;\; k = 0 \;\; \text{to} \;\; K; \\
v(\mathbf{X_j}, T_k) \to V_{\mathbf{j},k}; \\
\frac{\partial v}{\partial t}(\mathbf{X_j}, T_k) \to (V_{\mathbf{j},k+1/2+1/2} - V_{\mathbf{j},k+1/2-1/2})/(-2\cdot(1/2)\mathrm{DT}); \\
\nabla_x[v](\mathbf{X_j}, T_k) \to \mathbf{DV}_{\mathbf{j},k} = \left[(V_{\mathbf{j}+\mathbf{e}_i,k} - V_{\mathbf{j}-\mathbf{e}_i,k})/(2\mathrm{DX}_i)\right]_{m\times 1}, \mathbf{e}_l \equiv [\delta_{i,l}]_{m\times 1}; \\
\nabla_x[\nabla_x^\top[v]](\mathbf{X_j}, T_k) \to \mathrm{DDV}_{\mathbf{j},k} = \left[(V_{\mathbf{j}+\mathbf{e}_i,k} - 2V_{\mathbf{j},k} + V_{\mathbf{j}-\mathbf{e}_i,k})/\mathrm{DX}_i^2\right]_{m\times 1}; \\
v(\mathbf{X_j} + \mathbf{H}_{0,l}(\mathbf{X_j}, T_k) z_l) \to \mathrm{HV0}_{l,\mathbf{j},k} \\
\mathbf{u}_{\mathrm{reg}}(\mathbf{X_j}, T_k) \to \mathbf{UR}_{\mathbf{j},k} = \underset{\mathbf{u}}{\mathrm{Argmin}} \left[(F1_{\mathbf{j},k}\mathbf{u})^\top \mathbf{DV}_{\mathbf{j},k}; + \mathbf{C1}_{\mathbf{j},k}^\top \mathbf{u}(t) + \tfrac{1}{2}\mathbf{u}^\top(t) C2_{\mathbf{j},k}\mathbf{u}(t)\right]; \\
\mathbf{u}^*(\mathbf{X_j}, T_k) \to \mathbf{U}_{\mathbf{j},k} = \underset{\mathbf{u}\in\mathcal{D}_\mathbf{u}}{\mathrm{Argmin}} \left[(F1_{\mathbf{j},k}\mathbf{u})^\top \mathbf{DV}_{\mathbf{j},k}; + \mathbf{C1}_{\mathbf{j},k}^\top \mathbf{u}(t) + \tfrac{1}{2}\mathbf{u}^\top(t) C2_{\mathbf{j},k}\mathbf{u}(t)\right]; \\
\mathcal{S}^*(\mathbf{X_j}, T_k) \to \mathcal{S}_{\mathbf{j},k} = \underset{\mathbf{u}\in\mathcal{D}_\mathbf{u}}{\mathrm{Min}} \left[(F1_{\mathbf{j},k}\mathbf{u})^\top \mathbf{DV}_{\mathbf{j},k}; + \mathbf{C1}_{\mathbf{j},k}^\top \mathbf{u}(t) + \tfrac{1}{2}\mathbf{u}^\top(t) C2_{\mathbf{j},k}\mathbf{u}(t)\right].
\end{gathered} \tag{20}$$

In the above, the second order central difference approximation is shown for the derivatives of $v(\mathbf{x}, t)$ with respect to $\mathbf{x}$ at interior points, but one–sided differences of the same order are used for boundary points. The delay terms associated with the Poisson noise, $\mathrm{HV0}_{\mathbf{j},k}$ are computed using linear interpolation having the same order of accuracy as that used for the derivatives.

In the case of linear dynamics, quadratic costs and Gaussian noise (the LQG problem, see Bryson and Ho 1975, Lewis 1986, Dorato et al. 1995) formal solutions exist that require numerical solution for a state–independent matrix Riccati equation. Further, if there are Poisson noise terms (the LQGP problem, see Wonham 1970, Westman and Hanson 1997), then formal solutions exist that are quadratic state–space decompositions, that require the numerical solution for a system of genuine nonlinear differential equations in time. In order to obtain a numerical solution of the LQGP/U problem, a predictor–corrector method is used, since the HJB equation is quadratic in the gradient, $\nabla_x[v](\mathbf{x}, t)$, resulting from quadratic costs. The predictor–corrector method is combined with a Crank–Nicolson method so that the local truncation error is second order in both the state and time derivatives, coupled with linear interpolation of equivalent accuracy for the functional Poisson jump integrals. Extrapolation is also implemented at the starting iterations. This Crank–Nicolson, predictor–corrector method is a modification for stochastic dynamic programming described by Hanson et al. (1991, 1993, 1996). This method with modifications is based on an earlier (Douglas and Dupont 1970) Crank–Nicolson, predictor–corrector method for standard parabolic partial differential equations.

In this paper, we add several new improvements to the numerical method. These include a LQGP approximation to accelerate starting predictor–corrector iterations, where the approximate LQGP coefficients are estimated by multidimensional least squares in state space from the given coefficients in the HJB equation. Also, a systematic treatment of the Poisson jump integral term for proper arbitrary jump distributions using a Gauss quadrature type integration method with the amplitude mark density as a weight function. In deriving these new modifications, we have also developed some new multilinear extensions to linear algebra for cubic and quartic array forms that should be useful in other nonlinear problems.

2.1. LQGP least squares approximation

In this section, we newly derive the formal general theory of *Least Squares Approximation* (LSA) of coefficients for a nonlinear optimal control problem that is an LQGP problem in the control n–vector $\mathbf{u}$ only (i.e. LQGP/U), but can be arbitrarily nonlinear in the state m–vector $\mathbf{x}$. In LSA, the approximated coefficients define a genuine LQGP problem that is LQGP in both control and state vectors. The motivation is that the least squares approximate LQGP problem is a reasonable estimate of an averaged nonlinear problem. In absence of control constraints, the solution of the approximate LQGP problem is at least a reasonable approximation of the solution to the original problem using regular optimal control. This regular control solution has considerable computational savings over the full nonlinear problem since the solution of the state independent *Riccati-like* systems of nonlinear ordinary differential equations avoid the intensive search over the large dimension, discretized state space used in the iterative approximation of the solution of the nonlinear HJB equation. In LSA, the state–space search is replaced by a solution of the *Riccati-like* nonlinear system in time only, while the state dependence of the LSA optimal value can be recovered directly from the approximate LQGP quadratic form in state space. Thus, the *Curse of Dimensionality* (Bellman 1957) is not a concern for the LSA-LQGP approximation for the LQGP/U problem. The primary target of LSA is the first time step where there is no second value to extrapolate from the final condition.

The form of the least squares approximation of the LQGP/U problem (denoted with superscript (0)) varies with each of the original nonlinear coefficients, such that each approximate coefficient is valid for the genuine LQGP problem in both control and state. Depending on the coefficient, the LQGP least squares fit functions can be of three types: state independent, linear (affine) in the state, or quadratic in the state. Time, t, is merely a parameter held fixed in the least squares procedure. Below is a list of the LQGP form of each approximate state–dependent coefficient (superscripts '(0)' mark the LSA approximation and the second subscript indicates the order of state–space dependence):

- Linear Control–Independent Dynamics Coefficient ($m \times 1$): $\mathbf{F}_0^{(0)}(\mathbf{x},t) = \mathbf{F}_{0,0}^{(0)}(t) + F_{0,1}^{(0)}(t)\mathbf{x}$;

- State–Independent Linear Control Dynamics Coefficient ($m \times n$): $F_1^{(0)}(\mathbf{x},t) = F_{1,0}^{(0)}(t)$;

- Control–Independent, State–Dependent Gaussian Coefficient $(m \times r)$: $G_0^{(0)}(\mathbf{x},t) = G_{0,0}^{(0)}(t) + G_{0,1}^{(0)}(t) \cdot \mathbf{x}$; where '$G_{0,1}^{(0)} \cdot \mathbf{x}$' denotes the $(m \times r)$ array valued right dot product of the $(m \times r \times m)$ array $G_{0,1}^{(0)}$ and m–vector $\mathbf{x}$;

- Linear State–Dependent Poisson Amplitude Coefficient $(m \times q)$: $H_0^{(0)}(\mathbf{x},t) = H_{0,0}^{(0)}(t) + H_{0,1}^{(0)}(t) \cdot \mathbf{x}$, where '$H_{0,1}^{(0)} \cdot \mathbf{x}$' denotes the $(m \times q)$ array valued right dot product of the $(m \times q \times m)$ array $H_{0,1}^{(0)}$ and m–vector $\mathbf{x}$;

- Quadratic Control–Independent Cost Coefficient ((1×1), i.e. scalar):

$$C_0^{(0)}(\mathbf{x},t) = C_{0,0}^{(0)}(t) + \mathbf{C}_{0,1}^{(0)\top}(t)\mathbf{x} + \frac{1}{2} C_{0,2}^{(0)}(t) : \mathbf{x}\mathbf{x}^\top,$$

  where the ':' denotes here the trace related, scalar valued, double dot product $C_{0,2}^{(0)} : \mathbf{x}\mathbf{x}^\top = \mathrm{Trace}[C_{0,2}^{(0)}\mathbf{x}\mathbf{x}^\top] = \mathbf{x}^\top C_{0,2}^{(0)}\mathbf{x}$ for the $(m \times m)$ array $C_{0,2}^{(0)}(t)$;

- State–Linear Approximation of Control–Linear Cost Coefficient $(n \times 1)$: $\mathbf{C}_1^{(0)}(\mathbf{x},t) = \mathbf{C}_{1,0}^{(0)}(t) + C_{1,1}^{(0)}(t)\mathbf{x}$;

- State–independent Approximation of Control–Quadratic Cost Coefficient $(n \times n)$: $C_2^{(0)}(\mathbf{x},t) = C_{2,0}^{(0)}(t)$.

As above and in the following derivations, we will find limitations in the standard linear algebra when encountering cubic and quartic array forms, i.e. three and four dimensional arrays (here the array dimension is the same as the number of subscripts), so that we need to develop extensions of linear algebra for multilinear or even nonlinear algebra, which easily could be extended to higher dimensions. We derive the least squares approximations (LSA) for the three types of coefficients.

State Independent Coefficients: The two coefficients $F_{1,0}^{(0)}(t)$ and $C_{2,0}^{(0)}(t)$ are constant for fixed t. Therefore in these two cases, a general averaged least squares error objective is given in terms of a generic scalar valued function $f(\mathbf{x})$ and its generic constant approximation f_0:

$$\overline{E}_0 = \frac{1}{|\mathcal{D}_x|} \int_{\mathcal{D}_x} (f(\mathbf{x}) - f_0)^2 d\mathbf{x}, \tag{21}$$

where the size of the x–domain is $|\mathcal{D}_x| \equiv \int_{\mathcal{D}_x} d\mathbf{x}$, provided the x–domain, $\mathcal{D}_x$, is bounded. If $\mathcal{D}_x$ is discrete or has discrete components then we assume that the integrals are replaced or augmented by sums, $|\mathcal{D}_x| = \mathcal{N}_x \equiv \sum_{\mathbf{j}_x} 1$ with $\mathbf{j}_x = [j_i]_{m \times 1}$. The least squares value is determined by the critical condition:

$$-\frac{1}{2}\frac{\partial \overline{E}_0}{\partial f_0} = \frac{1}{|\mathcal{D}_x|} \int_{\mathcal{D}_x} (f(\mathbf{x}) - f_0) d\mathbf{x} = \overline{f} - f_0 = 0,$$

so that the least squares approximation for the constant case is

$$f_0 = \overline{f}, \tag{22}$$

the average function value. In the two actual cases of interest,

$$F_{1,0}^{(0)}(t) \;=\; [F_{1,0,i,j}^{(0)}(t)]_{m\times n} = \overline{F_1}(t) \equiv \frac{1}{|\mathcal{D}_x|}\int_{\mathcal{D}_x} F_1(\mathbf{x},t)d\mathbf{x}, \tag{23}$$

using the general state–space average, and

$$C_{2,0}^{(0)}(t) = \overline{C_2}(t) \equiv \frac{1}{|\mathcal{D}_x|}\int_{\mathcal{D}_x} C_2(\mathbf{x},t)d\mathbf{x}, \tag{24}$$

which is the mean value in each case.

Linear State Coefficients: The approximations for $\mathbf{F}_0^{(0)}(\mathbf{x},t)$, $G_0^{(0)}(\mathbf{x},t)$, $H_0^{(0)}(\mathbf{x},t)$ and $\mathbf{C}_1^{(0)}(\mathbf{x},t)$ are all linear in the state vector. Let the linear least squares approximation for the generic m_1–vector valued function $\mathbf{f}(\mathbf{x})$ be $\mathbf{f}_0 + f_1\mathbf{x}$ with linear regression quadratic objective:

$$\overline{E}_1 = \frac{1}{|\mathcal{D}_x|}\int_{\mathcal{D}_x} (\mathbf{f}(\mathbf{x}) - \mathbf{f}_0 - f_1\mathbf{x})^\top(\mathbf{f}(\mathbf{x}) - \mathbf{f}_0 - f_1\mathbf{x})d\mathbf{x}, \tag{25}$$

with the least squares location determined by the pair of critical conditions:

$$-\tfrac{1}{2}\nabla_{\mathbf{f}_0}[\overline{E}_1] = \tfrac{1}{|\mathcal{D}_x|}\textstyle\int_{\mathcal{D}_x}(\mathbf{f}(\mathbf{x}) - \mathbf{f}_0 - f_1\mathbf{x})d\mathbf{x} = \overline{\mathbf{f}} - \mathbf{f}_0 - f_1\overline{\mathbf{x}} = \mathbf{0}_{m_1\times 1},$$

$$-\tfrac{1}{2}\nabla_{f_1}[\overline{E}_1] = \tfrac{1}{|\mathcal{D}_x|}\textstyle\int_{\mathcal{D}_x}(\mathbf{f}(\mathbf{x}) - \mathbf{f}_0 - f_1\mathbf{x})\mathbf{x}^\top d\mathbf{x} = \overline{\mathbf{f}\mathbf{x}^\top} - \mathbf{f}_0\overline{\mathbf{x}^\top} - f_1\overline{\mathbf{x}\mathbf{x}^\top} = 0_{m_1\times m_2}.$$

These critical conditions lead to an algebraic set of equations, using forward elimination plus some algebra leads to the reduced equations:

$$\mathbf{f}_0 = \overline{\mathbf{f}} - f_1\overline{\mathbf{x}}, \tag{26}$$

defined recursively from f_1, where f_1 implicitly satisfies

$$f_1\overline{\delta\mathbf{x}\delta\mathbf{x}^\top} = \overline{\delta\mathbf{f}\delta\mathbf{x}^\top} \quad\text{or}\quad f_1\mathrm{Cov}[\mathbf{x},\mathbf{x}^\top] = \mathrm{Cov}[\mathbf{f},\mathbf{x}^\top], \tag{27}$$

where the deviation from the mean value is defined by $\delta F \equiv F - \overline{F}$, for some function F, and where the state–space covariance is defined by

$$\mathrm{Cov}[\mathbf{f},\mathbf{x}^\top] = \overline{\delta\mathbf{f}\delta\mathbf{x}^\top} = \frac{1}{|\mathcal{D}_x|}\int_{\mathcal{D}_x}\delta\mathbf{f}\delta\mathbf{x}^\top d\mathbf{x}.$$

Reducing the equation for f_1 to deviations creates a better posed numerical solution, but symbolic computation rather than numerical computation can be used as well.

The targeted linear approximations then have state–independent and linear coefficients that satisfy the following sets of equations:

$$\begin{array}{ll}
F_{0,1}^{(0)}(t)\overline{\delta\mathbf{x}\delta\mathbf{x}^\top} = \overline{\delta\mathbf{F}_0\delta\mathbf{x}^\top}, & \mathbf{F}_{0,0}^{(0)}(t) = \overline{\mathbf{F}_0}(t) - F_{0,1}^{(0)}(t)\overline{\mathbf{x}}; \\
H_{0,1}^{(0)}(t)\overline{\delta\mathbf{x}\delta\mathbf{x}^\top} = \overline{\delta H_0\delta\mathbf{x}^\perp}, & H_{0,0}^{(0)}(t) = \overline{H_0}(t) - H_{0,1}^{(0)}(t)\cdot\overline{\mathbf{x}}; \\
G_{0,1}^{(0)}(t)\overline{\delta\mathbf{x}\delta\mathbf{x}^\top} = \overline{\delta G_0\delta\mathbf{x}^\perp}, & G_{0,0}^{(0)}(t) = \overline{G_0}(t) - G_{0,1}^{(0)}(t)\cdot\overline{\mathbf{x}}; \\
C_{1,1}^{(0)}(t)\overline{\delta\mathbf{x}\delta\mathbf{x}^\top} = \overline{\delta\mathbf{C}_1\delta\mathbf{x}^\top}, & \mathbf{C}_{1,0}^{(0)}(t) = \overline{\mathbf{C}_1}(t) - C_{1,1}^{(0)}(t)\overline{\mathbf{x}};
\end{array} \tag{28}$$

where the new 'perp–posed' (symbol $\perp$) vector $\mathbf{x}^{\perp} = [x_k]_{1\times 1\times m}$ of the third array dimension changes column vectors into 'plane' vectors of the third array dimension (in analogy to the transpose $\mathbf{x}^{\top} = [x_j]_{1\times m\times 1}$, here, changing column vectors into row vectors of the second array dimension). The $\mathbf{x}^{\perp}$ appears with the $(m \times q)$ array H_0 in a covariance to properly balance the $(m \times q \times m)$ cubic (three dimensional) array form $H^{(0)}_{0,1}(t)\overline{\delta\mathbf{x}\delta\mathbf{x}^{\top}}$, which is not something that can be done by the customary transpose. The equation for $G^{(0)}_{0,1}(t)$ has a similar form.

Quadratic State Coefficient: The remaining approximate coefficient $C^{(0)}_0(\mathbf{x},t)$ is a quadratic function of the state vector. The averaged quadratic LSA for the generic scalar valued function $f(\mathbf{x})$ is taken to be $f_0 + \mathbf{f}_1^{\top}\mathbf{x} + \frac{1}{2}f_2 : \mathbf{x}\mathbf{x}^{\top}$ with the square objective:

$$\overline{E}_2 = \frac{1}{|\mathcal{D}_x|}\int_{\mathcal{D}_x}(f(\mathbf{x}) - f_0 - \mathbf{f}_1^{\top}\mathbf{x} - \frac{1}{2}f_2 : \mathbf{x}\mathbf{x}^{\top})^2 d\mathbf{x}. \tag{29}$$

The least squares location is determined by the triplet conditions:

$$-\tfrac{1}{2}\tfrac{\partial \overline{E}_2}{\partial f_0} = \tfrac{1}{|\mathcal{D}_x|}\textstyle\int_{\mathcal{D}_x}(f(\mathbf{x}) - f_0 - \mathbf{f}_1^{\top}\mathbf{x} - \tfrac{1}{2}f_2 : \mathbf{x}\mathbf{x}^{\top})d\mathbf{x} = \overline{f} - f_0 - \mathbf{f}_1^{\top}\overline{\mathbf{x}} - \tfrac{1}{2}f_2 : \overline{\mathbf{x}\mathbf{x}^{\top}} = 0,$$

$$-\tfrac{1}{2}\nabla_{\mathbf{f}_1}[\overline{E}_2] = \overline{f\mathbf{x}} - f_0\overline{\mathbf{x}} - \overline{\mathbf{x}\mathbf{x}^{\top}}\mathbf{f}_1 - \tfrac{1}{2}\overline{\mathbf{x}\mathbf{x}^{\top}\mathbf{x}^{\perp}} : f_2 = \mathbf{0}_{m\times 1},$$

$$-\nabla_{f_2}[\overline{E}_2] = \overline{f\mathbf{x}\mathbf{x}^{\top}} - f_0\overline{\mathbf{x}\mathbf{x}^{\top}} - \overline{\mathbf{x}\mathbf{x}^{\top}\mathbf{x}^{\perp}}\cdot\mathbf{f}_1 - \tfrac{1}{2}\overline{\mathbf{x}\mathbf{x}^{\top}\mathbf{x}^{\perp}\mathbf{x}^{\mathrm{I}}} : f_2 = 0_{m\times m},$$

where the elementary array valued cubic form is defined as $\mathbf{x}\mathbf{x}^{\top}\mathbf{x}^{\perp} \equiv [x_i x_j x_k]_{m\times m\times m}$, and the cubic right double dot product, with f_2 acting on the right, is given by

$$\mathbf{x}\mathbf{x}^{\top}\mathbf{x}^{\perp} : f_2 = (f_2 : \mathbf{x}\mathbf{x}^{\top})\mathbf{x} = (\sum_k\sum_l f_{2,k,l}x_k x_l)[x_i]_{m\times 1},$$

showing the relationship to the scalar valued quadratic double dot $f_2 : \mathbf{x}\mathbf{x}^{\top}$. The new transformed 'eye' vector $\mathbf{x}^{\mathrm{I}} = [x_l]_{1\times 1\times 1\times m}$, changes column vectors into 'temporal vectors' of the fourth array dimension (in the spirit of the time variable augmenting the three spatial dimensions in physical dynamics) such that $\mathbf{x}\mathbf{x}^{\top}\mathbf{x}^{\perp}\mathbf{x}^{\mathrm{I}} \equiv [x_i x_j x_k x_l]_{m\times m\times m\times m}$ is an elementary quartic form and the quartic right double dot product is related to the scalar valued quadratic form double dot product by

$$\mathbf{x}\mathbf{x}^{\top}\mathbf{x}^{\perp}\mathbf{x}^{\mathrm{I}} : f_2 = (f_2 : \mathbf{x}\mathbf{x}^{\top})\mathbf{x}\mathbf{x}^{\top} = (\sum_k\sum_l f_{2,k,l}x_k x_l)[x_i x_j]_{m\times m}.$$

Unlike quadratic forms for symmetric matrices, the double dot product is side dependent. Hence the critical conditions can be reduced to the following recursive–coupled system:

$$\begin{aligned} f_0 &= \overline{f} - -\mathbf{f}_1^{\top}\overline{\mathbf{x}} - \tfrac{1}{2}f_2 : \overline{\mathbf{x}\mathbf{x}^{\top}}, \\ \overline{\delta\mathbf{x}\delta\mathbf{x}^{\top}}\mathbf{f}_1 + \tfrac{1}{2}(\overline{\delta\mathbf{x}\delta(\mathbf{x}^{\top}\mathbf{x}^{\perp})}) : f_2 &= \overline{f\delta\mathbf{x}} \\ \overline{\delta(\mathbf{x}\mathbf{x}^{\top})\delta\mathbf{x}^{\perp}}\cdot\mathbf{f}_1 + \tfrac{1}{2}(\overline{\delta(\mathbf{x}\mathbf{x}^{\top})\delta(\mathbf{x}^{\perp}\mathbf{x}^{\mathrm{I}})}) : f_2 &= \overline{f\delta(\mathbf{x}\mathbf{x}^{\top})}. \end{aligned} \tag{30}$$

Thus, the reduced system of equations for the approximate quadratic cost of control coefficients follows the generic forms above, except with added matrix traces, here represented as sums:

$$\begin{aligned}
C^{(0)}_{0,0}(t) &= \overline{C_0}(t) - \textstyle\sum_k \overline{x_k} C^{(0)}_{0,1,k}(t) - \frac{1}{2}\sum_k \sum_l \overline{x_k x_l} C^{(0)}_{0,2,k,l}(t), \\
\textstyle\sum_k \overline{\delta \mathbf{x} \delta x_k} C^{(0)}_{0,1,k}(t) + \frac{1}{2}\sum_k \sum_l \overline{\delta \mathbf{x} \delta(x_k x_l)} C^{(0)}_{0,2,k,l}(t), &= \overline{C_0 \delta \mathbf{x}}, \\
\textstyle\sum_k \overline{\delta(\mathbf{x}\mathbf{x}^\top) \delta x_k} C^{(0)}_{0,1,k}(t) + \frac{1}{2}\sum_k \sum_l \overline{\delta(\mathbf{x}\mathbf{x}^\top) \delta(x_k x_l)} C^{(0)}_{0,2,k,l}(t), &= \overline{C_0 \delta(\mathbf{x}\mathbf{x}^\top)}.
\end{aligned} \tag{31}$$

Pivoted Gaussian elimination with high precision is suggested in order to find a solution to the above system of equations.

LQGP Approximation for Regular Control: In the LQGP least squares approximation (LSA), the original coefficient set, say $\{\mathcal{C}(\mathbf{x}, t)\}$ in Eqs. (1) and (3), is replaced by the above LSA set $\{\mathcal{C}^{(0)}(\mathbf{x}, t)\}$.

The LSA satisfies the optimal value LQGP quadratic decomposition in the state vector:

$$v^{(0)}_{\text{reg}}(\mathbf{x}, t) = E(t) + \mathbf{D}^\top(t)\mathbf{x} + \frac{1}{2} S(t) : \mathbf{x}\mathbf{x}^\top, \quad \mathbf{x} \in \mathcal{D}_x, \tag{32}$$

which assumes use of the regular control vector, $\mathbf{u}_{\text{reg}}(\mathbf{x}, t)$ in Eq. (17). Under regular control, the optimal value and control is continuously differentiable reflecting the similar properties of the model coefficient functions (see Anderson and Moore 1990). Under constrained control, their properties will be modified by the characteristics of the constraints, making it difficult to specify general continuity properties. The unknown LQGP coefficient set includes the scalar $E(t)$, the m–vector $\mathbf{D}(t)$ and symmetric $(m \times m)$ array $S(t)$. This coefficient set needs only to be computed *once* for all $\mathbf{x}$ in the $\mathcal{D}_x$ state domain for fixed t, since only integration in time is required, modulo the state–space floating point operations implied by Eq. (32). In fact, the time integration need only be for a single time step when model (32) is coordinated with another time stepping procedure. In the case of constrained control, the LQGP quadratic decomposition only holds piecewise in separate subdomain of the state–space $\mathcal{D}_x$ corresponding to the composite form for the hypercube constrained optimal control in (19), for example. The final coefficient conditions, using Eq. (5) and comparing the same orders of $\mathbf{x}$, are that

$$E(t_f) = 0, \qquad \mathbf{D}(t_f) = \mathbf{0}_{m\times 1}, \qquad S(t_f) = S_f. \tag{33}$$

The LQGP regular approximation, $v^{(0)}_{\text{reg}}(\mathbf{x}, t)$, satisfies the regular HJB equation:

$$\begin{aligned}
0 \;=\; & \left(v^{(0)}_{\text{reg},t} + C^{(0)}_0 + \left(\mathbf{F}^{(0)\top}_0 \nabla_x[v^{(0)}_{\text{reg}}]\right) + \frac{1}{2}\left(G^{(0)}_0 G^{(0)\top}_0 : \nabla_x[\nabla^\top_x[v^{(0)}_{\text{reg}}]]\right)\right)(\mathbf{x}, t) \\
- \; & \frac{1}{2}\left(\left(\mathbf{C}^{(0)}_1 + F^{(0)\top}_1 \nabla_x[v^{(0)}_{\text{reg}}]\right)^\top \left(C^{(0)}_2\right)^{-1} \left(\mathbf{C}^{(0)}_1 + F^{(0)\top}_1 \nabla_x[v^{(0)}_{\text{reg}}]\right)\right)(\mathbf{x}, t) \\
+ \; & \sum_{l=1}^{q} \lambda_l \int_{\mathcal{Z}_l} \left[v^{(0)}_{\text{reg}}(\mathbf{x} + \mathbf{H}^{(0)}_{0,l}(\mathbf{x}, t) z_l, t) - v^{(0)}_{\text{reg}}(\mathbf{x}, t)\right] \phi_l(z_l, t) dz_l.
\end{aligned} \tag{34}$$

Upon substitution of the quadratic decomposition, Eq. (32), the LQGP regular HJB equation becomes

$$
\begin{aligned}
0 &= \dot{E}(t)+\dot{\mathbf{D}}^\top(t)\mathbf{x}+\frac{1}{2}\dot{S}(t):\mathbf{x}\mathbf{x}^\top+C_{0,0}^{(0)}+\mathbf{C}_{0,1}^{(0)\top}\mathbf{x}+\frac{1}{2}C_{0,2}^{(0)}:\mathbf{x}\mathbf{x}^\top \\
&+ \left(\mathbf{F}_{0,0}^{(0)}+F_{0,1}^{(0)\top}\mathbf{x}\right)^\top(\mathbf{D}+S\mathbf{x})+\frac{1}{2}(G_{0,0}^{(0)}+G_{0,1}^{(0)}\cdot\mathbf{x})(G_{0,0}^{(0)}+G_{0,1}^{(0)}\cdot\mathbf{x})^\top:S \\
&- \frac{1}{2}\left(\mathbf{C}_{1,0}^{(0)}+C_{1,1}^{(0)}\mathbf{x}+\mathbf{F}_{1,0}^{(0)\top}(\mathbf{D}+S\mathbf{x})\right)^\top\left(C_{2,0}^{(0)}\right)^{-1}\left(\mathbf{C}_{1,0}^{(0)}+C_{1,1}^{(0)}\mathbf{x}+\mathbf{F}_{1,0}^{(0)\top}(\mathbf{D}+S\mathbf{x})\right) \\
&+ \sum_{l=1}^{q}\lambda_l\int_{\mathcal{Z}_l}\Big[\mathbf{D}^\top(\mathbf{H}_{0,0,l}+H_{0,1,l}\mathbf{x})z_l+\frac{1}{2}S:\Big(\big((\mathbf{H}_{0,0,l}+H_{0,1,l}\mathbf{x})\mathbf{x}^\top \\
&+ \mathbf{x}(\mathbf{H}_{0,0,l}+H_{0,1,l}\mathbf{x})^\top\big)z_l+(\mathbf{H}_{0,0,l}+H_{0,1,l}\mathbf{x})(\mathbf{H}_{0,0,l}+H_{0,1,l}\mathbf{x})^\top z_l^2\Big)\Big]\phi_l(z_l,t)dz_l,
\end{aligned}
\tag{35}
$$

where the linear jump vector coefficient is

$$
(\mathbf{H}_{0,0,l}+H_{0,1,l}\mathbf{x})=(\mathbf{H}_{0,0,l}(t)+H_{0,1,l}(t)\mathbf{x})=\mathbf{H}_{0,l}^{(0)}(\mathbf{x},t)\equiv[H_{0,i,l}^{(0)}(\mathbf{x},t)]_{m\times1},
$$

for each $l=1$ to q.

Setting the state dependent terms, the zeroth order (ord$((\bullet))$) terms, to zero, yields the following first order, coupled, scalar differential equation:

$$
\begin{aligned}
0 &= \dot{E}(t)+C_{0,0}^{(0)}+\mathbf{F}_{0,0}^{(0)\top}\mathbf{D}-\frac{1}{2}\left(\mathbf{C}_{1,0}^{(0)}+\mathbf{F}_{1,0}^{(0)\top}\mathbf{D}\right)^\top\left(C_{2,0}^{(0)}\right)^{-1}\left(\mathbf{C}_{1,0}^{(0)}+\mathbf{F}_{1,0}^{(0)\top}\mathbf{D}\right) \\
&+ \frac{1}{2}G_{0,0}^{(0)}G_{0,0}^{(0)\top}:S+\sum_{l=1}^{q}\lambda_l\left[\mathbf{D}^\top\mathbf{H}_{0,0,l}\overline{Z_l}+\frac{1}{2}S:\mathbf{H}_{0,0,l}\mathbf{H}_{0,0,l}^\top\overline{Z_l^2}\right],
\end{aligned}
\tag{36}
$$

where $\overline{Z_l}=\overline{Z_l}(t)\equiv\int_{\mathcal{Z}_l}z_l\phi_l(z_l,t)dz_l$ and $\overline{Z_l^2}=\overline{Z_l^2}(t)\equiv\int_{\mathcal{Z}_l}z_l^2\phi_l(z_l,t)dz_l$.

Upon equating like orders of powers of the state $\mathbf{x}$, first order (ord$(\mathbf{x}^\top(\bullet))$) terms satisfy the following vector differential equation:

$$
\begin{aligned}
\mathbf{0}_{m\times1} &= \dot{\mathbf{D}}(t)+\mathbf{C}_{0,1}^{(0)}+F_{0,1}^{(0)\top}\mathbf{D}+S\mathbf{F}_{0,0}^{(0)}-\left(C_{1,1}^{(0)}+\mathbf{F}_{1,0}^{(0)\top}S\right)^\top\left(C_{2,0}^{(0)}\right)^{-1}\left(\mathbf{C}_{1,0}^{(0)}+\mathbf{F}_{1,0}^{(0)\top}\mathbf{D}\right) \\
&+ G_{0,1}^{(0)\perp}:SG_{0,0}^{(0)}+\sum_{l=1}^{q}\lambda_l\left[\left(H_{0,1,l}^\top\mathbf{D}+S\mathbf{H}_{0,0,l}\right)\overline{Z_l}+H_{0,1,l}^\top S\mathbf{H}_{0,0,l}\overline{Z_l^2}\right],
\end{aligned}
\tag{37}
$$

where the x–gradient, $\nabla_x[\bullet]$, of the right hand side of LQGP HJB Eq. (35) was used, followed by setting remaining $\mathbf{x}$ terms to zero.

Finally, the second order (ord$(\frac{1}{2}\mathbf{x}^\top(\bullet)\mathbf{x})$) = (ord$(\frac{1}{2}(\bullet):\mathbf{x}\mathbf{x}^\top)$) satisfies the following matrix Riccati differential equation:

$$
\begin{aligned}
0_{m\times m} &= \dot{S}(t)+C_{0,2}^{(0)}+F_{0,1}^{(0)\top}S+SF_{0,1}^{(0)}-\left(C_{1,1}^{(0)}+\mathbf{F}_{1,0}^{(0)\top}S\right)^\top\left(C_{2,0}^{(0)}\right)^{-1}\left(C_{1,1}^{(0)}+\mathbf{F}_{1,0}^{(0)\top}S\right) \\
&+ G_{0,1}^{(0)\perp}:SG_{0,1}^{(0)}+\sum_{l=1}^{q}\lambda_l\left[\left(SH_{0,1,l}+H_{0,1,l}^\top S\right)\overline{Z_l}+H_{0,1,l}^\top SH_{0,1,l}\overline{Z_l^2}\right],
\end{aligned}
\tag{38}
$$

which can be calculated by taking the $\nabla_x[\nabla_x[\bullet]]$ operation of the right hand side of the LQGP HJB equation. Note that the 'perp' ($\perp$) array operation is used to put the three subscript, first order diffusion or Gaussian noise coefficient, $G_{0,1}^{(0)}$, in the proper form for the array dimensions of the $S(t)$ and $\mathbf{D}(t)$ equations.

In concluding this section, we again note that this Riccati set of three equations only needs to be solved once for all state vectors $\mathbf{x} \in \mathcal{D}_x$ for a given time t, so the corresponding extrapolated accelerator for $v_{\rm reg}(\mathbf{x}, t)$ is a good start for all predictor–corrector iterations under unconstrained regular control. For simplicity, only the unconstrained control is presented. Modifications can be made without much difficulty to extend the results to the constrained case.

2.2. Gauss–statistics quadrature for jump integrals

Since we seek a numerical integration method for an arbitrary jump density $\phi_l(z_l, T_k)$ in the HJB Eq. (15) and with sufficiently high accuracy, a Gauss type quadrature will be derived that is exact, modulo finite precision effects, for the first several moments of the density. Hence, this quadrature will be called *Gauss–Statistics quadrature* due to the property of integrating the maximum number of statistical moments exactly in infinite precision. For simplicity, we drop the jump subscript and time dependence, seeking an approximation of the mean jump defining integral:

$$\overline{f(Z)} \equiv \int_{\mathcal{Z}} f(z)\phi(z)dz \simeq \sum_l w_l f(z_l), \tag{39}$$

for some integrable function $f(z)$ with respect to the density $\phi(z)$, where $z_l \in \mathcal{Z}$ are the nodes and w_l are the corresponding weights. Discrete distributions are allowed, in which the density $\phi(z)$ can be interpreted as a generalized function or $\phi(z)dz$ is replaced by the corresponding distribution function differential $d\Phi(z)$ and the integral is replaced by a sum where appropriate.

The simplest Gauss–Statistics quadrature is the one point rule which satisfies the 0th and 1st moments exactly:

$$1 = \int_{\mathcal{Z}} \phi(z)dz = w_1 \quad \text{and} \quad \overline{Z} = \int_{\mathcal{Z}} z\phi(z)dz = w_1 z_1,$$

so $w_1 = 1$, conserving probability, and $z_1 = \overline{Z}$, the mean, given a proper density function $\phi(z)$.

However, it is necessary to use a two–point Gauss–Statistics quadrature, since the one–point version does not have sufficient accuracy. The two–point rule has the form:

$$\overline{f(Z)} \equiv \int_{\mathcal{Z}} f(z)\phi(z)dz \simeq w_1 f(z_1) + w_2 f(z_2), \quad z_1 < z_2 \in \mathcal{Z}, \tag{40}$$

and the quadrature possesses cubic moment accuracy such that

$$\begin{aligned} 1 &= w_1 + w_2, \\ \overline{Z} &= w_1 z_1 + w_2 z_2, \\ \overline{Z^2} &= w_1 z_1^2 + w_2 z_2^2 = (\overline{Z})^2 + \mathrm{Var}[Z], \end{aligned} \tag{41}$$

$$\overline{Z^3} = w_1 z_1^3 + w_2 z_2^3 = (\overline{Z})^3 + 3\overline{Z}\mathrm{Var}[Z] + \overline{(\delta Z)^3},$$

which comprise four equations in the four unknown quadrature parameters $\{w_1, w_2, z_1, z_2\}$. In Eq. (41), the basic moments have been related to the relevant central moments in terms of the deviation $\delta Z \equiv Z - \overline{Z}$, so that $\overline{\delta Z} = 0$, $\text{Var}[Z] \equiv \overline{(\delta Z)^2}$, to systematize the algebraic simplifications. After much algebra, the quadrature parameter formulas are found to be

$$\begin{array}{ll} z_1 = \overline{Z} + \left(\gamma/2 - \sqrt{(\gamma/2)^2 + 1}\right)\sigma, & z_2 = \overline{Z} + \left(\gamma/2 + \sqrt{(\gamma/2)^2 + 1}\right)\sigma, \\ w_1 = \frac{1}{2}\left(1 + \gamma/\left(2\sqrt{(\gamma/2)^2+1}\right)\right), & w_2 = \frac{1}{2}\left(1 - \gamma/\left(2\sqrt{(\gamma/2)^2+1}\right)\right), \end{array} \tag{42}$$

where we have used the notation that $\sigma \equiv \sqrt{\text{Var}[Z]}$ is the standard deviation and the normalized skewness of the density is $\gamma \equiv \text{Skew}[Z] \equiv \overline{(\delta Z)^3}/(\text{Var}[Z])^{\frac{3}{2}}$.

Since the two–point Gauss–Statistics quadrature has cubic accuracy, it has the same order of accuracy as the usual three–point Simpson's rule with even mesh spacing, and indeed the error is related to the error in the fourth moment and the fourth order derivative of $f(z)$:

$$E_2 \equiv \int_{\mathcal{Z}} f(z)\phi(z)dz - \sum_{l=1}^{2} w_l f(z_l) \simeq \frac{1}{3!} f^{(4)}(\overline{Z})\left[\overline{(\delta Z)^4} - \sum_{l=1}^{2} w_l(\delta z_l)^4\right] + O(\delta^5),$$

for sufficiently differentiable functions $f(z)$ and cubic accuracy.

Formulas for composite two–point Gauss–Statistics quadrature are similar to the simple two–point rule, with only renormalization of the partial densities over the composite intervals needed. In fact, the only structural change in the above formulas is the inclusion of a multiplicative renormalization factor in the weight parameters on the subinterval (Z_i, Z_{i+1}):

$$\begin{array}{ll} z_{i,1} = \overline{Z}_i + \left(\gamma_i/2 - \sqrt{(\gamma_i/2)^2 + 1}\right)\sigma_i, & z_{i,2} = \overline{Z}_i + \left(\gamma_i/2 + \sqrt{(\gamma_i/2)^2 + 1}\right)\sigma_i, \\ w_{i,1} = \frac{1}{2}\Phi_i\left(1 + \gamma_i/\left(2\sqrt{(\gamma_i/2)^2+1}\right)\right), & w_{i,2} = \frac{1}{2}\Phi_i\left(1 - \gamma_i/\left(2\sqrt{(\gamma_i/2)^2+1}\right)\right), \end{array}$$

where $\Phi_i \equiv \int_{Z_i}^{Z_{i+1}} \phi(z)dz$ is the ith subinterval renormalization factor, $\Phi_i\overline{Z}_i \equiv \int_{Z_i}^{Z_{i+1}} z\phi(z)dz$ is the ith subinterval mean, $\sigma_i \equiv \sqrt{\text{Var}_i[Z]} \equiv \sqrt{\overline{(\delta_i Z)^2}_i} \equiv \sqrt{\overline{(Z - \overline{Z}_i)^2}_i}$ is the ith subinterval standard deviation and $\gamma_i \equiv \text{Skew}_i[Z] \equiv \overline{(\delta_i Z)^3}_i/(\text{Var}_i[Z])^{\frac{3}{2}}$ is the ith subinterval skewness of the density. In the event of discrete components of the distribution corresponding to the density ϕ, they can be assigned to individual subintervals.

Combining the Gauss–Statistics quadrature for the Poisson jump integral with approximation of the state–space dependence by linear interpolation of the Poisson jump advanced optimal value leads to

$$\begin{aligned} &\sum_{l=0}^{q} \lambda_l \int_{\mathcal{Z}_l} \left[v(\mathbf{x_j} + \mathbf{H}_{0,l}(\mathbf{x_j}, T_{k-1/2})z_l, T_{k-1/2}) - v(\mathbf{x_j}, T_{k-1/2})\phi_l(z_l, T_{k-1/2})dz_l\right] \\ &\qquad \simeq \sum_{l=0}^{q} \lambda_l \left[\sum_{p=1}^{2} w_{p,l,k-1/2}\text{LIV}_{p,l,\mathbf{j},k-1/2} - V_{\mathbf{j},k-1/2}\right], \end{aligned}$$

where subscripts have been restored for consistency with the dependence in the HJB equation, e.g. $w_{p,l,k-1/2}$ exhibits the dependence of the Gauss–Statistics pth weight parameter on the full dependence of the density $\phi_l(z_l, T_{k-1/2})$, including the Poisson and time indices. Here, $\mathrm{LIV}_{p,l,\mathbf{j},k-1/2} = \mathrm{HV0}_{p,l,\mathbf{j},k-1/2}$ denotes the linear interpolation for $v(\mathbf{x_j} + \mathbf{H}_{0,l}(\mathbf{x_j}, T_{k-1/2})z_{p,l}, T_{k-1/2}) \simeq \mathrm{HV0}_{p,l,\mathbf{j},k-1/2}$ sufficiently close to $V_{\mathbf{j}'_{p,l,\mathbf{j},k-1/2},k-1/2}$ optimal values in the neighborhood of $(\mathbf{x_j} + \mathbf{H}_{0,l}(\mathbf{x_j}, T_{k-1/2})z_l) \simeq \mathbf{x}_{\mathbf{j}'_{p,l,\mathbf{j},k-1/2}}$. This last term completes the evaluation of the predictor–corrector version of HJB Eq. (15).

2.3. Predictor–corrector iteration scheme

It is assumed that the prior value for the optimal, expected performance index, $V_{\mathbf{j},k-1}$ is known prior to calculating the new time step k for $k = 1$ to K, this is also the case for the regular control. As a requirement of the Crank–Nicolson procedure used for the second order approximate time derivative, a number of quantities must be determined at the mid–point of the time step, $T_{k-\frac{1}{2}}$. For each new time step, $k = 1$ to K, the algorithm starts with a convergence accelerating *extrapolator* (x) *start:*

$$\begin{array}{c} V^{(\mathrm{x})}_{\mathbf{j},k-1/2} = \left\{ \begin{array}{ll} \frac{1}{2}(V^{(\mathrm{lqgp})}_{\mathbf{j},k} + V_{\mathbf{j},k-1}), & k=1 \\ \frac{1}{8}(3V^{(\mathrm{lqgp})}_{\mathbf{j},k} + 6V_{\mathbf{j},k-1} - V_{\mathbf{j},k-2}), & k \geq 2 \end{array} \right\}; \\ \text{Calculate new values}: \ \mathbf{DV}^{(\mathrm{x})}_{\mathbf{j},k-1/2}; \ \ \mathrm{DDV}^{(\mathrm{x})}_{\mathbf{j},k-1/2}; \ \ \mathrm{LIV}^{(\mathrm{x})}_{\mathbf{j},k-1/2}; \ \ \mathcal{S}^{(\mathrm{x})}_{\mathbf{j},k-1/2}, \end{array} \tag{43}$$

where $V^{(\mathrm{lqgp})}_{\mathbf{j},k} = v^{(0)}_{\mathrm{reg}}$ is the LQGP least squares approximation given in Eq. (32) and final condition $V_{\mathbf{j},0} = v(\mathbf{x_j}, t)$ from Eq. (3). In Hanson et al. (1991, 1993, 1996), the first time step does not use an accelerating extrapolator start since only the values at t_f are known. However, the first step is the most crucial time step and usually has the most difficulty converging and takes the most iterations to converge. In the treatment here, this problem has been addressed, using an LQGP start for the first new time value. After the first time step, $k \geq 2$, a quadratic extrapolation is used here that has cubic accuracy for evenly spaced state nodes.

The *corrector* (xpec) *step* determines the value of V at the new time time step T_k:

$$\begin{array}{rcl} V^{(\mathrm{xpec},\gamma)}_{\mathbf{j},k} & = & V_{\mathbf{j},k-1} + DT \cdot \left[\left(C0 + F0^\top \mathbf{DV} + \frac{1}{2}(G_0 G_0^\top) : DDV \right)^{(\mathrm{xpece},\gamma-1)}_{\mathbf{j},k-1/2} \right. \\ & + & \left. \mathcal{S}^{(\mathrm{xpece},\gamma-1)}_{\mathbf{j},k-1/2} + \sum_{l=1}^{q} \lambda_l \left[\sum_{p=1}^{2} w_{p,l,k-1/2} \mathrm{LIV}^{(\mathrm{xpece},\gamma-1)}_{p,l,\mathbf{j},k-1/2} - V^{(\mathrm{xpece},\gamma-1)}_{\mathbf{j},k-1/2} \right] \right], \end{array} \tag{44}$$

for $\gamma = 0$ to $\gamma_{\max}$ while the convergence criterion is still unsatisfied. The *extrapolated – predictor* (xp) *step* is the starting corrector step: $V^{(\mathrm{xp})}_{\mathbf{j},k} = V^{(\mathrm{xpec},0)}_{\mathbf{j},k}$, and the input is the extrapolation (x) start: $V^{(\mathrm{xpece},-1)}_{\mathbf{j},k-1/2} \equiv V^{(\mathrm{x})}_{\mathbf{j},k-1/2}$.

The *corrector evaluation* (xpece) *step* is used to determine the various quantities at the half time step, $T_{k-\frac{1}{2}}$, based on the corrector step $V^{(\mathrm{xpec},\gamma)}_{\mathbf{j},k}$ and previous step value $V_{\mathbf{j},k-1}$ to form the Crank–Nicolson average (i.e. precision consistent linear interpolation):

$$\begin{array}{c} V^{(\mathrm{xpece},\gamma)}_{\mathbf{j},k-1/2} = \frac{1}{2}(V_{\mathbf{j},k-1} + V^{(\mathrm{xpec},\gamma)}_{\mathbf{j},k}); \\ \text{Calculate new values}: \ \mathbf{DV}^{(\mathrm{xpece},\gamma)}_{\mathbf{j},k-1/2}; \ \ \mathrm{DDV}^{(\mathrm{xpece},\gamma)}_{\mathbf{j},k-1/2}; \ \ \mathrm{LIV}^{(\mathrm{xpece},\gamma)}_{\mathbf{j},k-1/2}; \ \ \mathcal{S}^{(\mathrm{xpece},\gamma)}_{\mathbf{j},k-1/2}. \end{array} \tag{45}$$

Upon convergence the final corrector step is used as the value for $v(\mathbf{X_j}, T_k)$, i.e. set $V_{\mathbf{j},k} = V^{(\mathrm{xpec},\gamma^*)}_{\mathbf{j},k}$, where $\gamma^* \leq \gamma_{\max}$ is the number of corrections when the corrections are stopped according to the stopping criterion. The regular control is determined from (17) using $V_{\mathbf{j},k}$.

2.4. Stopping criterion

The stopping criterion for corrections is when the norm of the difference between successive corrections is small compared to the current norm of the values:

$$||V_{\mathbf{j},k-1/2}^{(\text{xpece},\gamma)} - V_{\mathbf{j},k-1/2}^{(\text{xpece},\gamma-1)}|| \leq \text{tolrel}||V_{\mathbf{j},k-1/2}^{(\text{xpece},\gamma)}|| + \text{tolabs}, \quad \gamma \leq \gamma_{\max}, \tag{46}$$

where *tolrel* and *tolabs* are prescribed relative and absolute tolerances, respectively, depending on the arithmetic precision and the state dimension of the problem. The absolute tolerance *tolabs* should be very small since its main purpose is to avoid irregularities in the relative error for very small optimal values. The stopping correction is denoted by γ^*, $\gamma^* \leq \gamma_{\max}$. The norm is usually the $\mathcal{L}_\infty$ norm to save computation.

In order for this Cauchy–type stopping criterion to work, the time step DT must be scaled according to the size of the state step $\mathbf{DX} = [\text{DX}_i]_{m\times 1}$. One simple, but general criterion is

$$\text{DT} \cdot \sqrt{\left(\sum_{i=1}^{m} \frac{2A_{i,i}}{\text{DX}_i^2}\right)^2 + \left(\sum_{i=1}^{m} \frac{B_i}{\text{DX}_i}\right)^2} < 1 \ , \tag{47}$$

but preferably much less than unity to account for cumulative precision loss, where the maximal diffusion approximation coefficients are given by

$$\mathbf{B} = [B_i]_{m\times 1} = \max_{\mathbf{x},\mathbf{u},t}[\mathbf{F}_0 + F_1\mathbf{u} + \textstyle\sum_{l=1}^{q} \lambda_l(\overline{Z})_l \mathbf{H}_{0,l}],$$

$$A = [A_{i,j}]_{m\times m} = \max_{\mathbf{x},\mathbf{u},t}[G_0G_0^\top + \textstyle\sum_{l=1}^{q} \lambda_l \text{Var}_l[Z]\mathbf{H}_{0,l}\mathbf{H}_{0,l}^\top].$$

Only the diagonal values of A are used. See Hanson et al. (1991, 1993, 1996) for further information. Also see Kushner and Dupuis (1992) for another approach called the Markov Chain Approximation which implicitly treats the mesh ratio condition by making the finite differences consistent with the probabilities of a Markov chain (these methods are compared in Hanson 1996).

It is a general rule in numerical calculations that the higher the dimension of the problem, the higher the precision needs to be due to cumulative precision loss with each iteration: use at least double precision, but quadruple precision is advised.

3. Multistage manufacturing system application

Consider a multistage manufacturing system (MMS) that produces a single consumable commodity, that requires a process consisting of k sequential stages. For example, the assembly or manufacture of a printed circuit board or an automobile. The loading (initial) stage is where all of the raw materials for all stages are input into the system. The unloading (final) stage is the mechanism by which the consumable good is delivered to the consumer or distributer. For simplicity, the loading and unloading stages are not considered as part of the MMS. This model is similar to a flexible manufacturing system (FMS), but its perspective is global rather than local. Each stage of the MMS can be viewed as an FMS. Kimemia and Gershwin (1983), describe the similarities and differences between FMS and MMS, while presenting an algorithm for FMS control.

This example considered is a modification of the MMS presented by Westman and Hanson (1997). The example is meant to model a realistic application, such as the previously mentioned assembly of printed circuit boards, using a formal approach, in contrast to the purely theoretical approach. A linear dynamical system is employed with the state variables consisting of the number of operational workstations and the surplus aggregate level for each stage, where the production rate was a parameter. In the model presented here, the dynamics are nonlinear and the production rate is a state variable. The cost functional used is similar to that of Westman and Hanson (1997), with an additional term used to penalize shortfalls or surplus of production continuously during the planning horizon.

The objective of the MMS model is to determine the production rates necessary to achieve the production goal subject to an uncertain environment (random fluctuations of production and workstation repair and failure). For similar models with variations see Akella and Kumar (1986) for a treatment of optimal inventory levels, as well as Boukas and Haurie (1990), Boukas and Yang (1996), and Kenne and Boukas (1997), for a treatment that includes preventive maintenance and machine age structure, and Gershwin (1997), for a treatment that includes the production scheduling of several parts.

3.1. Model formulation

Let $n_i(t)$ denote the number of operational workstations for stage $i = 1$ to k at time t of the manufacture or planning horizon. The workstations for a given stage i are assumed to have identical properties, produce at the same rate $c_i(t)$, and have a capacity of producing M_i parts per unit time. Each workstation is subject to failure and can be repaired with the mean time between failures and the repair duration exponentially distributed. The production rate, $c_i(t)$, is the utilization, i.e. the fraction of time busy, of the workstations at stage i.

The state of the manufacturing system is described by the number of operational workstations, $n_i(t)$, the surplus aggregate levels, $a_i(t)$, and by the production rates, $c_i(t)$. The total number of workstations at stage i is N_i. Hence, $n_i(t)$ can be viewed as a birth and death process confined to the interval $0 \leq n_i(t) \leq N_i$ with random jump sizes. The defining equation for the number of operational workstations is given by

$$dn_i(t) = dP_i^R(t) - dP_i^F(t), \tag{48}$$

where $dP_i^R(t)$ and $dP_i^F(t)$ are Poisson processes used to model the repair (birth) and failure (death) processes, respectively. The surplus aggregate level, $a_i(t)$, represents the surplus (if positive) or shortfall (if negative) of the production of pieces that have successfully completed i stages. The state equation for the surplus aggregate level is given by

$$da_i(t) = [M_i n_i(t)(c_i(t) + u_i(t)) - d_i(t)]\, dt + g_i(t) dW_i(t). \tag{49}$$

A change in the surplus aggregate level, $da_i(t)$, is comprised of the number of pieces that have completed i stages of the manufacturing process, $M_i n_i(t)(c_i(t) + u_i(t))dt$, that are not consumed by stage $i + 1$, $d_i(t)dt$, subject to small random fluctuations, $g_i(t)dW_i(t)$, for example, the number of defective pieces, and by the status of the workstations. The control, $u_i(t)dt$, is used to adjust the production rate, $c_i(t)$, to compensate for workstation repair and failure and random fluctuations in the manufacturing process. The state equation for the production rate is given by

$$dc_i(t) = u_i(t)dt. \tag{50}$$

This equation allows for a dynamical determination of the production rate, which was not the case in Westman and Hanson (1997).

The demand rate, $d_i(t)$, is the number of pieces needed per unit time to insure that the manufacturing process is a continuous flow, so that the production goal, the number of pieces to be produced over the time horizon, is met. The demand rate must also take into account, based on past history, a minimal buffer level sufficient to compensate for defective pieces as well as workstation failures, and to insure that the proper start-up levels are present for the next planning horizon.

For this manufacturing application, the instantaneous cost coefficients will have the following form: $C_0(\mathbf{x},t) = \frac{1}{2}\mathbf{x}^\top C_{0,2}(t)\mathbf{x}$, $\mathbf{C}_1(\mathbf{x},t) \equiv \mathbf{0}_{n\times 1}$ and $C_2(\mathbf{x},t) = C_{2,2}(t) : \mathbf{x}\mathbf{x}^\top$. The state or plant vector is given by

$$\mathbf{X}(t) = [\mathbf{n}(t), \mathbf{a}(t), \mathbf{c}(t)]^\top, \tag{51}$$

i.e. the state–space dimension is $3k$, while the control–space dimension is k as is the production rate $\mathbf{c}(t)$. The salvage cost matrix S_f is a symmetric, positive semi–definite $3k \times 3k$ matrix. The performance index is based on the idea of *Just In Time* or *stockless production* (see Hall 1983). This type of manufacturing discipline is used to reduce overhead expenses by reducing the amount of raw materials kept on hand, as well as that for the storage of the finished product. In other words, the movement or transport of materials (raw and finished) is such that *only* the necessary materials are at the proper locations at the necessary times. Further motivation for this is given by Bielecki and Kumar (1988), who show that, for an unreliable manufacturing system, the optimal policy is a zero inventory policy. In order to implement this discipline for the MMS, a penalty is imposed if we have a surplus or shortfall of production at any time in planning horizon in any stage of the manufacturing system. These penalties can be viewed as the cost of storing additional pieces or loss due to reduced production. The quadratic cost term in control, $\mathbf{u}^\top C_2(\mathbf{x},t)\mathbf{u}$, is used to force a minimum effort discipline. The performance index Eqs. (2)–(3) is the same as the one used in Westman and Hanson (1997), but with an additional term, $\frac{1}{2}\mathbf{x}^\top C_{0,2}(t)\mathbf{x}$, that is used to force the *Just In Time* discipline for the entire planning horizon. The outputs of the MMS system (48)–(50) subject to the performance index (2)–(3) are various measures for the production rate and the estimated expected value for the surplus aggregate level at the end of the planning horizon, $\hat{a}_i(t,t_f)$. In this treatment of the MMS, the number of operational workstations is never allowed to be 0 (i.e. nondegenerate: $n_i(t) \neq 0$ for all i and t), and *only* the regular or unconstrained control is considered. The virtual production rate, $c_{\mathrm{vir},i}(t)$, does not anticipate workstation repair or failure or compensate for random fluctuations in the number of pieces produced, is defined as $c_{\mathrm{vir},i}(t) \equiv \int_t^{t_f} d_i(\tau)d\tau/(M_i n_i(t)(t_f - t))$. The regular controlled production rate, $c_{\mathrm{reg},i}(t)$, anticipates workstation repair and failure as well as fluctuations in the number of pieces produced, and is given by $c_{\mathrm{reg},i}(t) = c_i(t_f) + \int_t^{t_f} u_{\mathrm{reg},i}(\tau)d\tau$. It is based on the regular or unconstrained control $u_{\mathrm{reg},i}(t)$ and therefore may not be physically realizable, i.e. not admissible. The constrained controlled production rate, $c_i^*(t)$, is the modification of the regular controlled production rate, $c_{\text{reg},i}(t)$, in order to be admissible. It is determined by

$$c_i^*(t) = \text{Min}\left[c_i^*(t), c_{\max,i}(t)\right], \quad c_{\max,i}(t) \equiv \text{Min}\left[1, \frac{c_{i-1}(t) * M_{i-1} * n_{i-1}(t)}{M_i * n_i(t)}\right], \tag{52}$$

due to the physical (workstation) and production limitations, respectively. The estimated expected value for the surplus aggregate level at the end of the planning horizon, $\hat{a}_i(t, t_f)$, measures the deviation from the target production goal based on the current status of the manufacturing system, is determined by $\hat{a}_i(t, t_f) = a_i(t_f) + M_i n_i(t)(c_i^*(t) - c_{\text{reg},i}(t)(t_f - t)$.

3.2. Numerical example

For numerical concreteness, consider a MMS with $k = 3$ stages with a planning horizon of 40 hours. Hence in this example, the state space had dimension 9 and the control space has dimension 3. Let the initial surplus aggregate level for all stages be zero. Let the demand be $d_i(t) = 285$ pieces per hour for $i = 1$ to 3. The Gaussian random fluctuations of production will be assumed absent, i.e. $g_i(t) = 0$ for $i = 1$ to 3, which means $G_0 = 0$. Let the total number of workstations, N_i, for each stage be 3, 5, and 4, respectively. The individual characteristics for an individual given workstation is summarized in the table 1 below.

Stage i	Production Capacity, M_i (pieces/hour)	Mean Time Between Failure $1/\lambda_i^F$ (hours)	Mean Time to Repair $1/\lambda_i^R$ (hours)
1	117	25.0	2.5
2	71	21.5	4.0
3	88	23.0	1.5

Table 1: Numerical parameters of workstations for each stage.

For convenience, the amplitude mark density function is chosen to be discrete and to vary with the state of the system, a variation from the Poisson jump model described at the beginning of this paper to insure that jumps that occur do not force the state to become inadmissible. For example, if there are 2 operational workstations, then 3 workstations can not fail. Let $\Phi^R_{k,i,j}$ and $\Phi^F_{k,i,j}$ denote the discrete mark transition probabilities for the repair and failure, respectively, of $j - 1$ workstations for stage k when there are i operational workstations, with transition matrices given by

$$\Phi_1^R = \begin{bmatrix} 0.00 & 0.95 & 0.05 \\ 0.00 & 1.00 & 0.00 \\ 1.00 & 0.00 & 0.00 \end{bmatrix}, \qquad \Phi_1^F = \begin{bmatrix} 1.00 & 0.00 & 0.00 \\ 0.00 & 1.00 & 0.00 \\ 0.00 & 0.90 & 0.10 \end{bmatrix},$$

$$\Phi_2^R = \begin{bmatrix} 0.00 & 0.90 & 0.07 & 0.02 & 0.01 \\ 0.00 & 0.92 & 0.07 & 0.01 & 0.00 \\ 0.00 & 0.93 & 0.07 & 0.00 & 0.00 \\ 0.00 & 1.00 & 0.00 & 0.00 & 0.00 \\ 1.00 & 0.00 & 0.00 & 0.00 & 0.00 \end{bmatrix}, \quad \Phi_2^F = \begin{bmatrix} 1.00 & 0.00 & 0.00 & 0.00 & 0.00 \\ 0.00 & 1.00 & 0.00 & 0.00 & 0.00 \\ 0.00 & 0.95 & 0.05 & 0.00 & 0.00 \\ 0.00 & 0.94 & 0.05 & 0.01 & 0.00 \\ 0.00 & 0.92 & 0.05 & 0.02 & 0.01 \end{bmatrix},$$

$$\Phi_3^R = \begin{bmatrix} 0.00 & 0.96 & 0.03 & 0.01 \\ 0.00 & 0.97 & 0.03 & 0.00 \\ 0.00 & 1.00 & 0.00 & 0.00 \\ 1.00 & 0.00 & 0.00 & 0.00 \end{bmatrix}, \qquad \Phi_3^F = \begin{bmatrix} 1.00 & 0.00 & 0.00 & 0.00 \\ 0.00 & 1.00 & 0.00 & 0.00 \\ 0.00 & 0.95 & 0.05 & 0.00 \\ 0.00 & 0.90 & 0.07 & 0.03 \end{bmatrix};$$

By comparing equations (48), (49), and (50) to the defining SDE (1) the coefficients are given by

$$F_0(\mathbf{X}(t),t) = \begin{bmatrix} \mathbf{0}_{3\times 1} \\ \mathrm{diag}[\mathbf{M}]\mathrm{diag}[\mathbf{c}(t)]\mathbf{n}(t) - \mathbf{d}(t) \\ \mathbf{0}_{3\times 1} \end{bmatrix}, \quad F_1(\mathbf{X}(t),t) = \begin{bmatrix} \mathbf{0}_{3\times 3} \\ \mathrm{diag}[\mathbf{M}]\mathrm{diag}[\mathbf{n}(t)] \\ I_{3\times 3} \end{bmatrix},$$

$$H_0(t) = \begin{bmatrix} I_{3\times 3} & -I_{3\times 3} \\ 0_{3\times 3} & 0_{3\times 3} \\ 0_{3\times 3} & 0_{3\times 3} \end{bmatrix}, \qquad d\mathbf{P}_0(t) = \begin{bmatrix} d\mathbf{P}^R(t) \\ d\mathbf{P}^F(t) \end{bmatrix},$$

where, for example, $\mathrm{diag}[\mathbf{M}] = [\Sigma_j \delta_{i,j} M_j]_{3\times 3}$ being the diagonalization of the vector whose components are given by M_i for $i = 1$ to 3. The salvage cost and the instantaneous quadratic cost coefficient matrices are given by

$$S_f = \begin{bmatrix} 0_{3\times 3} & 0_{3\times 3} & 0_{3\times 3} \\ 0_{3\times 3} & \widehat{S}_f & 0_{3\times 3} \\ 0_{3\times 3} & 0_{3\times 3} & 0_{3\times 3} \end{bmatrix}, \quad \widehat{S}_f = \begin{bmatrix} 1.2 & 0 & 0 \\ 0 & 1.9 & 0 \\ 0 & 0 & 2.6 \end{bmatrix}, \tag{53}$$

$$C_{0,2}(t) = \begin{bmatrix} 0_{3\times 3} & 0_{3\times 3} & 0_{3\times 3} \\ 0_{3\times 3} & \widehat{C}_{0,2}(t) & 0_{3\times 3} \\ 0_{3\times 3} & 0_{3\times 3} & 0_{3\times 3} \end{bmatrix}, \quad \widehat{C}_{0,2}(t) = \begin{bmatrix} 0.5 & 0 & 0 \\ 0 & 1.1 & 0 \\ 0 & 0 & 1.9 \end{bmatrix},$$

$$C_2(\mathbf{x},t) = \frac{1}{2} C_{2,2}(t) : \mathbf{x}\mathbf{x}^\top = \begin{bmatrix} 1.3E4(M_1 n_1(t))^2 & 0 & 0 \\ 0 & 0.8E4(M_2 n_2(t))^2 & 0 \\ 0 & 0 & 0.6E4(M_3 n_3(t))^2 \end{bmatrix}.$$

The form, $\mathbf{u}^\top C_2(\mathbf{x},t)\mathbf{u}$, makes the performance index a highly nonlinear quartic in the state–control variables.

The state space for this manufacturing example is a hybrid space, since the n_i is the number of workstations, it must be discrete, while the a_i and c_i are integrals of functions of the control, u_i, so they must be continuous state variables. The number of workstations are specified by the discrete set, $n_i = 1$ to N_i, for stages $i = 1$ to 3. The surplus aggregate levels, a_i, are discretized into $NA_i = 3$ nodes in $[-1, 1]$ for $i = 1$ to 3. The production rates, c_i, are discretized into $NC_i = 3$ nodes in $[0.79, 0.81]$ for $i = 1$ to 3. Thus the global state space node count is given by $[NX_i]_{9\times1} = [[N_i]_{3\times1}, [3]_{3\times1}, [3]_{3\times1}]^T$.

The coefficients for least squares approximation are based on a hybrid of discrete and continuous values based on the properties of the state vectors $\mathbf{n}(t)$, $\mathbf{a}(t)$ and $\mathbf{c}(t)$. Therefore the general formulation of Section 2.1., needs to be modified by a hybrid of discrete sum and continuous integral averages:

$$\overline{F}(t) \equiv \sum_{j_1=1}^{N_1}\sum_{j_2=1}^{N_2}\sum_{j_3=1}^{N_3}\int_{\mathcal{D}_a}\int_{\mathcal{D}_c} F(\mathbf{n}_{j_1,j_2,j_3}, \mathbf{a}, \mathbf{c}, t)d\mathbf{a}d\mathbf{c}/(|\mathcal{D}_n|\cdot|\mathcal{D}_a|\cdot|\mathcal{D}_c|)$$

for some given integrable function $F(\mathbf{n}, \mathbf{a}, \mathbf{c}, t)$, where $|\mathcal{D}_n| = N_1N_2N_3$ is the total number of workstation combinations, $|\mathcal{D}_a| = \int_{\mathcal{D}_a} d\mathbf{a}$ is the size of the $\mathbf{a}$–space and $|\mathcal{D}_c| = \int_{\mathcal{D}_c} d\mathbf{c}$ is the size of the $\mathbf{c}$–space.

Then nonzero coefficients are given by

$$\mathbf{F}_{0,0}^{(0)} = \begin{bmatrix} \mathbf{0}_{3\times1} \\ \mathbf{F}_{0,0}^{\mathbf{a}} \\ \mathbf{0}_{3\times1} \end{bmatrix}, \qquad \mathbf{F}_{0,0}^{\mathbf{a}} = \begin{bmatrix} -472.2 \\ -455.4 \\ -461.0 \end{bmatrix},$$

$$F_{0,1}^{(0)} = \begin{bmatrix} 0_{3\times3} & 0_{3\times3} & 0_{3\times3} \\ F_{0,1}^{\mathbf{n}} & 0_{3\times3} & F_{0,1}^{\mathbf{c}} \\ 0_{3\times3} & 0_{3\times3} & 0_{3\times3}, \end{bmatrix}, \quad F_{0,1}^{\mathbf{n}} = \begin{bmatrix} 93.6 & 0 & 0 \\ 0 & 56.8 & 0 \\ 0 & 0 & 70.4 \end{bmatrix}, \quad F_{0,1}^{\mathbf{c}} = \begin{bmatrix} 234 & 0 & 0 \\ 0 & 213 & 0 \\ 0 & 0 & 220 \end{bmatrix},$$

$$F_{1,0}^{(0)} = \begin{bmatrix} \mathbf{0}_{3\times3} \\ F_{0,1}^{\mathbf{c}} \\ \mathbf{I}_{3\times3} \end{bmatrix}, \qquad C_{2,0}^{(0)} = \begin{bmatrix} 8.305E8 & 0 & 0 \\ 0 & 4.436E8 & 0 \\ 0 & 0 & 3.485E8 \end{bmatrix},$$

where $C_{0,2}^{(0)} = C_{0,2}$ and $H_{0,0}^{(0)} = H_0$. The moments for the amplitude mark density for the repair and failure processes are based on the average number of workstations keeping the spirit of the LSA are given by

$$\overline{\mathbf{Z}}^F = \begin{bmatrix} 1.000 \\ 1.050 \\ 1.025 \end{bmatrix}, \quad \overline{\mathbf{Z}^2}^F = \begin{bmatrix} 0.0000 \\ 0.0475 \\ 0.0237 \end{bmatrix}, \quad \overline{\mathbf{Z}}^R = \begin{bmatrix} 1.000 \\ 1.070 \\ 1.015 \end{bmatrix}, \quad \overline{\mathbf{Z}^2}^R = \begin{bmatrix} 0.0000 \\ 0.0657 \\ 0.1455 \end{bmatrix}.$$

The above coefficients are used to generate the system of equations (36), (37), and (38) which was solved by the *LSODE* programme of Hindmarsh (1980).

The system outputs are illustrated in figures 1 and 2 for the second stage with [3,4,*] operational workstations, i.e. there are $n_1 = 3$ operational workstations in stage 1, $n_2 = 4$ in stage 2 and the notation $n_3 = *$ means that the number of operational workstations in stage 3 does not matter at stage 2. In figure 3 for the third stage with $[n_1, n_2, n_3] = [3, 4, 4]$ operational workstations. In figure 1, the regular and constrained controlled production rates, exhibit the anticipation of workstation repair and failure, whereas the maximal, virtual production rate does not. The constrained controlled production rate saturates at 100% of capacity, since the unconstrained, regular controlled production rate exceeds 100% of capacity which is not physically realizable. In figure 2, the estimated expected value for the surplus aggregate level, $\hat{a}_2(t, t_f)$, becomes negative since the anticipated demand is not being met. In figure 3, the regular and constrained controlled production rates, exhibit the anticipation of workstation failure only since all of the workstations are operational, and the constrained controlled production rate exhibits the production constraints due to process limitations i.e. not enough pieces are coming through stage 2.

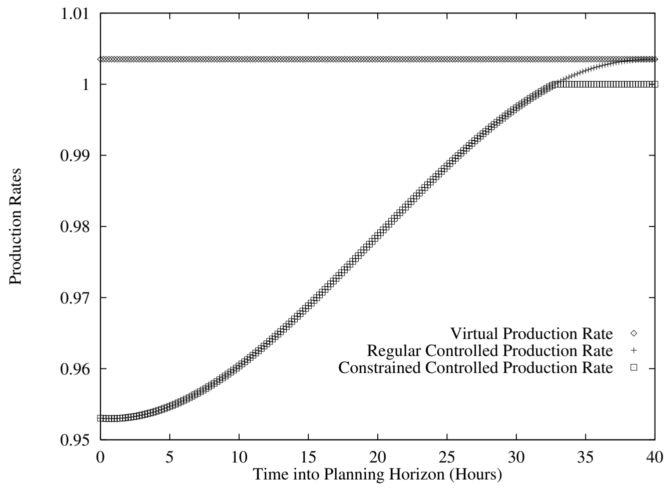

Figure 1: Virtual, regular and constrained controlled production rates $c_2(t)$, for stage 2 with $[n_1, n_2, n_3] = [3, 4, *]$ active workstations, '$*$' denoting any number of workstations.

The production rates illustrated in figures 1 and 3 are used by the plant manager to set the rates of production for the MMS. When an *event*, the repair or failure of a workstation, occurs the production rates need to be adjusted to reflect the change in the state of system. The production rates remain constant between events.

However, the production rates illustrated in figures 1 and 3 are based on the backward PDE for stochastic dynamic programming and do not account for the forward path realizations. These production rates assume that there is no surplus from the previous stage, that is all of the pieces from the previous stage are consumed. This will not be the case if the production rate saturates to full utility (1.00) because of workstation failure. This saturation is seen in figure 1, the regular controlled production rate indicates that stage 1 is producing enough pieces for stage 2 to meet its desired goal, but the constrained controlled production rate saturates to 1.00. In this case, when the failed workstation in stage 2 is repaired (repair event), the production rates will be adjusted, however this adjustment needs to account for pieces produced by stage 1 that were not processed by the rest of the MMS due to the diminished production of stage 2, this is referred to as a production bottleneck. The production rates for stages 2 and 3 need to be adjusted in this case.

The plant manager needs to decide the way in which these pieces will be processed by the remaining stages of the MMS. This decision can be fairly arbitrary, and depends on corporate

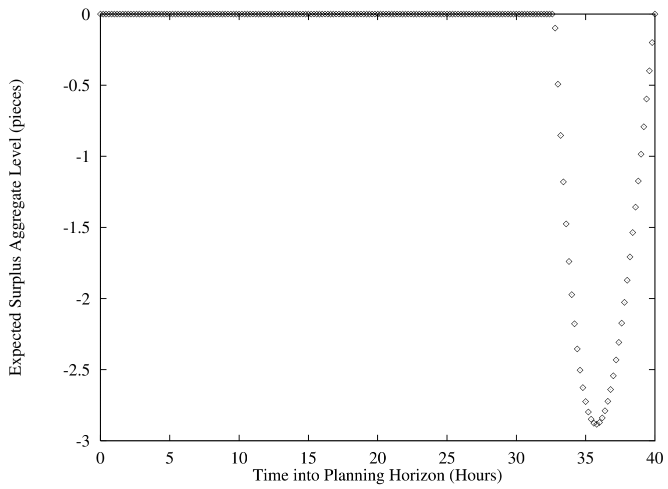

Figure 2: Estimated expected value for the surplus aggregate level, $\hat{a}_2(t, t_f)$, for stage 2 with $[n_1, n_2, n_3] = [3, 4, *]$ active workstations, '$*$' denoting any number of workstations.

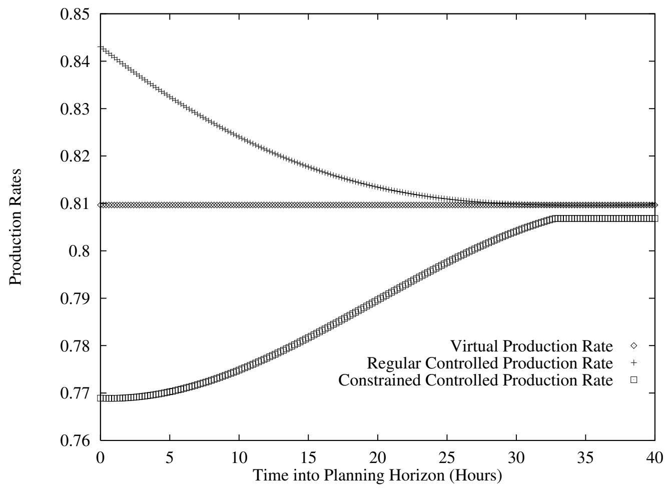

Figure 3: Virtual, regular and constrained controlled production rates $c_3(t)$, for stage 3 with $[n_1, n_2, n_3] = [3, 4, 4]$ active workstations.

planning and philosophy. Three possible choices would be to process the surplus pieces over a fixed time interval, for example one hour, over the remaining manufacturing horizon, or in the least amount of time. Note, that the adjustments begin at the production bottleneck stage and will effect all of the remaining stages of the MMS, this concept of surplus pieces does not apply to the first stage.

This model anticipates the repair and failure of workstations which will compensate for future events. Therefore, additional pieces may or may not be produced to compensate for workstation events, this may also lead to surplus pieces for various stages that should be processed after a workstation event occurs.

Figure 4 is a sample path realization for the workstation configuration in *forward time*. This sample path has workstation failures (failure events) and repairs (repair events) for all stages. Thus the production rates for all stages need to be adjusted for these events. The consumption of surplus pieces is to be completed in an hour after a repair event, the resulting production rates for the three stages is in figure 5. The relative error of the MMS production is given in figure 6, where positive values denote surplus and negative values denote shortfall of production. At the end of the manufacturing horizon, stage 3 has the greatest percent relative error of -0.0103% which corresponds to a shortfall of production by 1.174 (or 2) pieces. Notice that for the most part the percent relative error is positive or nearly zero, except when failure events occur, therefore the production demand is being satisfied. Thus, this approach to production scheduling is very good.

The computational and memory requirements for these calculations are quite large. The computational requirements apparently grow with memory requirements. The discrete space for the current example with (3,4,5) workstations initially available in the stages (1,2,3), each with 3 surplus aggregate levels and 3 production rates, leads to a basic array size of $3*4*5*3^3*3^3 = 43,740$ elements. There are 5 such arrays for several, past and present, value functions as well as for the control switching function, and 3 other larger arrays each with 3 additional components for the regular control and associated derivatives arrays, so that the total number of basic large array elements is $(5+3*3)*43,740 = 612,360$ elements. The use of quadruple precision at 16 bytes per element is critical and leads to a total of 9.3MB (MB denotes megabyte storage units) just for the basic large arrays. The code for this was run on the National Scalable Cluster Project IBM SP-2 (TM) Scalable Parallel Processor Cluster at the University of Pennsylvania via ATM switch there and in Chicago. The total elapsed computing time for the given example was 3532.61 seconds, and the maximum memory use was 17MB, which includes work arrays in addition to the above mentioned basic large arrays. The AIX F90 (TM) compiler was used since it allowed the use of larger number of subscripts than other compilers, and this was very useful for this highly multidimensional problem. Parallel execution scheduling was handled automatically by the Load Share Facility (LSF, TM), with a standard limit of 4 processors out of 20 per SP-2 in the cluster, but these computations were done at times when there was only one user on the system and the fastest nodes were targeted. An addition of one more workstation at stage 1, for example, would lead to $4/3*100 = 33.3\%$ more memory and a corresponding amount of more computational effort, with similar increases for additional workstations at other stages.

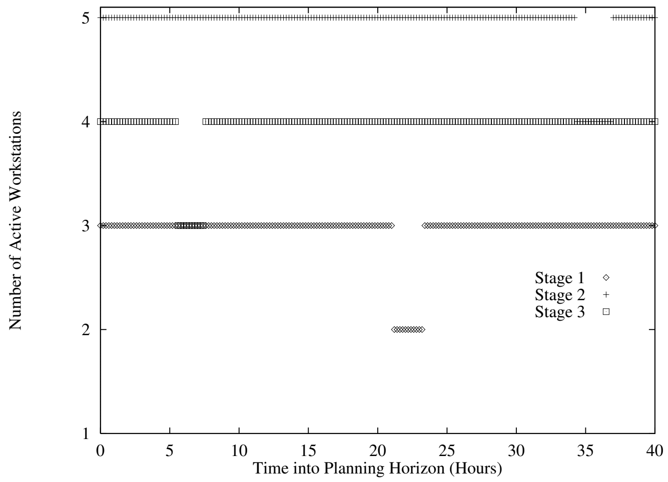

Figure 4: Sample path realization for workstation configuration.

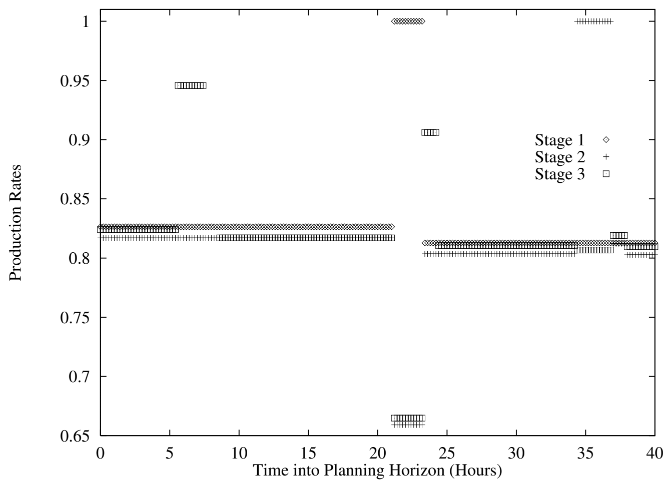

Figure 5: Adjusted constrained controlled production rates for the MMS.

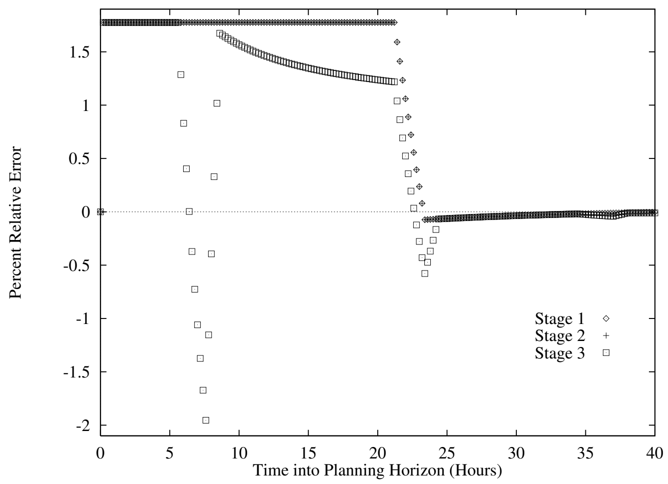

Figure 6: Relative percent error for adjusted constrained controlled production rates.

4. Conclusions

These results are of great interest because the LQGP/U problem is a *nonlinear* theoretical model and a canonical computational model. The Poisson components permit the model to account for a wide variety of large random fluctuations, whereas Gaussian noise is useful for relatively less severe background fluctuations. A new approach for accelerating the iterative convergence of solutions of the nonlinear partial differential equation of stochastic dynamic programming is presented using the method least squares to convert nonlinear control problems with linear control dynamics and quadratic control costs to a genuine LQGP problem in the state. The iterates following from this approach share with the genuine LQGP problem the state independence property of the coefficient equations and so do not suffer from the usual curse of dimensionality due to the computational complexity of searching the whole discretized state space. In addition, a generalized Gauss–Statistics quadrature is developed to integrate the Poisson functional integrals. These new modifications have led to new approaches to augment the usual linear algebraic forms.

The stochastic differential equation formulation used here makes the construction of models analogous to that for other dynamic systems. These features allow for greater realism without having to compromise aspects of physical reality than does the corresponding linear model. This problem formulation would be useful for investigating the response of systems subject to extreme random inputs or catastrophic failure. The multistage manufacturing system (MMS) application is a good test of our methods, since it is a nonlinear model with hybrid Poisson modeled catastrophic workstations failures.

Acknowledgments

Work supported in part by the National Science Foundation Grants DMS–96–26692, DMS–99–73231, CDA–94–13948, and CDA–96–01861. The authors would like to thank the National Scalable Cluster Project at the University of Illinois at Chicago for computational

support.

References

Akella, R., and Kumar, P. R., 1986, Optimal control of production rate in a failure prone manufacturing system. *I.E.E.E. Trans. Autom. Control*, **31**, 116–126.

Anderson, B. D. O., and Moore, J. B., 1990, *Optimal Control: Linear Quadratic Methods* (Englewood Cliffs, NJ : Prentice Hall).

Bellman, R. E., 1957, *Dynamic Programming* (Princeton, NJ : Princeton University Press).

Bielecki, T., and Kumar, P. R., 1988, Optimality of zero–inventory policies for unreliable manufacturing systems. *Operations Research*, **36**, 532–541.

Boukas, E. K., and Haurie, A., 1990, Manufacturing flow control and preventive maintenance: A stochastic control approach. *I.E.E.E. Trans. Autom. Control*, **35**, 1024–1031.

Boukas, E. K., and Yang, H., 1996, Optimal control of manufacturing flow and preventive maintenance. *I.E.E.E. Trans. Autom. Control*, **41**, 881–885.

Bryson, A. E., and Ho, Y., 1975, *Applied Optimal Control* (Waltham: Ginn).

Dorato, P., Abdallah, C., and Cerone, V., 1995, *Linear–Quadratic Control: An Introduction* (Englewood Cliffs, NJ: Prentice–Hall).

Douglas, Jr. J., and Dupont T., 1970, Galerkin methods for parabolic equations. *SIAM J. Num. Anal.*, **7**, 575–626.

Feller, W., 1996, *An Introduction to Probability Theory and Its Applications*, **II** (New York: Wiley).

Gershwin, S. B., 1997, Design and operation of manufacturing systems – Control– and system–Theoretical models and issues. *Proceedings of the 1997 American Control Conference*, **3**, pp. 1901–1913.

Gihman, I. I., and Skorohod, A. V., 1972, *Stochastic Differential Equations* (New York: Springer–Verlag).

Hall, R. W., 1983, *Zero Inventories* (Homewood, Illinois: Dow Jones–Irwin).

Hanson, F. B., 1991, Computational dynamic programming on a vector multiprocessor. *I.E.E.E. Trans. Autom. Control*, **36**, 507–511.

Hanson, F. B., 1996, Techniques in computational stochastic dynamic programming. In C. T. Leondes (ed) *Digital and Control System Techniques and Applications, Control and Dynamic Systems: Advances in Theory and Applications*, Vol. 76 (New York: Academic Press), pp. 103–162.

Hanson, F. B., and Ryan, D., 1998, Optimal harvesting with both population and price dynamics. *Mathematical Biosciences*, **148**, 129–146.

Hanson, F. B., and Tuckwell, H. C., 1997, Population growth with randomly distributed jumps. *Journal of Mathematical Biology*, **36**, 169–187.

Hindmarsh, A. C., 1980, Lsode and Lsodi, two new initial value ordinary differential equation solvers. *ACM–SigNum Newsletter*, **15**, 10–11.

Kenne, J. P., and Boukas, E. K., 1997, Production and corrective maintenance planning problem of a failure prone manufacturing system. *Proceedings of the 1997 American Control Conference*, **2**, pp. 1013–1014.

Kimemia, J., and Gershwin, S. B., 1983, An algorithm for the computer control of a flexible manufacturing system. *I.I.E. Trans.*, **15**, 353–362.

Kleinrock, L., 1975, *Queueing Systems: Theory*, **1** (New York: Wiley).

Kushner, H. J., and Dupuis, P. G., 1992, *Numerical Methods for Stochastic Control Problems in Continuous Time* (New York: Springer–Verlag).

Lewis, F. L., 1986, *Optimal Estimation with an Introduction to Stochastic Control Theory* (New York: Wiley).

Naimipour, K., and Hanson, F. B., 1993, Numerical convergence for the Bellman equation of stochastic optimal control with quadratic costs and constraints. *Dynamics and Control*, **3**, 237–259.

Stengel, R. F., 1986, *Stochastic Optimal Control* (New York: Wiley).

Westman, J. J., and Hanson, F. B., 1997, The LQGP problem: A manufacturing application. *Proceedings of the 1997 American Control Conference*, **1**, pp. 566–570.

Wonham, W. M., 1970, Random differential equations in control. In A. T. Bharucha–Reid (ed) *Probabilistic Methods in Applied Mathematics*, Vol. 2 (New York: Academic Press).