A Broader View of Brownian Network Models

Mike Harrison
June, 2000

Primary References


General Form of a Brownian Network Model

(1) $Z(t) = Z(0) + X(t) + \theta t + RY(t)$ \hspace{1cm} state dynamics

(2) $U(t) = KY(t) + \beta t$ \hspace{1cm} cumulative "idleness"

(3) $U(\cdot)$ is $\uparrow$ with $U(0) = 0$ \hspace{1cm} capacity constraints

(4) $Z(t) \in S$ (a convex set) \hspace{1cm} state-space constraints

(5) $\xi(t) = \int_0^t h(Z(s)) ds + v \cdot Y(t)$ \hspace{1cm} cumulative "cost"

where $X = \{X(t), t \geq 0\}$ is $(0, \Gamma)$ Brownian motion.

The system manager must choose a control $Y$, non-anticipating with respect to $X$, that satisfies (3) and (4). Generally speaking, lower costs $\xi(\cdot)$ are more desirable. For a complete formulation, more must be said about that, of course.
About the Model Class

• "Brownian networks" are applicable as models of *dynamic resource allocation* in a wide range of economic and technological systems.

• The virtues of this model class include *elegance* (minimality), *generality*, and analytical *tractability*.

• Brownian networks are appropriate as system models when the ambient mode of operation is characterized by *balanced, high-volume flows*.

Goals of Current Research

• *Generalize the definition of a Brownian network*, both for increased modeling flexibility and for greater parallelism between conventional and Brownian models.

• *Generalize the notion of "heavy traffic"* that is currently used to justify Brownian models. The more general view involves explicit consideration of costs and revenues from the outset.

• *Extend the analysis of Brownian networks to consider problems of optimal system design* (or optimal parameter selection), justifying the formulation by means of a formal limit.
Three Elements of a Stochastic Processing Network

- resources (servers)
- stocks (units of flow)
- processing activities

A Parallel-Server Example

\[ \lambda_1^* = 1.1 \]
\[ \lambda_2^* = 0.8 \]
\[ \mu_1 = 1 \]
\[ \mu_2 = 1 \]
\[ \mu_3 = 1/2 \]
\[ \mu_4 = 3 \]
\[ b_1^* = 1 \]
\[ b_2^* = 1 \]
First-Order Data for the Parallel-Server System

\[
R = \begin{bmatrix}
1 & \frac{1}{2} \\
1 & 3
\end{bmatrix}
\]

*input-output matrix*

\[
A = \begin{bmatrix}
1 \\
1 \\
1
\end{bmatrix}
\]

*capacity consumption matrix*

\[
v = \begin{bmatrix}
1 \\
1 \\
1 \\
\frac{1}{2}
\end{bmatrix}
\]

*value rate vector*

**Static Planning Problem for the Parallel-Server System**

\[
\text{maximize } \quad v \cdot x
\]

Subject to

\[
Rx = \lambda^*
\]

\[
Ax \leq b^*
\]

\[
x \geq 0
\]

\[
x^* = [1, \frac{8}{10}, \frac{2}{10}, 0]
\]

*nominal processing plan*

\((A1)\) The static planning problem has a unique solution \(x^*\), and moreover \(Ax^* = b^*\).
Dynamic Control Problem
for the Parallel-Server System

\( x(t) \) is the vector of actual activity rates at time \( t \),

\[ T(t) = \int_0^t x(s) \, ds \] is vector of cumulative activity levels,

\( E(t) \) is vector of cumulative external flows up to time \( t \), and

\( F_j(t) \) is vector of cumulative flows resulting from the first \( t \) units of activity \( j \).

So

\[ Z(t) = Z(0) + E(t) - \sum_{j=1}^4 F_j(T_j(t)) \quad \text{buffer contents process} \]

The control \( x(\cdot) \) or \( T(\cdot) \) must be suitably non-anticipating, and must
furthermore be chosen to satisfy the state-space constraint

\[ Z(t) \geq 0 \quad \text{for all } t \geq 0. \]

The processes \( E(\cdot) \) and \( F_j(\cdot) \) are the only stochastic model elements, and
they are taken as primitive. Think of these as a collection of independent
Poisson processes.
Given a holding cost function $H(z)$, we define the cumulative net value process

$$V(t) = \int_0^t [v \cdot x(s) - H(Z(s))] \, ds$$

$$= v \cdot T(t) - \int_0^t H(Z(s)) \, ds.$$ 

Now define the vector process of deviation controls

$$Y(t) = x^*t - T(t).$$

Then one has

$$V(t) = (v \cdot x^*) \, t - \xi(t),$$

where

$$\xi(t) = \int_0^t H(Z(s)) \, ds + v \cdot Y(t).$$

**Objective.** Find a control $Y$ to minimize $\mathcal{E}\{ \int_0^\infty e^{-pt} d\xi(t) \}.$
Consider a family of models parameterized by $r > 0$, each identical to the model described above except that

\begin{align*}
(A2) \quad \rho^r &= \alpha/r^2 , \\
(A3) \quad H^r(z) &= \frac{1}{r} h\left(\frac{1}{r} z\right) , \\
(A4) \quad \lambda^r &= \lambda^* + \frac{1}{r} \theta \\
\quad \text{and} \quad \\
(A5) \quad b^r &= b^* + \frac{1}{r} \beta .
\end{align*}

Finally, defining the nominal netflow process

\[
X^*(t) = E(t) - \sum_{j=1}^{4} F_j(x_j^* t) ,
\]

it is assumed that

\begin{align*}
(A6) \quad X^r &\Rightarrow X \quad \text{as} \quad r \to \infty ,
\end{align*}

where $X$ is a $(0, \Gamma)$ Brownian motion and

\[
X^r(t) = \frac{1}{r} X^*(r^2 t) .
\]
Brownian Network Model for the Parallel-Server System

As before, let $X$ be $(0, \Gamma)$ Brownian motion. To form $K$ from $A$ we must add one row for each non-basic activity:

$$K = \begin{bmatrix}
1 & 1 \\
1 & 1 \\
1 & -1 \\
\end{bmatrix}.$$

The system manager chooses a control $Y$, non-anticipating with respect to $X$, to minimize

$$\min \mathbb{E} \left\{ \int_0^\infty e^{-\alpha t} d\xi(t) \right\},$$

where

(1) $Z(t) = X(t) + \theta t + RY(t)$,

(2) $U(t) = KY(t) + \beta t$,

(3) $U(\cdot)$ is non-decreasing with $U(0) \geq 0$,

(4) $Z(t) \geq 0$, and

(5) $\xi(t) = \int_0^t h(Z(s)) ds + v \cdot Y(t)$.
Solution of the Brownian Control Problem for the Parallel-Server System

One can write out the solution of this problem explicitly, because its "equi-
valent workload formulation" is one-dimensional. The appropriate definition of
workload for this example is

\[ W(t) = 2Z_1(t) + Z_2(t) , \quad t \geq 0. \]

One aspect of the solution is the following: the control \( U_3 = -Y_4 \) is used to
impose an upper reflecting barrier at level \( w^* \) on the workload process \( W \).
Here \( w^* > 0 \) is a constant of order 1, easily computed from problem data,
including the covariance matrix \( \Gamma \).

Interpretation of the Brownian Solution

Consider the original system model with large parameter \( r \). When one reverses
the scaling involved in the formal Brownian limit, one obtains the following
policy description (again, only one aspect of the policy is described):

- server 1 should process jobs from buffer 1 (this is the basic activity 1)
  whenever (unscaled) workload \( \leq rw^* \)

- server 1 should process jobs from buffer 2 (this is the non-basic activity 4)
  whenever (unscaled) workload > \( rw^* \)

- the fraction of time spent using activity 4 is of order \( 1/r \)

- but this sparing use is enough to effectively confine (unscaled) system
  workload to the interval \( [0, rw^*] \)
To determine whether a Brownian model is appropriate, one must consider both physical and economic data.

- In my parallel-server example there exists a non-basic activity which, if introduced into the nominal processing plan, would allow strictly more inputs to be processed. Thus we have an imbalance between exogenous input rates and server capabilities.

  But a Brownian model is still appropriate, because introducing that activity is economically undesirable, except in small amounts during the dynamic control phase.

Recall that in terms of original (unscaled) units of measurement, the "optimal policy" derived from our Brownian model allows workload to vary within the interval \([0, \text{rw}^*]\).

- The assumption that \( H'(z) = \frac{1}{r} h\left(\frac{1}{r} z\right) \) is crucial. This means that backlogs of order \( r \) produce holding cost rates of order \( \frac{1}{r} \). If backlogs of order \( r \) produce holding cost rates of a higher order than that, then the system manager will typically find such backlogs intolerable (for example, steady use of activity 4 to maintain backlogs of order 1 may be justified). In such cases a Brownian model is inappropriate.
Expanded Version of the Static Planning Problem
(Parallel-Server Example)

Now suppose that $\lambda$ and $b$ are under the system manager’s control, but they must be fixed at $t = 0$ (they are design parameters). This choice generates costs at a fixed rate $c(\lambda, b)$ ever afterward. The expanded planning problem is to choose $\lambda$, $b$ and $x$ so as to

$$\text{maximize} \quad \pi = v \cdot x - c(\lambda, b)$$

subject to

$$Rx = \lambda$$
$$Ax \leq b$$
$$x \geq 0.$$ 

$(A1)$ This static planning problem has a unique solution $(\lambda^*, b^*, x^*)$, and moreover $Ax^* = b^*$. Hereafter let

$$\pi^* = v \cdot x^* - c(\lambda^*, b^*).$$

Also let $(f, g) = \nabla c(\lambda^*, b^*)$, so that one has

$$c(\lambda^* + \varepsilon\theta, b^* + \varepsilon\beta) = c(\lambda^*, b^*) + (f \cdot \theta + g \cdot \beta) \varepsilon + o(\varepsilon)$$

for small $\varepsilon$. 
Design and Control of the Brownian Network  
(Parallel-Server Example)

The system manager chooses parameters $\theta$ and $\beta$ at $t = 0$, then chooses controls $Y(t)$ dynamically given state observations. Cumulative cost incurred up to time $t$ is

$$\eta(t) = (f \cdot \theta + g \cdot \beta) t + \int_{0}^{t} h(Z(s)) \, ds + \nu \cdot Y(t).$$

Interpretation in the Original Model Context  
(Parallel-Server Example)

If, in the original model with large parameter $r$, the system manager chooses $\lambda = \lambda^* + \frac{1}{r} \theta$ and $b = b^* + \frac{1}{r} \beta$, then cumulative net gain up to time $r^2 t$ is approximated by

$$r^2 \pi^* t - r \eta(t).$$

To refine the naïve design choice ($\lambda^*, b^*$), we choose vectors $\theta$ and $b$ to optimize the tradeoff embodied in the second (lower order) term.

Stochastic variability leads to congestion costs of order $r$ over time spans of order $r^2$. Consideration of those costs may motivate "unbalancing" the design parameters $\lambda$ and $b$, relative to the nominal values $\lambda^*$ and $b^*$, but only adjustments of order $\frac{1}{r}$ are worthy of consideration.
Three Stages of System Management

- Using only first-order system data, determine nominal design parameters $\lambda^*$ and $b^*$, and a nominal processing plan $x^*$, that maximize the higher-order performance rate $\pi^*$.

- Given arbitrary perturbations $\theta$ and $\beta$ from nominal design parameters, determine dynamic control policy to minimize lower-order performance degradation due to stochastic variability.

- Choose $\theta$ and $\beta$ to optimize the tradeoff between costs associated with system design and performance degradation due to stochastic variability.

An Issue not Touched Upon

"Interpretation" of the optimal policy for the Brownian control problem may be quite subtle. That is, constructing a parametric family of policies that achieve optimal performance in the limit as $r \to \infty$ is a subject unto itself.