Bermudan options by simulation

Chris Rogers
Statistical Laboratory, University of Cambridge

July 15, 2015

Optimal stopping problems

Stop Markov process X_t at time τ and get reward $g(\tau, X_\tau)$.

Optimal stopping problems

Stop Markov process X_t at time τ and get reward $g(\tau, X_\tau)$. We want

$$\sup_{0 \le \tau \le T} E[\, g(\tau, X_\tau)\,].$$

Optimal stopping problems

Stop Markov process X_t at time τ and get reward $g(\tau, X_\tau)$. We want

$$\sup_{0 \leq \tau \leq T} E[\,g(\tau, X_\tau)\,].$$

Find the value function

Optimal stopping problems

Stop Markov process X_t at time τ and get reward $g(\tau, X_\tau)$. We want

$$\sup_{0 \leq \tau \leq T} E[\, g(\tau, X_\tau)\,].$$

Find the value function

$$V(t, x) \quad \equiv \quad \sup_{t \leq \tau \leq T} E[\, g(\tau, X_\tau) \mid X_t = x\,]$$

Optimal stopping problems

Stop Markov process X_t at time τ and get reward $g(\tau, X_\tau)$. We want

$$\sup_{0\leq\tau\leq T} E[\, g(\tau, X_\tau)\,].$$

Find the value function

$$\begin{aligned} V(t,x) &\equiv \sup_{t\leq\tau\leq T} E[\, g(\tau, X_\tau) \mid X_t = x\,] \\ &= \max\{\, g(t,x), E[\, V(t+1, X_{t+1}) \mid X_t = x\,]\, \} \end{aligned}$$

Optimal stopping problems

Stop Markov process X_t at time τ and get reward $g(\tau, X_\tau)$. We want

$$\sup_{0 \le \tau \le T} E[\, g(\tau, X_\tau)\,].$$

Find the value function

$$\begin{aligned} V(t,x) &\equiv \sup_{t \le \tau \le T} E[\, g(\tau, X_\tau) \mid X_t = x\,] \\ &= \max\{\, g(t,x), E[\, V(t+1, X_{t+1}) \mid X_t = x\,]\, \} \end{aligned}$$

by backward induction from $V(T, \cdot) = g(T, \cdot)$.

Optimal stopping problems

Stop Markov process X_t at time τ and get reward $g(\tau, X_\tau)$. We want

$$\sup_{0 \le \tau \le T} E[\, g(\tau, X_\tau)\,].$$

Find the value function

$$\begin{aligned} V(t,x) &\equiv \sup_{t \le \tau \le T} E[\, g(\tau, X_\tau) \mid X_t = x \,] \\ &= \max\{\, g(t,x), E[\, V(t+1, X_{t+1}) \mid X_t = x \,] \,\} \end{aligned}$$

by backward induction from $V(T, \cdot) = g(T, \cdot)$.

But what if the statespace of X is high-dimensional?

Optimal stopping problems

Stop Markov process X_t at time τ and get reward $g(\tau, X_\tau)$. We want

$$\sup_{0 \le \tau \le T} E[\,g(\tau, X_\tau)\,].$$

Find the value function

$$\begin{aligned} V(t,x) &\equiv \sup_{t \le \tau \le T} E[\,g(\tau, X_\tau) \mid X_t = x\,] \\ &= \max\{\,g(t,x), E[\,V(t+1, X_{t+1}) \mid X_t = x\,]\,\} \end{aligned}$$

by backward induction from $V(T,\cdot) = g(T,\cdot)$.

But what if the statespace of X is high-dimensional?

How do we calculate the expectations?

Optimal stopping problems

Stop Markov process X_t at time τ and get reward $g(\tau, X_\tau)$. We want

$$\sup_{0 \le \tau \le T} E[\, g(\tau, X_\tau)\,].$$

Find the value function

$$\begin{aligned} V(t,x) &\equiv \sup_{t \le \tau \le T} E[\, g(\tau, X_\tau) \mid X_t = x\,] \\ &= \max\{\, g(t,x), E[\, V(t+1, X_{t+1}) \mid X_t = x\,]\, \} \end{aligned}$$

by backward induction from $V(T, \cdot) = g(T, \cdot)$.

But what if the statespace of X is high-dimensional?

How do we calculate the expectations?

How do we store the numerically-computed value function?

Examples.

Examples.

Consider a d-vector of log-Brownian assets $S_t^i = \exp(x_t^i)$, where

$$x_t = \sigma W_t - \mu t,$$

taking $\mu^i = r - \frac{1}{2}(\sigma\sigma^T)_{ii}$.

Examples.

Consider a d-vector of log-Brownian assets $S_t^i = \exp(x_t^i)$, where

$$x_t = \sigma W_t - \mu t,$$

taking $\mu^i = r - \frac{1}{2}(\sigma\sigma^T)_{ii}$.

Example: min put. Reward function is

$$g(t,x) = e^{-rt}(\ K - \exp(\min_i x^i)\ )^+.$$

Examples.

Consider a d-vector of log-Brownian assets $S^i_t = \exp(x^i_t)$, where

$$x_t = \sigma W_t - \mu t,$$

taking $\mu^i = r - \frac{1}{2}(\sigma\sigma^T)_{ii}$.

Example: min put. Reward function is

$$g(t,x) = e^{-rt}(\, K - \exp(\min_i x^i)\,)^+.$$

Example: fixed window lookback. Here, take $d = 1$ and the reward function is

$$g(t,X) = \sup_{t-a \le u \le t} S_u,$$

Examples.

Consider a d-vector of log-Brownian assets $S_t^i = \exp(x_t^i)$, where

$$x_t = \sigma W_t - \mu t,$$

taking $\mu^i = r - \frac{1}{2}(\sigma\sigma^T)_{ii}$.

Example: min put. Reward function is

$$g(t,x) = e^{-rt}(\ K - \exp(\min_i x^i)\ )^+.$$

Example: fixed window lookback. Here, take $d = 1$ and the reward function is

$$g(t,X) = \sup_{t-a \le u \le t} S_u,$$

where $a \in \mathbb{N}$.

Examples.

Consider a d-vector of log-Brownian assets $S_t^i = \exp(x_t^i)$, where

$$x_t = \sigma W_t - \mu t,$$

taking $\mu^i = r - \frac{1}{2}(\sigma\sigma^T)_{ii}$.

Example: min put. Reward function is

$$g(t,x) = e^{-rt}(\,K - \exp(\min_i x^i)\,)^+.$$

Example: fixed window lookback. Here, take $d = 1$ and the reward function is

$$g(t,X) = \sup_{t-a \le u \le t} S_u,$$

where $a \in \mathbb{N}$. State variable is $X_t = (x_{t-a}, x_{t-a+1}, \ldots, x_t)$, of dimension $a + 1$.

Observations.

Observations.

- Pricing a European option in $d = 50$ can only be done by simulation -

Observations.

- Pricing a European option in $d = 50$ can only be done by simulation - so pricing Bermudan/American options can only be done by simulation!

Observations.

- Pricing a European option in $d = 50$ can only be done by simulation - so pricing Bermudan/American options can only be done by simulation!
- We cannot have the value function -

Observations.

- Pricing a European option in $d = 50$ can only be done by simulation - so pricing Bermudan/American options can only be done by simulation!
- We cannot have the value function -
  - Can we calculate it?

Observations.

- Pricing a European option in $d = 50$ can only be done by simulation - so pricing Bermudan/American options can only be done by simulation!
- We cannot have the value function -
  - Can we calculate it?
  - Can we store it?

Observations.

- Pricing a European option in $d = 50$ can only be done by simulation - so pricing Bermudan/American options can only be done by simulation!
- We cannot have the value function -
  - Can we calculate it?
  - Can we store it?
  - Can we access it?

Observations.

- Pricing a European option in $d = 50$ can only be done by simulation - so pricing Bermudan/American options can only be done by simulation!
- We cannot have the value function -
  - Can we calculate it?
  - Can we store it?
  - Can we access it?
- What do we *need*?

Observations.

- Pricing a European option in $d = 50$ can only be done by simulation - so pricing Bermudan/American options can only be done by simulation!
- We cannot have the value function -
  - Can we calculate it?
  - Can we store it?
  - Can we access it?
- What do we *need*?

1. At any time, we need to be able to decide stop/continue

Observations.

- Pricing a European option in $d = 50$ can only be done by simulation - so pricing Bermudan/American options can only be done by simulation!
- We cannot have the value function -
  - Can we calculate it?
  - Can we store it?
  - Can we access it?
- What do we *need*?
  1. At any time, we need to be able to decide stop/continue
  2. At any time, need to be able to bound the price

Observations.

- Pricing a European option in $d = 50$ can only be done by simulation - so pricing Bermudan/American options can only be done by simulation!
- We cannot have the value function -
  - Can we calculate it?
  - Can we store it?
  - Can we access it?
- What do we *need*?
  1. At any time, we need to be able to decide stop/continue
  2. At any time, need to be able to bound the price
  3. At any time, need to provide a hedging rule

Observations.

- Pricing a European option in $d = 50$ can only be done by simulation - so pricing Bermudan/American options can only be done by simulation!
- We cannot have the value function -
  - Can we calculate it?
  - Can we store it?
  - Can we access it?
- What do we *need*?
  1. At any time, we need to be able to decide stop/continue
  2. At any time, need to be able to bound the price
  3. At any time, need to provide a hedging rule
- The value function is *sufficient*, but not necessary -

Observations.

- Pricing a European option in $d = 50$ can only be done by simulation - so pricing Bermudan/American options can only be done by simulation!
- We cannot have the value function -
  - Can we calculate it?
  - Can we store it?
  - Can we access it?
- What do we *need*?
  1. At any time, we need to be able to decide stop/continue
  2. At any time, need to be able to bound the price
  3. At any time, need to provide a hedging rule
- The value function is *sufficient*, but not necessary - less may be enough!

Observations.

- Pricing a European option in $d = 50$ can only be done by simulation - so pricing Bermudan/American options can only be done by simulation!
- We cannot have the value function -
  - Can we calculate it?
  - Can we store it?
  - Can we access it?
- What do we *need*?
  1. At any time, we need to be able to decide stop/continue
  2. At any time, need to be able to bound the price
  3. At any time, need to provide a hedging rule
- The value function is *sufficient*, but not necessary - less may be enough!

Simple approximations are often good enough to give 1, 2, 3.

Markov coercion (Barraquand & Martineau 1995).

Markov coercion (Barraquand & Martineau 1995).

The ONLY thing we need for X is to be able to simulate steps.

Markov coercion (Barraquand & Martineau 1995).

The ONLY thing we need for X is to be able to simulate steps.

The process $Z_t \equiv g(t, X_t)$ is *not* Markovian

Markov coercion (Barraquand & Martineau 1995).

The ONLY thing we need for X is to be able to simulate steps.

The process $Z_t \equiv g(t, X_t)$ is *not* Markovian - but we pretend that it is!

Markov coercion (Barraquand & Martineau 1995).

The ONLY thing we need for X is to be able to simulate steps.

The process $Z_t \equiv g(t, X_t)$ is *not* Markovian - but we pretend that it is!
For each $t \in \mathbb{T} \equiv \{0, 1, \ldots, T\}$ take partition

$$B_t^{(1)} = (-\infty, y_t^{(1)}] < \ldots < B_t^{(N)} = (y_t^{(N-1)}, \infty)$$

Markov coercion (Barraquand & Martineau 1995).

The ONLY thing we need for X is to be able to simulate steps.

The process $Z_t \equiv g(t, X_t)$ is *not* Markovian - but we pretend that it is!
For each $t \in \mathbb{T} \equiv \{0, 1, \ldots, T\}$ take partition

$$B_t^{(1)} = (-\infty, y_t^{(1)}] \;<\; \ldots \;<\; B_t^{(N)} = (y_t^{(N-1)}, \infty)$$

and values $\eta_j^{(k)} \in B_t^{(k)}$,

Markov coercion (Barraquand & Martineau 1995).

The ONLY thing we need for X is to be able to simulate steps.

The process $Z_t \equiv g(t, X_t)$ is *not* Markovian - but we pretend that it is!
For each $t \in \mathbb{T} \equiv \{0, 1, \ldots, T\}$ take partition

$$B_t^{(1)} = (-\infty, y_t^{(1)}] < \ldots < B_t^{(N)} = (y_t^{(N-1)}, \infty)$$

and values $\eta_j^{(k)} \in B_t^{(k)}$, and approximate Z_t by

$$Y_t = \sum_{k=1}^{N} \eta_t^{(k)} \; I_{\{g(t,X_t) \in B_t^{(k)}\}} \; .$$

Markov coercion (Barraquand & Martineau 1995).

The ONLY thing we need for X is to be able to simulate steps.

The process $Z_t \equiv g(t, X_t)$ is *not* Markovian - but we pretend that it is!
For each $t \in \mathbb{T} \equiv \{0, 1, \dots, T\}$ take partition

$$B_t^{(1)} = (-\infty, y_t^{(1)}] < \dots < B_t^{(N)} = (y_t^{(N-1)}, \infty)$$

and values $\eta_j^{(k)} \in B_t^{(k)}$, and approximate Z_t by

$$Y_t = \sum_{k=1}^{N} \eta_t^{(k)} \, I_{\{g(t,X_t) \in B_t^{(k)}\}} \, .$$

Now simulate paths $X^{(i)}$, $i = 1, \dots, N_{\mathrm{sim}}$,

Markov coercion (Barraquand & Martineau 1995).

The ONLY thing we need for X is to be able to simulate steps.

The process $Z_t \equiv g(t, X_t)$ is *not* Markovian - but we pretend that it is!
For each $t \in \mathbb{T} \equiv \{0, 1, \ldots, T\}$ take partition

$$B_t^{(1)} = (-\infty, y_t^{(1)}] \; < \ldots < \; B_t^{(N)} = (y_t^{(N-1)}, \infty)$$

and values $\eta_j^{(k)} \in B_t^{(k)}$, and approximate Z_t by

$$Y_t = \sum_{k=1}^{N} \eta_t^{(k)} \; I_{\{g(t,X_t) \in B_t^{(k)}\}} \; .$$

Now simulate paths $X^{(i)}$, $i = 1, \ldots, N_{\text{sim}}$, and postulate Markov dynamics:

$$P\left[\; Y_{t+1} = \eta_{t+1}^{(k)} \mid Y_t = \eta_t^{(j)} \; \right] = \frac{\text{Number of paths in } B_t^{(j)}, \text{ then in } B_{t+1}^{(k)}}{\text{Number of paths in } B_t^{(j)}}$$

Continuing the pretence.

Now have discrete-time Markovian proxy taking values in a finite set of size N;

Continuing the pretence.

Now have discrete-time Markovian proxy taking values in a finite set of size N; trivial to compute value $\tilde{V}$ and stopping rule $\tilde{\mathcal{S}}$ for this!

The stopping rule says

$$\text{At time } t\text{, stop iff } g(t, X_t) \in \tilde{\mathcal{S}}_t$$

Continuing the pretence.

Now have discrete-time Markovian proxy taking values in a finite set of size N; trivial to compute value $\tilde{V}$ and stopping rule $\tilde{\mathcal{S}}$ for this!

The stopping rule says

$$\text{At time } t, \text{ stop iff } g(t, X_t) \in \tilde{\mathcal{S}}_t$$

- and we can evaluate this by simulation.

Continuing the pretence.

Now have discrete-time Markovian proxy taking values in a finite set of size N; trivial to compute value $\tilde{V}$ and stopping rule $\tilde{\mathcal{S}}$ for this!

The stopping rule says

$$\text{At time } t\text{, stop iff } g(t, X_t) \in \tilde{\mathcal{S}}_t$$

- and we can evaluate this by simulation. This gives a lower bound for the price of the option,

Continuing the pretence.

Now have discrete-time Markovian proxy taking values in a finite set of size N; trivial to compute value $\tilde{V}$ and stopping rule $\tilde{\mathcal{S}}$ for this!

The stopping rule says

$$\text{At time } t \text{, stop iff } g(t, X_t) \in \tilde{\mathcal{S}}_t$$

- and we can evaluate this by simulation. This gives a lower bound for the price of the option, and a simple stopping rule.

Continuing the pretence.

Now have discrete-time Markovian proxy taking values in a finite set of size N; trivial to compute value $\tilde{V}$ and stopping rule $\tilde{S}$ for this!

The stopping rule says

$$\text{At time } t, \text{ stop iff } g(t, X_t) \in \tilde{\mathcal{S}}_t$$

- and we can evaluate this by simulation. This gives a lower bound for the price of the option, and a simple stopping rule.

What about an upper bound for the price?

Continuing the pretence.

Now have discrete-time Markovian proxy taking values in a finite set of size N; trivial to compute value $\tilde{V}$ and stopping rule $\tilde{\mathcal{S}}$ for this!

The stopping rule says

$$\text{At time } t, \text{ stop iff } g(t, X_t) \in \tilde{\mathcal{S}}_t$$

- and we can evaluate this by simulation. This gives a lower bound for the price of the option, and a simple stopping rule.

What about an upper bound for the price? Use the dual characterization of the value (R. (2002), Haugh & Kogan (2004)):

$$\sup_{0 \le \tau \le T} E[\, g(\tau, X_\tau)\,] = \min_{M \in \mathcal{M}_0} E[\, \sup_{0 \le t \le T} (g(t, X_t) - M_t)\,],$$

Continuing the pretence.

Now have discrete-time Markovian proxy taking values in a finite set of size N; trivial to compute value $\tilde{V}$ and stopping rule $\tilde{\mathcal{S}}$ for this!

The stopping rule says

$$\text{At time } t\text{, stop iff } g(t, X_t) \in \tilde{\mathcal{S}}_t$$

- and we can evaluate this by simulation. This gives a lower bound for the price of the option, and a simple stopping rule.

What about an upper bound for the price? Use the dual characterization of the value (R. (2002), Haugh & Kogan (2004)):

$$\sup_{0 \le \tau \le T} E[\, g(\tau, X_\tau)\,] = \min_{M \in \mathcal{M}_0} E[\, \sup_{0 \le t \le T} (g(t, X_t) - M_t)\,],$$

achieved when M is the martingale of the Snell envelope process

Continuing the pretence.

Now have discrete-time Markovian proxy taking values in a finite set of size N; trivial to compute value $\tilde{V}$ and stopping rule $\tilde{\mathcal{S}}$ for this!

The stopping rule says

$$\text{At time } t, \text{ stop iff } g(t, X_t) \in \tilde{\mathcal{S}}_t$$

- and we can evaluate this by simulation. This gives a lower bound for the price of the option, and a simple stopping rule.

What about an upper bound for the price? Use the dual characterization of the value (R. (2002), Haugh & Kogan (2004)):

$$\sup_{0 \leq \tau \leq T} E[\, g(\tau, X_\tau)\,] = \min_{M \in \mathcal{M}_0} E[\, \sup_{0 \leq t \leq T} (g(t, X_t) - M_t)\,],$$

achieved when M is the martingale of the Snell envelope process - which here is simply $V(t, X_t)$.

Upper bounds.

Upper bounds.

If we knew V, the optimal martingale difference sequence would be

$$\Delta M_{t+1}^* \equiv M_{t+1}^* - M_t^* = V(t+1, X_{t+1}) - E[\ V(t+1, X_{t+1}) \mid X_t\ ].$$

Upper bounds.

If we knew V, the optimal martingale difference sequence would be

$$\Delta M^*_{t+1} \equiv M^*_{t+1} - M^*_t = V(t+1, X_{t+1}) - E[\ V(t+1, X_{t+1}) \mid X_t\ ].$$

We don't know V, but we *do* know $\tilde{V}$, so this suggests we use the martingale difference sequence

$$\Delta M_{t+1} = \tilde{V}(t+1, X_{t+1}) - E[\ \tilde{V}(t+1, X_{t+1}) \mid X_t\ ].$$

Upper bounds.

If we knew V, the optimal martingale difference sequence would be

$$\Delta M^*_{t+1} \equiv M^*_{t+1} - M^*_t = V(t+1, X_{t+1}) - E[\ V(t+1, X_{t+1}) \mid X_t\ ].$$

We don't know V, but we *do* know $\tilde{V}$, so this suggests we use the martingale difference sequence

$$\Delta M_{t+1} = \tilde{V}(t+1, X_{t+1}) - E[\ \tilde{V}(t+1, X_{t+1}) \mid X_t\ ].$$

When we simulate, we know $\tilde{V}(t+1, X_{t+1})$, but $E[\ \tilde{V}(t+1, X_{t+1}) \mid X_t\ ]$ is not known directly.

Upper bounds.

If we knew V, the optimal martingale difference sequence would be

$$\Delta M^*_{t+1} \equiv M^*_{t+1} - M^*_t = V(t+1, X_{t+1}) - E[\ V(t+1, X_{t+1}) \mid X_t\ ].$$

We don't know V, but we *do* know $\tilde{V}$, so this suggests we use the martingale difference sequence

$$\Delta M_{t+1} = \tilde{V}(t+1, X_{t+1}) - E[\ \tilde{V}(t+1, X_{t+1}) \mid X_t\ ].$$

When we simulate, we know $\tilde{V}(t+1, X_{t+1})$, but $E[\ \tilde{V}(t+1, X_{t+1}) \mid X_t\ ]$ is not known directly.

We estimate this by simulating N_{sub} possible continuations of X_t;

Upper bounds.

If we knew V, the optimal martingale difference sequence would be

$$\Delta M^*_{t+1} \equiv M^*_{t+1} - M^*_t = V(t+1, X_{t+1}) - E[\ V(t+1, X_{t+1}) \mid X_t\ ].$$

We don't know V, but we *do* know $\tilde{V}$, so this suggests we use the martingale difference sequence

$$\Delta M_{t+1} = \tilde{V}(t+1, X_{t+1}) - E[\ \tilde{V}(t+1, X_{t+1}) \mid X_t\ ].$$

When we simulate, we know $\tilde{V}(t+1, X_{t+1})$, but $E[\ \tilde{V}(t+1, X_{t+1}) \mid X_t\ ]$ is not known directly.

We estimate this by simulating $N_{\rm sub}$ possible continuations of X_t; generally 'simulations within simulations' are slow,

Upper bounds.

If we knew V, the optimal martingale difference sequence would be

$$\Delta M^*_{t+1} \equiv M^*_{t+1} - M^*_t = V(t+1, X_{t+1}) - E[\; V(t+1, X_{t+1}) \mid X_t \;].$$

We don't know V, but we *do* know $\tilde{V}$, so this suggests we use the martingale difference sequence

$$\Delta M_{t+1} = \tilde{V}(t+1, X_{t+1}) - E[\; \tilde{V}(t+1, X_{t+1}) \mid X_t \;].$$

When we simulate, we know $\tilde{V}(t+1, X_{t+1})$, but $E[\; \tilde{V}(t+1, X_{t+1}) \mid X_t \;]$ is not known directly.

We estimate this by simulating $N_{\rm sub}$ possible continuations of X_t; generally 'simulations within simulations' are slow, but here we are only searching a one-dimensional space of values

Upper bounds.

If we knew V, the optimal martingale difference sequence would be

$$\Delta M^*_{t+1} \equiv M^*_{t+1} - M^*_t = V(t+1, X_{t+1}) - E[\ V(t+1, X_{t+1}) \mid X_t\ ].$$

We don't know V, but we *do* know $\tilde{V}$, so this suggests we use the martingale difference sequence

$$\Delta M_{t+1} = \tilde{V}(t+1, X_{t+1}) - E[\ \tilde{V}(t+1, X_{t+1}) \mid X_t\ ].$$

When we simulate, we know $\tilde{V}(t+1, X_{t+1})$, but $E[\ \tilde{V}(t+1, X_{t+1}) \mid X_t\ ]$ is not known directly.

We estimate this by simulating $N_{\rm sub}$ possible continuations of X_t; generally 'simulations within simulations' are slow, but here we are only searching a one-dimensional space of values so we mainly used $N_{\rm sub} = 25$.

Upper bounds.

If we knew V, the optimal martingale difference sequence would be

$$\Delta M^*_{t+1} \equiv M^*_{t+1} - M^*_t = V(t+1, X_{t+1}) - E[\ V(t+1, X_{t+1}) \mid X_t\ ].$$

We don't know V, but we *do* know $\tilde{V}$, so this suggests we use the martingale difference sequence

$$\Delta M_{t+1} = \tilde{V}(t+1, X_{t+1}) - E[\ \tilde{V}(t+1, X_{t+1}) \mid X_t\ ].$$

When we simulate, we know $\tilde{V}(t+1, X_{t+1})$, but $E[\ \tilde{V}(t+1, X_{t+1}) \mid X_t\ ]$ is not known directly.

We estimate this by simulating N_{sub} possible continuations of X_t; generally 'simulations within simulations' are slow, but here we are only searching a one-dimensional space of values so we mainly used $N_{\mathrm{sub}} = 25$.

How close are the upper and lower bounds?

Example: min put.

Example: min put.

d	low	MC price	high	gap(%)	time
2	24.77	25.16	26.63	6.98	5.00
3	31.26	31.76	33.00	5.40	5.74
4	35.57	36.28	37.48	5.08	6.82
5	39.11	39.47	40.22	2.74	7.66
10	47.95	48.33	49.06	2.26	12.42
15	51.97	52.14	52.86	1.68	17.34
30	57.66	-	58.23	0.98	35.86
60	62.10	-	62.62	0.82	68.32

Table : Min put prices. The d assets are independent, $S^i(0) = 100$. Other parameters are $K = 100$, $r = 0.06$, $T = 0.5$, $\sigma_{ii} = 0.6$. MC price is from R. (2002).

Example: max call.

Example: max call.

Stopping reward is

$$g(t, X_t) = e^{-rt}\Big(\exp\big(\max_{1\le i\le d} x_t^i\big) - K\Big)^+.$$

Example: max call.

Stopping reward is

$$g(t, X_t) = e^{-rt}\left(\exp(\max_{1\le i\le d} x_t^i) \; - K \right)^+.$$

m	S_0	low	BG price	high	gap(%)	time
3	90	15.29	16.006	16.11	5.08	4.21
	100	24.41	25.284	25.35	3.71	4.20
	110	34.74	35.695	35.84	3.07	4.20
6	90	15.52	16.474	16.79	7.59	8.47
	100	24.63	25.92	26.50	7.04	8.57
	110	34.81	36.497	37.23	6.48	8.57
9	90	15.81	16.659	16.95	6.71	12.67
	100	24.78	26.158	26.75	7.35	12.82
	110	35.25	36.782	37.63	6.30	12.66

Table : Max call prices on 5 independent assets with common volatility $\sigma = 0.2\%$ and expiry $T = 3$. There are $m = 3, 6, 9$ exercise opportunities at times iT/m, $i = 0, \ldots, m$. Other parameters are $K = 100$, $r = 0.05$, $\delta = 0.1$. BG price from Broadie & Glasserman 2004.

Example: basket put.

Stopping reward is

$$g(t, X_t) = e^{-rt}\left(K - d^{-1}\sum_{i=1}^{d} \exp(x_t^i) \right)^+ .$$

d	low	KLM	high	gap(%)	time
2	3.05	3.14	3.14	6.11	27.20
3	2.74	2.94	2.94	6.80	36.66
4	2.64	2.84	2.84	7.03	45.56
5	2.57	2.77	2.77	7.24	53.85
6	2.54	2.72	2.72	6.73	63.01
12	2.43	-	2.61	7.07	131.8

Table : Basket put. All assets start at 100, $K = 100$, $T = 0.25$, $r = 0.05$. All assets have volatility 20%, and the correlation between assets is 0.5. Here, $N_{\text{sub}} = 125$. KLM prices from Kovalov, Linetsky & Marcozzi (2007).

Example: Fixed window lookback.

Stopping reward is

$$g(t, X) = \sup_{t-a \leq u \leq t} S_u.$$

a	low	high	gap(%)	time
5	116.07	116.74	0.57	115
10	121.8	123.2	1.18	109
15	125.2	127.2	1.57	110
20	127.7	130.1	1.84	114
25	129.6	132.4	2.17	115

Table : Fixed window lookback option. Parameters were $T = 1$, $\sigma = 0.5$, $r = 0.05$, $S_0 = 100$, and the time interval was divided into 250 equal time steps. The calculations were done in the numeraire of the discounted asset price.

Min puts with stochastic volatility and interest.

Min puts with stochastic volatility and interest.

'Market' Brownian motion W^M;

Min puts with stochastic volatility and interest.

'Market' Brownian motion W^M; log prices $x_t^i = \log S_t^i$ evolve as

$$dx_t^i = \sigma_t^i \{ \rho_S \, dW_t^M + \rho_S' \, dW_t^{S,i} \} + (r_t - \tfrac{1}{2} (\sigma_t^i)^2) \, dt, \qquad i = 1, \ldots, d$$

Min puts with stochastic volatility and interest.

'Market' Brownian motion W^M; log prices $x_t^i = \log S_t^i$ evolve as

$$dx_t^i = \sigma_t^i \{ \rho_S \, dW_t^M + \rho_S' \, dW_t^{S,i} \} + (r_t - \tfrac{1}{2} (\sigma_t^i)^2) \, dt, \qquad i = 1, \dots, d$$

and OU processes ξ and z (in dimensions d, 1, respectively):

$$\begin{aligned} d\xi_t &= -\beta_\xi \, \xi_t \, dt + \sigma_\xi \, (\rho_\xi \, dW_t^M + \rho_\xi' \, dW_t^\xi) \\ dz_t &= -\beta_r \, dt + \sigma_r \, (\rho_r \, dW_t^M + \rho_r' \, dW_t^r) \end{aligned}$$

Min puts with stochastic volatility and interest.

'Market' Brownian motion W^M; log prices $x_t^i = \log S_t^i$ evolve as

$$dx_t^i = \sigma_t^i \{ \rho_S \, dW_t^M + \rho_S' \, dW_t^{S,i} \} + (r_t - \tfrac{1}{2} (\sigma_t^i)^2) \, dt, \qquad i = 1, \ldots, d$$

and OU processes ξ and z (in dimensions d, 1, respectively):

$$\begin{aligned} d\xi_t &= -\beta_\xi \, \xi_t \, dt + \sigma_\xi \, (\rho_\xi \, dW_t^M + \rho_\xi' \, dW_t^\xi) \\ dz_t &= -\beta_r \, dt + \sigma_r \, (\rho_r \, dW_t^M + \rho_r' \, dW_t^r) \end{aligned}$$

in terms of which

$$\sigma_t^i = \bar{\sigma}^i \exp(\xi_t^i), \qquad r_t = \bar{r} \exp(z_t).$$

Min puts with stochastic volatility and interest.

'Market' Brownian motion W^M; log prices $x_t^i = \log S_t^i$ evolve as

$$dx_t^i = \sigma_t^i \{ \rho_S \, dW_t^M + \rho_S' \, dW_t^{S,i} \} + (r_t - \tfrac{1}{2} (\sigma_t^i)^2) \, dt, \qquad i = 1, \ldots, d$$

and OU processes ξ and z (in dimensions d, 1, respectively):

$$\begin{aligned} d\xi_t &= -\beta_\xi \, \xi_t \, dt + \sigma_\xi \, (\rho_\xi \, dW_t^M + \rho_\xi' \, dW_t^\xi) \\ dz_t &= -\beta_r \, dt + \sigma_r \, (\rho_r \, dW_t^M + \rho_r' \, dW_t^r) \end{aligned}$$

in terms of which

$$\sigma_t^i = \bar{\sigma}^i \exp(\xi_t^i), \qquad r_t = \bar{r} \exp(z_t).$$

The riskless rate is a Black-Karasinski model;

Min puts with stochastic volatility and interest.

'Market' Brownian motion W^M; log prices $x_t^i = \log S_t^i$ evolve as

$$dx_t^i = \sigma_t^i \{ \rho_S \, dW_t^M + \rho_S' \, dW_t^{S,i} \} + (r_t - \tfrac{1}{2} (\sigma_t^i)^2) \, dt, \qquad i = 1, \ldots, d$$

and OU processes ξ and z (in dimensions d, 1, respectively):

$$\begin{aligned} d\xi_t &= -\beta_\xi \, \xi_t \, dt + \sigma_\xi \, (\rho_\xi \, dW_t^M + \rho_\xi' \, dW_t^\xi) \\ dz_t &= -\beta_r \, dt + \sigma_r \, (\rho_r \, dW_t^M + \rho_r' \, dW_t^r) \end{aligned}$$

in terms of which

$$\sigma_t^i = \bar{\sigma}^i \exp(\xi_t^i), \qquad r_t = \bar{r} \exp(z_t).$$

The riskless rate is a Black-Karasinski model; the vol is similar.

Constant σ, r, varying ρ_S.

ρ_S	low	high	gap (%)	time
-0.15	38.75	40.27	3.77	27
0.00	39.06	40.65	3.92	28
0.15	38.61	40.11	3.73	27
0.30	37.47	38.99	3.89	27
0.45	35.58	37.26	4.52	27
0.60	32.98	34.58	4.61	27

Table : Prices of min puts as ρ_S varies. Volatility is constant at $\bar{\sigma} = 0.6$, interest is constant at $r = 0.06$. 40 timesteps are used, $N_{\text{bins}} = 200$, $b = 50$. To establish the lower bound, 50000 paths are used, and for the upper bound, 4000 paths are used with $N_{\text{sub}} = 25$ representatives in each subsimulation.

Constant σ, varying r.

Constant σ, varying r.

r_0	low	high	gap (%)	time
0.00	40.31	41.74	3.43	27
0.025	39.10	40.59	3.66	27
0.06	37.47	39.17	4.35	27
0.10	35.63	37.52	5.01	27

Table : Prices of min puts as r_0 varies. Volatility is constant at $\bar{\sigma} = 0.6$, correlation between assets is $\rho_S = 0.3$. Parameters are $\rho_r = 0.3$, $\bar{r} = 0.06$, $\beta_r = 0.02$, $\sigma_r = 0.12$. 40 timesteps are used, $N_{\text{bins}} = 200$, $b = 50$. To establish the lower bound, 50000 paths are used, and for the upper bound, 4000 paths are used with $N_{\text{sub}} = 25$ representatives in each subsimulation.

Constant r, varying σ.

Constant r, varying σ.

σ_0	low	high	gap (%)	time
0.10	22.09	23.10	4.36	26
0.20	26.56	27.74	4.25	26
0.30	29.88	31.22	4.29	27
0.40	32.74	34.11	4.03	27
0.50	35.20	36.79	4.32	27
0.60	37.46	39.12	4.23	27

Table : Prices of min puts as σ_0 varies. Interest is constant at $\bar{r} = 0.06$, correlation between assets is $\rho_S = 0.3$. Parameters are $\rho_\xi = 0.3$, $\bar{\sigma}^i = 0.6$ for all i, $\beta_\xi = 4.5$, $\sigma_\xi = 0.3$, 40 timesteps are used, $N_{\rm bins} = 200$, $b = 50$. To establish the lower bound, 50000 paths are used, and for the upper bound, 4000 paths are used with $N_{\rm sub} = 25$ representatives in each subsimulation.

Both r and σ varying.

Both r and σ varying.

r_0	σ_0	low	high	gap (%)	time
0.00	0.2	29.41	30.50	3.59	26
	0.4	35.53	36.76	3.35	27
	0.6	40.35	41.44	2.62	26
0.06	0.2	26.58	27.74	4.21	26
	0.4	32.66	34.10	4.23	27
	0.6	37.58	39.08	3.84	27

Table : Prices of min puts as σ_0 and r_0 vary. Parameters are $\rho_S = \rho_\xi = \rho_r = 0.3$, $\bar{\sigma}^i = 0.6$ for all i, $\bar{r} = 0.06$, $\beta_\xi = 4.5$, $\sigma_\xi = 0.3$, $\beta_r = 0.02$, $\sigma_r = 0.12$. 40 timesteps are used, $N_{\mathrm{bins}} = 200$, $b = 50$. To establish the lower bound, 50000 paths are used, and for the upper bound, 4000 paths are used with $N_{\mathrm{sub}} = 25$ representatives in each subsimulation.

Accuracy.

Accuracy.

With Monte Carlo, expect 2 sig fig, perhaps 3;

Accuracy.

With Monte Carlo, expect 2 sig fig, perhaps 3; not *basis point* accuracy.

Accuracy.

With Monte Carlo, expect 2 sig fig, perhaps 3; not *basis point* accuracy.
Here we get at best 2 sig fig

Accuracy.

With Monte Carlo, expect 2 sig fig, perhaps 3; not *basis point* accuracy.
Here we get at best 2 sig fig - errors more 3-6% than 1-2%.

Accuracy.

With Monte Carlo, expect 2 sig fig, perhaps 3; not *basis point* accuracy. Here we get at best 2 sig fig - errors more 3-6% than 1-2%. Errors from

1. Markovian assumption;

Accuracy.

With Monte Carlo, expect 2 sig fig, perhaps 3; not *basis point* accuracy. Here we get at best 2 sig fig - errors more 3-6% than 1-2%. Errors from

1. Markovian assumption;
2. discretization of g values;

Accuracy.

With Monte Carlo, expect 2 sig fig, perhaps 3; not *basis point* accuracy. Here we get at best 2 sig fig - errors more 3-6% than 1-2%. Errors from

1. Markovian assumption;
2. discretization of g values;
3. simulation error.

Accuracy.

With Monte Carlo, expect 2 sig fig, perhaps 3; not *basis point* accuracy. Here we get at best 2 sig fig - errors more 3-6% than 1-2%. Errors from

1. Markovian assumption;
2. discretization of g values;
3. simulation error.

Can reduce 2, 3, but not 1 ...

Accuracy.

With Monte Carlo, expect 2 sig fig, perhaps 3; not *basis point* accuracy. Here we get at best 2 sig fig - errors more 3-6% than 1-2%. Errors from

1. Markovian assumption;
2. discretization of g values;
3. simulation error.

Can reduce 2, 3, but not 1 ... But what errors come from parameter uncertainty?

Accuracy.

With Monte Carlo, expect 2 sig fig, perhaps 3; not *basis point* accuracy. Here we get at best 2 sig fig - errors more 3-6% than 1-2%. Errors from

1. Markovian assumption;
2. discretization of g values;
3. simulation error.

Can reduce 2, 3, but not 1 ... But what errors come from parameter uncertainty? Is the volatility *really* 20%?

Accuracy.

With Monte Carlo, expect 2 sig fig, perhaps 3; not *basis point* accuracy. Here we get at best 2 sig fig - errors more 3-6% than 1-2%. Errors from

1. Markovian assumption;
2. discretization of g values;
3. simulation error.

Can reduce 2, 3, but not 1 ... But what errors come from parameter uncertainty? Is the volatility *really* 20%? Might it not be 19%? Or 21%?

Accuracy.

With Monte Carlo, expect 2 sig fig, perhaps 3; not *basis point* accuracy. Here we get at best 2 sig fig - errors more 3-6% than 1-2%. Errors from

1. Markovian assumption;
2. discretization of g values;
3. simulation error.

Can reduce 2, 3, but not 1 ... But what errors come from parameter uncertainty? Is the volatility *really* 20%? Might it not be 19%? Or 21%?

d	low (19%)	high (19%)	low (20%)	high (20%)	low (21%)	high (21%)
2	2.77	2.97	3.05	3.14	3.10	3.32
3	2.58	2.75	2.74	2.94	2.90	3.10
4	2.49	2.66	2.64	2.84	2.79	3.01
5	2.43	2.62	2.57	2.77	2.73	2.92
6	2.39	2.54	2.54	2.72	2.70	2.88

Table : Basket put. Parameters as before, except the volatility parameter which takes values 19%, 20%, 21%.

Accuracy.

With Monte Carlo, expect 2 sig fig, perhaps 3; not *basis point* accuracy. Here we get at best 2 sig fig - errors more 3-6% than 1-2%. Errors from

1. Markovian assumption;
2. discretization of g values;
3. simulation error.

Can reduce 2, 3, but not 1 ... But what errors come from parameter uncertainty? Is the volatility *really* 20%? Might it not be 19%? Or 21%?

d	low (19%)	high (19%)	low (20%)	high (20%)	low (21%)	high (21%)
2	2.77	2.97	3.05	3.14	3.10	3.32
3	2.58	2.75	2.74	2.94	2.90	3.10
4	2.49	2.66	2.64	2.84	2.79	3.01
5	2.43	2.62	2.57	2.77	2.73	2.92
6	2.39	2.54	2.54	2.72	2.70	2.88

Table : Basket put. Parameters as before, except the volatility parameter which takes values 19%, 20%, 21%.

Why 1bp accuracy?

Accuracy.

With Monte Carlo, expect 2 sig fig, perhaps 3; not *basis point* accuracy. Here we get at best 2 sig fig - errors more 3-6% than 1-2%. Errors from

1. Markovian assumption;
2. discretization of g values;
3. simulation error.

Can reduce 2, 3, but not 1 ... But what errors come from parameter uncertainty? Is the volatility *really* 20%? Might it not be 19%? Or 21%?

d	low (19%)	high (19%)	low (20%)	high (20%)	low (21%)	high (21%)
2	2.77	2.97	3.05	3.14	3.10	3.32
3	2.58	2.75	2.74	2.94	2.90	3.10
4	2.49	2.66	2.64	2.84	2.79	3.01
5	2.43	2.62	2.57	2.77	2.73	2.92
6	2.39	2.54	2.54	2.72	2.70	2.88

Table : Basket put. Parameters as before, except the volatility parameter which takes values 19%, 20%, 21%.

Why 1bp accuracy? To delta-hedge ...

Fixed strike Bermudan-Asian.

Fixed strike Bermudan-Asian.

Stopping reward is

$$g(t, X_t) = e^{-rt}(A_t - K)^+,$$

where we define the average price

$$A_t = \frac{\int_{-\delta}^{t} S_u \, du}{t + \delta}.$$

Fixed strike Bermudan-Asian.

Stopping reward is

$$g(t, X_t) = e^{-rt}(A_t - K)^+,$$

where we define the average price

$$A_t = \frac{\int_{-\delta}^{t} S_u \, du}{t + \delta}.$$

There is also some initial lock-out time $t^* \geq 0$.

Fixed strike Bermudan-Asian.

Stopping reward is

$$g(t, X_t) = e^{-rt}(A_t - K)^+,$$

where we define the average price

$$A_t = \frac{\int_{-\delta}^{t} S_u \, du}{t + \delta}.$$

There is also some initial lock-out time $t^* \geq 0$. Notice that

$$Z_t \equiv e^{-rt}(A_t - K)$$

is differentiable with derivative

$$\dot{Z}_t = -rZ_t + \frac{e^{-rt}(S_t - A_t)}{t + \delta}$$

Fixed strike Bermudan-Asian.

Stopping reward is

$$g(t, X_t) = e^{-rt}(A_t - K)^+,$$

where we define the average price

$$A_t = \frac{\int_{-\delta}^{t} S_u \, du}{t + \delta}.$$

There is also some initial lock-out time $t^* \geq 0$. Notice that

$$Z_t \equiv e^{-rt}(A_t - K)$$

is differentiable with derivative

$$\dot{Z}_t = -rZ_t + \frac{e^{-rt}(S_t - A_t)}{t + \delta}$$

so you would *never* exercise when this is positive and $g > 0$.

Fixed strike Bermudan-Asian.

Stopping reward is

$$g(t, X_t) = e^{-rt}(A_t - K)^+,$$

where we define the average price

$$A_t = \frac{\int_{-\delta}^{t} S_u \, du}{t + \delta}.$$

There is also some initial lock-out time $t^* \geq 0$. Notice that

$$Z_t \equiv e^{-rt}(A_t - K)$$

is differentiable with derivative

$$\dot{Z}_t = -rZ_t + \frac{e^{-rt}(S_t - A_t)}{t + \delta}$$

so you would *never* exercise when this is positive and $g > 0$. A stopping rule that only looks at A_t will likely do rather badly here.

Fixed strike Bermudan-Asian results.

A_0	S_0	low	FD price	high	gap(%)	time
90	80	0.919	0.949	0.991	7.33	4.12
	90	3.20	3.267	3.35	4.46	4.15
	100	7.49	7.889	8.03	6.71	4.15
	110	13.68	14.538	14.89	8.16	4.12
	120	21.01	22.423	22.99	8.61	4.10
100	80	1.052	1.108	1.147	8.28	4.13
	90	3.53	3.710	3.82	7.52	4.14
	100	8.07	8.658	8.83	8.61	20
	110	14.95	15.717	16.22	7.83	20
	120	23.33	23.811	24.42	4.46	20
110	80	1.21	1.288	1.33	9.02	20
	90	3.87	4.136	4.35	11.03	11.21
	100	8.7	9.821	10.66	18.4	20
	110	15.76	17.399	18.34	14.1	20
	120	24.50	25.453	26.32	6.91	20

Table : Fixed strike Bermudan Asian call. Parameters are $\sigma = 0.2$, $K = 100$, $t^* = 0.25$, $\delta = 0.25$, $T = 2$. Finite difference price from Longstaff & Schwartz (2001).

Floating strike Bermudan-Asian.

Floating strike Bermudan-Asian.

Stopping reward is

$$g(t, X_t) = e^{-rt}(A_t - S_t)^+.$$

Floating strike Bermudan-Asian.

Stopping reward is

$$g(t, X_t) = e^{-rt}(A_t - S_t)^+.$$

S_0	low	high	gap(%)	time
80	10.60	10.86	2.50	8.5
90	8.33	8.67	3.98	8.40
100	7.14	7.50	4.88	8.51
110	6.51	6.85	5.00	8.81
120	6.16	6.57	6.26	8.32

Table : Floating strike Bermudan Asian call. Parameters are $\sigma = 0.2$, $A_0 = 100$, $t^* = 0.25$, $\delta = 0.25$, $T = 2$.

Summary.

Summary.

- Generic methodology for Bermudan options

Summary.

- Generic methodology for Bermudan options
- Only requirement is ability to simulate Markov process

Summary.

- Generic methodology for Bermudan options
- Only requirement is ability to simulate Markov process
- Computational cost is largely insensitive to dimension

Summary.

- Generic methodology for Bermudan options
- Only requirement is ability to simulate Markov process
- Computational cost is largely insensitive to dimension
- Gap between upper and lower bounds of the order 5-8% typically, comparable to parameter uncertainty

Summary.

- Generic methodology for Bermudan options
- Only requirement is ability to simulate Markov process
- Computational cost is largely insensitive to dimension
- Gap between upper and lower bounds of the order 5-8% typically, comparable to parameter uncertainty
- Stopping/hedging rules obtained are very simple

Summary.

- Generic methodology for Bermudan options
- Only requirement is ability to simulate Markov process
- Computational cost is largely insensitive to dimension
- Gap between upper and lower bounds of the order 5-8% typically, comparable to parameter uncertainty
- Stopping/hedging rules obtained are very simple
- Computational times rarely more than 10-20s

Summary.

- Generic methodology for Bermudan options
- Only requirement is ability to simulate Markov process
- Computational cost is largely insensitive to dimension
- Gap between upper and lower bounds of the order 5-8% typically, comparable to parameter uncertainty
- Stopping/hedging rules obtained are very simple
- Computational times rarely more than 10-20s