Pricing American options using martingale bases

Jérôme Lelong

Grenoble Alpes University

International Monte Carlo Conference, Paris – July 8, 2016
Outline

1. American Options
2. An optimization point of view
3. How to effectively solve the optimization problem
4. Numerical experiments
Framework (1)

Consider a $d'$—dimensional financial market driven by a $d$—dimensional Brownian motion $B$, with $d' \leq d$.

The discounted payoff process writes
\[
Z_t = e^{-\int_0^t r_s \, ds} \phi(S_t), \quad t \leq T.
\]
Assume $\mathbb{E} \left[ \sup_t Z_t^2 \right] < \infty$.

Consider an American option. Its time–$t$ discounted price is given by
\[
U_t = \text{esssup}_{\tau \in \mathcal{T}_t} \mathbb{E}[Z_\tau | \mathcal{F}_t]
\]
where $\mathcal{T}_t$ is the set of all $\mathcal{F}$— stopping times with values in $[t, T]$. 
Dual price (1)

The *Snell envelope* process \((U_t)_{0 \leq t \leq T}\) admits a Doob–Meyer decomposition

\[
U_t = U_0 + M_t^* - A_t^*.
\]

[Rogers, 2002]:

\[
U_0 = \inf_{M \in H_0^1} \mathbb{E} \left[ \sup_{0 \leq t \leq T} (Z_t - M_t) \right] = \mathbb{E} \left[ \sup_{0 \leq t \leq T} (Z_t - M_t^*) \right]
\]

- This problem admits more than a single solution.
- For any stopping time \(\tau\) smaller than the largest optimal strategy,

\[
U_0 = \inf_{M \in H_0^1} \mathbb{E} \left[ \sup_{\tau \leq t \leq T} (Z_t - M_t) \right] = \mathbb{E} \left[ \sup_{\tau \leq t \leq T} (Z_t - M_t^*) \right].
\]
Dual price (2)

- Some of the martingales $M$ attaining the infimum are surely optimal

$$U_0 = \sup_{0 \leq t \leq T} (Z_t - M_t) \quad a.s.$$  

- From [Schoenmakers et al., 2013], any martingale satisfying

$$\text{Var} \left( \sup_{0 \leq t \leq T} (Z_t - M_t) \right) = 0$$

is surely optimal.

- From [Jamshidian, 2007], for any optimal stopping time $\tau$ and any surely optimal optimal martingale $M$,

$$(M_t \wedge \tau)_t = (M^*_t \wedge \tau)_t.$$
Dual price (3)

With our square integrability assumption, we can rewrite the minimization problem as

$$U_0 = \inf_{X \in L^2(\Omega, \mathcal{F}_T, \mathbb{P})} \mathbb{E} \left[ \sup_{0 \leq t \leq T} (Z_t - \mathbb{E}[X|\mathcal{F}_t]) \right].$$

How to approximate $L^2(\Omega, \mathcal{F}_T, \mathbb{P})$ by a finite dimensional vector space in which conditional expectations are tractable in a closed form?
Wiener chaos expansion ($d = 1$)

Let $H_i$ be the $i$th Hermite polynomial defined by

$$H_0(x) = 1; \quad H_i(x) = (-1)^i e^{x^2/2} \frac{d^i}{dx^i}(e^{-x^2/2}), \text{ for } i \geq 1.$$  

- $H'_i = H_{i-1}$ with the convention $H_{-1} = 0$.
- If $X, Y \sim \mathcal{N}(0, 1)$ and form a Gaussian vector,

$$\mathbb{E}[H_i(X)H_j(Y)] = i! \left(\mathbb{E}[XY]\right)^i \mathbf{1}_{\{i=j\}}.$$
Truncated Wiener chaos expansion \((d = 1)\)

Take a regular grid \(0 = t_0 < t_1 < \cdots < t_n\) with step \(h\).

Define the truncated Wiener chaos space of order \(p\)

\[
\mathcal{H}_p = \text{span}\left\{ \prod_{i=1}^n H_{\alpha_i}(G_i) : \alpha \in \mathbb{N}^n, \|\alpha\|_1 = p \right\}
\]

with \(G_i = \frac{B_{t_i} - B_{t_{i-1}}}{\sqrt{h}}\).

For \(F \in L^2(\Omega, \mathcal{F}_T)\), we introduce the truncated chaos expansion of order \(p\)

\[
C_{p,n}(F) = \sum_{\alpha \in A_{p,n}} \lambda_{\alpha} \prod_{i \geq 1} H_{\alpha_i}(G_i)
\]

where \(A_{p,n} = \{ \alpha \in \mathbb{N}^n : \|\alpha\|_1 \leq p \}\) with \(\|\alpha\|_1 = \sum_{i \geq 0} \alpha_i\).

In the following we write,

\[
C_{p,n}(F) = \sum_{\alpha \in A_{p,n}} \lambda_{\alpha} \hat{H}_{\alpha}(G_1, \ldots, G_n)
\]
Key property of the truncated Wiener chaos expansion

For \( k \leq n \),

\[
\mathbb{E}[C_{p,n}(F) | \mathcal{F}_{t_k}] = \sum_{\alpha \in A_{p,n}^k} \lambda_\alpha \hat{H}_\alpha(G_1, \ldots, G_n)
\]

with

\[
A_{p,n}^k = \{ \alpha \in \mathbb{N}^n : \|\alpha\|_1 \leq p, \; \alpha_\ell = 0 \; \forall \ell > k \}.
\]

"Computing \( \mathbb{E}[\cdot | \mathcal{F}_{t_k}] \)" \( \Leftrightarrow \) "Dropping all non \( \mathcal{F}_{t_k} \) – measurable terms"
Extension to the multi–dimensional case

The truncated Wiener chaos of order $p \geq 0$ is given by

$$\left\{ \prod_{j=1}^{d} \hat{H}_{\alpha_j}(G_1^j, \ldots, G_n^j) : \alpha \in (\mathbb{N}^n)^d, \|\alpha\|_1 \leq p \right\}.$$

We introduce the truncated chaos expansion of order $p$ of $F \in L^2(\Omega, \mathcal{F}_T)$

$$C_{p,n}(F) = \sum_{\alpha \in A_{p,n}^{\otimes d}} \lambda_\alpha \hat{H}_{\alpha}^{\otimes d}(G_1, \ldots, G_n) = C_{p,n}(\lambda)$$

where

$$A_{p,n}^{\otimes d} = \left\{ \alpha \in (\mathbb{N}^n)^d : \|\alpha\|_1 \leq p \right\},$$

$$\hat{H}_{\alpha}^{\otimes d}(G_1, \ldots, G_n) = \prod_{j=1}^{d} \hat{H}_{\alpha_j}(G_1^j, \ldots, G_n^j) \quad \forall \alpha \in (\mathbb{N}^n)^d.$$
Return to the American option price

We approximate the original problem

\[
\inf_{X \in L^2(\Omega, \mathcal{F}_T, \mathbb{P})} \mathbb{E} \left[ \sup_{0 \leq t \leq T} (Z_t - \mathbb{E}[X|\mathcal{F}_t]) \right] \quad \text{s.t.} \quad \mathbb{E}[X] = 0
\]

by

\[
\inf_{\lambda \in \mathbb{R}^{A \otimes d}_{p,n}} V_{p,n}(\lambda) \quad \text{(1)}
\]

with

\[
V_{p,n}(\lambda) = \mathbb{E} \left[ \max_{0 \leq k \leq n} (Z_{t_k} - \mathbb{E}[C_{p,n}(\lambda)|\mathcal{F}_{t_k}]) \right].
\]
Properties of the minimization problem (1)

The minimization problem (1) has at least one solution.

- The function $V_{p,n}$ is clearly convex (maximum of affine functions).
- Not strongly convex but,

$$V_{p,n}(\lambda) \geq \frac{|\lambda|}{2} \inf_{\mu \in \mathbb{R}^{A_p,n} \otimes^d, |\mu| = 1} \mathbb{E} [ |C_{p,n}(\mu)| ] .$$
Properties of the minimization problem (2)

\[ \mathcal{I}(\lambda, Z, G) = \{ 0 \leq k \leq n : \text{the pathwise maximum is attained at time } k \} . \]

**Proposition 2**

Let \( p \geq 1 \). Assume that

\[ \forall 1 \leq r \leq k \leq n, \forall F \mathcal{F}_{tk} - \text{measurable}, \ F \in \mathcal{C}_{p-1,n}, \ F \neq 0, \]

\[ \exists 1 \leq q \leq d \text{ s.t.} \ \mathbb{P} \left( \forall t \in ]t_{r-1}, t_r], \ D^q_t Z_{tk} + F = 0 \mid Z_{tk} > 0 \right) = 0. \]

Then, the function \( V_{p,n} \) is differentiable at all points \( \lambda \in \mathbb{R}^{A_{p,n}} \) with no zero component and its gradient \( \nabla V_{p,n} \) is given by

\[ \nabla V_{p,n}(\lambda) = \mathbb{E} \left[ \mathbb{E} \left[ \hat{H}^{\otimes d}(G_1, \ldots, G_n) \mid \mathcal{F}_{t_i} \right] \right]_{i=\mathcal{I}(\lambda,Z,G)}. \]
Properties of the minimization problem (3)

- Differentiability is ensured as soon as \( \mathcal{I}(\lambda, Z, G) \) is a.s. reduced to a unique element: purpose of the blue condition.

- Alternative approach by [Belomestny, 2013]: use smoothing techniques instead (see [Nesterov, 2004]). General idea:

  \[
  \text{Replace } \max_k a_k \text{ by } p^{-1} \log \left( \sum_k \exp(p \cdot a_k) \right). 
  \]

- Let \( \lambda^\# \) be a solution, \( V_{p,n}(\lambda^\#_{p,n}) = \inf_\lambda V_{p,n}(\lambda) \). Then \( \nabla V_{p,n}(\lambda^\#_{p,n}) = 0 \).
Convergence to the true solution (1)

**Proposition 3**

The solution of the minimization problem (1), $V_{p,n}(\lambda_{p,n}^\#)$, converges to the price of the American options when both $p$ and $n$ go to infinity and moreover

$$0 \leq V_{p,n}(\lambda_{p,n}^\#) - U_0 \leq 2 \| M_T^* - C_{p,n}(M_T^*) \|_2.$$

- Consider a **Bermudan** option with exercising dates $t_0, \ldots, t_n$ and discounted payoff $(Z_{t_k})_k$ adapted to the discrete time filtration generated by the Brownian increments only. Then, $V_{p,n}(\lambda_{p,n}^\#)$ converges to the price of the Bermudan option when $p$ only goes to infinity.
Practically solving the optimization problem (1)

We approximate the solution of

\[ V_{p,n}(\lambda^\#_{p,n}) = \inf_{\lambda \in A_{p,n}^{\otimes d}} V_{p,n}(\lambda) = \inf_{\lambda \in A_{p,n}^{\otimes d}} \mathbb{E} \left[ \max_{0 \leq k \leq n} \left( Z_{t_k} - \mathbb{E}[C_{p,n}(\lambda) | \mathcal{F}_{t_k}] \right) \right] \]

by introducing the well–known Sample Average Approximation (see [Rubinstein and Shapiro, 1993]) of \( V_{p,n} \) defined by

\[ V_{p,n}^{m}(\lambda) = \frac{1}{m} \sum_{i=1}^{m} \max_{0 \leq k \leq n} \left( Z_{t_k}^{(i)} - \mathbb{E}[C_{p,n}^{(i)}(\lambda) | \mathcal{F}_{t_k}] \right). \]

Note that the conditional expectation boils down to truncating the chaos expansion and hence is tractable in a closed form.
Practically solving the optimization problem (2)

For large enough $m$, $V_{p,N}^m$ is convex, a.s. differentiable and tends to infinity at infinity. Then, there exists $\lambda_{p,n}^m$ such that

$$V_{p,n}^m(\lambda_{p,n}^m) = \inf_{\lambda \in \mathbb{R}^{A_{p,n}}} V_{p,n}^m(\lambda).$$

**Proposition 4**

$V_{p,n}^m(\lambda_{p,n}^m)$ converges a.s. to $V_{p,n}(\lambda_{p,N}^\#)$ when $m \to \infty$.

The distance from $\lambda_{p,n}^m$ to the set of minimizers of $V_{p,n}$ converges to zero as $m$ goes to infinity.
Practically solving the optimization problem (3)

Write $M_k(\lambda) = \mathbb{E}[C_{p,n}(\lambda) | \mathcal{F}_{t_k}]$ for $0 \leq k \leq n$.

**Proposition 5**

Assume $\lambda_{p,n}^\#$ is unique. Then,

$$
\frac{1}{m} \sum_{i=1}^{m} \left( \max_{0 \leq k \leq n} Z_{t_k}^{(i)} - M_k^{(i)}(\lambda_{p,n}^m) \right)^2 - V_{p,n}(\lambda_{p,n}^m)^2
$$

is a convergent estimator of $\text{Var}(\max_{k \leq 0 \leq n} Z_{t_k} - M_k(\lambda_{p,n}^\#))$ and moreover, if $\lambda_{p,n}^m$ is bounded,

$$
\lim_{m \to \infty} m \text{Var}\left( V_{p,n}(\lambda_{p,n}^m) \right) = \text{Var}\left( \max_{k \leq 0 \leq n} Z_{t_k} - M_k(\lambda_{p,n}^\#) \right).
$$
The algorithm: bespoke martingales

Define the first time the option goes in the money by

$$\tau_0 = \inf\{k \geq 0 : Z_{t_k} > 0\} \land n.$$ 

Consider martingales only starting once the option has been in the money

$$N_k(\lambda) = M_k(\lambda) - M_{k \land \tau_0}(\lambda).$$

In the dual price, “max\(_{0\leq k \leq n}\)” can be shrunk to “max\(_{\tau_0 \leq k \leq n}\)”.

Using Doob’s stopping theorem, we have

$$\mathbb{E}\left[\max_{\tau_0 \leq k \leq n} (Z_{t_k} - M_k(\lambda))\right] = \mathbb{E}\left[\max_{\tau_0 \leq k \leq n} (Z_{t_k} - (M_k(\lambda) - M_{\tau_0}(\lambda)))\right]$$

The martingales \(M(\lambda)\) or \(N(\lambda)\) lead to the same minimum value.

The set of martingales \(N^\lambda\) is far more efficient from a practical point of view.
The algorithm: a gradient descent with line search

\[ x_0 \leftarrow 0, \quad k \leftarrow 0, \quad \gamma \leftarrow 1, \quad d_0 \leftarrow 0, \quad v_0 \leftarrow \infty; \]

\textbf{while} True \textbf{do}

\hspace{1em} \textbf{Compute} \( v_{k+1/2} \leftarrow \tilde{V}_{m}^{p,n}(x_k - \gamma \alpha_k d_k) \);

\hspace{1em} \textbf{if} \( v_{k+1/2} < v_k \) \textbf{then}

\hspace{2em} \( x_{k+1} \leftarrow x_k - \gamma \alpha_k d_k; \)

\hspace{2em} \( v_{k+1} \leftarrow v_{k+1/2}; \)

\hspace{2em} \( d_{k+1} \leftarrow \nabla \tilde{V}_{mp,n}^{m}(x_{k+1}); \)

\hspace{2em} \textbf{if} \( \frac{|v_{k+1} - v_k|}{v_k} \leq \varepsilon \) \textbf{then return};

\hspace{1em} \textbf{else}

\hspace{2em} \( \gamma \leftarrow \gamma / 2; \)

\hspace{1em} \textbf{end}

\textbf{end}
The algorithm: a gradient descent with line search

\[ x_0 \leftarrow 0, \, k \leftarrow 0, \, \gamma \leftarrow 1, \, d_0 \leftarrow 0, \, v_0 \leftarrow \infty; \]

\textbf{while True do}

Compute \( v_{k+1/2} \leftarrow \tilde{V}_{m,p,n}^n(x_k - \gamma \alpha_k d_k); \)

\textbf{if} \( v_{k+1/2} < v_k \) \textbf{then}

\[ x_{k+1} \leftarrow x_k - \gamma \alpha_k d_k; \]

\[ v_{k+1} \leftarrow v_{k+1/2}; \]

\[ d_{k+1} \leftarrow \nabla \tilde{V}_{m,p,n}^n(x_{k+1}); \]

\textbf{if} \( \frac{|v_{k+1} - v_k|}{v_k} \leq \varepsilon \) \textbf{then return;}

\textbf{else}

\[ \gamma \leftarrow \gamma/2; \]

\textbf{end}

\textbf{end}

Take \( \alpha_\ell = \frac{\tilde{V}_{m,p,n}^n(x_\ell) - v^\#}{\left\| \nabla \tilde{V}_{m,p,n}^n(x_\ell) \right\|^2}, \) see [Polyak, 1987], but with the European price instead of the American one for \( v^\#. \)
Some remarks on the algorithm

- Given the expression of $V_{p,n}^m$, both the value function and its gradient are computed at the same time without extra cost.

\[
V_{p,n}(\lambda) = \mathbb{E} \left[ \max_{\tau_0 \leq k \leq n} (Z_{t_k} - \mathbb{E}[\lambda \cdot H^{\otimes d}(G_1, \cdots, G_n) | \mathcal{F}_{t_k}]) \right],
\]
\[
= \mathbb{E}[Z_{t_{\mathcal{I}(\lambda,z,G)}}] - \lambda \cdot \nabla \tilde{V}_{p,n}(\lambda).
\]

- Checking the admissibility of a step $\gamma$ costs as much as updating $x_k$.

- The algorithm is *almost* embarrassingly parallel:
  - Few iterations of the gradient descent are required ($\approx 10$).
  - Each iteration is fully parallel: each process treats its bunch of paths.
  - No demanding centralized computations
  - Very little communication: a few broadcasts only.
Parallel implementation

In parallel Generate \((G^{(1)}, Z^{(1)}), \ldots, (G^{(m)}, Z^{(m)})\) \(m x_0 \leftarrow 0 \in \mathbb{R}^{A_{p,n}}\);

while True do

Broadcast \(x_\ell, d_\ell, \gamma, \alpha_\ell\);

In parallel Compute \(\max_{0 \leq k \leq n}(Z_{t_k}^{(i)} - N_k^{(i)}(x_\ell - \gamma \alpha_\ell d_\ell))\);

Make a reduction of the above contributions to obtain \(\tilde{V}_m^p(x_{\ell+1/2})\) and \(\nabla \tilde{V}_m^p(x_{\ell+1/2})\);

\(v_{\ell+1/2} \leftarrow \tilde{V}_m^p(x_\ell - \gamma \alpha_\ell d_\ell)\);

if \(v_{\ell+1/2} < v_\ell\) then

\(x_{\ell+1} \leftarrow x_\ell - \gamma \alpha_\ell d_\ell\);

\(v_{\ell+1} \leftarrow v_{\ell+1/2}; \quad d_{\ell+1} \leftarrow \nabla \tilde{V}_m^p(x_{\ell+1})\);

if \(|v_{\ell+1} - v_\ell| / v_\ell \leq \varepsilon\) then return;

else

\(\gamma \leftarrow \gamma / 2\);

end

end
Basket option in the BS model

<table>
<thead>
<tr>
<th>p</th>
<th>n</th>
<th>$S_0$</th>
<th>price</th>
<th>Stdev</th>
<th>time (sec.)</th>
<th>reference price</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>3</td>
<td>100</td>
<td>2.27</td>
<td>0.029</td>
<td>0.17</td>
<td>2.17</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>100</td>
<td>2.23</td>
<td>0.025</td>
<td>0.9</td>
<td>2.17</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>110</td>
<td>0.56</td>
<td>0.014</td>
<td>0.07</td>
<td>0.55</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>110</td>
<td>0.53</td>
<td>0.012</td>
<td>0.048</td>
<td>0.55</td>
</tr>
<tr>
<td>2</td>
<td>6</td>
<td>100</td>
<td>2.62</td>
<td>0.021</td>
<td>0.91</td>
<td>2.43</td>
</tr>
<tr>
<td>3</td>
<td>6</td>
<td>100</td>
<td>2.42</td>
<td>0.021</td>
<td>14</td>
<td>2.43</td>
</tr>
<tr>
<td>2</td>
<td>6</td>
<td>110</td>
<td>0.61</td>
<td>0.012</td>
<td>0.33</td>
<td>0.61</td>
</tr>
<tr>
<td>3</td>
<td>6</td>
<td>110</td>
<td>0.55</td>
<td>0.008</td>
<td>10</td>
<td>0.61</td>
</tr>
</tbody>
</table>

**Tab.**: Prices for the put basket option with parameters $T = 3$, $r = 0.05$, $K = 100$, $\rho = 0$, $\sigma^j = 0.2$, $\delta^j = 0$, $d = 5$, $\omega^j = 1/d$, $m = 20,000$. 
Call option on the maximum of a basket

<table>
<thead>
<tr>
<th>d</th>
<th>p</th>
<th>m</th>
<th>S_0</th>
<th>price</th>
<th>Stdev</th>
<th>time (sec.)</th>
<th>reference price</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>2</td>
<td>20,000</td>
<td>90</td>
<td>10.18</td>
<td>0.07</td>
<td>0.4</td>
<td>8.15</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>20,000</td>
<td>90</td>
<td>8.5</td>
<td>0.05</td>
<td>4.1</td>
<td>8.15</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>20,000</td>
<td>100</td>
<td>16.2</td>
<td>0.06</td>
<td>0.54</td>
<td>14.01</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>20,000</td>
<td>100</td>
<td>14.4</td>
<td>0.06</td>
<td>5.6</td>
<td>14.01</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>20,000</td>
<td>90</td>
<td>21.2</td>
<td>0.09</td>
<td>2</td>
<td>16.77</td>
</tr>
<tr>
<td>5</td>
<td>3</td>
<td>40,000</td>
<td>90</td>
<td>16.3</td>
<td>0.05</td>
<td>210</td>
<td>16.77</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>20,000</td>
<td>100</td>
<td>30.7</td>
<td>0.09</td>
<td>3.4</td>
<td>26.34</td>
</tr>
<tr>
<td>5</td>
<td>3</td>
<td>40,000</td>
<td>100</td>
<td>26.0</td>
<td>0.05</td>
<td>207</td>
<td>26.34</td>
</tr>
</tbody>
</table>

**Tab.**: Prices for the call option on the maximum of $d$ assets with parameters $T = 3$, $r = 0.05$, $K = 100$, $\rho = 0$, $\sigma^j = 0.2$, $\delta^j = 0.1$, $n = 9$. 
## Scalability of the parallel algorithm

The tests were run on a BullX DLC supercomputer containing 3204 cores.

<table>
<thead>
<tr>
<th>#processes</th>
<th>time (sec.)</th>
<th>efficiency</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4365</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>2481</td>
<td>0.99</td>
</tr>
<tr>
<td>4</td>
<td>1362</td>
<td>0.90</td>
</tr>
<tr>
<td>16</td>
<td>282</td>
<td>0.84</td>
</tr>
<tr>
<td>32</td>
<td>272</td>
<td>0.75</td>
</tr>
<tr>
<td>64</td>
<td>87</td>
<td>0.78</td>
</tr>
<tr>
<td>128</td>
<td>52</td>
<td>0.73</td>
</tr>
<tr>
<td>256</td>
<td>34</td>
<td>0.69</td>
</tr>
<tr>
<td>512</td>
<td>10.7</td>
<td>0.59</td>
</tr>
</tbody>
</table>

**Tab.:** Scalability of the parallel algorithm on the 40—dimensional geometric put option described above with $T = 1$, $r = 0.0488$, $K = 100$, $\sigma^j = 0.3$, $\rho = 0.1$, $\delta^j = 0$, $n = 9$, $p = 2$, $m = 200,000$. 
Conclusion

- Purely optimization approach. No need of an optimal strategy.
- The problem is in large dimension but convex.
- *Almost* embarrassingly parallel and scales very well.
- Can deal with path dependent options
Solving optimal stopping problems via empirical dual optimization.

The duality of optimal exercise and domineering claims: a Doob-Meyer decomposition approach to the Snell envelope.

Smooth minimization of non-smooth functions.

Introduction to optimization.
*Optimization Software*.

Monte Carlo valuation of American options.