Moving boundary approaches for solving free-boundary problems

Kumar Muthuraman
McCombs School of Business, University of Texas - Austin

Workshop on Computational Methods in Finance, Fields Institute
March 2010

Overview

- **Goal**: To introduce a class of computational methods for solving free boundary problems
- **Idea**: Convert the arising free-boundary problem into a sequence of fixed boundary problems, which are easier to solve
- Talk Outline:
    - American option pricing - optimal stopping
    - Portfolio optimization with transaction costs - singular control, higher dimensional
    - (Cash management - impulse control)
    - Overview of the theoretical guarantees established

American Options

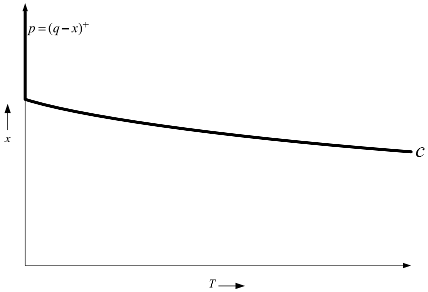
- Asset price process S_t follows a Geometric Brownian Motion.
- Risk-free rate r and constant asset volatility σ.

- A put option: A contract that pays $\max(q - S_\tau, 0)$.
- American put: The holder can choose any $\tau \in [0, T]$.
- q: Strike price, T time to expiry.

- Option price denoted by $p(T, x)$. (x underlying asset price)

$$p(T, x) = \sup_{\tau \in [0,T]} E\left\{e^{-r\tau}(q - S_\tau)^+\right\}$$

Optimal Exercise Policy

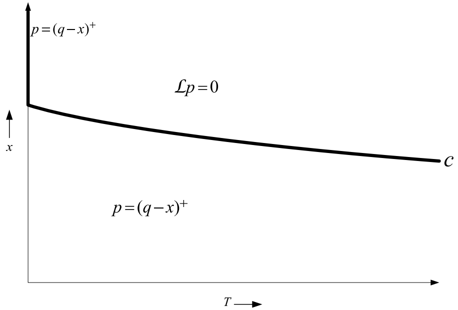


- Optimal exercise policy is characterized by an exercise boundary $c(T)$
- Exercise if $x \leq c(T)$ else hold.

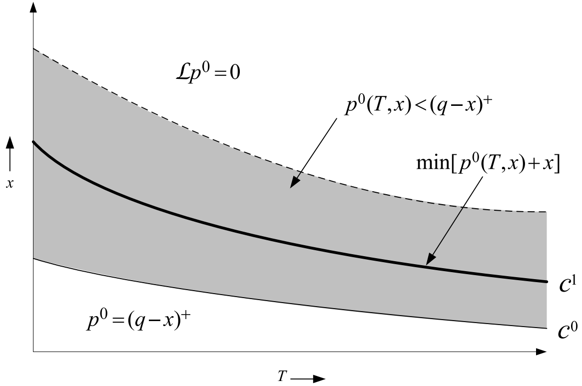
The Related Free-boundary PDE Problem

- Standard dynamic programming arguments and the Ito's formula yield a free-boundary PDE: Hamilton Jacobi Bellman (HJB) equation.
- Find $p(T,x)$, $c(T)$ such that
  - $\mathcal{L}p = 0$ if $x > c(T)$,
  - $p = (q-x)^+$ if $x \le c(T)$ and
  - $\max\{\mathcal{L}p, (q-x)^+ - p\} = 0$ for all $(x,T) \in (0,\infty)^2$.

  Here $\mathcal{L}p \equiv \frac{1}{2}\sigma^2 x^2 p_{xx} + rxp_x - rp - p_T$



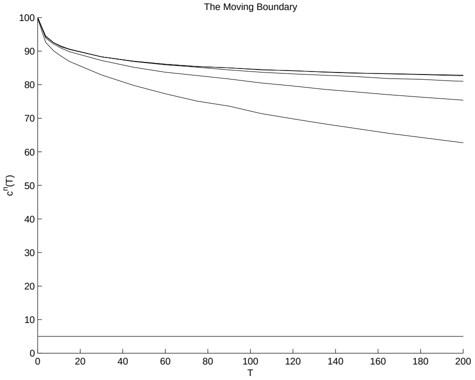
The Moving Boundary Method



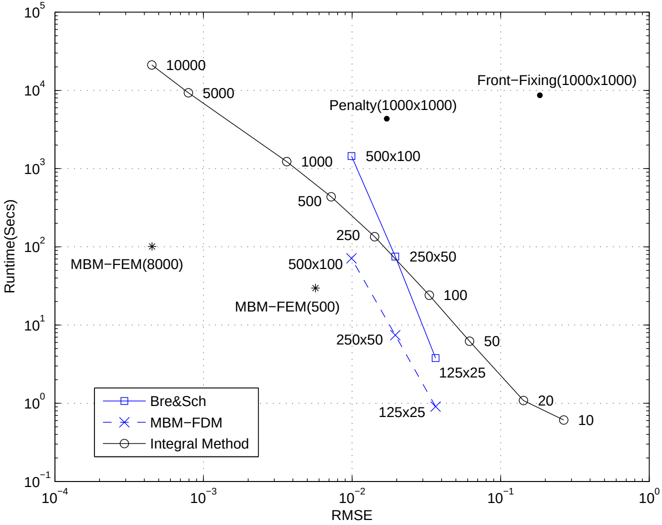
- Guess a c^0 such that $c^0 < c^*$: $\max\{\mathcal{L}p, (q-x)^+ - p\} = 0$ would be violated.
- Any c^1 in the shaded region: Policy improvement $(\mathbf{p^1} > \mathbf{p^0})$, but possibly $c^1 > c^*$.
- If c^1 is the contour of $\min(p^0 + x)$ along x, then $\mathbf{c^1} < \mathbf{c}^*$ as well.

An Example

- $q = 100, r = 8\%, \sigma = 20\%$



Runtime and Error comparisons



Portfolio optimization with transaction costs

- Buying (selling) a unit of stock i costs : $1 + \lambda_i \quad (-(1 - \mu_i))$.
- To maximize Long-term growth rate
  (Taksar et.al. ('88), Akian et.al. ('01))

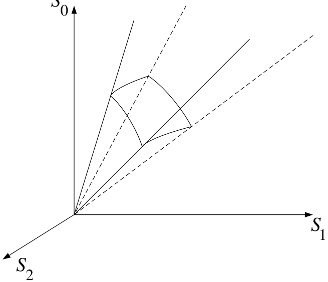
$$\liminf_{t \to \infty} E \left\{ \frac{\log W(t)}{t} \right\}.$$

- Dynamics: Value of stock (S_i) and bank (S_0),

$$\begin{aligned} dS_i &= \alpha_i S_i \, dt + \sigma_i S_i \, dB_i \quad + dL_i - dU_i \\ dS_0 &= r S_0 \, dt \quad + \sum_i [-(1 + \lambda_i) dL_i + (1 - \mu_i) dU_i] \end{aligned}$$

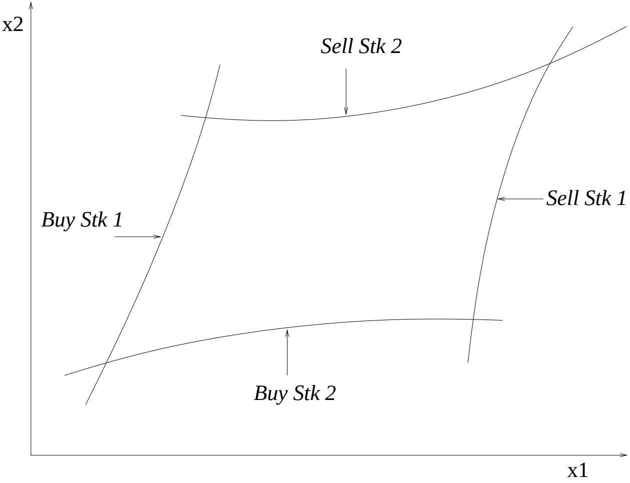
Portfolio optimization with transaction costs

- With no transaction costs: optimal to stick to the Merton line
- With transaction costs: too expensive to stay on the Merton line
- Optimal policy is characterized by a *No transaction region* - A cone



State $x_i = S_i/W$, fraction of wealth in each stock.

Structure of optimal policy



The Hamilton-Jacobi-Bellman Equation

- Change of var. and dynamic prog. arguments switch the objective to a cost minimization problem.
- To find $V(x)$ the differential cost function and d the long-term asymptotic cost growth rate, such that,

$$\min\{\ \mathcal{L}V, \quad \mathcal{B}_1V, \ldots, \mathcal{B}_NV, \quad \mathcal{S}_NV, \ldots, \mathcal{S}_NV\} = 0$$

Here

$$\mathcal{L}V(x) \quad = \quad \frac{1}{2}\,\mathrm{tr}\left\{D^2V\, z\sigma\sigma^T\, z\right\} + \nabla V \cdot \left[z(\alpha - re - \sigma\sigma^T x)\right] + h(x) - d.$$

and the i-th component of the vectors $\mathcal{B}V(x)$ and $\mathcal{S}V(x)$ are

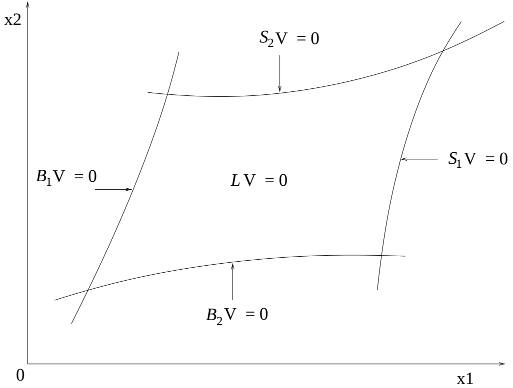
$\lambda_i \sum_{j=1}^N x_j \frac{\partial V}{\partial x_j} + \frac{\partial V}{\partial x_i} + \lambda_i$ and $\mu_i \sum_{j=1}^N x_j \frac{\partial V}{\partial x_j} - \frac{\partial V}{\partial x_i} + \mu_i$.

- State space is partitioned into $2N+1$ regions by the tight terms.

The HJB: Graphically

- Two stock case: Looking for
  - a function $V(x_1, x_2)$, a constant d
  - and the optimal region of inaction Ω with 4 boundaries

such that

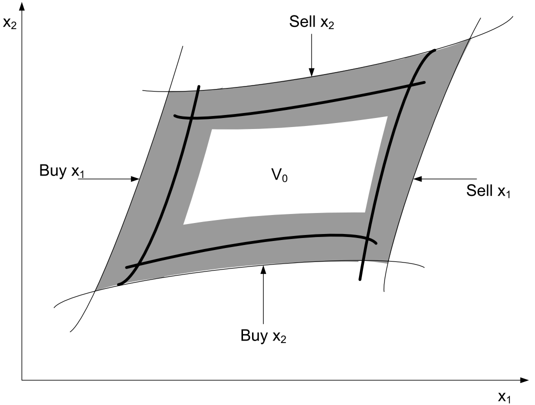


and $\min[\ \mathcal{L}V\ ,\ \mathcal{B}_1 V\ ,\ \mathcal{S}_1 V\ ,\ \mathcal{B}_2 V\ ,\ \mathcal{S}_2 V\ ] = 0$

- N-assets: search for $2N$ hyper surfaces in N dimension.

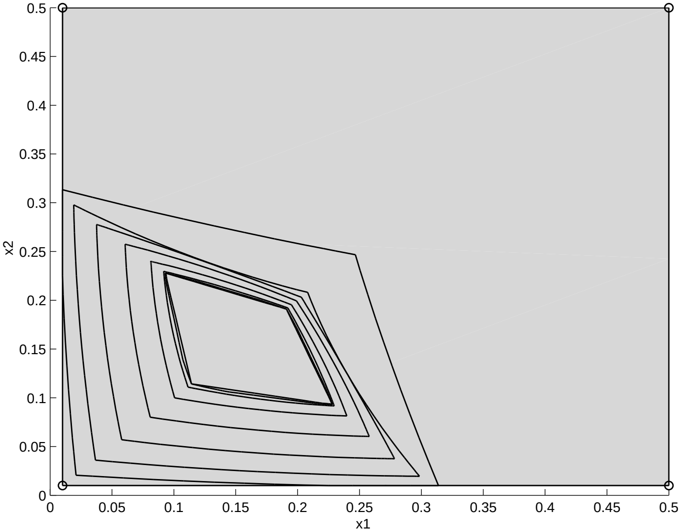
Boundary Update

- Guess a large Ω^0, (i.e. assume $\Omega^* \subset \Omega^0$). Solve for V_0, d_0.



- $\min[\ \mathcal{L}V_0\ ,\ \mathcal{B}_1V_0\ ,\ \mathcal{S}_1V_0\ ,\ \mathcal{B}_2V_0\ ,\ \mathcal{S}_2V_0\ ] = 0$, would be violated.
- Any Ω^1 in grey area $\Rightarrow d_1 < d_0 \Rightarrow$ Policy imp. $\Omega^* \subset \Omega^1$?
- $\Omega^* \subset \Omega^1$ guaranteed if new boundaries are defined by $\min(\lambda_i \sum_j x_j \frac{\partial V}{\partial x_j} + \frac{\partial V}{\partial x_i} + \lambda_i)$ and $\min(\mu_i \sum_j x_j \frac{\partial V}{\partial x_j} - \frac{\partial V}{\partial x_i} + \mu_i)$

Results - Boundary iteration



Parameters: $\sigma = [0.4 \quad 0.1; 0.1 \quad 0.4]$, $r = 7\%$

$\alpha_1 = \alpha_2 = 15\%$, $\lambda_1 = \mu_1 = \lambda_2 = \mu_2 = 1\%$

Increasing portfolio sizes

- The dimensionality of the underlying problem increases with number of stocks in the portfolio.
- Consider independent assets with $\alpha_i = 0.14$, $\sigma_i = 0.3$, $\lambda_i = \mu_i = 5\%$ and $r = 10\%$. Increasing N.
- Finite element based implementation on Matlab.

Increasing portfolio sizes

- The dimensionality of the underlying problem increases with number of stocks in the portfolio.
- Consider independent assets with $\alpha_i = 0.14$, $\sigma_i = 0.3$, $\lambda_i = \mu_i = 5\%$ and $r = 10\%$. Increasing N.
- Finite element based implementation on Matlab.

# of stocks, N	Runtime
1	34 sec
2	20 mins
3	45 hrs
4	???
5	???
6	???
7	???

Revisiting the Scheme

- Guess Ω^0. $n = 0$.

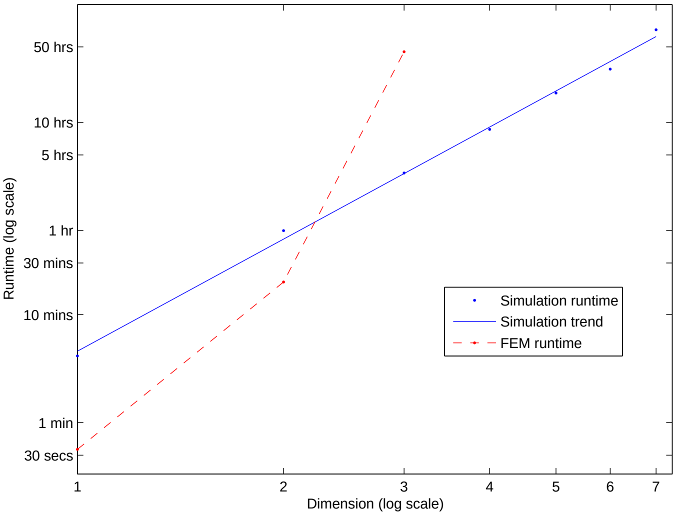
$$\begin{tikzcd}
\Omega^n \arrow[r, "\text{Solve PDE}"] & V_n, d_n \arrow[r, "\text{Bd update}"] & \Omega^{n+1} \arrow[ll, bend left, "\text{Iterate (n=n+1)}"]
\end{tikzcd}$$

- Most of the computational time is spent in solving the PDE
- Can use simulation instead of solving PDEs to estimate V, d.
- Issues:
  - Estimation errors
  - Monotonicity breaks
  - Simplest discretization of Ω: each axis by P points. P^N points.
- Fixes:
  - Use increasing sample paths
  - With a procedure that allows backing out
  - Approximate policy space: Eg. Hyper polygonal regions require only PN^3 points.

Runtime comparisons

# of stocks	PDE based	Simulation
1	34 sec	4 mins
2	20 mins	58 mins
3	45 hrs	3.4 hrs
4	???	8.6 hrs
5	???	18.7 hrs
6	???	36.6 hrs
7	???	62.3 hrs

Runtime comparisons



Impulse control: The Cash Management Problem

- A firm with stochastic cash flows.
- The cash level at time t is $Y(t)$.
- $h(Y(t)) > 0$ captures to opportunity costs and penalty costs.
- Cash level can be controlled by buying/selling short-term securities.
- Transaction costs have a fixed component, which justifies the use of non-infinitesimal control application. Causing discontinuities in state evolution (Constantinides and Richard (1978))
- Applications also exist in Portfolio optimization , Foreign exchange Rate models, Index tracking, Inventory management and Healthcare services management

The Control Band Policy

- In several cases including this model, optimal control policies take a simple form: (d, D, U, u).

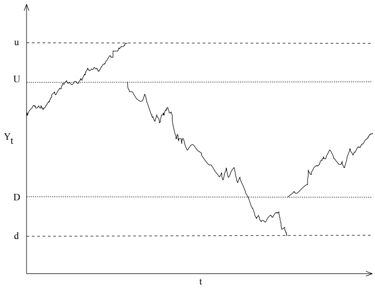


Figure: An Illustration of a (d, D, U, u) Policy

The HJB Equation

$$\min\{\mathcal{L}V(x),\ \mathcal{M}V(x),\ mV(x)\} = 0.$$

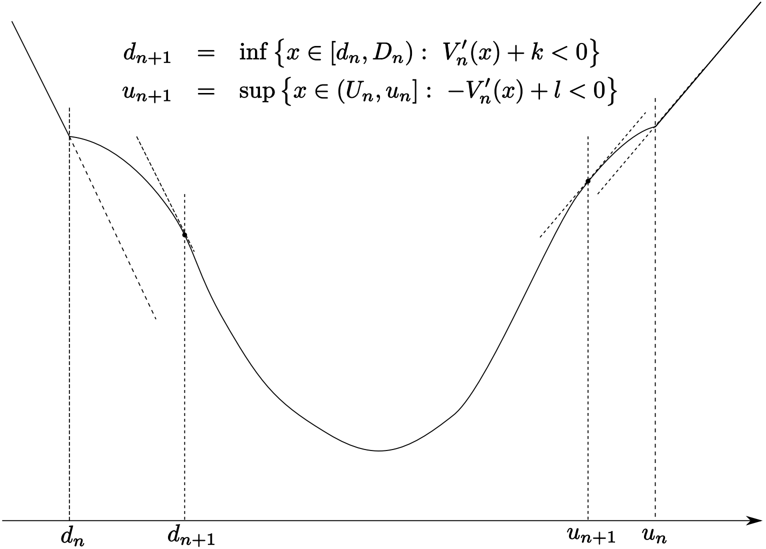
where

- $\mathcal{M}f(x) := \inf_{\eta > x} \Big\{ f(\eta) + K + k \cdot (\eta - x) \Big\} - f(x)$
- $mf(x) = \inf_{\eta < x} \Big\{ f(\eta) + L + l \cdot (x - \eta) \Big\} - f(x).$
- $\mathcal{L}f(x) := \frac{1}{2}\sigma^2 f''(x) + \mu \cdot f'(x) - \beta \cdot f(x) + h(x)$

The Moving boundary iteration

- We will start with an initial guess $d_0 < D_0 \le U_0 < u_0$, and solve the fixed-boundary problem to get $V_0(x)$.

$$
\begin{aligned}
d_{n+1} &= \inf\left\{x \in [d_n, D_n) : \; V_n'(x) + k < 0\right\} \\
u_{n+1} &= \sup\left\{x \in (U_n, u_n] : \; -V_n'(x) + l < 0\right\}
\end{aligned}
$$



Updating D and U

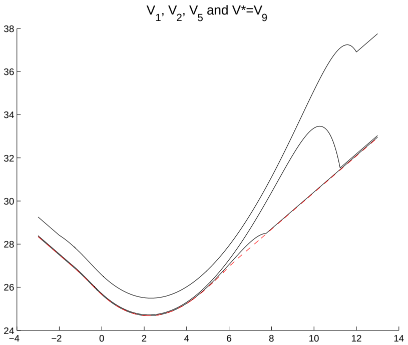
We solve the fixed boundary problem $(d_{n+1}, D_n, U_n, u_{n+1})$ to obtains its value function $\hat{V}_n(x)$, and then update D_n and U_n using

- $D_{n+1} = \arg\min_{x \in (d_{n+1}, u_{n+1})} \left\{ \hat{V}_n(x) + K + k \cdot (x - d_{n+1}) \right\}$

- $U_{n+1} = \arg\min_{x \in (d_{n+1}, u_{n+1})} \left\{ \hat{V}_n(x) + L + l \cdot (u_{n+1} - x) \right\}$

That is, whenever a control is exerted, we choose the most efficient jump-to point (inner boundary).

An Example of Stochastic Cash Management



Parameters chosen: $\mu = -0.2$, $\sigma = 0.6$, $\beta = 0.01$, $K = 0.14$, $k = 0.85$, $L = 0.14$, $l = 0.85$, $p = 0.12$, $q = 0.08$. The optimal (d, D, U, u) obtained is $(-1.315, 0.117, 4.838, 6.492)$.

Theoritical Guarantees for the Moving Boundary Approach

- One dimensional problems:
  - Monotone convergence
  - Optimality of the converged value function
  - ϵ-optimality: $|(1 + c_1)V(x) - f(x)| \leq c_2\epsilon$.

- Multiple dimensions
  - Almost entirely numerical evidence based
  - Recently: For American options under a stochastic volatility setting
    - for the first time established convergence proofs
    - works for all popular stochastic volatility models

Conclusion

- If you are looking to solve a free-boundary problem, transforming them to a sequence of fixed boundary problems is very likely possible
- Very efficient compared to general solution techniques like transformation to controlled Markov chains or boundary mapping transformation procedures
- By clubbing the procedure with simulation, it is possible to approximately solve large dimensional problems

Questions, Comments?