Risk Averse

Computational Stochastic Programming

Somayeh Moazeni
Princeton University

July 9, 2013

Joint work with Warren B. Powell.

The XIII International Conference on Stochastic Programming (ICSP2013)
Stochastic Models in Energy Planning
Optimization Problems in Energy Planning

Unit Commitment Problem and Optimal Dispatch

Renewable Energy Planning

Management of Energy Storage Devices

Optimal Revenue Hedging with Power Forwards and Options
Uncertainties and Risk

Uncertainty in Electricity and Fuel Prices

Uncertainty in Loads

Uncertainty in Electricity Generation

Uncertainty in Transmission
Uncertainties and Risk

- Uncertainty in Electricity Price
- Uncertainty in Oil Price
- Uncertainty in Gas Price
- Uncertainty in Coal Price
- Uncertainty in Loads
- Uncertainty in Electricity Generation
- Uncertainty in Transmission
- ...

possible negative divergence in relation to corporate planning:

Risk

TOO MUCH UNCERTAINTY!
Main Features of a Risk Management Process in the Energy Industry

**Price Risk**
- Energy Prices
- Emission Certificates Prices
- Foreign Exchange Rates
- Interest Rates

**Volume Risk**
- Consumption
- Production or Generation
- Optionality

**Credit Risk**
- Risk of Non-fulfillment of forward contracts
- Payment risk

**Other Risks**
- Liquidity risk
- Operational risk
- IT risk
- Legal risk

**Total Risk**

Main Features of a Risk Management Process in the Energy Industry

**Price Risk**
- Energy Prices
- Emission Certificates Prices
- Foreign Exchange Rates
- Interest Rates

**Volume Risk**
- Consumption
- Production or Generation
- Optionality

**Credit Risk**
- Risk of Non-fulfillment of forward contracts
- Payment risk

**Other Risks**
- Liquidity risk
- Operational risk
- IT risk
- Legal risk

Total Risk

Among the quantifiable risks, the **price risk** is the most significant risk.
Main Features of a Risk Management Process in the Energy Industry

**Price Risk**
- Energy Prices
- Emission Certificates Prices
- Foreign Exchange Rates
- Interest Rates

**Volume Risk**
- Consumption
- Production or Generation
- Optionality

**Credit Risk**
- Risk of Non-fulfillment of forward contracts
- Payment risk

**Other Risks**
- Liquidity risk
- IT risk
- Operational risk
- Legal risk

Total Risk

Among the quantifiable risks, the **price risk** is the most significant risk.
Outline

• Motivation
  o Energy Storage Management
    o Myopic Policy
    o Stochastic Dynamic Programming Approach
    o Rolling Horizon Approach

• Risk Averse State Dependent Policy
  o Direct Policy Search
  o Smoothing and Parametric Decision Rules for Minimizing CVaR
  o SMART-parallel policy search (SMART PPS)

• Preliminary Numerical Experiments

• Conclusions
Energy Storage Management

- **Goal:** To optimally control the energy flows among the source, grid, load, and a storage device.
  - Planning horizon: 168 hours (1 week)
- Three sources of (stochastic) uncertainty:
  - Electricity spot price ($\tilde{P}_t$)
  - Electricity demand ($\tilde{D}_t$)
  - Wind energy ($\tilde{E}_t$)
Energy Storage Management

• **Goal:** To optimally control the energy flows among the source, grid, load, and a **storage device**.
  - Planning horizon: 168 hours (1 week)

• Three sources of (stochastic) uncertainty:
  - Electricity spot price ($\tilde{P}_t$)
  - Electricity demand ($\tilde{D}_t$)
  - Wind energy ($\tilde{E}_t$)
Energy Storage Management

- **Goal:** To optimally control the energy flows among the source, grid, load, and a storage device.
  - Planning horizon: 168 hours (1 week)
- Three sources of (stochastic) uncertainty:
  - Electricity spot price ($\tilde{P}_t$)
  - Electricity demand ($\tilde{D}_t$)
  - Wind energy ($\tilde{E}_t$)
Energy Storage Management

• **Goal**: To optimally control the energy flows among the source, grid, load, and a storage device.
  - Planning horizon: 168 hours (1 week)

• Three sources of (stochastic) uncertainty:
  - Electricity spot price ($\tilde{P}_t$)
  - Electricity demand ($\tilde{D}_t$)
  - Wind energy ($\tilde{E}_t$)
Energy Storage Management

• J. P. Barton and D. G. Infield (2004)
• R. Carmona and M. Ludkovski (2009)
• P. Harsha and M. Dahleh (2011)
• I. Koutsopoulos et al. (2011)
• J. H. Kim and W. B. Powell (2011)
• P. M. van de Ven et al. (2012)
• W. R. Scott and W. B. Powell (2012)
• C. Budischak et al. (2013)
• ...
Energy Storage Management: Admissible Policies

\[ \eta^C, \eta^D \] : Charging (Discharging) efficiency rates
\[ \Delta R^C, \Delta R^D \] : Maximum Charging (Discharging) rates
\[ R_{\text{min}}, R_{\text{max}} \] : Minimum (Maximum) storage levels
\[ \gamma \Delta t \] : Energy rate lost over a time interval of length \( \Delta t \)

\[
\begin{align*}
    x_t^{WD} &= \min\{ \bar{E}_t, \bar{D}_t \}, \\
    x_t^{WR} + x_t^{WD} + x_t^{WG} &= \bar{E}_t, \\
    x_t^{GD} + \eta^D x_t^{RD} + x_t^{WD} &= \bar{D}_t, \\
    x_t^{RD} + x_t^{RG} &\leq ((1 - \gamma \Delta t)R_t - R_{\text{min}}) R_{\text{cap}} + \eta^C (x_t^{GR} + x_t^{WR}), \\
    \eta^C (x_t^{GR} + x_t^{WR}) - (x_t^{RD} + x_t^{RG}) &\leq (R_{\text{max}} - (1 - \gamma \Delta t)R_t) R_{\text{cap}}, \\
    x_t^{RD} + x_t^{RG} &\leq \Delta R^D R_{\text{cap}}, \\
    \eta^C (x_t^{GR} + x_t^{WR}) &\leq \Delta R^C R_{\text{cap}}. 
\end{align*}
\]
Energy Storage Management: Admissible Policies

\[ \eta^C, \eta^D, \Delta R^C, \Delta R^D, R^\text{min}, R^\text{max} \]

: Charging (Discharging) efficiency rates
: Maximum Charging (Discharging) rates
: Minimum (Maximum) storage levels
: Energy rate lost over a time interval of length \( \Delta t \)

\[
x_t^{WD} = \min \{ \bar{E}_t, \bar{D}_t \}, \\
x_t^{WR} + x_t^{WD} + x_t^{WG} = \bar{E}_t, \\
x_t^{GD} + \eta^D x_t^{RD} + x_t^{WD} = \bar{D}_t, \\
x_t^{RD} + x_t^{RG} \leq ((1 - \gamma_{\Delta t}) R_t - R^\text{min}) R^{\text{cap}} + \eta^C (x_t^{GR} + x_t^{WR}), \\
\eta^C (x_t^{GR} + x_t^{WR}) - (x_t^{RD} + x_t^{RG}) \leq (R^\text{max} - (1 - \gamma_{\Delta t}) R_t) R^{\text{cap}}, \\
x_t^{RD} + x_t^{RG} \leq \Delta R^D R^{\text{cap}}, \\
\eta^C (x_t^{GR} + x_t^{WR}) \leq \Delta R^C R^{\text{cap}}.
\]
Energy Storage Management: Admissible Policies

\[ \eta^C, \eta^D \] : Charging (Discharging) efficiency rates
\[ \Delta R^C, \Delta R^D \] : Maximum Charging (Discharging) rates
\[ R^\text{min}, R^\text{max} \] : Minimum (Maximum) storage levels
\[ \gamma \Delta t \] : Energy rate lost over a time interval of length $\Delta t$

\[
x_t^{WD} = \min\{\bar{E}_t, \bar{D}_t\},
\]
\[
x_t^{WR} + x_t^{WD} + x_t^{WG} = \bar{E}_t,
\]
\[
x_t^{GD} + \eta^D x_t^{RD} + x_t^{WD} = \bar{D}_t,
\]
\[
x_t^{RD} + x_t^{RG} \leq ((1 - \gamma \Delta t) R_t - R^\text{min}) R^{\text{cap}} + \eta^C (x_t^{GR} + x_t^{WR}),
\]
\[
\eta^C (x_t^{GR} + x_t^{WR}) - (x_t^{RD} + x_t^{RG}) \leq (R^\text{max} - (1 - \gamma \Delta t) R_t) R^{\text{cap}},
\]
\[
x_t^{RD} + x_t^{RG} \leq \Delta R^D R^{\text{cap}},
\]
\[
\eta^C (x_t^{GR} + x_t^{WR}) \leq \Delta R^C R^{\text{cap}}.
\]
Energy Storage Management: Admissible Policies

\[ \eta^C, \eta^D \] : Charging (Discharging) efficiency rates
\[ \Delta R^C, \Delta R^D \] : Maximum Charging (Discharging) rates
\[ R^{\min}, R^{\max} \] : Minimum (Maximum) storage levels
\[ \gamma \Delta t \] : Energy rate lost over a time interval of length \( \Delta t \)

\[ x_t^{WD} = \min\{ \tilde{E}_t, \tilde{D}_t \}, \]
\[ x_t^{WR} + x_t^{WD} + x_t^{WG} = \tilde{E}_t, \]
\[ x_t^{GD} + \eta^D x_t^{RD} + x_t^{WD} = \tilde{D}_t, \]
\[ x_t^{RD} + x_t^{RG} \leq ((1 - \gamma \Delta t) R_t - R^{\min}) R^{\text{cap}} + \eta^C (x_t^{GR} + x_t^{WR}), \]
\[ \eta^C (x_t^{GR} + x_t^{WR}) - (x_t^{RD} + x_t^{RG}) \leq (R^{\max} - (1 - \gamma \Delta t) R_t) R^{\text{cap}}, \]
\[ x_t^{RD} + x_t^{RG} \leq \Delta R^D R^{\text{cap}}, \]
\[ \eta^C (x_t^{GR} + x_t^{WR}) \leq \Delta R^C R^{\text{cap}}. \]
Energy Storage Management:
Admissible Policies

$\eta^C, \eta^D$: Charging (Discharging) efficiency rates

$\Delta R^C, \Delta R^D$: Maximum Charging (Discharging) rates

$R^\text{min}, R^\text{max}$: Minimum (Maximum) storage levels

$\gamma \Delta t$: Energy rate lost over a time interval of length $\Delta t$

$$Ax_t \leq b_t,$$
Energy Storage Management: Transaction Costs

Stage Cost Function:

\[ C_t(\tilde{P}_t, X^*_t) = (\tilde{P}_t^{GR} + \alpha_t^{GR}) x_t^{GR} + (\tilde{P}_t^{GD} + \alpha_t^{GD}) x_t^{GD} - \eta_t^D (\tilde{P}_t^{RG} - \alpha_t^{RG}) x_t^{RG} - (\tilde{P}_t^{WG} - \alpha_t^{WG}) x_t^{WG}. \]

Transaction costs of trading with the grid:
Myopic Policy

- Solve the problem myopically by ignoring the decision’s impact on the future!
- The myopic problem:

\[
x_t^{myopic} \overset{\text{def}}{=} \arg \min_{x_t \in \{x \in \mathbb{R}^4_+: Ax \leq b_t\}} C_t(\tilde{P}_t, x_t)
\]

- This policy discharges the storage device as quickly as possible.
- New battery level, \( R_{t+1} = \max \left\{ (1 - \gamma_{\Delta t})R_t - \Delta R^D, R_{\text{min}} \right\} \)
Stochastic Dynamic Programming Policy: Exact Approach

- The Bellman’s equation

\[ V_t(S_t) = \min_{x_t} \mathbb{E} \left[ C_t(\tilde{P}_t, x_t) + V_{t+1}(S_{t+1}) \mid S_t \right], \quad t = 0, \cdots, T - 1, \]
\[ V_T(S_T) = 0. \]

- Closed-form solution (under some conditions):
  
  - Charge the battery as much as possible, when
    \[ \tilde{P}_t \leq \mathbb{E}_t[\tilde{P}_{t+1}] \]
  
  - Discharge the battery as much as possible, when
    \[ \tilde{P}_t > \mathbb{E}_t[\tilde{P}_{t+1}] \]
Rolling Horizon Approach

Expected Price Path vs a Simulated Price Path

Risk Neutral Rolling Horizon Policy
Rolling Horizon Approach

Expected Price Path vs a Simulated Price Path

Risk Neutral Rolling Horizon Policy
## Relative Improvement of Risk-Neutral State-Dependent Policy versus Deterministic Policy

<table>
<thead>
<tr>
<th>Price Path</th>
<th>Real Cost of Rolling Horizon Policy [$]</th>
<th>Real Cost of Static Policy [$]</th>
<th>Relative Improvement [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>79,825,036.82</td>
<td>83,940,794.93</td>
<td>5.1560</td>
</tr>
<tr>
<td>2</td>
<td>71,579,479.37</td>
<td>76,253,993.65</td>
<td>6.5305</td>
</tr>
<tr>
<td>3</td>
<td>83,249,207.73</td>
<td>88,344,108.99</td>
<td>6.1201</td>
</tr>
<tr>
<td>4</td>
<td>68,673,816.35</td>
<td>74,928,003.82</td>
<td>9.1071</td>
</tr>
<tr>
<td>5</td>
<td>77,519,925.56</td>
<td>83,073,913.68</td>
<td>7.1646</td>
</tr>
<tr>
<td>6</td>
<td>61,076,208.44</td>
<td>65,459,435.92</td>
<td>7.1767</td>
</tr>
<tr>
<td>7</td>
<td>74,303,008.53</td>
<td>78,639,075.82</td>
<td>5.8357</td>
</tr>
<tr>
<td>8</td>
<td>78,305,905.55</td>
<td>81,834,130.53</td>
<td>4.5057</td>
</tr>
<tr>
<td>9</td>
<td>78,383,452.94</td>
<td>83,813,765.90</td>
<td>6.9279</td>
</tr>
<tr>
<td>10</td>
<td>83,092,056.07</td>
<td>86,972,791.26</td>
<td>4.6704</td>
</tr>
</tbody>
</table>
Relative Improvement of Risk-Neutral State-Dependent Policy versus Deterministic Policy

<table>
<thead>
<tr>
<th>Price Path</th>
<th>Real Cost of Rolling Horizon Policy [$]</th>
<th>Real Cost of Static Policy [$]</th>
<th>Relative Improvement [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>79,822,026.82</td>
<td>83,940,794.93</td>
<td>5.1560</td>
</tr>
<tr>
<td>2</td>
<td>71,579,479.37</td>
<td>76,253,993.65</td>
<td>6.5305</td>
</tr>
<tr>
<td>3</td>
<td>83,249,207.73</td>
<td>88,344,108.99</td>
<td>6.1201</td>
</tr>
<tr>
<td>4</td>
<td><strong>68,673,816.35</strong></td>
<td><strong>74,928,003.82</strong></td>
<td><strong>9.1071</strong></td>
</tr>
<tr>
<td>5</td>
<td>77,519,925.56</td>
<td>83,073,913.68</td>
<td>7.1646</td>
</tr>
<tr>
<td>6</td>
<td>61,076,208.44</td>
<td>65,459,435.92</td>
<td>7.1767</td>
</tr>
<tr>
<td>7</td>
<td>74,303,008.53</td>
<td>78,639,075.82</td>
<td>5.8357</td>
</tr>
<tr>
<td>8</td>
<td>78,305,905.55</td>
<td>81,834,130.53</td>
<td>4.5057</td>
</tr>
<tr>
<td>9</td>
<td>78,383,452.94</td>
<td>83,813,765.90</td>
<td>6.9279</td>
</tr>
<tr>
<td>10</td>
<td>83,092,056.07</td>
<td>86,972,791.26</td>
<td>4.6704</td>
</tr>
</tbody>
</table>
Risk Averse State-Dependent Strategy
Computational Stochastic Programming

- Planning horizon: [0, T)

\[
\min_{\pi} \omega_E \mathbb{E}\left[ \sum_{t=0}^{T-1} C_t(\tilde{P}_t, X_t^\pi(S_t)) \right] + \omega_\rho \rho \left[ \sum_{t=0}^{T-1} C_t(\tilde{P}_t, X_t^\pi(S_t)) \right]
\]

- Most typical stochastic dynamic programming approaches are unamenable to risk consideration. Here,

\( \rho[\cdot] \) : Risk measure
\( \omega_E, \omega_\rho \geq 0 \) : Risk aversion parameters
**Why Risk Matters?**

\[ \text{VaR}_\beta (\text{cost}) = \inf \{ \ell \in \mathbb{R} : \Pr(\text{cost} \leq \ell) \geq \beta \} \]

<table>
<thead>
<tr>
<th>(\beta[%])</th>
<th>(\text{VaR}_\beta)</th>
<th>Absolute Difference</th>
<th>Relative Difference [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>75</td>
<td>88,329,335.07</td>
<td>6,422,731.19</td>
<td>7.84</td>
</tr>
<tr>
<td>80</td>
<td>90,068,845.54</td>
<td>8,162,241.66</td>
<td>9.97</td>
</tr>
<tr>
<td>85</td>
<td>92,060,362.08</td>
<td>10,153,758.20</td>
<td>12.40</td>
</tr>
<tr>
<td>90</td>
<td>94,670,388.29</td>
<td>12,763,784.41</td>
<td>15.58</td>
</tr>
<tr>
<td>95</td>
<td>98,751,253.29</td>
<td>16,844,649.41</td>
<td>20.57</td>
</tr>
<tr>
<td>99</td>
<td>105,722,089.92</td>
<td>23,815,486.04</td>
<td>29.08</td>
</tr>
<tr>
<td>99.90</td>
<td>114,941,626.50</td>
<td>33,035,022.62</td>
<td>40.33</td>
</tr>
</tbody>
</table>

\[
\tilde{P}_t = P_t^{\text{hour}} + P_t^{\text{day}} + P_t^{\text{month}} + \tilde{Y}_t^P
\]

\[
\tilde{Y}_t^P = \phi_P \tilde{Y}_{t-\Delta t} + \sigma_P \sqrt{\Delta t} \tilde{\epsilon}_t, \quad \tilde{\epsilon}_t \sim \mathcal{N}(0, 1)
\]
Why Risk Matters?

A reasonable evolution process should be able to capture important properties of energy prices such as:

- Spikes
- Mean reversion
- Fat tails of the price distributions
- Seasonality

Energy prices have an extraordinarily high volatility! The volatility of energy prices, especially electricity, is an order of magnitude higher than the volatility of some financial assets such as interest rates or equity prices.

Risk Measures and Dynamic Programming

- Computing an exact stochastic dynamic programming solution is challenging, mainly due to the curse of dimensionality. The size of the state space grows very quickly.
- When risk is also concerned, an appropriate optimality equation does not exist for most risk measures, e.g., see (Shapiro, 2009, 2012).
- Tractable computational methods for risk averse stochastic dynamic programming is scarce, especially when objective function is non-differentiable.

Value-at-Risk:
How bad can things get?

Conditional Value-at-Risk

\[ CVaR_\alpha = E[X: X \geq VaR_\alpha(X)] \]
Risk Averse
Computational Stochastic Programming

• The aim is to develop an efficient computational method to solve the problem for effective handling of
  ▪ broad range of risk measures, particularly VaR
  ▪ possibly, very large number of constraints
  ▪ non-stationary multi-dimensional policies

• **Direct Policy Search:** Decision Rule + Sampling + Optimization

\[
\min_{\theta_0, \theta_1, \ldots, \theta_{T-1}} \omega_E \left[ \sum_{t=0}^{T-1} C_t(\tilde{P}_t, x_t(S_t|\theta_t)) \right] + \omega_{\rho} \left[ \sum_{t=0}^{T-1} C_t(\tilde{P}_t, x_t(S_t|\theta_t)) \right]
\]
Smoothing and Parametric Decision Rules for Stochastic Mean-CVaR Optimal Strategy

• Variable Reduction through parametric decision rules, \( x_t(S_t|\theta_t) = f_{\theta_t}(S_t) \) where \( f_{\theta_t}() \) is defined by simple algebraic operators, e.g., an affine parametric model
  \[
  x_t(S_t|\theta_t) = \theta_t'S_t
  \]

• Monte Carlo Simulation

\[
\min_{\theta_0, \theta_1, \ldots, \theta_{T-1}, \beta} \frac{\omega_E}{N} \sum_{n=1}^{N} \sum_{t=0}^{T-1} C_t(\bar{P}_t^{(n)}, x_t(S_t^{(n)}|\theta_t)) + \frac{\omega_\rho}{N(1-\alpha)} \sum_{n=1}^{N} \sum_{t=0}^{T-1} C_t\left(\bar{P}_t^{(n)}, x_t(S_t^{(n)}|\theta_t)\right) - \beta
\]

• Eliminating Non-differentiability by approximating the function \([\cdot]^+ = \max(\cdot, 0)\) with a smooth function.

✓ Handling Inequality Constraints: smoothed exact penalty function.

Smoothing and Parametric Decision Rules for Stochastic Mean-CVaR Optimal Strategy

• The objective function is continuously differentiable.
• The trust region method with derivative evaluations using automatic differentiation.
• The computational complexity of the resulting method does not depend on the number of simulations.
• The proposed computational method is applicable to several other risk measures, e.g., semi-standard deviation risk measure $\mathbb{E}(\cdot^+)$ as well as variance.

⚠️ Limitations:

• An increase in the number of constraints reduces the efficiency of the approach.
• The trust region method along with derivative evaluations using AD is not directly applicable for some risk measures such as VaR or for more complex decision rule parametric approximations.
A Parallel Computational Stochastic Programming Approach: SMART-parallel policy search (SMART PPS)

- SMART-parallel policy search (SMART PPS)
  - Decision rule defined by an $\arg\min$ operator and basis functions.
  - Spline approximation
  - Pattern search framework (nested parallel implementation)

For every $t = 0, \cdots, T - 1$, and every $S_t$:

$$x_t(S_t|\theta_t) = \arg \min_{x \in \chi_t} C_t(S_t, x) + \sum_{k=1}^{K} \theta_{tk} \cdot \phi_k(S_t)$$

$\chi_t$: The set of feasible vectors $x_t$, given the state at period $t$

The number of decision variables is now of the order of $T \times K$. 
A Parallel Computational Stochastic Programming Approach: SMART-parallel policy search (SMART PPS)

- SMART-parallel policy search (SMART PPS)
  - Decision rule defined by an arg-min operator and basis functions.
  - Spline approximation
  - Pattern search framework (nested parallel implementation)

For every \( t = 0, \ldots, T - 1 \), and every \( S_t \):

\[
x_t(S_t | \theta_t) = \arg \min_{x \in \chi_t} C_t(S_t, x) + \sum_{k=1}^{K} \theta_{tk} \cdot \phi_k(S_t)
\]

\( \chi_t \): The set of feasible vectors \( x_t \), given the state at period \( t \)

The number of decision variables is now of the order of \( T \times K \).
SMART-parallel policy search (SMART PPS)

\[
\theta_0 = (\theta_{01}, \theta_{02}, \ldots, \theta_{0K}) \\
\vdots \\
\theta_T = (\theta_{1T}, \theta_{2T}, \ldots, \theta_{kT}) \\
\theta_{T-1} = (\theta_{(T-1)1}, \ldots, \theta_{(T-1)K})
\]

\[
\omega_E \mathbb{E} \left[ \sum_{t=0}^{T-1} C_t(\tilde{P}_t, x_t(S_t|\theta_t)) \right] + \omega_\rho \mathbb{P} \left[ \sum_{t=0}^{T-1} C_t(\tilde{P}_t, x_t(S_t|\theta_t)) \right]
\]
SMART-parallel policy search (SMART PPS)

Each $\theta_t$ is a (unique) natural cubic spline with respect to time.

Optimal Decision Variables determined by the Optimization Problem

We choose.

$$\min \quad \theta_0 = (\theta_{01}, \theta_{02}, \ldots, \theta_{0K})$$

$$\theta_t = (\theta_{t1}, \theta_{t2}, \ldots, \theta_{tK})$$

$$\theta_{T-1} = (\theta_{(T-1)1}, \ldots, \theta_{(T-1)K})$$

$$\omega_E \mathbb{E} \left[ \sum_{t=0}^{T-1} C_t(\tilde{P}_t, x_t(S_t|\theta_t)) \right] + \omega_P \rho \left[ \sum_{t=0}^{T-1} C_t(\tilde{P}_t, x_t(S_t|\theta_t)) \right]$$
SMART-parallel policy search (SMART PPS)

- We choose a set of $S \ll T$ spline points $t_1, t_2, \cdots, t_S$ over $[0, T)$.

\[
\begin{align*}
\min_{(\theta_{f_1}(t_1), \theta_{f_2}(t_1), \cdots, \theta_{f_K}(t_1))} & \quad \omega_E \mathbb{E} \left[ \sum_{t=0}^{T-1} C_t(S_t, x_t(S_t|\theta_t)) \right] + \omega_\rho \mathbb{P} \left[ \sum_{t=0}^{T-1} C_t(S_t, x_t(S_t|\theta_t)) \right] \\
(\theta_{f_1}(t_i), \theta_{f_2}(t_i), \cdots, \theta_{f_K}(t_i)) & \\
(\theta_{f_1}(t_S), \theta_{f_2}(t_S), \cdots, \theta_{f_K}(t_S)) &
\end{align*}
\]

- Where $\theta_t$ includes values of the spline functions $\theta_{f_1}(\cdot), \cdots, \theta_{f_K}(\cdot)$ at $t$:

\[
\theta_t = (\theta_{f_1}(t), \theta_{f_2}(t), \cdots, \theta_{f_K}(t))
\]

- The number of decision variables is now of the order of $S \times K$ which is much smaller than $T \times N$. 
SMART-parallel policy search (SMART PPS)

- Monte Carlo simulations to estimate the expected value and risk measure, at every $\theta_{t_1}, \theta_{t_2}, \ldots, \theta_{t_S}$.

<table>
<thead>
<tr>
<th>N</th>
<th>$\varepsilon_N(\text{mean})$ [%]</th>
<th>$\varepsilon_N(\text{VaR}_{95%})$ [%]</th>
<th>$\varepsilon_N(\text{CVaR}_{95%})$ [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>500</td>
<td>1.2997</td>
<td>3.8543</td>
<td>2.8977</td>
</tr>
<tr>
<td>1,000</td>
<td>1.2449</td>
<td>2.5335</td>
<td>2.2226</td>
</tr>
<tr>
<td>5,000</td>
<td>0.3597</td>
<td>1.0597</td>
<td>1.1881</td>
</tr>
<tr>
<td>10,000</td>
<td>0.3065</td>
<td>0.6735</td>
<td>0.8380</td>
</tr>
<tr>
<td>15,000</td>
<td>0.2715</td>
<td>0.5272</td>
<td>0.6044</td>
</tr>
<tr>
<td>20,000</td>
<td>0.2174</td>
<td>0.4440</td>
<td>0.3636</td>
</tr>
<tr>
<td>30,000</td>
<td>0.1483</td>
<td>0.4370</td>
<td>0.3058</td>
</tr>
</tbody>
</table>

$$
\varepsilon_N(\mathbf{f}) \overset{\text{def}}{=} \frac{\max_{i=1}^{10} f_M^{(i)} - \min_{i=1}^{10} f_M^{(i)}}{\frac{1}{10} \sum_{i=1}^{10} f_M^{(i)}}
$$
SMART-parallel policy search (SMART PPS)

- Monte Carlo simulations to estimate the expected value and risk measure, at every $\theta_{t_1}, \theta_{t_2}, \ldots, \theta_{t_S}$.

- The resulting unconstrained (or box constrained) optimization problem probably has a non-differentiable objective function.

- Derivative free optimization
  - Pattern Search Framework
  - Direct Search methods may very well be used to solve non-continues, non-differentiable and multi-modal, i.e. multiple local optima, optimization problems.

Pattern Search Framework

Iteration 1
Successful!
Pattern Search Framework

Iteration 2
Unsuccessful!
Pattern Search Framework

Iteration 3
Successful!
Challenges with Sequential Implementation

• When dimension of the stochastic search problem, $S \times K$, increases, each pattern search iteration becomes more computationally demanding.

• A large number of Monte Carlo Paths is needed for a reasonably accurate estimation of the expected value and risk.

• Each function evaluation involves solving $N$ minimization problem, due to the arg-min operator, where $N$ is the number of Monte Carlo paths. Whence, very computationally expensive!
### Run Time of Each Function Evaluation

<table>
<thead>
<tr>
<th>$N$</th>
<th>Objective mean</th>
<th>VaR$_{95%}$</th>
<th>CVaR$_{95%}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>500</td>
<td>20.15</td>
<td>20.40</td>
<td>20.01</td>
</tr>
<tr>
<td>1,000</td>
<td>38.25</td>
<td>38.40</td>
<td>38.08</td>
</tr>
<tr>
<td>5,000</td>
<td>183.70</td>
<td>184.66</td>
<td>187.61</td>
</tr>
<tr>
<td>10,000</td>
<td>370.48</td>
<td>364.10</td>
<td>372.33</td>
</tr>
<tr>
<td>15,000</td>
<td>540.62</td>
<td>543.34</td>
<td>541.03</td>
</tr>
<tr>
<td>20,000</td>
<td>752.31</td>
<td>760.71</td>
<td>723.34</td>
</tr>
<tr>
<td>30,000</td>
<td>1147.07</td>
<td>1476.60</td>
<td>1300.64</td>
</tr>
</tbody>
</table>

Table: Required time (in seconds) to evaluate mean, VaR$_{95\%}$, and CVaR$_{95\%}$ corresponding to the myopic policy over 168 hours, derived using $N$ Monte Carlo paths.
## Sequential Implementation

<table>
<thead>
<tr>
<th>$N$</th>
<th>$\tau + 1$</th>
<th>4</th>
<th>10</th>
<th>15</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>100</td>
<td>2,623.85</td>
<td>6,574.69</td>
</tr>
<tr>
<td></td>
<td></td>
<td>500</td>
<td>17,856.32</td>
<td>30,816.21</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1,000</td>
<td>26,228.92</td>
<td>60,617.88</td>
</tr>
<tr>
<td></td>
<td></td>
<td>5,000</td>
<td>$&gt; 86,340.00$</td>
<td>$&gt; 86,340.00$</td>
</tr>
<tr>
<td>Policy &amp; $N$</td>
<td>100</td>
<td>2,907.77</td>
<td>6,113.48</td>
<td>8,988.13</td>
</tr>
<tr>
<td></td>
<td></td>
<td>500</td>
<td>17,729.50</td>
<td>30,567.15</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1,000</td>
<td>26,298.11</td>
<td>61,480.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td>5,000</td>
<td>$&gt; 86,340.00$</td>
<td>$&gt; 86,340.00$</td>
</tr>
<tr>
<td>Policy $N_{C, 95%}$</td>
<td>100</td>
<td>2,898.93</td>
<td>8,408.95</td>
<td>8,986.76</td>
</tr>
<tr>
<td></td>
<td></td>
<td>500</td>
<td>18,142.97</td>
<td>30,479.51</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1,000</td>
<td>25,939.26</td>
<td>60,601.94</td>
</tr>
<tr>
<td></td>
<td></td>
<td>5,000</td>
<td>$&gt; 86,340.00$</td>
<td>$&gt; 86,340.00$</td>
</tr>
<tr>
<td>Policy $N_{V, 95%}$</td>
<td>100</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>500</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>1,000</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>5,000</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Implemented on Hyperion cluster from the Livermore National Lab
Not yet done!

• Global convergence theory has been established for (most) pattern search methods with continuously differentiable objective function.

• How about non-differentiable objectives such as Value-at-Risk?
Multiple Starting Points are needed for non-differentiable criteria

Figure 1: $N = 1,000$ Monte Carlo samples, $I_{Max} = 50$ iterations, $\tau + 1 = 6$ spline knots. Solid red line indicates the starting point $e_6$. Dashed lines are for 4 randomly generated starting points.
(Nested) Parallel Implementation

• Three layers of parallelization
  – Starting Points
  – Directions in Pattern Search Framework
  – Monte Carlo Simulation Paths

When more resources are available, other directions and random points in the positive spanning set of the coordinate points can also be included in the pool of directions.
Parallel Programming (Implementation) Model

• A hybrid model is used which combines a distributed parallel computing Message Passing Model (MPI) with the threads model.
• We use a MPI-like Java library, called MPJ Express, to implement Java messaging system.

$$\theta^*(\theta^0)$$

Manager Node

Client Nodes

2n+1 nodes

At most $I_{Max}$ times

(MPI)

Only one client node handles all $N$ LPs, one for each Monte Carlo Path.
Parallel Programming (Implementation) Model

\[ \theta^* \text{ to } \theta^{(0)} \]

Manager Node

Head-Client Nodes

Client Nodes

2nc_N + 1 nodes

\( (C_N - 1) \) Client nodes, along with their associated Head-Client node, handle the \( N \) LPs, one for each Monte Carlo Path. Only the Head-Client node communicates with the Manager node.
Parallel Programming (Implementation) Model

Multiple Starting Points

Head Node

Manager Nodes

\(\theta(i)\)

\(\theta^*\)

\(\theta^{(0,1)}, \theta^{(0,2)}, \ldots, \theta^{(0,S)}\)

\((c_s(2nc_N+1)+1)\) nodes

Each of the \(c_N\) Manager nodes handles a subset of the given set of starting points.

Multi-threading parallelization is only the lowest level of the chain!
Parallel Implementation

\[ \text{with}\ c_{S} = 1,\ c_{N} = 1,\ \text{and}\ c_{I} = 2(\tau + 1) + 1 \]

<table>
<thead>
<tr>
<th>( N )</th>
<th>( \tau + 1 )</th>
<th>4</th>
<th>10</th>
<th>15</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td></td>
<td>826.00</td>
<td>819.46</td>
<td>607.60</td>
</tr>
<tr>
<td>500</td>
<td></td>
<td>2,980.78</td>
<td>2,970.80</td>
<td>2,993.20</td>
</tr>
<tr>
<td>1,000</td>
<td></td>
<td>8,116.79</td>
<td>7,990.87</td>
<td>5,954.25</td>
</tr>
<tr>
<td>5,000</td>
<td></td>
<td>( &gt; 18,000.00 )</td>
<td>( &gt; 18,000.00 )</td>
<td>( &gt; 18,000.00 )</td>
</tr>
<tr>
<td>Policy_{E}(6)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>100</td>
<td></td>
<td>605.05</td>
<td>819.31</td>
<td>610.58</td>
</tr>
<tr>
<td>500</td>
<td></td>
<td>2,985.26</td>
<td>4,063.24</td>
<td>3,012.06</td>
</tr>
<tr>
<td>1,000</td>
<td></td>
<td>8,126.29</td>
<td>5,988.94</td>
<td>5,986.86</td>
</tr>
<tr>
<td>5,000</td>
<td></td>
<td>( &gt; 18,000.00 )</td>
<td>( &gt; 18,000.00 )</td>
<td>( &gt; 18,000.00 )</td>
</tr>
<tr>
<td>Policy_{C}(6)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>100</td>
<td></td>
<td>604.65</td>
<td>604.52</td>
<td>826.80</td>
</tr>
<tr>
<td>500</td>
<td></td>
<td>4,005.76</td>
<td>4,030.66</td>
<td>2,980.00</td>
</tr>
<tr>
<td>1,000</td>
<td></td>
<td>8,079.50</td>
<td>5,976.54</td>
<td>5,956.60</td>
</tr>
<tr>
<td>5,000</td>
<td></td>
<td>( &gt; 18,000.00 )</td>
<td>( &gt; 18,000.00 )</td>
<td>( &gt; 18,000.00 )</td>
</tr>
<tr>
<td>Policy_{V}(6)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Implemented on Hyperion cluster from the Livermore National Lab
Parallel Implementation
(with $c_N > 1$)

<table>
<thead>
<tr>
<th>N</th>
<th>$\tau + 1$</th>
<th>$c_N$</th>
<th>Total Number of Processes</th>
<th>Total Run Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>4</td>
<td>5</td>
<td>42</td>
<td>34.51 [s]</td>
</tr>
<tr>
<td>500</td>
<td>4</td>
<td>10</td>
<td>82</td>
<td>172.87 [s]</td>
</tr>
<tr>
<td>1000</td>
<td>4</td>
<td>20</td>
<td>162</td>
<td>232.41 [s]</td>
</tr>
</tbody>
</table>
Preliminary Numerical Experiments for the Battery Storage Problem

- Only one basis function:

\[ x_t = \arg \min_x C(S_t, x) - \eta^D \theta_t R_{Cap} E[P_{t+1}] R_{t+1} \]

- This decision rule can accurately approximate the exact solution, when it is known (e.g., when \( \eta^D = \eta^C = 1, \Delta R^D = \Delta R^C = 1 \), and the objective is to minimize the expected cost):

<table>
<thead>
<tr>
<th>( \theta_i )</th>
<th>Exact Solution</th>
<th>PCSP Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \theta_1 )</td>
<td>1</td>
<td>1.000</td>
</tr>
<tr>
<td>( \theta_2 )</td>
<td>1</td>
<td>1.032</td>
</tr>
<tr>
<td>( \theta_3 )</td>
<td>1</td>
<td>1.032</td>
</tr>
<tr>
<td>( \theta_4 )</td>
<td>1</td>
<td>1.004</td>
</tr>
<tr>
<td>Objective Value</td>
<td>( 2.821 \times 10^9 )</td>
<td>( 2.733 \times 10^9 )</td>
</tr>
</tbody>
</table>

1,000 simulations, initial point \( \theta_i = 1.5 \), initial step sizes \( \gamma_i = 0.5 \), 5 periods
run time of sequential implementation 49.27 mins with 17 iterations in pattern search
when $\eta^D = 0.85, \eta^C = 0.75, \Delta R^D = 0.85, \Delta R^C = 0.90$
Conclusions

- A computationally tractable approach is developed based on:
  - direct policy search,
  - an arg-min type decision rule
  - spline approximation
  - nested parallel pattern search framework
  to
    - mitigate the curses of dimensionality
    - handle risk

- Effectiveness of the devised approach is illustrated for the energy storage management problem.
Stationary versus Non-Stationary Policies
Impact of Number of Spline Knots

<table>
<thead>
<tr>
<th>Number of Periods</th>
<th>Objective Function Value</th>
<th>Obtained Policy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stationary Policy ((\theta_1 = \cdots = \theta_T))</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Myopic Policy</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Non-Stationary Policy ((S = T))</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Non-Stationary Policy (with (S = T/2) knots)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Non-Stationary Policy (with (S = T/4) knots)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Non-Stationary Policy (with (S = T/10) knots)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

only first layer of parallelization, initial point \(\theta_i = 1.5\), initial step sizes \(\gamma_i = 0.5\) parallelization using \(?\) machines, 1 for each coordinate direction minimizing expected value of the cost, \(T=10\)