Decision Rule Approach with Transformation Technique

Zhan Lin
linzhan@mail.ustc.edu.cn

October 8, 2016

Original Problems

$$
\begin{aligned}
\max_{\mathbf{x_t}} \quad & \mathbb{E}_{\xi}\left(\sum_{t=1}^{T} \mathbf{v}^T \mathbf{x_t}\left(\xi^{t-1}, \xi_t\right)\right) \\
s.t. \quad & \sum_{t=1}^{T} A\mathbf{x_t}\left(\xi^{t-1}, \xi_t\right) \leq \mathbf{c} \\
& \mathbf{x_t}(\xi^{t-1}, \xi_t) \geq 0 \\
& \forall \xi \in \Xi, t = 1, \ldots, T
\end{aligned} \tag{1}
$$

- Capacity levels $\mathbf{c} = (c_1, \ldots, c_m)^T$.
- Corresponding prices $\mathbf{v} = (v_1, \ldots, v_n)^T$.
- Each products needs at most one unit of each resource. Let $A = (a_{ij})$ be the resource coefficient matrix.
- ξ_t is demands on period t and ξ^{t-1} is information of demands before period t.

Transformation

$$
\begin{aligned}
\max_{\mathcal{X}_{\mathbf{t}}} \quad & \mathbb{E}_{\xi}\left(\sum_{t=1}^{T} \mathbf{v}^{T} \mathcal{X}_{\mathbf{t}}\left(\xi^{t-1}, \xi_{t}\right)\right) \\
s.t. \quad & \mathcal{X}_{t}(\xi^{t-1}, \xi_{t}) \leq \xi_{t} \\
& \{\mathcal{X}_{\mathbf{t}}(\xi^{t-1}, \xi_{t}) = (\mathcal{X}_{t,1}(\xi^{t-1}, \xi_{t,1}), \mathcal{X}_{t,2}(\xi^{t-1}, \xi_{t,2}) \ldots \\
& , \mathcal{X}_{t,j}(\xi^{t-1}, \xi_{t,j}))^{T}\} \in \mathcal{F} \\
& \forall \xi \in \Xi, t = 1, \ldots, T
\end{aligned}
\tag{2}
$$

where $\mathcal{F} = \{\{f(\xi^{t-1}, \xi_{t})\} : f(\xi^{t-1}, \xi_{t}) = \mathbf{u}_{t}(\xi^{t-1}) \bigwedge \xi_{t}, \mathbf{u}_{\mathbf{t}}(\xi^{t-1}) \in \mathbb{R}^{+}, \sum_{t=1}^{T} A\mathbf{u}_{\mathbf{t}}(\xi^{t-1}) \leq \mathbf{c}\}$.

Functional Optimization

- How to optimize object with $\{\mathcal{X}_t(\xi^{t-1}, \xi_t)\} \in \mathcal{F}$?
- How to implement polices when we have no knowledge of ξ_t?
- As lower time complexity as possible with acceptable numerical accuracy.

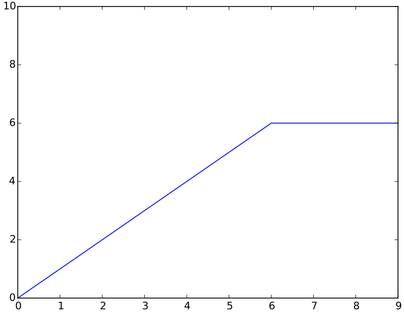
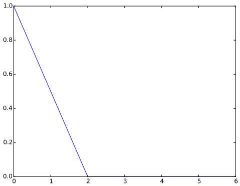


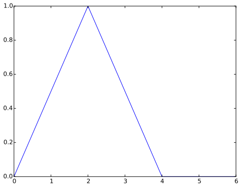
Figure: values of $\mathcal{X}_t(\xi^{t-1}, \xi_t)$ when ξ_t varies and keep ξ^{t-1}

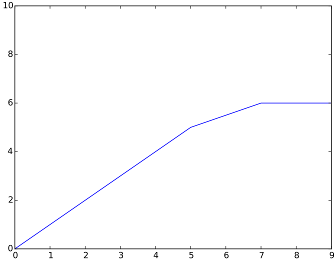
Generalized Decision Rule Approach

Put decisions as a linear combination of historic information?

No! Put it as a linear combination of basis functions!







Approach

Forget about historic information temporarily and use linear programming,

$$
\begin{aligned}
\max_{\mathcal{X}_t} \quad & \mathbb{E}_\xi\left(\sum_{t=1}^{T}\sum_{j=1}^{J} v_j X_{t,j} L(\xi)\right) \\
s.t. \quad & 0 \le X_{t,j} L(\xi) \le \xi_{t,j} \\
& \sum_{t=1}^{T}\sum_{j=1}^{J} A X_{t,j} L(\xi) \le 2\mathbf{c} - \mathbb{E}\left(\sum_{t=1}^{T}\sum_{j=1}^{J} A X_{t,j} L(\xi)\right) \\
& \forall \xi \in \Xi, t = 1, \ldots, T
\end{aligned} \tag{3}
$$

where $L(\xi)$ lift ξ to values of basis functions,and $X_{t,j}$ is actually coefficient matrixes.

Details in above equation

- It's not hard to prove that shapes of $X_{t,j}L(\xi)$ will be something like we required. So we can get booking limits from maximum of $X_{t,j}L(\xi)$ and then implement polices.
- $\mathbf{c} - \mathbb{E}(\sum_{t=1}^{T}\sum_{j=1}^{J} AX_{t,j}L(\xi))$ is added to ensure that we can make use of resources as much as possible. In fact, $\sum_{t=1}^{T}\sum_{j=1}^{J} AX_{t,j}L(\xi) \approx 2\mathbf{c} - \mathbb{E}(\sum_{t=1}^{T}\sum_{j=1}^{J} AX_{t,j}L(\xi))$.So expectations of used resources will be c.

Now we turn to problems of making use of historic information.

Historic information

A usual way is using linear combinations of historic information,that is, $X(\xi^{t-1}, \xi_t) = X_1(\xi_t) + X_2(\xi^{t-1})$. However,it couldn't give a good results,as we have constraints $X(\xi^{t-1}, \xi_t) \leq \xi_t$ and $X_1(\xi_t) \leq \xi_t$. It suggests that we cannot use linear way to approach the multivariate function.

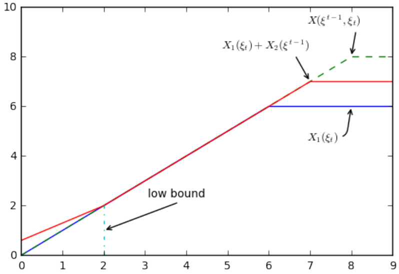


Figure: Cases of low bound of $\xi_t \geq 0$, we can see the limitation of the method

Historic information

- Remember that we didn't give $L(\cdot)$ a specific form.
- Now we are going to set elements of $L(\cdot)$ be functions of two variables and deal with problems in 2-dimension seems like in 1-dimension.
- Simple triangulations are required to approach with piecewise linear functions in 2-dimension.

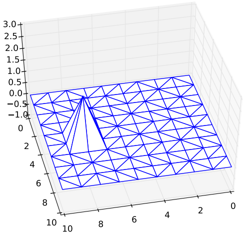


Figure: a basis function in 2-dimension

Solving Nonlinear Optimization in 2-d with Linear Programming

- What we are actually doing is solving a nonlinear optimization in 2-dimension with linear programming !

Computational Results

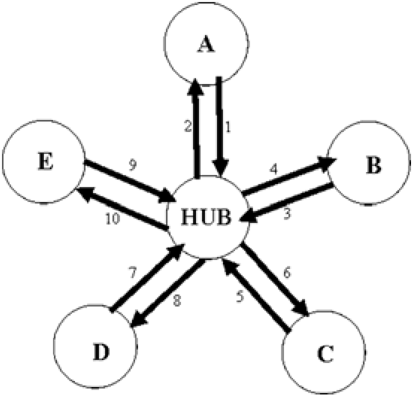


Figure: Network in Re-solving stochastic programming models for airline revenue management

Computational Results

Re-solving:

Computational Results

No re-solving	Uniform re-solving	Trick re-solving
415410	419262	421894

Table: Results from re-solving paper

Stages	5	10	20
	420364	419474	418527

Table: Results from our paper

And actually our method are much faster and can also use re-solving to promote results and use ideas from Reductions of Approximate Linear Programs for Network Revenue Management!

Computational Results

- Number of variables is only $O(T \times J)$.
- Our method is less customize and more intelligent.
- Decision rule approach can give a fractional estimation of booking limits.After rounding, we can use Zhang's method to give a better solution in need of a higher accuracy.

Future Work

- As we can see, we can generalize original problems into higher dimension. In the meantime, we have to face curse of dimensionality.
- Not only $X(\sum_{\tau=1}^{t-1} \xi_\tau, \xi_t)$ but also $X(\max_{\tau=1,\ldots,t-1} \xi_\tau, \xi_t)$ is available. It hints us that the methodology is suitable to approximate dynamic programs in other scenrios.
- Rounding policy.