Integer Programming Models: Constructing an Index Fund

Ranya Almohsen[1]

[1] Lane Department Of Computer Science And Electrical Engineering
West Virginia University
Morgantown, WV USA

April 14, 2015

Outline

Outline

Outline

Outline

Outline

Outline

Classification of Auctions

Classification of Auctions

- An Auction is a mechanism to allocate a set of goods to a set of bidders on the basis of their bids.

Classification of Auctions

- An Auction is a mechanism to allocate a set of goods to a set of bidders on the basis of their bids.
- Framework for classifying auctions based on the requirements that need to be considered to set up an auction:

Classification of Auctions

- An Auction is a mechanism to allocate a set of goods to a set of bidders on the basis of their bids.
- Framework for classifying auctions based on the requirements that need to be considered to set up an auction:
  - Resources

Classification of Auctions

- An Auction is a mechanism to allocate a set of goods to a set of bidders on the basis of their bids.
- Framework for classifying auctions based on the requirements that need to be considered to set up an auction:
  - Resources
  - Market Structure

Classification of Auctions

- An Auction is a mechanism to allocate a set of goods to a set of bidders on the basis of their bids.
- Framework for classifying auctions based on the requirements that need to be considered to set up an auction:
  - Resources
  - Market Structure
  - Preference Structure

Classification of Auctions

- An Auction is a mechanism to allocate a set of goods to a set of bidders on the basis of their bids.
- Framework for classifying auctions based on the requirements that need to be considered to set up an auction:
  - Resources
  - Market Structure
  - Preference Structure
  - Bid Structure

Classification of Auctions

- An Auction is a mechanism to allocate a set of goods to a set of bidders on the basis of their bids.
- Framework for classifying auctions based on the requirements that need to be considered to set up an auction:
  - Resources
  - Market Structure
  - Preference Structure
  - Bid Structure
  - Matching Supply to Demand

Classification of Auctions

- An Auction is a mechanism to allocate a set of goods to a set of bidders on the basis of their bids.
- Framework for classifying auctions based on the requirements that need to be considered to set up an auction:
  - Resources
  - Market Structure
  - Preference Structure
  - Bid Structure
  - Matching Supply to Demand
  - Information Feedback

Classification of Auctions

- An Auction is a mechanism to allocate a set of goods to a set of bidders on the basis of their bids.
- Framework for classifying auctions based on the requirements that need to be considered to set up an auction:
  - Resources
  - Market Structure
  - Preference Structure
  - Bid Structure
  - Matching Supply to Demand
  - Information Feedback

Combinatorial Auctions

Combinatorial Auctions

- In combinatorial auctions (CAs), bidders can bid on combinations of items.

Combinatorial Auctions

- In combinatorial auctions (CAs), bidders can bid on combinations of items.
- For example, if A, B, and C are three different items, a bidder can place separate bids on seven possible combinations, namely, $\{A\}, \{B\}, \{C\}, \{A, B\}, \{B, C\}, \{C, A\}, and \{A, B, C\}$.

Combinatorial Auctions

- In combinatorial auctions (CAs), bidders can bid on combinations of items.
- For example, if A, B, and C are three different items, a bidder can place separate bids on seven possible combinations, namely, $\{A\}, \{B\}, \{C\}, \{A, B\}, \{B, C\}, \{C, A\}, and \{A, B, C\}$.
- In the case of CAs, the value of an item a bidder wins depends on other items that he wins. The notions of complementarity and substitutability are very important in CAs.

Combinatorial Auctions

- In combinatorial auctions (CAs), bidders can bid on combinations of items.
- For example, if A, B, and C are three different items, a bidder can place separate bids on seven possible combinations, namely, $\{A\}, \{B\}, \{C\}, \{A, B\}, \{B, C\}, \{C, A\}, and \{A, B, C\}$.
- In the case of CAs, the value of an item a bidder wins depends on other items that he wins. The notions of complementarity and substitutability are very important in CAs.
- Complementarity:

Combinatorial Auctions

- In combinatorial auctions (CAs), bidders can bid on combinations of items.
- For example, if A, B, and C are three different items, a bidder can place separate bids on seven possible combinations, namely, $\{A\}, \{B\}, \{C\}, \{A, B\}, \{B, C\}, \{C, A\}, and \{A, B, C\}$.
- In the case of CAs, the value of an item a bidder wins depends on other items that he wins. The notions of complementarity and substitutability are very important in CAs.
- Complementarity:
  - Suppose an auctioneer is selling different goods.

Combinatorial Auctions

- In combinatorial auctions (CAs), bidders can bid on combinations of items.
- For example, if A, B, and C are three different items, a bidder can place separate bids on seven possible combinations, namely, $\{A\}, \{B\}, \{C\}, \{A, B\}, \{B, C\}, \{C, A\}, and \{A, B, C\}$.
- In the case of CAs, the value of an item a bidder wins depends on other items that he wins. The notions of complementarity and substitutability are very important in CAs.
- Complementarity:
  - Suppose an auctioneer is selling different goods.
  - A bidder might be willing to pay more for the whole than the sum of what he is willing to pay for the parts.

Combinatorial Auctions

- In combinatorial auctions (CAs), bidders can bid on combinations of items.
- For example, if A, B, and C are three different items, a bidder can place separate bids on seven possible combinations, namely, $\{A\}, \{B\}, \{C\}, \{A, B\}, \{B, C\}, \{C, A\}, and \{A, B, C\}$.
- In the case of CAs, the value of an item a bidder wins depends on other items that he wins. The notions of complementarity and substitutability are very important in CAs.
- Complementarity:
  - Suppose an auctioneer is selling different goods.
  - A bidder might be willing to pay more for the whole than the sum of what he is willing to pay for the parts.

Combinatorial Auctions

- In combinatorial auctions (CAs), bidders can bid on combinations of items.
- For example, if A, B, and C are three different items, a bidder can place separate bids on seven possible combinations, namely, $\{A\}, \{B\}, \{C\}, \{A, B\}, \{B, C\}, \{C, A\}, and \{A, B, C\}$.
- In the case of CAs, the value of an item a bidder wins depends on other items that he wins. The notions of complementarity and substitutability are very important in CAs.
- Complementarity:
  - Suppose an auctioneer is selling different goods.
  - A bidder might be willing to pay more for the whole than the sum of what he is willing to pay for the parts.
- Substitutability:

Combinatorial Auctions

- In combinatorial auctions (CAs), bidders can bid on combinations of items.
- For example, if A, B, and C are three different items, a bidder can place separate bids on seven possible combinations, namely, $\{A\}, \{B\}, \{C\}, \{A, B\}, \{B, C\}, \{C, A\}, and \{A, B, C\}$.
- In the case of CAs, the value of an item a bidder wins depends on other items that he wins. The notions of complementarity and substitutability are very important in CAs.
- Complementarity:
  - Suppose an auctioneer is selling different goods.
  - A bidder might be willing to pay more for the whole than the sum of what he is willing to pay for the parts.
- Substitutability:
  - A bidder may be willing to pay for the whole only less than the sum of what he is willing to pay for parts.

Combinatorial Auctions

- In combinatorial auctions (CAs), bidders can bid on combinations of items.
- For example, if A, B, and C are three different items, a bidder can place separate bids on seven possible combinations, namely, $\{A\}, \{B\}, \{C\}, \{A, B\}, \{B, C\}, \{C, A\}, and \{A, B, C\}$.
- In the case of CAs, the value of an item a bidder wins depends on other items that he wins. The notions of complementarity and substitutability are very important in CAs.
- Complementarity:
  - Suppose an auctioneer is selling different goods.
  - A bidder might be willing to pay more for the whole than the sum of what he is willing to pay for the parts.
- Substitutability:
  - A bidder may be willing to pay for the whole only less than the sum of what he is willing to pay for parts.
  - This is the case if the bidder has a limited budget or the goods are similar or interchangeable.

Motivation Example: eBay

Motivation Example: eBay

Example (1)

Motivation Example: eBay

Example (1)

- On-line auctions such as eBay.

Motivation Example: eBay

Example (1)

- On-line auctions such as eBay.
- The Online Auction Format is a type of selling format.

Motivation Example: eBay

Example (1)

- On-line auctions such as eBay.
- The Online Auction Format is a type of selling format.
- A seller lists an item for a set amount of time.

Motivation Example: eBay

Example (1)

- On-line auctions such as eBay.
- The Online Auction Format is a type of selling format.
- A seller lists an item for a set amount of time.
- Buyers must place a bid to purchase.

Motivation Example: eBay

Example (1)

- On-line auctions such as eBay.
- The Online Auction Format is a type of selling format.
- A seller lists an item for a set amount of time.
- Buyers must place a bid to purchase.
- Buyers must place a bid higher than the last bid placed.

Motivation Example: eBay

Example (1)

- On-line auctions such as eBay.
- The Online Auction Format is a type of selling format.
- A seller lists an item for a set amount of time.
- Buyers must place a bid to purchase.
- Buyers must place a bid higher than the last bid placed.
- At the end of the auction the buyer with the highest bid wins that item.

Motivation Example: eBay

Example (1)

- On-line auctions such as eBay.
- The Online Auction Format is a type of selling format.
- A seller lists an item for a set amount of time.
- Buyers must place a bid to purchase.
- Buyers must place a bid higher than the last bid placed.
- At the end of the auction the buyer with the highest bid wins that item.

Combinatorial Auctions

Combinatorial Auctions

Problem description

Combinatorial Auctions

Problem description

- Auctioneer sells items $M = \{1, ..., m\}$.

Combinatorial Auctions

Problem description

- Auctioneer sells items $M = \{1, ..., m\}$.
- Bid is a pair $B_j = (S_j, p_j)$, where $S_j \subseteq M$ is a nonempty set of items and p_j is the price offer for this set.

Combinatorial Auctions

Problem description

- Auctioneer sells items $M = \{1, ..., m\}$.
- Bid is a pair $B_j = (S_j, p_j)$, where $S_j \subseteq M$ is a nonempty set of items and p_j is the price offer for this set.
- Suppose that the aucioneer has received n bids $B_1, ..., B_n$.

Combinatorial Auctions

Problem description

- Auctioneer sells items $M = \{1, ..., m\}$.
- Bid is a pair $B_j = (S_j, p_j)$, where $S_j \subseteq M$ is a nonempty set of items and p_j is the price offer for this set.
- Suppose that the aucioneer has received n bids $B_1, ..., B_n$.
- Question: How should auctioneer determine winners and losers in order to maximize his revenue?

Combinatorial Auctions

Combinatorial Auctions

Formulating Integer Programs

Combinatorial Auctions

Formulating Integer Programs

- Let x_j be a $0, 1$ variable that takes the value1 if bid B_j wins, and 0 if it loses.

Combinatorial Auctions

Formulating Integer Programs

- Let x_j be a $0, 1$ variable that takes the value1 if bid B_j wins, and 0 if it loses.
- The auctioneer maximizes his revenue by solving the integer program:

Combinatorial Auctions

Formulating Integer Programs

- Let x_j be a $0, 1$ variable that takes the value1 if bid B_j wins, and 0 if it loses.
- The auctioneer maximizes his revenue by solving the integer program:

$$\max \sum_{i=1}^{n} p_j x_j$$

Combinatorial Auctions

Formulating Integer Programs

- Let x_j be a $0, 1$ variable that takes the value1 if bid B_j wins, and 0 if it loses.
- The auctioneer maximizes his revenue by solving the integer program:

$$\max \sum_{i=1}^{n} p_j x_j$$

$$Subject \ \ to :$$

Combinatorial Auctions

Formulating Integer Programs

- Let x_j be a $0, 1$ variable that takes the value1 if bid B_j wins, and 0 if it loses.
- The auctioneer maximizes his revenue by solving the integer program:

$$\max \sum_{i=1}^{n} p_j x_j$$

$$Subject\ to: \sum_{j:i \in S_j} x_j \leq 1, \quad \forall\, i = 1, \ldots, m$$

Combinatorial Auctions

Formulating Integer Programs

- Let x_j be a $0, 1$ variable that takes the value1 if bid B_j wins, and 0 if it loses.
- The auctioneer maximizes his revenue by solving the integer program:

$$\max \sum_{i=1}^{n} p_j x_j$$

$$Subject\ to: \sum_{j:i \in S_j} x_j \leq 1, \quad \forall\, i = 1, \ldots, m$$

$$x_j = 0\ \ or\ \ 1, \quad \forall\, j = 1, \ldots, n$$

Combinatorial Auctions

Combinatorial Auctions

Example

Combinatorial Auctions

Example

- If there are four items for sale and the following bids have been received:

Combinatorial Auctions

Example

- If there are four items for sale and the following bids have been received:
- $B_1 = (\{1\}, 6), B_2 = (\{2\}, 3), B_3 = (\{3, 4\}, 12), B_4 = (\{1, 3\}, 12), B_5 = (\{2, 4\}, 8), B_6 = (\{1, 3, 4\}, 16),$

Combinatorial Auctions

Example

- If there are four items for sale and the following bids have been received:
- $B_1 = (\{1\}, 6), B_2 = (\{2\}, 3), B_3 = (\{3, 4\}, 12), B_4 = (\{1, 3\}, 12), B_5 = (\{2, 4\}, 8), B_6 = (\{1, 3, 4\}, 16)$,
- The winners can be determined by the following integer program:

Combinatorial Auctions

Example

- If there are four items for sale and the following bids have been received:
- $B_1 = (\{1\}, 6), B_2 = (\{2\}, 3), B_3 = (\{3, 4\}, 12), B_4 = (\{1, 3\}, 12), B_5 = (\{2, 4\}, 8), B_6 = (\{1, 3, 4\}, 16)$,
- The winners can be determined by the following integer program:

$$\max \quad 6x_1 + 3x_2 + 12x_3 + 12x_4 + 8x_5 + 16x_6$$

Combinatorial Auctions

Example

- If there are four items for sale and the following bids have been received:
- $B_1 = (\{1\}, 6), B_2 = (\{2\}, 3), B_3 = (\{3, 4\}, 12), B_4 = (\{1, 3\}, 12), B_5 = (\{2, 4\}, 8), B_6 = (\{1, 3, 4\}, 16)$,
- The winners can be determined by the following integer program:

$$\max \; 6x_1 + 3x_2 + 12x_3 + 12x_4 + 8x_5 + 16x_6$$

$$x_1 + x_4 + x_6 \leq 1$$

Combinatorial Auctions

Example

- If there are four items for sale and the following bids have been received:
- $B_1 = (\{1\}, 6), B_2 = (\{2\}, 3), B_3 = (\{3, 4\}, 12), B_4 = (\{1, 3\}, 12), B_5 = (\{2, 4\}, 8), B_6 = (\{1, 3, 4\}, 16),$
- The winners can be determined by the following integer program:

$$\begin{aligned} \max \quad 6x_1 + 3x_2 + 12x_3 + 12x_4 + 8x_5 + 16x_6 \\ x_1 + x_4 + x_6 \leq 1 \\ x_2 + x_5 \leq 1 \end{aligned}$$

Combinatorial Auctions

Example

- If there are four items for sale and the following bids have been received:
- $B_1 = (\{1\}, 6), B_2 = (\{2\}, 3), B_3 = (\{3, 4\}, 12), B_4 = (\{1, 3\}, 12), B_5 = (\{2, 4\}, 8), B_6 = (\{1, 3, 4\}, 16),$
- The winners can be determined by the following integer program:

$$
\begin{aligned}
\max \quad 6x_1 + 3x_2 + 12x_3 + 12x_4 + 8x_5 + 16x_6 \\
x_1 + x_4 + x_6 \leq 1 \\
x_2 + x_5 \leq 1 \\
x_3 + x_4 + x_6 \leq 1
\end{aligned}
$$

Combinatorial Auctions

Example

- If there are four items for sale and the following bids have been received:
- $B_1 = (\{1\}, 6), B_2 = (\{2\}, 3), B_3 = (\{3, 4\}, 12), B_4 = (\{1, 3\}, 12), B_5 = (\{2, 4\}, 8), B_6 = (\{1, 3, 4\}, 16)$,
- The winners can be determined by the following integer program:

$$\begin{aligned} \max \quad 6x_1 + 3x_2 + 12x_3 + 12x_4 + 8x_5 + 16x_6 \\ x_1 + x_4 + x_6 \leq 1 \\ x_2 + x_5 \leq 1 \\ x_3 + x_4 + x_6 \leq 1 \\ x_3 + x_5 + x_6 \leq 1 \end{aligned}$$

Combinatorial Auctions

Example

- If there are four items for sale and the following bids have been received:
- $B_1 = (\{1\}, 6), B_2 = (\{2\}, 3), B_3 = (\{3, 4\}, 12), B_4 = (\{1, 3\}, 12), B_5 = (\{2, 4\}, 8), B_6 = (\{1, 3, 4\}, 16),$
- The winners can be determined by the following integer program:

$$
\begin{aligned}
\max \quad 6x_1 + 3x_2 + 12x_3 + 12x_4 + 8x_5 + 16x_6 \\
x_1 + x_4 + x_6 \leq 1 \\
x_2 + x_5 \leq 1 \\
x_3 + x_4 + x_6 \leq 1 \\
x_3 + x_5 + x_6 \leq 1 \\
x_j = 0 \;\; or \;\; 1 \quad , \forall\, j = 1, \ldots, 6
\end{aligned}
$$

Combinatorial Auctions

Example

- If there are four items for sale and the following bids have been received:
- $B_1 = (\{1\}, 6), B_2 = (\{2\}, 3), B_3 = (\{3, 4\}, 12), B_4 = (\{1, 3\}, 12), B_5 = (\{2, 4\}, 8), B_6 = (\{1, 3, 4\}, 16),$
- The winners can be determined by the following integer program:

$$
\begin{aligned}
\max \quad 6x_1 + 3x_2 + 12x_3 + 12x_4 + 8x_5 + 16x_6 \\
x_1 + x_4 + x_6 \leq 1 \\
x_2 + x_5 \leq 1 \\
x_3 + x_4 + x_6 \leq 1 \\
x_3 + x_5 + x_6 \leq 1 \\
x_j = 0 \;\; or \;\; 1 \quad , \forall\, j = 1, \ldots, 6
\end{aligned}
$$

Combinatorial Auctions

Combinatorial Auctions

Several indistinguishable items

Combinatorial Auctions

Several indistinguishable items

- In some auctions, there are multiple indistinguishable units of each item for sale.

Combinatorial Auctions

Several indistinguishable items

- In some auctions, there are multiple indistinguishable units of each item for sale.
- A bid in this setting is defined as $B_j = (\lambda_1^j, \lambda_2^j, \ldots, \lambda_m^j; p_j)$.

Combinatorial Auctions

Several indistinguishable items

- In some auctions, there are multiple indistinguishable units of each item for sale.
- *A* bid in this setting is defined as $B_j = (\lambda_1^j, \lambda_2^j, \ldots, \lambda_m^j; p_j)$.
- Where λ_i^j is the desired number of units of item i and p_j is the price offer.

Combinatorial Auctions

Combinatorial Auctions

Several indistinguishable items

Combinatorial Auctions

Several indistinguishable items

Combinatorial Auctions

Several indistinguishable items

- The auctioneer maximizes his revenue by solving the integer program:

$$\max \sum_{i=1}^{n} p_j x_j$$

Combinatorial Auctions

Several indistinguishable items

- The auctioneer maximizes his revenue by solving the integer program:

$$\max \sum_{i=1}^{n} p_j x_j$$

Subject to:

$$\sum_{j:i\in S_j} \lambda_i^j x_j \leq u_i \quad , \forall\, i = 1, \ldots, m$$

Combinatorial Auctions

Several indistinguishable items

- The auctioneer maximizes his revenue by solving the integer program:

$$\max \sum_{i=1}^{n} p_j x_j$$

Subject to:

$$\sum_{j:i\in S_j} \lambda_i^j x_j \leq u_i \quad , \forall\, i = 1, \ldots, m$$

$$x_j = 0 \;\; or \;\; 1 \quad , \forall\, j = 1, \ldots, n$$

Combinatorial Auctions

Several indistinguishable items

- The auctioneer maximizes his revenue by solving the integer program:

$$\max \sum_{i=1}^{n} p_j x_j$$

Subject to:

$$\sum_{j:i\in S_j} \lambda_i^j x_j \leq u_i \quad , \forall\, i = 1, \ldots, m$$

$$x_j = 0 \;\; or \;\; 1 \quad , \forall\, j = 1, \ldots, n$$

- Where u_i is the number of units of item i for sale.

Combinatorial Auctions

Combinatorial Auctions

Exercise

Combinatorial Auctions

Exercise

- In a combinatorial exchange, both buyers and sellers can submit combinatorial bids.

Combinatorial Auctions

Exercise

- In a combinatorial exchange, both buyers and sellers can submit combinatorial bids.
- Bids are like in the multiple item case, except that the λ_i^j values can be negative, as can the prices p_j , representing selling instead of buying.

Combinatorial Auctions

Exercise

- In a combinatorial exchange, both buyers and sellers can submit combinatorial bids.
- Bids are like in the multiple item case, except that the λ_i^j values can be negative, as can the prices p_j , representing selling instead of buying.
- Note that a single bid can be buying some items while selling other items.

Combinatorial Auctions

Exercise

- In a combinatorial exchange, both buyers and sellers can submit combinatorial bids.
- Bids are like in the multiple item case, except that the λ_i^j values can be negative, as can the prices p_j , representing selling instead of buying.
- Note that a single bid can be buying some items while selling other items.
- Write an integer linear program that will maximize the surplus generated by the combinatorial exchange.

Combinatorial Auctions

Exercise

- In a combinatorial exchange, both buyers and sellers can submit combinatorial bids.
- Bids are like in the multiple item case, except that the λ_i^j values can be negative, as can the prices p_j , representing selling instead of buying.
- Note that a single bid can be buying some items while selling other items.
- Write an integer linear program that will maximize the surplus generated by the combinatorial exchange.

Combinatorial Auctions

Combinatorial Auctions

Exercise

Combinatorial Auctions

Exercise

- Let $B = \{1, \ldots, m\}$ be the set of buyer items, and the bid is $b_j = (\lambda_1^j, \lambda_2^j, \ldots, \lambda_m^j; p_j)$, such that: $b_j \subseteq B$.

Combinatorial Auctions

Exercise

- Let $B = \{1, \ldots, m\}$ be the set of buyer items, and the bid is $b_j = (\lambda_1^j, \lambda_2^j, \ldots, \lambda_m^j; p_j)$, such that: $b_j \subseteq B$.
- Let $S = \{1, \ldots, n\}$ be the set of seller items, and and the bid is $s_j = (\lambda_1^j, \lambda_2^j, \ldots, \lambda_m^j; p_j)$, such that: $s_j \subseteq S$.

Combinatorial Auctions

Exercise

- Let $B = \{1, \ldots, m\}$ be the set of buyer items, and the bid is $b_j = (\lambda_1^j, \lambda_2^j, \ldots, \lambda_m^j; p_j)$, such that: $b_j \subseteq B$.
- Let $S = \{1, \ldots, n\}$ be the set of seller items, and and the bid is $s_j = (\lambda_1^j, \lambda_2^j, \ldots, \lambda_m^j; p_j)$, such that: $s_j \subseteq S$.
- $x_j = 0$ or 1 indicates b_j wins, $y_j = 0$ or 1 indicates s_j wins.

Combinatorial Auctions

Exercise

- Let $B = \{1, \ldots, m\}$ be the set of buyer items, and the bid is $b_j = (\lambda_1^j, \lambda_2^j, \ldots, \lambda_m^j; p_j)$, such that: $b_j \subseteq B$.
- Let $S = \{1, \ldots, n\}$ be the set of seller items, and and the bid is $s_j = (\lambda_1^j, \lambda_2^j, \ldots, \lambda_m^j; p_j)$, such that: $s_j \subseteq S$.
- $x_j = 0$ or 1 indicates b_j wins, $y_j = 0$ or 1 indicates s_j wins.

$$\max \sum_{j \in B}^{n} p_j x_j - \sum_{j \in S}^{n} p_j y_j$$

Combinatorial Auctions

Exercise

- Let $B = \{1, \ldots, m\}$ be the set of buyer items, and the bid is $b_j = (\lambda_1^j, \lambda_2^j, \ldots, \lambda_m^j; p_j)$, such that: $b_j \subseteq B$.
- Let $S = \{1, \ldots, n\}$ be the set of seller items, and and the bid is $s_j = (\lambda_1^j, \lambda_2^j, \ldots, \lambda_m^j; p_j)$, such that: $s_j \subseteq S$.
- $x_j = 0$ or 1 indicates b_j wins, $y_j = 0$ or 1 indicates s_j wins.

$$\max \sum_{j \in B}^{n} p_j x_j - \sum_{j \in S}^{n} p_j y_j$$

$$\textit{Subject to}: \sum_{j:i \in B_j} \lambda_i^j x_j = \sum_{j:i \in S_j} \lambda_i^j y_j$$

Combinatorial Auctions

Exercise

- Let $B = \{1, \ldots, m\}$ be the set of buyer items, and the bid is $b_j = (\lambda^j_1, \lambda^j_2, \ldots, \lambda^j_m; p_j)$, such that: $b_j \subseteq B$.
- Let $S = \{1, \ldots, n\}$ be the set of seller items, and and the bid is $s_j = (\lambda^j_1, \lambda^j_2, \ldots, \lambda^j_m; p_j)$, such that: $s_j \subseteq S$.
- $x_j = 0$ or 1 indicates b_j wins, $y_j = 0$ or 1 indicates s_j wins.

$$\max \sum_{j \in B}^{n} p_j x_j - \sum_{j \in S}^{n} p_j y_j$$

$$\textit{Subject to}: \sum_{j: i \in B_j} \lambda^j_i x_j = \sum_{j: i \in S_j} \lambda^j_i y_j$$

$$x_j = 0 \ \ \textit{or} \ \ 1 \quad , \forall\, j = 1, \ldots, m,$$

$$y_j = 0 \ \ \textit{or} \ \ 1 \quad , \forall\, j = 1, \ldots, n$$

The Lockbox Problem

The Lockbox Problem

Construct the Problem

The Lockbox Problem

Construct the Problem

- National firm in US receives checks from all over the country.

The Lockbox Problem

Construct the Problem

- National firm in US receives checks from all over the country.
- Delay from obligation of customer (check postmarked) to clearing (check arrives).

The Lockbox Problem

Construct the Problem

- National firm in US receives checks from all over the country.
- Delay from obligation of customer (check postmarked) to clearing (check arrives).
- Money should be available as soon as possible.

The Lockbox Problem

Construct the Problem

- National firm in US receives checks from all over the country.
- Delay from obligation of customer (check postmarked) to clearing (check arrives).
- Money should be available as soon as possible.
- Idea: Open offices all over country to receive checks and to minimize delay.

The Lockbox Problem

The Lockbox Problem

Example

The Lockbox Problem

Example

- Suppose we receive payments from 4 regions (West, Midwest, East, and South).

The Lockbox Problem

Example

- Suppose we receive payments from 4 regions (West, Midwest, East, and South).
- Average daily value from each region is: $600 K, $240 K, $720 K, $360 K respectively.

The Lockbox Problem

Example

- Suppose we receive payments from 4 regions (West, Midwest, East, and South).
- Average daily value from each region is: $600 K, $240 K, $720 K, $360 K respectively.
- We are considering opening lockboxes in Los Angeles, Pittsburgh, Boston, and/or Houston.

The Lockbox Problem

Example

- Suppose we receive payments from 4 regions (West, Midwest, East, and South).
- Average daily value from each region is: \$600 K, \$240 K, \$720 K, \$360 K respectively.
- We are considering opening lockboxes in Los Angeles, Pittsburgh, Boston, and/or Houston.
- Operating a lockbox costs $\$90,000$ per year.

The Lockbox Problem

The Lockbox Problem

Example

The Lockbox Problem

Example

Clearing times				
From	L.A	Pittsburgh	Boston	Houston
West	2	4	6	6
Midwest	4	2	5	5
East	6	5	2	5
South	7	5	6	3

The Lockbox Problem

The Lockbox Problem

Example Cont.

The Lockbox Problem

Example Cont.

- First we must calculate the lost interest for each possible assignment.

The Lockbox Problem

Example Cont.

- First we must calculate the lost interest for each possible assignment.
- For example, if the West sends its checks to a lockbox in Boston, then on average there will be $\$3,600,000 = (6 \times \$600,000)$ in process on any given day.

The Lockbox Problem

Example Cont.

- First we must calculate the lost interest for each possible assignment.
- For example, if the West sends its checks to a lockbox in Boston, then on average there will be $\$3,600,000 = (6 \times \$600,000)$ in process on any given day.
- Assuming an investment rate of 5%, this corresponds to a yearly loss of $\$180,000$.

The Lockbox Problem

The Lockbox Problem

Example Cont.

The Lockbox Problem

Example Cont.

Lost Interest				
From	L.A	Pittsburgh	Boston	Houston
West	60	120	180	180
Midwest	48	24	60	60
East	216	180	72	180
South	126	90	108	54

The Lockbox Problem

The Lockbox Problem

Example Cont.,Integer Programming Formulation

The Lockbox Problem

Example Cont.,Integer Programming Formulation

- Let $y_j \in \{0, 1\}$ indicates whether lockbox j is open or not.

The Lockbox Problem

Example Cont.,Integer Programming Formulation

- Let $y_j \in \{0, 1\}$ indicates whether lockbox j is open or not.
- Let $x_{ij} = 1$ if region i sends checks to lockbox j.

The Lockbox Problem

Example Cont.,Integer Programming Formulation

- Let $y_j \in \{0, 1\}$ indicates whether lockbox j is open or not.
- Let $x_{ij} = 1$ if region i sends checks to lockbox j.
- What is the objective function?

The Lockbox Problem

Example Cont.,Integer Programming Formulation

- Let $y_j \in \{0, 1\}$ indicates whether lockbox j is open or not.
- Let $x_{ij} = 1$ if region i sends checks to lockbox j.
- What is the objective function?
- The objective is to minimize total yearly costs:

The Lockbox Problem

Example Cont.,Integer Programming Formulation

- Let $y_j \in \{0, 1\}$ indicates whether lockbox j is open or not.
- Let $x_{ij} = 1$ if region i sends checks to lockbox j.
- What is the objective function?
- The objective is to minimize total yearly costs:
  $60x_{11} + 120x_{12} + 180x_{13} + 180x_{14} + 48x_{21} + \dots$
  $+90y_1 + 90y_2 + 90y_3 + 90y_4$.

The Lockbox Problem

Example Cont.,Integer Programming Formulation

- Let $y_j \in \{0, 1\}$ indicates whether lockbox j is open or not.
- Let $x_{ij} = 1$ if region i sends checks to lockbox j.
- What is the objective function?
- The objective is to minimize total yearly costs:
  $60x_{11} + 120x_{12} + 180x_{13} + 180x_{14} + 48x_{21} + \dots$
  $+90y_1 + 90y_2 + 90y_3 + 90y_4$.
- Each region is assigned to exactly one lockbox:

The Lockbox Problem

Example Cont.,Integer Programming Formulation

- Let $y_j \in \{0, 1\}$ indicates whether lockbox j is open or not.
- Let $x_{ij} = 1$ if region i sends checks to lockbox j.
- What is the objective function?
- The objective is to minimize total yearly costs:
  $60x_{11} + 120x_{12} + 180x_{13} + 180x_{14} + 48x_{21} + \dots$
  $+90y_1 + 90y_2 + 90y_3 + 90y_4$.
- Each region is assigned to exactly one lockbox: $\sum_j x_{ij} = 1$ for all i.

The Lockbox Problem

The Lockbox Problem

Example cont.

The Lockbox Problem

Example cont.

- Regions can only send to open lockboxes:

The Lockbox Problem

Example cont.

- Regions can only send to open lockboxes:
  $\sum_i x_{ij} \leq 4y_j$ for all j.

The Lockbox Problem

Example cont.

- Regions can only send to open lockboxes:
  $\sum_i x_{ij} \leq 4y_j$ for all j.
- For lockbox 1 (Los Angeles), this can be written as:

The Lockbox Problem

Example cont.

- Regions can only send to open lockboxes:
  $\sum_i x_{ij} \leq 4y_j$ for all j.
- For lockbox 1 (Los Angeles), this can be written as:
  $x_{11} + x_{21} + x_{31} + x_{41} \leq 4y_1$.

The Lockbox Problem

The Lockbox Problem

Integer Programming Formulation

The Lockbox Problem

Integer Programming Formulation

- min $60x_{11} + 120x_{12} + 180x_{13} + 180x_{14} + 48x_{21} + 24x_{22} + 60x_{23} + 60x_{24} + 216x_{31} + 180x_{32} + 72x_{33} + 180x_{34} + 126x_{41} + 90x_{42} + 108x_{43} + 54x_{44} + 90y_1 + 90y_2 + 90y_3 + 90y_4$.

The Lockbox Problem

Integer Programming Formulation

- min $60x_{11} + 120x_{12} + 180x_{13} + 180x_{14} + 48x_{21} + 24x_{22} + 60x_{23} + 60x_{24} + 216x_{31} + 180x_{32} + 72x_{33} + 180x_{34} + 126x_{41} + 90x_{42} + 108x_{43} + 54x_{44} + 90y_1 + 90y_2 + 90y_3 + 90y_4$.
- Subject to:

The Lockbox Problem

Integer Programming Formulation

- min $60x_{11} + 120x_{12} + 180x_{13} + 180x_{14} + 48x_{21} + 24x_{22} + 60x_{23} + 60x_{24} + 216x_{31} + 180x_{32} + 72x_{33} + 180x_{34} + 126x_{41} + 90x_{42} + 108x_{43} + 54x_{44} + 90y_1 + 90y_2 + 90y_3 + 90y_4$.
- Subject to:

$$x_{11} + x_{12} + x_{13} + x_{14} = 1$$
$$x_{21} + x_{22} + x_{23} + x_{24} = 1$$
$$x_{31} + x_{32} + x_{33} + x_{34} = 1$$
$$x_{41} + x_{42} + x_{43} + x_{44} = 1$$
$$x_{11} + x_{21} + x_{31} + x_{41} - 4y_1 \leq 0$$
$$x_{12} + x_{22} + x_{32} + x_{42} - 4y_2 \leq 0$$
$$x_{13} + x_{23} + x_{33} + x_{43} - 4y_3 \leq 0$$
$$x_{14} + x_{24} + x_{34} + x_{44} - 4y_4 \leq 0$$

The Lockbox Problem

Integer Programming Formulation

- min $60x_{11} + 120x_{12} + 180x_{13} + 180x_{14} + 48x_{21} + 24x_{22} + 60x_{23} + 60x_{24} + 216x_{31} + 180x_{32} + 72x_{33} + 180x_{34} + 126x_{41} + 90x_{42} + 108x_{43} + 54x_{44} + 90y_1 + 90y_2 + 90y_3 + 90y_4$.
- Subject to:

$$x_{11} + x_{12} + x_{13} + x_{14} = 1$$
$$x_{21} + x_{22} + x_{23} + x_{24} = 1$$
$$x_{31} + x_{32} + x_{33} + x_{34} = 1$$
$$x_{41} + x_{42} + x_{43} + x_{44} = 1$$
$$x_{11} + x_{21} + x_{31} + x_{41} - 4y_1 \leq 0$$
$$x_{12} + x_{22} + x_{32} + x_{42} - 4y_2 \leq 0$$
$$x_{13} + x_{23} + x_{33} + x_{43} - 4y_3 \leq 0$$
$$x_{14} + x_{24} + x_{34} + x_{44} - 4y_4 \leq 0$$

- Al variables binary.

The Lockbox Problem

The Lockbox Problem

Integer Programming Formulation

The Lockbox Problem

Integer Programming Formulation

- If we ignore integrality, then we can solve it as linear program.

The Lockbox Problem

Integer Programming Formulation

- If we ignore integrality, then we can solve it as linear program.
- We get the solution $x_{11} = x_{22} = x_{33} = x_{44} = 1$, $y_1 = y_2 = y_3 = y_4 = 0.25$ and the rest equals 0.

The Lockbox Problem

Integer Programming Formulation

- If we ignore integrality, then we can solve it as linear program.
- We get the solution $x_{11} = x_{22} = x_{33} = x_{44} = 1$, $y_1 = y_2 = y_3 = y_4 = 0.25$ and the rest equals 0.
- Note that we get no useful information out of this linear programming solution: all 4 regions look the same.

The Lockbox Problem

Integer Programming Formulation

- If we ignore integrality, then we can solve it as linear program.
- We get the solution $x_{11} = x_{22} = x_{33} = x_{44} = 1$, $y_1 = y_2 = y_3 = y_4 = 0.25$ and the rest equals 0.
- Note that we get no useful information out of this linear programming solution: all 4 regions look the same.

Other Formulations

The Lockbox Problem

Integer Programming Formulation

- If we ignore integrality, then we can solve it as linear program.
- We get the solution $x_{11} = x_{22} = x_{33} = x_{44} = 1$, $y_1 = y_2 = y_3 = y_4 = 0.25$ and the rest equals 0.
- Note that we get no useful information out of this linear programming solution: all 4 regions look the same.

Other Formulations

- Consider the sixteen constraints of the form:

The Lockbox Problem

Integer Programming Formulation

- If we ignore integrality, then we can solve it as linear program.
- We get the solution $x_{11} = x_{22} = x_{33} = x_{44} = 1$, $y_1 = y_2 = y_3 = y_4 = 0.25$ and the rest equals 0.
- Note that we get no useful information out of this linear programming solution: all 4 regions look the same.

Other Formulations

- Consider the sixteen constraints of the form:
  $x_{ij} \leq y_j$

The Lockbox Problem

Integer Programming Formulation

- If we ignore integrality, then we can solve it as linear program.
- We get the solution $x_{11} = x_{22} = x_{33} = x_{44} = 1$, $y_1 = y_2 = y_3 = y_4 = 0.25$ and the rest equals 0.
- Note that we get no useful information out of this linear programming solution: all 4 regions look the same.

Other Formulations

- Consider the sixteen constraints of the form:

  $x_{ij} \leq y_j$
- These constraints also force a region to only use open lockboxes.

The Lockbox Problem

Integer Programming Formulation

- If we ignore integrality, then we can solve it as linear program.
- We get the solution $x_{11} = x_{22} = x_{33} = x_{44} = 1$, $y_1 = y_2 = y_3 = y_4 = 0.25$ and the rest equals 0.
- Note that we get no useful information out of this linear programming solution: all 4 regions look the same.

Other Formulations

- Consider the sixteen constraints of the form:
  $x_{ij} \leq y_j$
- These constraints also force a region to only use open lockboxes.
- If we solve the linear program with the above constraints, we get the solution $x_{11} = x_{21} = x_{33} = x_{43} = y_1 = y_3 = 1$, with the rest equal to zero.

The Lockbox Problem

Integer Programming Formulation

- If we ignore integrality, then we can solve it as linear program.
- We get the solution $x_{11} = x_{22} = x_{33} = x_{44} = 1$, $y_1 = y_2 = y_3 = y_4 = 0.25$ and the rest equals 0.
- Note that we get no useful information out of this linear programming solution: all 4 regions look the same.

Other Formulations

- Consider the sixteen constraints of the form:
  $x_{ij} \leq y_j$
- These constraints also force a region to only use open lockboxes.
- If we solve the linear program with the above constraints, we get the solution $x_{11} = x_{21} = x_{33} = x_{43} = y_1 = y_3 = 1$, with the rest equal to zero.
- In fact, we have an integer solution, which must therefore be optimal!

The Lockbox Problem

The Lockbox Problem

Exercise

The Lockbox Problem

Exercise

- Consider a lockbox problem where C_{ij} is the cost of assigning region i to a lockbox in region j, for $j = 1, \ldots, n$. Suppose that we wish to open exactly q lockboxes where q is a given integer, $1 \leq q \leq n$.

The Lockbox Problem

Exercise

- Consider a lockbox problem where C_{ij} is the cost of assigning region i to a lockbox in region j, for $j = 1, \ldots, n$. Suppose that we wish to open exactly q lockboxes where q is a given integer, $1 \leq q \leq n$.
  1. Formulate as an integer linear program the problem of opening q lockboxes so as to minimize the total cost of assigning each region to an open lockbox.

The Lockbox Problem

Exercise

- Consider a lockbox problem where C_{ij} is the cost of assigning region i to a lockbox in region j, for $j = 1, \ldots, n$. Suppose that we wish to open exactly q lockboxes where q is a given integer, $1 \leq q \leq n$.
  1. Formulate as an integer linear program the problem of opening q lockboxes so as to minimize the total cost of assigning each region to an open lockbox.
  2. Formulate in two different ways the constraint that regions cannot send checks to closed lockboxes.

The Lockbox Problem

Exercise

- Consider a lockbox problem where C_{ij} is the cost of assigning region i to a lockbox in region j, for $j = 1, \ldots, n$. Suppose that we wish to open exactly q lockboxes where q is a given integer, $1 \leq q \leq n$.
  1. Formulate as an integer linear program the problem of opening q lockboxes so as to minimize the total cost of assigning each region to an open lockbox.
  2. Formulate in two different ways the constraint that regions cannot send checks to closed lockboxes.
  3. For the following data,

The Lockbox Problem

Exercise

- Consider a lockbox problem where C_{ij} is the cost of assigning region i to a lockbox in region j, for $j = 1, \ldots, n$. Suppose that we wish to open exactly q lockboxes where q is a given integer, $1 \leq q \leq n$.
  1. Formulate as an integer linear program the problem of opening q lockboxes so as to minimize the total cost of assigning each region to an open lockbox.
  2. Formulate in two different ways the constraint that regions cannot send checks to closed lockboxes.
  3. For the following data,

$$q = 2, \mathbf{C}_{ij} = \begin{pmatrix} 0 & 4 & 5 & 8 & 2 \\ 4 & 0 & 3 & 4 & 6 \\ 5 & 3 & 0 & 1 & 7 \\ 8 & 4 & 1 & 0 & 4 \\ 2 & 6 & 7 & 4 & 0 \end{pmatrix}$$

- Compare the linear programming relaxations of your two formulations in question (2).

The Lockbox Problem

Exercise Solution

The Lockbox Problem

Exercise Solution

- For (1). Use decision variables

$$x_{ij} = \begin{cases} 1 & \text{if region } i \text{ is assigned to lock-box } j \\ 0 & \text{otherwise} \end{cases}$$

The Lockbox Problem

Exercise Solution

- For (1). Use decision variables

$$x_{ij} = \begin{cases} 1 & \text{if region } i \text{ is assigned to lock-box } j \\ 0 & \text{otherwise} \end{cases}$$

$$y_{ij} = \begin{cases} 1 & \text{if lock-box } j \text{ is opened} \\ 0 & \text{otherwise} \end{cases}$$

The Lockbox Problem

Exercise Solution Cont.

The Lockbox Problem

Exercise Solution Cont.

- The lock-box ILP then is

The Lockbox Problem

Exercise Solution Cont.

- The lock-box ILP then is

$$\max \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij}$$

The Lockbox Problem

Exercise Solution Cont.

- The lock-box ILP then is

$$\max \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij}$$

$$\sum_{j=1}^{n} y_j = q$$

The Lockbox Problem

Exercise Solution Cont.

- The lock-box ILP then is

$$\max \sum_{i=1}^{n}\sum_{j=1}^{n} c_{ij}x_{ij}$$

$$\sum_{j=1}^{n} y_j = q$$

$$\sum x_{ij} \geq 1 \quad , \forall i = 1, \ldots, n$$

The Lockbox Problem

Exercise Solution Cont.

- The lock-box ILP then is

$$\max \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij}$$

$$\sum_{j=1}^{n} y_j = q$$

$$\sum x_{ij} \geq 1 \quad , \forall i = 1, \ldots, n$$

$$\sum x_{ij} \leq n \cdot y_j \quad , \forall j = 1, \ldots, n$$

The Lockbox Problem

Exercise Solution Cont.

- The lock-box ILP then is

$$\max \sum_{i=1}^{n}\sum_{j=1}^{n} c_{ij}x_{ij}$$

$$\sum_{j=1}^{n} y_j = q$$

$$\sum x_{ij} \geq 1 \quad ,\forall i = 1,\ldots,n$$

$$\sum x_{ij} \leq n \cdot y_j \quad ,\forall j = 1,\ldots,n$$

$$x_{ij}, y_j \in [0,1].$$

The Lockbox Problem

Exercise Solution Cont.

The Lockbox Problem

Exercise Solution Cont.

- For (2). The 3rd inequality in the ILP forbids that we can assign regions to closed lock-boxes. Alternatively it can be expressed with

The Lockbox Problem

Exercise Solution Cont.

- For (2). The 3rd inequality in the ILP forbids that we can assign regions to closed lock-boxes. Alternatively it can be expressed with $x_{ij} \leq y_j \; \forall i = 1, \ldots, n, \;\; and \; \forall j = 1, \ldots, n$

The Lockbox Problem

Exercise Solution Cont.

- For (2). The 3rd inequality in the ILP forbids that we can assign regions to closed lock-boxes. Alternatively it can be expressed with $x_{ij} \leq y_j \; \forall i = 1, \ldots, n, \; and \; \forall j = 1, \ldots, n$
- Note that the integer solutions to both systems are exactly the same.

The Lockbox Problem

Exercise Solution Cont.

- For (2). The 3rd inequality in the ILP forbids that we can assign regions to closed lock-boxes. Alternatively it can be expressed with $x_{ij} \leq y_j \; \forall i = 1, \ldots, n, \; \textit{and} \; \forall j = 1, \ldots, n$
- Note that the integer solutions to both systems are exactly the same.
- For (3). One obtains the LP relaxation for above ILP by replacing the constraint x_{ij} ; $y_j \in [0, 1]$ by $0 \leq x_{ij}$;$y_j \leq 1$.

The Lockbox Problem

Exercise Solution Cont.

- For (2). The 3rd inequality in the ILP forbids that we can assign regions to closed lock-boxes. Alternatively it can be expressed with $x_{ij} \leq y_j \; \forall i = 1, \ldots, n, \;\; and \; \forall j = 1, \ldots, n$
- Note that the integer solutions to both systems are exactly the same.
- For (3). One obtains the LP relaxation for above ILP by replacing the constraint x_{ij} ; $y_j \in [0, 1]$ by $0 \leq x_{ij}$;$y_j \leq 1$.
- An optimum fractional solution of the system in (a) is $x_{ii} = 1$ and $y_i = 0.4$ for $i = 1, \ldots, n$ (other values are 0). The objective value of this solution is 0. On optimum solution of the same LP but with the constraint from (2) gives $y_3 = y_5 = 1; x_{15} = x_{55} = x_{23} = x_{33} = x_{43} = 1$ with a value of 6. This is even an integer solution.

The Lockbox Problem

Exercise Solution Cont.

- For (2). The 3rd inequality in the ILP forbids that we can assign regions to closed lock-boxes. Alternatively it can be expressed with $x_{ij} \leq y_j \; \forall i = 1, \dots, n, \; and \; \forall j = 1, \dots, n$
- Note that the integer solutions to both systems are exactly the same.
- For (3). One obtains the LP relaxation for above ILP by replacing the constraint x_{ij} ; $y_j \in [0, 1]$ by $0 \leq x_{ij}$;$y_j \leq 1$.
- An optimum fractional solution of the system in (a) is $x_{ii} = 1$ and $y_i = 0.4$ for $i = 1, \dots, n$ (other values are 0). The objective value of this solution is 0. On optimum solution of the same LP but with the constraint from (2) gives $y_3 = y_5 = 1; x_{15} = x_{55} = x_{23} = x_{33} = x_{43} = 1$ with a value of 6. This is even an integer solution.
- Conclusion: The constraint from (2) is stronger (and therefore better).

Definitions

Definitions

Active and Passive Portfolio

Definitions

Active and Passive Portfolio

- **Active portfolio** management tries to achieve superior performance by using technical and fundamental analysis as well as forecasting techniques.

Definitions

Active and Passive Portfolio

- **Active portfolio** management tries to achieve superior performance by using technical and fundamental analysis as well as forecasting techniques.
- **Passive portfolio** management avoids any forecasting techniques and rather relies on diversification to achieve a desired performance.

Definitions

Active and Passive Portfolio

- **Active portfolio** management tries to achieve superior performance by using technical and fundamental analysis as well as forecasting techniques.
- **Passive portfolio** management avoids any forecasting techniques and rather relies on diversification to achieve a desired performance.

Types of Passive Portfolio

Definitions

Active and Passive Portfolio

- **Active portfolio** management tries to achieve superior performance by using technical and fundamental analysis as well as forecasting techniques.
- **Passive portfolio** management avoids any forecasting techniques and rather relies on diversification to achieve a desired performance.

Types of Passive Portfolio

- Buy and hold: where assets are selected on the basis of some fundamental criteria and there is no active selling or buying of these stocks afterwards.

Definitions

Active and Passive Portfolio

- **Active portfolio** management tries to achieve superior performance by using technical and fundamental analysis as well as forecasting techniques.
- **Passive portfolio** management avoids any forecasting techniques and rather relies on diversification to achieve a desired performance.

Types of Passive Portfolio

- Buy and hold: where assets are selected on the basis of some fundamental criteria and there is no active selling or buying of these stocks afterwards.
- Indexing: absolutely no attempt is made to identify mispriced securities.

Definitions

Active and Passive Portfolio

- **Active portfolio** management tries to achieve superior performance by using technical and fundamental analysis as well as forecasting techniques.
- **Passive portfolio** management avoids any forecasting techniques and rather relies on diversification to achieve a desired performance.

Types of Passive Portfolio

- Buy and hold: where assets are selected on the basis of some fundamental criteria and there is no active selling or buying of these stocks afterwards.
- Indexing: absolutely no attempt is made to identify mispriced securities.
- The goal is to choose a portfolio that mirrors the movements of a broad market population or a market index. Such a portfolio is called an index fund.

Constructing an Index Fund

Constructing an Index Fund

Define an Index Fund

Constructing an Index Fund

Define an Index Fund

- An index fund is a portfolio designed to track the movement of the market as a whole or some selected broad market segment.

Constructing an Index Fund

Define an Index Fund

- An index fund is a portfolio designed to track the movement of the market as a whole or some selected broad market segment.
- The Rising Popularity of Index Funds.

Constructing an Index Fund

Define an Index Fund

- An index fund is a portfolio designed to track the movement of the market as a whole or some selected broad market segment.
- The Rising Popularity of Index Funds.
  - Market Efficiency.

Constructing an Index Fund

Define an Index Fund

- An index fund is a portfolio designed to track the movement of the market as a whole or some selected broad market segment.
- The Rising Popularity of Index Funds.
  - Market Efficiency.
  - Empirical Performance.

Constructing an Index Fund

Define an Index Fund

- An index fund is a portfolio designed to track the movement of the market as a whole or some selected broad market segment.
- The Rising Popularity of Index Funds.
  - Market Efficiency.
  - Empirical Performance.
  - Transaction Cost.

Constructing an Index Fund

Define an Index Fund

- An index fund is a portfolio designed to track the movement of the market as a whole or some selected broad market segment.
- The Rising Popularity of Index Funds.
  - Market Efficiency.
  - Empirical Performance.
  - Transaction Cost.

Constructing an Index Fund

Define an Index Fund

- An index fund is a portfolio designed to track the movement of the market as a whole or some selected broad market segment.
- The Rising Popularity of Index Funds.
  - Market Efficiency.
  - Empirical Performance.
  - Transaction Cost.

Constructing an Index Fund

Constructing an Index Fund

Strategies for Forming Index Funds

Constructing an Index Fund

Strategies for Forming Index Funds

- Choosing a broad market index as a proxy for an entire market, for example the Standard and Poor list of 500 stocks (S & P 500).

Constructing an Index Fund

Strategies for Forming Index Funds

- Choosing a broad market index as a proxy for an entire market, for example the Standard and Poor list of 500 stocks (S & P 500).
- A pure indexing approach consists in purchasing all the issues in the index, with the same exact weights as in the index, (impractical).

Constructing an Index Fund

Strategies for Forming Index Funds

- Choosing a broad market index as a proxy for an entire market, for example the Standard and Poor list of 500 stocks (S & P 500).
- A pure indexing approach consists in purchasing all the issues in the index, with the same exact weights as in the index, (impractical).
- An index fund with q stocks, where q is substantially smaller than the size n of the target population seems desirable.

Constructing an Index Fund

Strategies for Forming Index Funds

- Choosing a broad market index as a proxy for an entire market, for example the Standard and Poor list of 500 stocks (S & P 500).
- A pure indexing approach consists in purchasing all the issues in the index, with the same exact weights as in the index, (impractical).
- An index fund with q stocks, where q is substantially smaller than the size n of the target population seems desirable.

Constructing an Index Fund

Constructing an Index Fund

A Large-Scale Deterministic Model

Constructing an Index Fund

A Large-Scale Deterministic Model

- Suppose a measure of similarity is available.

Constructing an Index Fund

A Large-Scale Deterministic Model

- Suppose a measure of similarity is available.
- $\rho_{ij} =$ similarity between stock i and stock j.

Constructing an Index Fund

A Large-Scale Deterministic Model

- Suppose a measure of similarity is available.
- ρ_{ij} = similarity between stock i and stock j.
- For example, $\rho_{ii} = 1$, $\rho_{ij} \leq 1$ for $i \neq j$ and ρ_{ij} is larger for more similar stocks.

Constructing an Index Fund

A Large-Scale Deterministic Model

- Suppose a measure of similarity is available.
- ρ_{ij} = similarity between stock i and stock j.
- For example, $\rho_{ii} = 1$, $\rho_{ij} \leq 1$ for $i \neq j$ and ρ_{ij} is larger for more similar stocks.

Constructing an Index Fund

Example

Constructing an Index Fund

Example

- The correlation between the returns of stocks i and j. But one could choose other similarity indices ρ_{ij}, where ρ_{ij} = similarity between stock i and stock j.

Constructing an Index Fund

Example

- The correlation between the returns of stocks i and j. But one could choose other similarity indices ρ_{ij}, where $\rho_{ij} =$ similarity between stock i and stock j.
- For example, $\rho_{ii} = 1$, $\rho_{ij} \leq 1$ for $i \neq j$ and ρ_{ij} is larger for more similar stocks.

Constructing an Index Fund

Example

- The correlation between the returns of stocks i and j. But one could choose other similarity indices ρ_{ij}, where $\rho_{ij} =$ similarity between stock i and stock j.
- For example, $\rho_{ii} = 1$, $\rho_{ij} \leq 1$ for $i \neq j$ and ρ_{ij} is larger for more similar stocks.

$$(M) \quad Z = \max \sum_{i=1}^{n} \sum_{j=1}^{n} \rho_{ij} x_{ij}$$

Constructing an Index Fund

Example

- The correlation between the returns of stocks i and j. But one could choose other similarity indices ρ_{ij}, where ρ_{ij} = similarity between stock i and stock j.
- For example, $\rho_{ii} = 1$, $\rho_{ij} \leq 1$ for $i \neq j$ and ρ_{ij} is larger for more similar stocks.

$$(M) \quad Z = \max \sum_{i=1}^{n} \sum_{j=1}^{n} \rho_{ij} x_{ij}$$

$$\sum_{j=1}^{n} y_j = q$$

Constructing an Index Fund

Example

- The correlation between the returns of stocks i and j. But one could choose other similarity indices ρ_{ij}, where ρ_{ij} = similarity between stock i and stock j.
- For example, $\rho_{ii} = 1$, $\rho_{ij} \leq 1$ for $i \neq j$ and ρ_{ij} is larger for more similar stocks.

$$(M) \quad Z = \max \sum_{i=1}^{n} \sum_{j=1}^{n} \rho_{ij} x_{ij}$$

$$\sum_{j=1}^{n} y_j = q$$

$$\sum x_{ij} = 1 \quad , \forall\, i = 1, \ldots, n$$

Constructing an Index Fund

Example

- The correlation between the returns of stocks i and j. But one could choose other similarity indices ρ_{ij}, where ρ_{ij} = similarity between stock i and stock j.
- For example, $\rho_{ii} = 1$, $\rho_{ij} \leq 1$ for $i \neq j$ and ρ_{ij} is larger for more similar stocks.

$$(M) \quad Z = \max \sum_{i=1}^{n} \sum_{j=1}^{n} \rho_{ij} x_{ij}$$

$$\sum_{j=1}^{n} y_j = q$$

$$\sum x_{ij} = 1 \ , \forall\, i = 1, \ldots, n$$

$$\sum x_{ij} \leq y_j \ , \forall\, i = 1, \ldots, n, \ \ and \ j = 1, \ldots, n$$

Constructing an Index Fund

Example

- The correlation between the returns of stocks i and j. But one could choose other similarity indices ρ_{ij}, where ρ_{ij} = similarity between stock i and stock j.
- For example, $\rho_{ii} = 1$, $\rho_{ij} \leq 1$ for $i \neq j$ and ρ_{ij} is larger for more similar stocks.

$$
\begin{aligned}
(M) \quad Z = \max \sum_{i=1}^{n} \sum_{j=1}^{n} \rho_{ij} x_{ij} \\
\sum_{j=1}^{n} y_j = q \\
\sum x_{ij} = 1 \ , \forall\, i = 1, \ldots, n \\
\sum x_{ij} \leq y_j \ , \forall\, i = 1, \ldots, n, \ \textit{and} \ j = 1, \ldots, n \\
x_{ij}, y_j = 0 \ \textit{or} \ 1 \ , \forall\, i = 1, \ldots, n, \ \textit{and} \ j = 1, \ldots, n
\end{aligned}
$$

Constructing an Index Fund

Constructing an Index Fund

Example Cont.

Constructing an Index Fund

Example Cont.

- A weight w_j is calculated for each j in the fund:

Constructing an Index Fund

Example Cont.

- A weight w_j is calculated for each j in the fund:
- $w_j = \sum_{j=1}^{n} V_i x_{ij}$, where V_i is the market value of stock i.

Constructing an Index Fund

Example Cont.

- A weight w_j is calculated for each j in the fund:
- $w_j = \sum_{j=1}^{n} V_i x_{ij}$, where V_i is the market value of stock i.
- w_j is the total market value of the stocks represented by stock j in the fund.

Constructing an Index Fund

Example Cont.

- A weight w_j is calculated for each j in the fund:
- $w_j = \sum_{j=1}^{n} V_i x_{ij}$, where V_i is the market value of stock i.
- w_j is the total market value of the stocks represented by stock j in the fund.
- The fraction of the index fund to be invested in stock j is proportional to the weight w_j, such that: $\frac{w_j}{\sum_{f=1}^{n} w_f}$

Constructing an Index Fund

Example Cont.

- A weight w_j is calculated for each j in the fund:
- $w_j = \sum_{j=1}^{n} V_i x_{ij}$, where V_i is the market value of stock i.
- w_j is the total market value of the stocks represented by stock j in the fund.
- The fraction of the index fund to be invested in stock j is proportional to the weight w_j, such that: $\frac{w_j}{\sum_{f=1}^{n} w_f}$

Constructing an Index Fund

Constructing an Index Fund

Data Requirements

Constructing an Index Fund

Data Requirements

- We need a coefficient ρ_{ij} which measures the similarity between stocks i and j.

Constructing an Index Fund

Data Requirements

- We need a coefficient ρ_{ij} which measures the similarity between stocks i and j.
- One approach is to consider the time series of stock prices over a calibration period T and to compute the correlation between each pair of assets.

Constructing an Index Fund

Data Requirements

- We need a coefficient ρ_{ij} which measures the similarity between stocks i and j.
- One approach is to consider the time series of stock prices over a calibration period T and to compute the correlation between each pair of assets.

Testing the Model

Constructing an Index Fund

Data Requirements

- We need a coefficient ρ_{ij} which measures the similarity between stocks i and j.
- One approach is to consider the time series of stock prices over a calibration period T and to compute the correlation between each pair of assets.

Testing the Model

- Stocks comprising the S&P 500 were chosen as the target population to test the model.

Constructing an Index Fund

Data Requirements

- We need a coefficient ρ_{ij} which measures the similarity between stocks i and j.
- One approach is to consider the time series of stock prices over a calibration period T and to compute the correlation between each pair of assets.

Testing the Model

- Stocks comprising the S&P 500 were chosen as the target population to test the model.
- A calibration period of sixty months was used.

Constructing an Index Fund

Data Requirements

- We need a coefficient ρ_{ij} which measures the similarity between stocks i and j.
- One approach is to consider the time series of stock prices over a calibration period T and to compute the correlation between each pair of assets.

Testing the Model

- Stocks comprising the S&P 500 were chosen as the target population to test the model.
- A calibration period of sixty months was used.
- Then a portfolio of 25 stocks was constructed using model (M) and held for periods ranging from three months to three years.

Constructing an Index Fund

Solution Strategy

Constructing an Index Fund

Solution Strategy

- Branch-and-bound is a natural candidate for solving model (M).

Constructing an Index Fund

Solution Strategy

- Branch-and-bound is a natural candidate for solving model (M).
- Note however that the formulation is very large.

Constructing an Index Fund

Solution Strategy

- Branch-and-bound is a natural candidate for solving model (M).
- Note however that the formulation is very large.
- Indeed, for the S&P 500, there are $250,000$ variables x_{ij} and $250,000$ constraints $x_{ij} \leq y_j$.

Constructing an Index Fund

Solution Strategy

- Branch-and-bound is a natural candidate for solving model (M).
- Note however that the formulation is very large.
- Indeed, for the S&P 500, there are $250,000$ variables x_{ij} and $250,000$ constraints $x_{ij} \leq y_j$.
- So the linear programming relaxation needed to get upper bounds in the branch-and-bound algorithm is a very large linear program to solve.

Constructing an Index Fund

Solution Strategy

- Branch-and-bound is a natural candidate for solving model (M).
- Note however that the formulation is very large.
- Indeed, for the S&P 500, there are $250,000$ variables x_{ij} and $250,000$ constraints $x_{ij} \leq y_j$.
- So the linear programming relaxation needed to get upper bounds in the branch-and-bound algorithm is a very large linear program to solve.
- It turns out, however, that one does not need to solve this large linear program to obtain good upper bounds.

Constructing an Index Fund

Performance of Stocks

Constructing an Index Fund

Performance of Stocks

Performance of a 25 stock index fund	
Length	Ratio
1 QTR	1.006
2 QTR	.99
1 YR	.985
3 YR	.982

Constructing an Index Fund

Lagrangian Relaxation

Constructing an Index Fund

Lagrangian Relaxation

- Cornuejols, Fisher and Nemhauser proposed using the following:

Constructing an Index Fund

Lagrangian Relaxation

- Cornuejols, Fisher and Nemhauser proposed using the following:

$$L(u) = \max \sum_{i=1}^{n} \sum_{j=1}^{n} \rho_{ij} x_{ij} + \sum_{i=1}^{n} u_i (1 - \sum_{j=1}^{n} x_{ij})$$

Constructing an Index Fund

Lagrangian Relaxation

- Cornuejols, Fisher and Nemhauser proposed using the following:

$$L(u) = \max \sum_{i=1}^{n} \sum_{j=1}^{n} \rho_{ij} x_{ij} + \sum_{i=1}^{n} u_i (1 - \sum_{j=1}^{n} x_{ij})$$

$$\sum_{j=1}^{n} y_j = q$$

Constructing an Index Fund

Lagrangian Relaxation

- Cornuejols, Fisher and Nemhauser proposed using the following:

$$L(u) = \max \sum_{i=1}^{n}\sum_{j=1}^{n} \rho_{ij} x_{ij} + \sum_{i=1}^{n} u_i(1 - \sum_{j=1}^{n} x_{ij})$$

$$\sum_{j=1}^{n} y_j = q$$

$$\sum x_{ij} \leq y_j \quad , \forall\, i = 1, \ldots, n, \;\; and\; j = 1, \ldots, n$$

Constructing an Index Fund

Lagrangian Relaxation

- Cornuejols, Fisher and Nemhauser proposed using the following:

$$L(u) = \max \sum_{i=1}^{n}\sum_{j=1}^{n} \rho_{ij}x_{ij} + \sum_{i=1}^{n} u_i(1 - \sum_{j=1}^{n} x_{ij})$$

$$\sum_{j=1}^{n} y_j = q$$

$$\sum x_{ij} \leq y_j \ ,\forall\, i = 1,\dots,n,\ \ and\ j = 1,\dots,n$$

$$x_{ij}, y_j = 0\ \ or\ 1\ \ ,\forall\, i = 1,\dots,n,\ \ and\ j = 1,\dots,n$$

Constructing an Index Fund

Properties

Constructing an Index Fund

Properties

- **Property 1**: $L(u) \geq Z$, where Z is the maximum for model (M).

Constructing an Index Fund

Properties

- **Property 1**: $L(u) \geq Z$, where Z is the maximum for model (M).
- **Property 2**:

Constructing an Index Fund

Properties

- **Property 1**: $L(u) \geq Z$, where Z is the maximum for model (M).
- **Property 2**:

$$L(u) = \max \sum_{j=1}^{n} C_j y_j + \sum_{i=1}^{n} u_i$$

Constructing an Index Fund

Properties

- **Property 1**: $L(u) \geq Z$, where Z is the maximum for model (M).
- **Property 2**:

$$L(u) = \max \sum_{j=1}^{n} C_j y_j + \sum_{i=1}^{n} u_i$$

$$\sum_{j=1}^{n} y_j = q$$

Constructing an Index Fund

Properties

- **Property 1**: $L(u) \geq Z$, where Z is the maximum for model (M).
- **Property 2**:

$$L(u) = \max \sum_{j=1}^{n} C_j y_j + \sum_{i=1}^{n} u_i$$

$$\sum_{j=1}^{n} y_j = q$$

$$y_j = 0 \ or \ 1 \ \ , \forall\, j = 1, \ldots, n$$

Constructing an Index Fund

Properties Cont.

Constructing an Index Fund

Properties Cont.

Constructing an Index Fund

Properties Cont.

- **Property 3**: In an optimal solution of the Lagrangian relaxation, y_j is equal to 1 for the q largest values of C_j , and the remaining y_j are equal to 0.

Constructing an Index Fund

A Linear Programming Model

Constructing an Index Fund

A Linear Programming Model

Constructing an Index Fund

A Linear Programming Model

- This approach assumes that we have identified important characteristics of the market index to be tracked.

Constructing an Index Fund

A Linear Programming Model

- This approach assumes that we have identified important characteristics of the market index to be tracked.
- Let us assume that there are m such characteristics that we would like our index fund to track as well as possible.

Constructing an Index Fund

A Linear Programming Model

- This approach assumes that we have identified important characteristics of the market index to be tracked.
- Let us assume that there are m such characteristics that we would like our index fund to track as well as possible.
- Let $a_{ij} = 1$ if company j has characteristic i and 0 if it does not.

Constructing an Index Fund

A Linear Programming Model

- This approach assumes that we have identified important characteristics of the market index to be tracked.
- Let us assume that there are m such characteristics that we would like our index fund to track as well as possible.
- Let $a_{ij} = 1$ if company j has characteristic i and 0 if it does not.
- Let x_j denote the optimum weight of asset j in the portfolio.

Constructing an Index Fund

A Linear Programming Model

- This approach assumes that we have identified important characteristics of the market index to be tracked.
- Let us assume that there are m such characteristics that we would like our index fund to track as well as possible.
- Let $a_{ij} = 1$ if company j has characteristic i and 0 if it does not.
- Let x_j denote the optimum weight of asset j in the portfolio.
- Assume that initially, the portfolio has weights x_j^0.

Constructing an Index Fund

A Linear Programming Model

- This approach assumes that we have identified important characteristics of the market index to be tracked.
- Let us assume that there are m such characteristics that we would like our index fund to track as well as possible.
- Let $a_{ij} = 1$ if company j has characteristic i and 0 if it does not.
- Let x_j denote the optimum weight of asset j in the portfolio.
- Assume that initially, the portfolio has weights x_j^0.
- Let y_j denote the fraction of asset j bought and z_j the fraction sold.

Constructing an Index Fund

Rebalancing the Portfolio

Constructing an Index Fund

Rebalancing the Portfolio

Constructing an Index Fund

Rebalancing the Portfolio

$$\min \sum_{j=1}^{n} y_j + z_j$$

Constructing an Index Fund

Rebalancing the Portfolio

$$\min \sum_{j=1}^{n} y_j + z_j$$

$$\sum_{j=1}^{n} a_{ij} x_j = f_i \quad , \forall\, i = 1, \ldots, m$$

Constructing an Index Fund

Rebalancing the Portfolio

$$\min \sum_{j=1}^{n} y_j + z_j$$

$$\sum_{j=1}^{n} a_{ij} x_j = f_i \quad , \forall\, i = 1, \ldots, m$$

$$\sum_{j=1}^{n} x_j = 1$$

Constructing an Index Fund

Rebalancing the Portfolio

$$\min \sum_{j=1}^{n} y_j + z_j$$

$$\sum_{j=1}^{n} a_{ij} x_j = f_i \quad , \forall\, i = 1, \ldots, m$$

$$\sum_{j=1}^{n} x_j = 1$$

$$x_j - x_j^0 \leq y_j \quad , \forall\, j = 1, \ldots, n$$

Constructing an Index Fund

Rebalancing the Portfolio

$$\min \sum_{j=1}^{n} y_j + z_j$$

$$\sum_{j=1}^{n} a_{ij} x_j = f_i \quad , \forall\, i = 1, \ldots, m$$

$$\sum_{j=1}^{n} x_j = 1$$

$$x_j - x_j^0 \leq y_j \quad , \forall\, j = 1, \ldots, n$$

$$x_j^0 - x_j \leq z_j \quad , \forall\, j = 1, \ldots, n$$

Constructing an Index Fund

Rebalancing the Portfolio

$$\min \sum_{j=1}^{n} y_j + z_j$$

$$\sum_{j=1}^{n} a_{ij} x_j = f_i \quad , \forall\, i = 1, \ldots, m$$

$$\sum_{j=1}^{n} x_j = 1$$

$$x_j - x_j^0 \leq y_j \quad , \forall\, j = 1, \ldots, n$$

$$x_j^0 - x_j \leq z_j \quad , \forall\, j = 1, \ldots, n$$

$$y_j \geq 0, z_j \geq 0, x_j \geq 0 \quad , \forall\, j = 1, \ldots, n$$

Portfolio Optimization with Minimum Transaction Levels

Portfolio Optimization with Minimum Transaction Levels

Markowitz Model

Portfolio Optimization with Minimum Transaction Levels

Markowitz Model

- It assists in the selection of the most efficient portfolio.

Portfolio Optimization with Minimum Transaction Levels

Markowitz Model

- It assists in the selection of the most efficient portfolio.
- By choosing securities that do not 'move' exactly together.

Portfolio Optimization with Minimum Transaction Levels

Markowitz Model

- It assists in the selection of the most efficient portfolio.
- By choosing securities that do not 'move' exactly together.
- HM model shows investors how to reduce their risk.

Portfolio Optimization with Minimum Transaction Levels

Markowitz Model

- It assists in the selection of the most efficient portfolio.
- By choosing securities that do not 'move' exactly together.
- HM model shows investors how to reduce their risk.
- The HM model is also called Mean-Variance Model.

Portfolio Optimization with Minimum Transaction Levels

Markowitz Model

- It assists in the selection of the most efficient portfolio.
- By choosing securities that do not 'move' exactly together.
- HM model shows investors how to reduce their risk.
- The HM model is also called Mean-Variance Model.

Portfolio Optimization with Minimum Transaction Levels

Portfolio Optimization with Minimum Transaction Levels

Markowitz Model Assumptions

Portfolio Optimization with Minimum Transaction Levels

Markowitz Model Assumptions

- Risk of a portfolio is based on the variability of returns from the said portfolio.

Portfolio Optimization with Minimum Transaction Levels

Markowitz Model Assumptions

- Risk of a portfolio is based on the variability of returns from the said portfolio.
- An investor is risk averse.

Portfolio Optimization with Minimum Transaction Levels

Markowitz Model Assumptions

- Risk of a portfolio is based on the variability of returns from the said portfolio.
- An investor is risk averse.
- An investor prefers to increase consumption.

Portfolio Optimization with Minimum Transaction Levels

Markowitz Model Assumptions

- Risk of a portfolio is based on the variability of returns from the said portfolio.
- An investor is risk averse.
- An investor prefers to increase consumption.
- The investor's utility function is concave and increasing, due to his risk aversion and consumption preference.

Portfolio Optimization with Minimum Transaction Levels

Markowitz Model Assumptions

- Risk of a portfolio is based on the variability of returns from the said portfolio.
- An investor is risk averse.
- An investor prefers to increase consumption.
- The investor's utility function is concave and increasing, due to his risk aversion and consumption preference.
- Analysis is based on single period model of investment.

Portfolio Optimization with Minimum Transaction Levels

Markowitz Model Assumptions

- Risk of a portfolio is based on the variability of returns from the said portfolio.
- An investor is risk averse.
- An investor prefers to increase consumption.
- The investor's utility function is concave and increasing, due to his risk aversion and consumption preference.
- Analysis is based on single period model of investment.
- An investor either maximizes his portfolio return for a given level of risk or maximizes his return for the minimum risk.

Portfolio Optimization with Minimum Transaction Levels

Markowitz Model Assumptions

- Risk of a portfolio is based on the variability of returns from the said portfolio.
- An investor is risk averse.
- An investor prefers to increase consumption.
- The investor's utility function is concave and increasing, due to his risk aversion and consumption preference.
- Analysis is based on single period model of investment.
- An investor either maximizes his portfolio return for a given level of risk or maximizes his return for the minimum risk.
- An investor is rational in nature.

Portfolio Optimization with Minimum Transaction Levels

Portfolio Optimization with Minimum Transaction Levels

Determining the Efficient Set

Portfolio Optimization with Minimum Transaction Levels

Determining the Efficient Set

- What is the efficient portfolio?

Portfolio Optimization with Minimum Transaction Levels

Determining the Efficient Set

- What is the efficient portfolio?
- A portfolio that gives maximum return for a given risk, or minimum risk for given return is an efficient portfolio.

Portfolio Optimization with Minimum Transaction Levels

Determining the Efficient Set

- What is the efficient portfolio?
- A portfolio that gives maximum return for a given risk, or minimum risk for given return is an efficient portfolio.
- Portfolios are selected as follows:

Portfolio Optimization with Minimum Transaction Levels

Determining the Efficient Set

- What is the efficient portfolio?
- A portfolio that gives maximum return for a given risk, or minimum risk for given return is an efficient portfolio.
- Portfolios are selected as follows:
    - From the portfolios that have the same return, the investor will prefer the portfolio with lower risk.

Portfolio Optimization with Minimum Transaction Levels

Determining the Efficient Set

- What is the efficient portfolio?
- A portfolio that gives maximum return for a given risk, or minimum risk for given return is an efficient portfolio.
- Portfolios are selected as follows:
  - From the portfolios that have the same return, the investor will prefer the portfolio with lower risk.
  - From the portfolios that have the same risk level, an investor will prefer the portfolio with higher rate of return.

Portfolio Optimization with Minimum Transaction Levels

Portfolio Optimization with Minimum Transaction Levels

Solve for the Optimal Portfolio

Portfolio Optimization with Minimum Transaction Levels

Solve for the Optimal Portfolio

- When solving the classical Markowitz model, the optimal portfolio often contains positions x_i that are too small to execute.

Portfolio Optimization with Minimum Transaction Levels

Solve for the Optimal Portfolio

- When solving the classical Markowitz model, the optimal portfolio often contains positions x_i that are too small to execute.
- $$\min_{x} \cdot \frac{1}{2} \cdot x^T \cdot Q \cdot x$$

Portfolio Optimization with Minimum Transaction Levels

Solve for the Optimal Portfolio

- When solving the classical Markowitz model, the optimal portfolio often contains positions x_i that are too small to execute.
- $$\min_x \cdot \frac{1}{2} \cdot x^T \cdot Q \cdot x$$
  $$\mu^T \cdot x \geq R$$

Portfolio Optimization with Minimum Transaction Levels

Solve for the Optimal Portfolio

- When solving the classical Markowitz model, the optimal portfolio often contains positions x_i that are too small to execute.
-

$$\min_x \cdot \frac{1}{2} \cdot x^T \cdot Q \cdot x$$
$$\mu^T \cdot x \geq R$$
$$A \cdot x = b$$

Portfolio Optimization with Minimum Transaction Levels

Solve for the Optimal Portfolio

- When solving the classical Markowitz model, the optimal portfolio often contains positions x_i that are too small to execute.
- $$\begin{aligned} \min_x \cdot \frac{1}{2} \cdot x^T \cdot Q \cdot x \\ \mu^T \cdot x \geq R \\ A \cdot x = b \\ C \cdot x \geq d \end{aligned}$$

Portfolio Optimization with Minimum Transaction Levels

Solve for the Optimal Portfolio

- When solving the classical Markowitz model, the optimal portfolio often contains positions x_i that are too small to execute.
- $$\begin{aligned} \min_x \cdot \frac{1}{2} \cdot x^T \cdot Q \cdot x \\ \mu^T \cdot x \geq R \\ A \cdot x = b \\ C \cdot x \geq d \end{aligned}$$
- with the additional property that:

Portfolio Optimization with Minimum Transaction Levels

Solve for the Optimal Portfolio

- When solving the classical Markowitz model, the optimal portfolio often contains positions x_i that are too small to execute.
- $$\begin{aligned} \min_x \cdot \frac{1}{2} \cdot x^T \cdot Q \cdot x \\ \mu^T \cdot x \geq R \\ A \cdot x = b \\ C \cdot x \geq d \end{aligned}$$
- with the additional property that:
  $x_j > 0 \Rightarrow x_j \geq l_j$, where l_j are given minimum transaction levels.

Portfolio Optimization with Minimum Transaction Levels

Portfolio Optimization with Minimum Transaction Levels

Solve for the Optimal Portfolio

Portfolio Optimization with Minimum Transaction Levels

Solve for the Optimal Portfolio

- Because the prevous constraint is not a simple linear constraint, it cannot be handled directly by quadratic programming.

Portfolio Optimization with Minimum Transaction Levels

Solve for the Optimal Portfolio

- Because the prevous constraint is not a simple linear constraint, it cannot be handled directly by quadratic programming.
- This problem is considered by Bienstock.

Portfolio Optimization with Minimum Transaction Levels

Solve for the Optimal Portfolio

- Because the prevous constraint is not a simple linear constraint, it cannot be handled directly by quadratic programming.
- This problem is considered by Bienstock.
- The portfolio optimization problem where there is an upper bound on the number of positive variables, that is:
  $x_j > 0$ for at most K distinct $j = 1, \ldots, n$.

Portfolio Optimization with Minimum Transaction Levels

Solve for the Optimal Portfolio

- Because the prevous constraint is not a simple linear constraint, it cannot be handled directly by quadratic programming.
- This problem is considered by Bienstock.
- The portfolio optimization problem where there is an upper bound on the number of positive variables, that is:
  $x_j > 0$ for at most K distinct $j = 1, \ldots, n$.
- The constraint can easily be incorporated within a branch-and-bound algorithm.

Portfolio Optimization with Minimum Transaction Levels

Portfolio Optimization with Minimum Transaction Levels

Solve for the Optimal Portfolio

Portfolio Optimization with Minimum Transaction Levels

Solve for the Optimal Portfolio

- First solve the basic Markowitz model.

Portfolio Optimization with Minimum Transaction Levels

Solve for the Optimal Portfolio

- First solve the basic Markowitz model.
- Let x^* be the optimal solution found.

Portfolio Optimization with Minimum Transaction Levels

Solve for the Optimal Portfolio

- First solve the basic Markowitz model.
- Let x^* be the optimal solution found.
- If no minimum transaction level , then the constraint is violated by x^*.

Portfolio Optimization with Minimum Transaction Levels

Solve for the Optimal Portfolio

- First solve the basic Markowitz model.
- Let x^* be the optimal solution found.
- If no minimum transaction level , then the constraint is violated by x^*.
- x^* is also optimum to Markowitz model and the constraint, so we can stop.

Portfolio Optimization with Minimum Transaction Levels

Solve for the Optimal Portfolio

- First solve the basic Markowitz model.
- Let x^* be the optimal solution found.
- If no minimum transaction level , then the constraint is violated by x^*.
- x^* is also optimum to Markowitz model and the constraint, so we can stop.
- Otherwise, let j be an index for which the constraint is violated by x^*.

Portfolio Optimization with Minimum Transaction Levels

Solve for the Optimal Portfolio

- First solve the basic Markowitz model.
- Let x^* be the optimal solution found.
- If no minimum transaction level , then the constraint is violated by x^*.
- x^* is also optimum to Markowitz model and the constraint, so we can stop.
- Otherwise, let j be an index for which the constraint is violated by x^*.

Portfolio Optimization with Minimum Transaction Levels

Portfolio Optimization with Minimum Transaction Levels

Solve for the Optimal Portfolio

Portfolio Optimization with Minimum Transaction Levels

Solve for the Optimal Portfolio

- Form two subproblems.

Portfolio Optimization with Minimum Transaction Levels

Solve for the Optimal Portfolio

- Form two subproblems.
- One obtained from the Markowitz model by adding the constraint $x_j = 0$.

Portfolio Optimization with Minimum Transaction Levels

Solve for the Optimal Portfolio

- Form two subproblems.
- One obtained from the Markowitz model by adding the constraint $x_j = 0$.
- The other obtained from the Markowitz model by adding the constraint $x_j \geq l_j$.

Portfolio Optimization with Minimum Transaction Levels

Solve for the Optimal Portfolio

- Form two subproblems.
- One obtained from the Markowitz model by adding the constraint $x_j = 0$.
- The other obtained from the Markowitz model by adding the constraint $x_j \geq l_j$.
- Both are quadratic programs that can be solved using the usual algorithms.

Portfolio Optimization with Minimum Transaction Levels

Solve for the Optimal Portfolio

- Form two subproblems.
- One obtained from the Markowitz model by adding the constraint $x_j = 0$.
- The other obtained from the Markowitz model by adding the constraint $x_j \geq l_j$.
- Both are quadratic programs that can be solved using the usual algorithms.

Check the Optimum Solutions

Portfolio Optimization with Minimum Transaction Levels

Solve for the Optimal Portfolio

- Form two subproblems.
- One obtained from the Markowitz model by adding the constraint $x_j = 0$.
- The other obtained from the Markowitz model by adding the constraint $x_j \geq l_j$.
- Both are quadratic programs that can be solved using the usual algorithms.

Check the Optimum Solutions

- Now check whether the optimum solutions to these two problems satisfy the transaction level constraint.

Portfolio Optimization with Minimum Transaction Levels

Solve for the Optimal Portfolio

- Form two subproblems.
- One obtained from the Markowitz model by adding the constraint $x_j = 0$.
- The other obtained from the Markowitz model by adding the constraint $x_j \geq l_j$.
- Both are quadratic programs that can be solved using the usual algorithms.

Check the Optimum Solutions

- Now check whether the optimum solutions to these two problems satisfy the transaction level constraint.
- If a solution violates the constraint for index k, then the corresponding problem is further divided by adding the constraint $x_k = 0$ on one side and $x_k \geq l_k$ on the other.

Portfolio Optimization with Minimum Transaction Levels

Solve for the Optimal Portfolio

- Form two subproblems.
- One obtained from the Markowitz model by adding the constraint $x_j = 0$.
- The other obtained from the Markowitz model by adding the constraint $x_j \geq l_j$.
- Both are quadratic programs that can be solved using the usual algorithms.

Check the Optimum Solutions

- Now check whether the optimum solutions to these two problems satisfy the transaction level constraint.
- If a solution violates the constraint for index k, then the corresponding problem is further divided by adding the constraint $x_k = 0$ on one side and $x_k \geq l_k$ on the other.

Portfolio Optimization with Minimum Transaction Levels

Portfolio Optimization with Minimum Transaction Levels

Check the Upper Bound on the Number of Positive Variables

Portfolio Optimization with Minimum Transaction Levels

Check the Upper Bound on the Number of Positive Variables

- Assume that there is a given upper bound u_j on how much can be invested in stock j.

Portfolio Optimization with Minimum Transaction Levels

Check the Upper Bound on the Number of Positive Variables

- Assume that there is a given upper bound u_j on how much can be invested in stock j.
- Assume $x_j \leq u_j$ are part of the formulation of Markowitz model .

Portfolio Optimization with Minimum Transaction Levels

Check the Upper Bound on the Number of Positive Variables

- Assume that there is a given upper bound u_j on how much can be invested in stock j.
- Assume $x_j \leq u_j$ are part of the formulation of Markowitz model .
- Then, clearly, the upper bound constraint implies the weaker constraint:

Portfolio Optimization with Minimum Transaction Levels

Check the Upper Bound on the Number of Positive Variables

- Assume that there is a given upper bound u_j on how much can be invested in stock j.
- Assume $x_j \leq u_j$ are part of the formulation of Markowitz model .
- Then, clearly, the upper bound constraint implies the weaker constraint:
- $$\sum_j \frac{x_j}{u_j} \leq K$$

Portfolio Optimization with Minimum Transaction Levels

Check the Upper Bound on the Number of Positive Variables

- Assume that there is a given upper bound u_j on how much can be invested in stock j.
- Assume $x_j \leq u_j$ are part of the formulation of Markowitz model .
- Then, clearly, the upper bound constraint implies the weaker constraint:
-
$$\sum_j \frac{x_j}{u_j} \leq K$$
- We add this constraint to Markowitz model and solve the resulting quadratic program.

Portfolio Optimization with Minimum Transaction Levels

Check the Upper Bound on the Number of Positive Variables

- Assume that there is a given upper bound u_j on how much can be invested in stock j.
- Assume $x_j \leq u_j$ are part of the formulation of Markowitz model .
- Then, clearly, the upper bound constraint implies the weaker constraint:
- $$\sum_j \frac{x_j}{u_j} \leq K$$
- We add this constraint to Markowitz model and solve the resulting quadratic program.

Portfolio Optimization with Minimum Transaction Levels

Portfolio Optimization with Minimum Transaction Levels

Check the Upper Bound on the Number of Positive Variables

Portfolio Optimization with Minimum Transaction Levels

Check the Upper Bound on the Number of Positive Variables

- Let x^* be the optimal solution found.

Portfolio Optimization with Minimum Transaction Levels

Check the Upper Bound on the Number of Positive Variables

- Let x^* be the optimal solution found.
- If x^* satisfies the constraint,then it is optimum to Markowitz model and we can stop.

Portfolio Optimization with Minimum Transaction Levels

Check the Upper Bound on the Number of Positive Variables

- Let x^* be the optimal solution found.
- If x^* satisfies the constraint,then it is optimum to Markowitz model and we can stop.
- Otherwise, let k be an index for which $x_k > 0$.

Portfolio Optimization with Minimum Transaction Levels

Check the Upper Bound on the Number of Positive Variables

- Let x^* be the optimal solution found.
- If x^* satisfies the constraint,then it is optimum to Markowitz model and we can stop.
- Otherwise, let k be an index for which $x_k > 0$.
- Form two subproblems:

Portfolio Optimization with Minimum Transaction Levels

Check the Upper Bound on the Number of Positive Variables

- Let x^* be the optimal solution found.
- If x^* satisfies the constraint,then it is optimum to Markowitz model and we can stop.
- Otherwise, let k be an index for which $x_k > 0$.
- Form two subproblems:
- One obtained from Markowitz model by adding the constraint $x_k = 0$ (down branch),

Portfolio Optimization with Minimum Transaction Levels

Check the Upper Bound on the Number of Positive Variables

- Let x^* be the optimal solution found.
- If x^* satisfies the constraint,then it is optimum to Markowitz model and we can stop.
- Otherwise, let k be an index for which $x_k > 0$.
- Form two subproblems:
- One obtained from Markowitz model by adding the constraint $x_k = 0$ (down branch),
- The other obtained from Markowitz model by adding the constraint $\sum_{j \neq k} \frac{x_j}{u_j} \leq K - 1$ (up branch).

Portfolio Optimization with Minimum Transaction Levels

Portfolio Optimization with Minimum Transaction Levels

Check the Upper Bound on the Number of Positive Variables

Portfolio Optimization with Minimum Transaction Levels

Check the Upper Bound on the Number of Positive Variables

- The branch-and-bound tree is developped recursively.

Portfolio Optimization with Minimum Transaction Levels

Check the Upper Bound on the Number of Positive Variables

- The branch-and-bound tree is developped recursively.
- When a set T of variables has been branched up, the constraint added to the basic Markowitz model becomes:

Portfolio Optimization with Minimum Transaction Levels

Check the Upper Bound on the Number of Positive Variables

- The branch-and-bound tree is developped recursively.
- When a set T of variables has been branched up, the constraint added to the basic Markowitz model becomes:
- $$\sum_{j \notin T} \frac{x_j}{u_j} \leq K - |T|.$$

References

References

References

References

References

1. Y. Narahari, Pankaj Dayama,Combinatorial Auctions for Electronic Business, game theory, survey.

References

References

1. Y. Narahari, Pankaj Dayama,Combinatorial Auctions for Electronic Business, game theory, survey.
2. G. Cornuejols, R. Tütüncü, Optimization methods in Finance, Cambridge University Press, 2007.

References

References

1. Y. Narahari, Pankaj Dayama,Combinatorial Auctions for Electronic Business, game theory, survey.
2. G. Cornuejols, R. Tütüncü, Optimization methods in Finance, Cambridge University Press, 2007.
3. S. Biswas and Y. Narahari. An iterative auction mechanism for combinatorial exchanges. Technical report, Electronic Enterprises Lab, Dept. of Computer Science and Automation, Indian Institute of Science, 2003.

References

References

1. Y. Narahari, Pankaj Dayama,Combinatorial Auctions for Electronic Business, game theory, survey.
2. G. Cornuejols, R. Tütüncü, Optimization methods in Finance, Cambridge University Press, 2007.
3. S. Biswas and Y. Narahari. An iterative auction mechanism for combinatorial exchanges. Technical report, Electronic Enterprises Lab, Dept. of Computer Science and Automation, Indian Institute of Science, 2003.
4. Robert M. Nauss and Robert E. Markland, Solving Lock Box Location Problems, Vol. 8, No. 1 (Spring, 1979)

References

References

1. Y. Narahari, Pankaj Dayama,Combinatorial Auctions for Electronic Business, game theory, survey.
2. G. Cornuejols, R. Tütüncü, Optimization methods in Finance, Cambridge University Press, 2007.
3. S. Biswas and Y. Narahari. An iterative auction mechanism for combinatorial exchanges. Technical report, Electronic Enterprises Lab, Dept. of Computer Science and Automation, Indian Institute of Science, 2003.
4. Robert M. Nauss and Robert E. Markland, Solving Lock Box Location Problems, Vol. 8, No. 1 (Spring, 1979)