Crane Scheduling in Container Terminals at Seaports

Erwin Pesch

University of Siegen, Department of Management Information Science, Germany

IARIA BUSTECH 2023, June 26-30, 2023

Universität Siegen

https://www.wiwi.uni-siegen.de/mis/team/pesch/

erwin[dot]pesch[at]uni[minus]siegen[dot]de

Outline

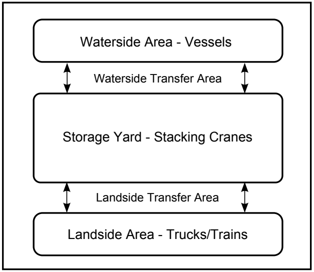
1. Problem Setting: Automated Container Terminal
    - Terminal Layouts
    - Yard Layouts
    - Container Flow
    - Yard Crane Systems
2. Twin Cranes
3. Policies and Results
4. Crossover Crane Scheduling Problem (CCSP)
5. Solution Approach: Logic-Based Benders Decomposition
6. Computational Results

Problem Setting

Problem Setting

Container Terminal Layouts

- **Waterside Area:** Vessels are moored at the berth and quay cranes are used to load and unload containers.
- **Landside Area:** Handles the hinterland container transportation on trucks and trains.
- **Storage Yard:** Containers are temporarily stored by yard cranes and are exchanged between the waterside and the landside.
- **Waterside Transfer Area:** Operated by internal vehicles (e.g., straddle carriers, automated guided vehicles, trucks) and performs the container transport from the waterside to the storage yard and vice versa.

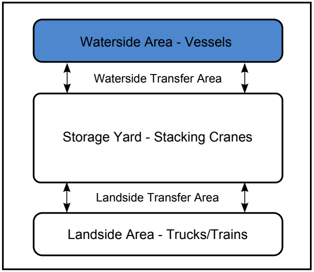


- **Landside Transfer Area:** Containers are picked up/delivered by external trucks or internal trucks are employed to deliver (receive) containers to (from) trains.

Problem Setting

Container Terminal Layouts

- **Waterside Area:** Vessels are moored at the berth and quay cranes are used to load and unload containers.
- **Landside Area:** Handles the hinterland container transportation on trucks and trains.
- **Storage Yard:** Containers are temporarily stored by yard cranes and are exchanged between the waterside and the landside.
- **Waterside Transfer Area:** Operated by internal vehicles (e.g., straddle carriers, automated guided vehicles, trucks) and performs the container transport from the waterside to the storage yard and vice versa.

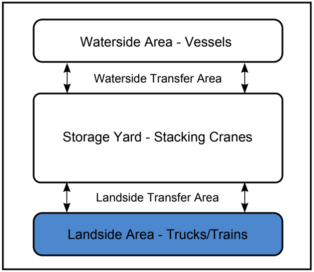


- **Landside Transfer Area:** Containers are picked up/delivered by external trucks or internal trucks are employed to deliver (receive) containers to (from) trains.

Problem Setting

Container Terminal Layouts

- **Waterside Area:** Vessels are moored at the berth and quay cranes are used to load and unload containers.
- **Landside Area:** Handles the hinterland container transportation on trucks and trains.
- **Storage Yard:** Containers are temporarily stored by yard cranes and are exchanged between the waterside and the landside.
- **Waterside Transfer Area:** Operated by internal vehicles (e.g., straddle carriers, automated guided vehicles, trucks) and performs the container transport from the waterside to the storage yard and vice versa.

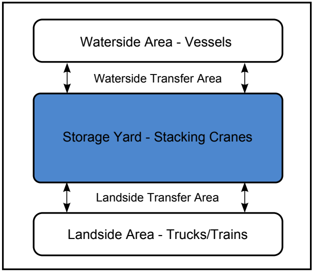


- **Landside Transfer Area:** Containers are picked up/delivered by external trucks or internal trucks are employed to deliver (receive) containers to (from) trains.

Problem Setting

Container Terminal Layouts

- **Waterside Area:** Vessels are moored at the berth and quay cranes are used to load and unload containers.
- **Landside Area:** Handles the hinterland container transportation on trucks and trains.
- **Storage Yard:** Containers are temporarily stored by yard cranes and are exchanged between the waterside and the landside.
- **Waterside Transfer Area:** Operated by internal vehicles (e.g., straddle carriers, automated guided vehicles, trucks) and performs the container transport from the waterside to the storage yard and vice versa.

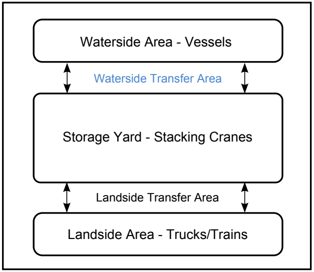


- **Landside Transfer Area:** Containers are picked up/delivered by external trucks or internal trucks are employed to deliver (receive) containers to (from) trains.

Problem Setting

Container Terminal Layouts

- **Waterside Area:** Vessels are moored at the berth and quay cranes are used to load and unload containers.
- **Landside Area:** Handles the hinterland container transportation on trucks and trains.
- **Storage Yard:** Containers are temporarily stored by yard cranes and are exchanged between the waterside and the landside.
- **Waterside Transfer Area:** Operated by internal vehicles (e.g., straddle carriers, automated guided vehicles, trucks) and performs the container transport from the waterside to the storage yard and vice versa.

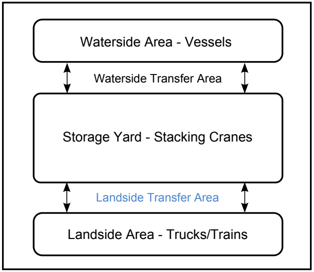


- **Landside Transfer Area:** Containers are picked up/delivered by external trucks or internal trucks are employed to deliver (receive) containers to (from) trains.

Problem Setting

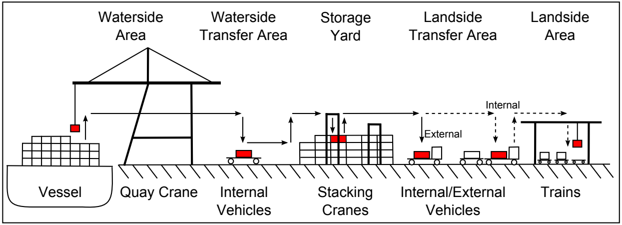
Container Terminal Layouts

- **Waterside Area:** Vessels are moored at the berth and quay cranes are used to load and unload containers.
- **Landside Area:** Handles the hinterland container transportation on trucks and trains.
- **Storage Yard:** Containers are temporarily stored by yard cranes and are exchanged between the waterside and the landside.
- **Waterside Transfer Area:** Operated by internal vehicles (e.g., straddle carriers, automated guided vehicles, trucks) and performs the container transport from the waterside to the storage yard and vice versa.



- **Landside Transfer Area:** Containers are picked up/delivered by external trucks or internal trucks are employed to deliver (receive) containers to (from) trains.

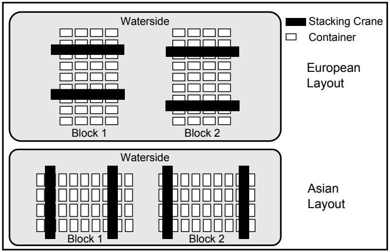
Container Flow



Container Flow (motivated by Steenken et al. 2004)

Problem Setting

Yard Layouts



- **European Layout:** Terminal layouts with a storage yard *perpendicular* to the waterside.
- **Asian Layout:** Terminal layouts with a storage yard *parallel* to the waterside.

Problem Setting

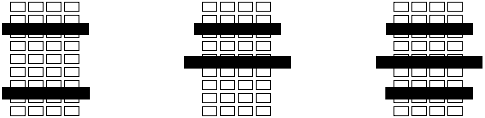
Yard Crane Systems

Different yard crane systems are employed at a storage yard. Systems vary, e.g., in their installation of **rail-mounted (RMG)** or **rubber-tyred (RTG) gantry cranes**.

Problem Setting

Rail-Mounted Gantry Cranes

- **Twin Cranes:** Denote two RMGs of equal size which operate on the same rail tracks and cannot pass each other.
- **Crossover or Dual Cranes:** Refer to two RMGs (outer and inner crane) of different sizes that run on different tracks and have the possibility to cross each other.
- **Triple Cranes:** Consist of two twin cranes and one larger crane that moves on a different track and can pass both twin cranes.

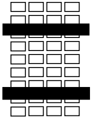


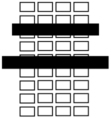
Twin Cranes   Dual Cranes   Triple Cranes

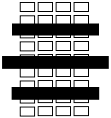
Problem Setting

Rail-Mounted Gantry Cranes

- **Twin Cranes:** Denote two RMGs of equal size which operate on the same rail tracks and cannot pass each other.
- **Crossover or Dual Cranes:** Refer to two RMGs (outer and inner crane) of different sizes that run on different tracks and have the possibility to cross each other.
- **Triple Cranes:** Consist of two twin cranes and one larger crane that moves on a different track and can pass both twin cranes.


Twin Cranes

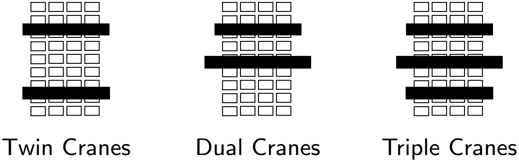

Dual Cranes


Triple Cranes

Problem Setting

Rail-Mounted Gantry Cranes

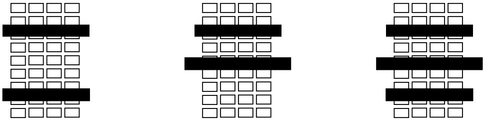
- **Twin Cranes:** Denote two RMGs of equal size which operate on the same rail tracks and cannot pass each other.
- **Crossover or Dual Cranes:** Refer to two RMGs (outer and inner crane) of different sizes that run on different tracks and have the possibility to cross each other.
- **Triple Cranes:** Consist of two twin cranes and one larger crane that moves on a different track and can pass both twin cranes.



Problem Setting

Rail-Mounted Gantry Cranes

- **Twin Cranes:** Denote two RMGs of equal size which operate on the same rail tracks and cannot pass each other.
- **Crossover or Dual Cranes:** Refer to two RMGs (outer and inner crane) of different sizes that run on different tracks and have the possibility to cross each other.
- **Triple Cranes:** Consist of two twin cranes and one larger crane that moves on a different track and can pass both twin cranes.

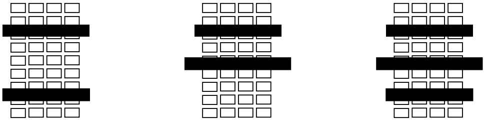


Twin Cranes   Dual Cranes   Triple Cranes

Problem Setting

Rail-Mounted Gantry Cranes

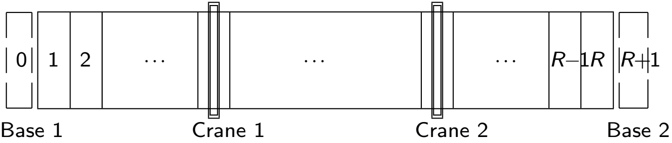
- **Twin Cranes:** Denote two RMGs of equal size which operate on the same rail tracks and cannot pass each other.
- **Crossover or Dual Cranes:** Refer to two RMGs (outer and inner crane) of different sizes that run on different tracks and have the possibility to cross each other.
- **Triple Cranes:** Consist of two twin cranes and one larger crane that moves on a different track and can pass both twin cranes.

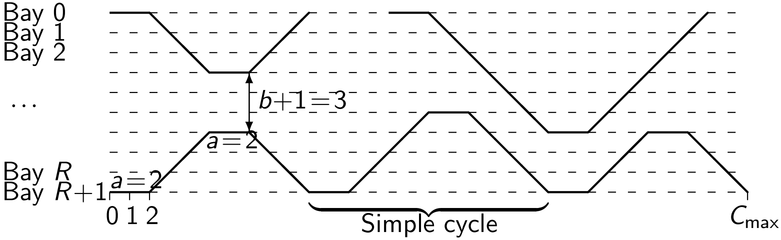


Twin Cranes Dual Cranes Triple Cranes

Twin Cranes

Example of a Solution





Policies

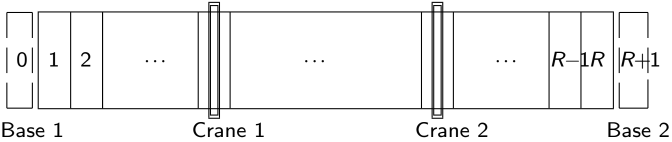


Figure: Kovalyov, Pesch, Ryzhikov, *Networks* (2018)

Containers are assigned to the cranes according to one of the following policies:

(1) two fixed sequences policy where a container processing sequence is given for each crane, $C2|FixFix|C_{\max}$

(2) dedicated crane policy where containers are pre-assigned to the cranes, $C2|Dedic|C_{\max}$

(3) one fixed, one arbitrary sequence policy where a container processing sequence is given for one crane and it can be arbitrary for the other crane, $C2|FixAny|C_{\max}$

(4) flexible policy where any container can be assigned to any crane at any time, $C2|Flex|C_{\max}$

(5) global fixed sequence policy where the container sequence is given and the relative processing order of containers in this sequence must be preserved by any crane, $C2|GlobFix|C_{\max}$

Policies

Figure: Kovalyov, Pesch, Ryzhikov, *Networks* (2018)

Containers are assigned to the cranes according to one of the following policies:

(1) two fixed sequences policy where a container processing sequence is given for each crane, $C2|FixFix|C_{\max}$

(2) dedicated crane policy where containers are pre-assigned to the cranes, $C2|Dedic|C_{\max}$

(3) one fixed, one arbitrary sequence policy where a container processing sequence is given for one crane and it can be arbitrary for the other crane, $C2|FixAny|C_{\max}$

(4) flexible policy where any container can be assigned to any crane at any time, $C2|Flex|C_{\max}$

(5) global fixed sequence policy where the container sequence is given and the relative processing order of containers in this sequence must be preserved by any crane, $C2|GlobFix|C_{\max}$

Policies

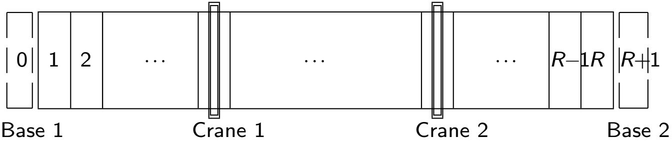


Figure: Kovalyov, Pesch, Ryzhikov, *Networks* (2018)

Containers are assigned to the cranes according to one of the following policies:

(1) two fixed sequences policy where a container processing sequence is given for each crane, $C2|FixFix|C_{\max}$

(2) dedicated crane policy where containers are pre-assigned to the cranes, $C2|Dedic|C_{\max}$

(3) one fixed, one arbitrary sequence policy where a container processing sequence is given for one crane and it can be arbitrary for the other crane, $C2|FixAny|C_{\max}$

(4) flexible policy where any container can be assigned to any crane at any time, $C2|Flex|C_{\max}$

(5) global fixed sequence policy where the container sequence is given and the relative processing order of containers in this sequence must be preserved by any crane, $C2|GlobFix|C_{\max}$

Policies

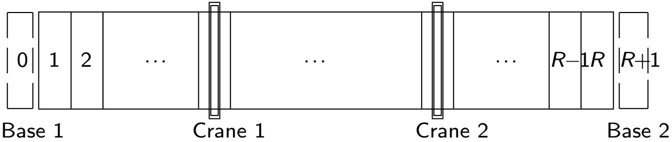


Figure: Kovalyov, Pesch, Ryzhikov, *Networks* (2018)

Containers are assigned to the cranes according to one of the following policies:

(1) two fixed sequences policy where a container processing sequence is given for each crane, $C2|FixFix|C_{\max}$

(2) dedicated crane policy where containers are pre-assigned to the cranes, $C2|Dedic|C_{\max}$

(3) one fixed, one arbitrary sequence policy where a container processing sequence is given for one crane and it can be arbitrary for the other crane, $C2|FixAny|C_{\max}$

(4) flexible policy where any container can be assigned to any crane at any time, $C2|Flex|C_{\max}$

(5) global fixed sequence policy where the container sequence is given and the relative processing order of containers in this sequence must be preserved by any crane, $C2|GlobFix|C_{\max}$

Policies

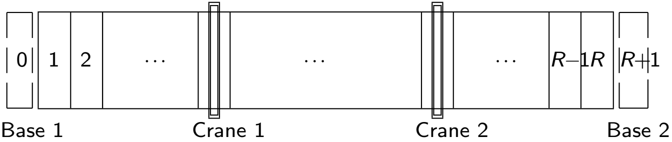


Figure: Kovalyov, Pesch, Ryzhikov, *Networks* (2018)

Containers are assigned to the cranes according to one of the following policies:

(1) two fixed sequences policy where a container processing sequence is given for each crane, $C2|FixFix|C_{\max}$

(2) dedicated crane policy where containers are pre-assigned to the cranes, $C2|Dedic|C_{\max}$

(3) one fixed, one arbitrary sequence policy where a container processing sequence is given for one crane and it can be arbitrary for the other crane, $C2|FixAny|C_{\max}$

(4) flexible policy where any container can be assigned to any crane at any time, $C2|Flex|C_{\max}$

(5) global fixed sequence policy where the container sequence is given and the relative processing order of containers in this sequence must be preserved by any crane, $C2|GlobFix|C_{\max}$

Complexity Results

- Problem $C2|FixFix|C_{\max}$: Briskorn and Angeloudis (2016) suggested a reduction of this more general problem to finding shortest path in a specially designed acyclic digraph with arc weights in $O(n^4)$ time.
- Problem $C2|Dedic|C_{\max}$: strongly NP-hard; Erdogan et al. (2014) proved NP-hardness in the ordinary sense
- Problem $C2|FixAny|C_{\max}$: NP-hard in the strong sense; Boysen, Briskorn, Emde (2015) proved for a special case $a = b = 0$, strong NP-hardness. The proof can be adjusted to show strong NP-hardness of $C2|Dedic|C_{\max}$ and $C2|FixAny|C_{\max}$ even if $a = 1$ and $b = 0$
- Problem $C2|Flex|C_{\max}$
- Problem $C2|GlobFix|C_{\max}$

Complexity Results

- Problem $C2|FixFix|C_{\max}$: Briskorn and Angeloudis (2016) suggested a reduction of this more general problem to finding shortest path in a specially designed acyclic digraph with arc weights in $O(n^4)$ time.
- Problem $C2|Dedic|C_{\max}$: strongly NP-hard; Erdogan et al. (2014) proved NP-hardness in the ordinary sense
- Problem $C2|FixAny|C_{\max}$: NP-hard in the strong sense; Boysen, Briskorn, Emde (2015) proved for a special case $a = b = 0$, strong NP-hardness. The proof can be adjusted to show strong NP-hardness of $C2|Dedic|C_{\max}$ and $C2|FixAny|C_{\max}$ even if $a = 1$ and $b = 0$
- Problem $C2|Flex|C_{\max}$
- Problem $C2|GlobFix|C_{\max}$

Complexity Results

- Problem $C2|FixFix|C_{\max}$: Briskorn and Angeloudis (2016) suggested a reduction of this more general problem to finding shortest path in a specially designed acyclic digraph with arc weights in $O(n^4)$ time.
- Problem $C2|Dedic|C_{\max}$: strongly NP-hard; Erdogan et al. (2014) proved NP-hardness in the ordinary sense
- Problem $C2|FixAny|C_{\max}$: NP-hard in the strong sense; Boysen, Briskorn, Emde (2015) proved for a special case $a = b = 0$, strong NP-hardness. The proof can be adjusted to show strong NP-hardness of $C2|Dedic|C_{\max}$ and $C2|FixAny|C_{\max}$ even if $a = 1$ and $b = 0$
- Problem $C2|Flex|C_{\max}$
- Problem $C2|GlobFix|C_{\max}$

Complexity Results

- Problem $C2|FixFix|C_{\max}$: Briskorn and Angeloudis (2016) suggested a reduction of this more general problem to finding shortest path in a specially designed acyclic digraph with arc weights in $O(n^4)$ time.
- Problem $C2|Dedic|C_{\max}$: strongly NP-hard; Erdogan et al. (2014) proved NP-hardness in the ordinary sense
- Problem $C2|FixAny|C_{\max}$: NP-hard in the strong sense; Boysen, Briskorn, Emde (2015) proved for a special case $a = b = 0$, strong NP-hardness. The proof can be adjusted to show strong NP-hardness of $C2|Dedic|C_{\max}$ and $C2|FixAny|C_{\max}$ even if $a = 1$ and $b = 0$
- Problem $C2|Flex|C_{\max}$
- Problem $C2|GlobFix|C_{\max}$

Complexity Results

- Problem $C2|FixFix|C_{\max}$: Briskorn and Angeloudis (2016) suggested a reduction of this more general problem to finding shortest path in a specially designed acyclic digraph with arc weights in $O(n^4)$ time.
- Problem $C2|Dedic|C_{\max}$: strongly NP-hard; Erdogan et al. (2014) proved NP-hardness in the ordinary sense
- Problem $C2|FixAny|C_{\max}$: NP-hard in the strong sense; Boysen, Briskorn, Emde (2015) proved for a special case $a = b = 0$, strong NP-hardness. The proof can be adjusted to show strong NP-hardness of $C2|Dedic|C_{\max}$ and $C2|FixAny|C_{\max}$ even if $a = 1$ and $b = 0$
- Problem $C2|Flex|C_{\max}$
- Problem $C2|GlobFix|C_{\max}$

- Problem $C2|Flex|C_{\max}$
  - There exists an optimal schedule E for problem $C2|Flex|C_{\max}$ such that $\max\{r_j \mid j \in X_1^{(E)}\} \leq \min\{r_j \mid j \in X_2^{(E)}\}$.
  - There exists an optimal schedule for the problem $C2|Flex|C_{\max}$ and a separation number k, $1 \leq k \leq n-1$, such that containers $1, \ldots, k$ are assigned to crane 1 and containers $k+1, \ldots, n$ are assigned to crane 2.
  - relaxed problem $C2|Flex, meet|C_{\max}$ can be solved in $O(n \log n)$ time
  - There is feasible schedule for the problem $C2|Flex|C_{\max}$ with the makespan $C^{I}_{\max} \leq C^{Flex}_{\max} + (a+1)n_{\max}/2$. Run time is $O(n \log n)$, is a 3/2-approximation algorithm for $C2|Flex|C_{\max}$.
  - There is an optimal algorithm for the problem $C2|Flex|C_{\max}$ if $n_{\max} \leq n/2$, with a running time $O(n \log n)$.

$C2|Flex|C_{\max}$

- Problem $C2|Flex|C_{\max}$
  - There exists an optimal schedule E for problem $C2|Flex|C_{\max}$ such that $\max\{r_j \mid j \in X_1^{(E)}\} \leq \min\{r_j \mid j \in X_2^{(E)}\}$.
  - There exists an optimal schedule for the problem $C2|Flex|C_{\max}$ and a separation number k, $1 \leq k \leq n-1$, such that containers $1,\dots,k$ are assigned to crane 1 and containers $k+1,\dots,n$ are assigned to crane 2.
  - relaxed problem $C2|Flex,meet|C_{\max}$ can be solved in $O(n \log n)$ time
  - There is feasible schedule for the problem $C2|Flex|C_{\max}$ with the makespan $C^I_{\max} \leq C^{Flex}_{\max} + (a+1)n_{\max}/2$. Run time is $O(n \log n)$, is a 3/2-approximation algorithm for $C2|Flex|C_{\max}$.
  - There is an optimal algorithm for the problem $C2|Flex|C_{\max}$ if $n_{\max} \leq n/2$, with a running time $O(n \log n)$.

$C2|Flex|C_{max}$

- Problem $C2|Flex|C_{max}$
  - There exists an optimal schedule E for problem $C2|Flex|C_{max}$ such that $\max\{r_j \mid j \in X_1^{(E)}\} \leq \min\{r_j \mid j \in X_2^{(E)}\}$.
  - There exists an optimal schedule for the problem $C2|Flex|C_{max}$ and a separation number k, $1 \leq k \leq n-1$, such that containers $1, \ldots, k$ are assigned to crane 1 and containers $k+1, \ldots, n$ are assigned to crane 2.
  - relaxed problem $C2|Flex, meet|C_{max}$ can be solved in $O(n \log n)$ time
  - There is feasible schedule for the problem $C2|Flex|C_{max}$ with the makespan $C_{max}^I \leq C_{max}^{Flex} + (a+1)n_{max}/2$. Run time is $O(n \log n)$, is a 3/2-approximation algorithm for $C2|Flex|C_{max}$.
  - There is an optimal algorithm for the problem $C2|Flex|C_{max}$ if $n_{max} \leq n/2$, with a running time $O(n \log n)$.

$C2|Flex|C_{\max}$

- Problem $C2|Flex|C_{\max}$
  - There exists an optimal schedule E for problem $C2|Flex|C_{\max}$ such that $\max\{r_j \mid j \in X_1^{(E)}\} \leq \min\{r_j \mid j \in X_2^{(E)}\}$.
  - There exists an optimal schedule for the problem $C2|Flex|C_{\max}$ and a separation number k, $1 \leq k \leq n-1$, such that containers $1, \ldots, k$ are assigned to crane 1 and containers $k+1, \ldots, n$ are assigned to crane 2.
  - relaxed problem $C2|Flex, meet|C_{\max}$ can be solved in $O(n \log n)$ time
  - There is feasible schedule for the problem $C2|Flex|C_{\max}$ with the makespan $C^I_{\max} \leq C^{Flex}_{\max} + (a+1)n_{\max}/2$. Run time is $O(n \log n)$, is a 3/2-approximation algorithm for $C2|Flex|C_{\max}$.
  - There is an optimal algorithm for the problem $C2|Flex|C_{\max}$ if $n_{\max} \leq n/2$, with a running time $O(n \log n)$.

$C2|Flex|C_{\max}$

- Problem $C2|Flex|C_{\max}$
  - There exists an optimal schedule E for problem $C2|Flex|C_{\max}$ such that $\max\{r_j \mid j \in X_1^{(E)}\} \leq \min\{r_j \mid j \in X_2^{(E)}\}$.
  - There exists an optimal schedule for the problem $C2|Flex|C_{\max}$ and a separation number k, $1 \leq k \leq n-1$, such that containers $1, \ldots, k$ are assigned to crane 1 and containers $k+1, \ldots, n$ are assigned to crane 2.
  - relaxed problem $C2|Flex, meet|C_{\max}$ can be solved in $O(n \log n)$ time
  - There is feasible schedule for the problem $C2|Flex|C_{\max}$ with the makespan $C_{\max}^{I} \leq C_{\max}^{Flex} + (a+1)n_{\max}/2$. Run time is $O(n \log n)$, is a $3/2$-approximation algorithm for $C2|Flex|C_{\max}$.
  - There is an optimal algorithm for the problem $C2|Flex|C_{\max}$ if $n_{\max} \leq n/2$, with a running time $O(n \log n)$.

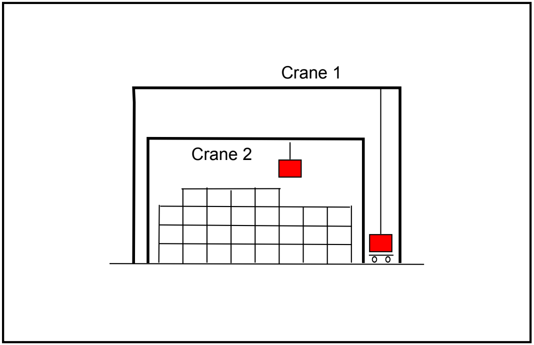
$C2|Flex|C_{\max}$

- Problem $C2|Flex|C_{\max}$
  - There exists an optimal schedule E for problem $C2|Flex|C_{\max}$ such that $\max\{r_j \mid j \in X_1^{(E)}\} \leq \min\{r_j \mid j \in X_2^{(E)}\}$.
  - There exists an optimal schedule for the problem $C2|Flex|C_{\max}$ and a separation number k, $1 \leq k \leq n-1$, such that containers $1, \dots, k$ are assigned to crane 1 and containers $k+1, \dots, n$ are assigned to crane 2.
  - relaxed problem $C2|Flex, meet|C_{\max}$ can be solved in $O(n \log n)$ time
  - There is feasible schedule for the problem $C2|Flex|C_{\max}$ with the makespan $C^{I}_{\max} \leq C^{Flex}_{\max} + (a+1)n_{\max}/2$. Run time is $O(n \log n)$, is a 3/2-approximation algorithm for $C2|Flex|C_{\max}$.
  - There is an optimal algorithm for the problem $C2|Flex|C_{\max}$ if $n_{\max} \leq n/2$, with a running time $O(n \log n)$.

$C2|GlobFix|C_{max}$

- Problem $C2|GlobFix|C_{max}$
  Similar algorithms guarantee an absolute deviation of $(a+1)n_{max}/2$ and a relative deviation of $3/2$ from the optimal value.

Sea Port



J. Nossack, D. Briskorn, E. Pesch: Container Dispatching and Conflict-Free Yard Crane Routing in an Automated Container Terminal, *Transportation Science* 52 (2018), 1059–1076

Crossover Crane Scheduling Problem

Crossover Crane Scheduling Problem

Problem Setting: European Layout, Crossover Cranes

Crossover Crane Scheduling Problem

Problem Setting: European Layout, Crossover Cranes

Transportation Request

- Transportation requests arise at the water- and landside transfer area and have to be handled by the yard cranes.
- Each transportation request defines an origin and destination.
- An **inbound request** is initially located at the waterside/landside transfer area and has to be transported to a predefined yard location. This yard location is typically determined beforehand by a pre-executed stacking algorithm (cf. Dorndorf / Schneider 2010).
- An **outbound request** starts at a well-defined position in the storage yard and has to be transported to the waterside/landside transfer area.
- **Housekeeping requests** are handled inside the storage yard and improve the storage location of containers in the block (Kemme 2011).

Crossover Crane Scheduling Problem

Problem Setting: European Layout, Crossover Cranes

Transportation Request

- Transportation requests arise at the water- and landside transfer area and have to be handled by the yard cranes.
- Each transportation request defines an origin and destination.
- An **inbound request** is initially located at the waterside/landside transfer area and has to be transported to a predefined yard location. This yard location is typically determined beforehand by a pre-executed stacking algorithm (cf. Dorndorf / Schneider 2010).
- An **outbound request** starts at a well-defined position in the storage yard and has to be transported to the waterside/landside transfer area.
- **Housekeeping requests** are handled inside the storage yard and improve the storage location of containers in the block (Kemme 2011).

Crossover Crane Scheduling Problem

Problem Setting: European Layout, Crossover Cranes

Transportation Request

- Transportation requests arise at the water- and landside transfer area and have to be handled by the yard cranes.
- Each transportation request defines an origin and destination.
- An **inbound request** is initially located at the waterside/landside transfer area and has to be transported to a predefined yard location. This yard location is typically determined beforehand by a pre-executed stacking algorithm (cf. Dorndorf / Schneider 2010).
- An **outbound request** starts at a well-defined position in the storage yard and has to be transported to the waterside/landside transfer area.
- **Housekeeping requests** are handled inside the storage yard and improve the storage location of containers in the block (Kemme 2011).

Crossover Crane Scheduling Problem

Problem Setting: European Layout, Crossover Cranes

Transportation Request

- Transportation requests arise at the water- and landside transfer area and have to be handled by the yard cranes.
- Each transportation request defines an origin and destination.
- An **inbound request** is initially located at the waterside/landside transfer area and has to be transported to a predefined yard location. This yard location is typically determined beforehand by a pre-executed stacking algorithm (cf. Dorndorf / Schneider 2010).
- An **outbound request** starts at a well-defined position in the storage yard and has to be transported to the waterside/landside transfer area.
- **Housekeeping requests** are handled inside the storage yard and improve the storage location of containers in the block (Kemme 2011).

Crossover Crane Scheduling Problem

Problem Setting: European Layout, Crossover Cranes

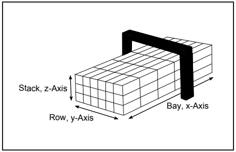
Transportation Request

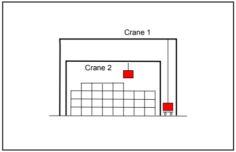
- Transportation requests arise at the water- and landside transfer area and have to be handled by the yard cranes.
- Each transportation request defines an origin and destination.
- An **inbound request** is initially located at the waterside/landside transfer area and has to be transported to a predefined yard location. This yard location is typically determined beforehand by a pre-executed stacking algorithm (cf. Dorndorf / Schneider 2010).
- An **outbound request** starts at a well-defined position in the storage yard and has to be transported to the waterside/landside transfer area.
- **Housekeeping requests** are handled inside the storage yard and improve the storage location of containers in the block (Kemme 2011).

Crossover Crane Scheduling Problem

Crane Interferences - Crossover Cranes

- If dual cranes are employed in a single yard block, crane interferences have to be prevented.
- Since both cranes run on different rail tracks, they can move freely from water- to landside and in the reverse direction.
- Interferences may occur if the outer crane works (i.e. picks up or delivers a container) in a certain bay and the inner crane wants to pass or work in the same bay as well.

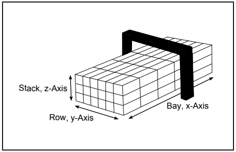


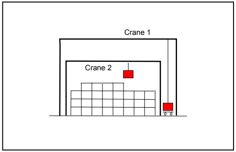


Crossover Crane Scheduling Problem

Crane Interferences - Crossover Cranes

- If dual cranes are employed in a single yard block, crane interferences have to be prevented.
- Since both cranes run on different rail tracks, they can move freely from water- to landside and in the reverse direction.
- Interferences may occur if the outer crane works (i.e. picks up or delivers a container) in a certain bay and the inner crane wants to pass or work in the same bay as well.

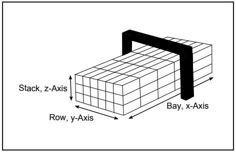


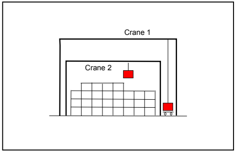


Crossover Crane Scheduling Problem

Crane Interferences - Crossover Cranes

- If dual cranes are employed in a single yard block, crane interferences have to be prevented.
- Since both cranes run on different rail tracks, they can move freely from water- to landside and in the reverse direction.
- Interferences may occur if the outer crane works (i.e. picks up or delivers a container) in a certain bay and the inner crane wants to pass or work in the same bay as well.





Crossover Crane Scheduling Problem

Crossover Crane Scheduling Problem (CCSP)

- evaluates in which order (i.e., **crane routing**) and
- by which crane (i.e., **container dispatching**) the transportation requests are carried out
- such that crane interferences (i.e., **conflict-free crane scheduling**) are prevented, and
- the **makespan** is minimized.

Crossover Crane Scheduling Problem

Crossover Crane Scheduling Problem (CCSP)

- evaluates in which order (i.e., **crane routing**) and
- by which crane (i.e., **container dispatching**) the transportation requests are carried out
- such that crane interferences (i.e., **conflict-free crane scheduling**) are prevented, and
- the **makespan** is minimized.

Theorem

The Crossover Crane Scheduling Problem is strongly NP-hard.

Proof.

Reduction to 3-PARTITION. □

Solution Approach

Solution Approach: Logic-Based Benders Decomposition

The CCSP simultaneously solves

- a dispatch and routing problem,
- and a conflict-free scheduling problem.

Master Problem: Dispatch and Routing Problem

The dispatch and routing problem evaluates in which order and by which crane the requests are conducted.

Subproblem: Conflict-Free Scheduling Problem

The conflict-free scheduling problem guarantees that cranes do not interfere.

Solution Approach: Logic-Based Benders Decomposition

The CCSP simultaneously solves

- a dispatch and routing problem,
- and a conflict-free scheduling problem.

Master Problem: Dispatch and Routing Problem

The dispatch and routing problem evaluates in which order and by which crane the requests are conducted.

Subproblem: Conflict-Free Scheduling Problem

The conflict-free scheduling problem guarantees that cranes do not interfere.

Solution Approach: Logic-Based Benders Decomposition

The CCSP simultaneously solves

- a dispatch and routing problem,
- and a conflict-free scheduling problem.

Master Problem: Dispatch and Routing Problem

The dispatch and routing problem evaluates in which order and by which crane the requests are conducted.

Subproblem: Conflict-Free Scheduling Problem

The conflict-free scheduling problem guarantees that cranes do not interfere.

Solution Approach: Example

Example: Storage area with 8 bays, outer crane starts at bay 1, inner crane starts at bay 8

Table of Requests:

Request No.	Origin (Service Time)	Destination (Service Time)
1	3 (1)	5 (1)
2	3 (1)	5 (2)
3	4 (2)	3 (2)

Solution Approach: Example

Example: Storage area with 8 bays, outer crane starts at bay 1, inner crane starts at bay 8

Table of Requests:

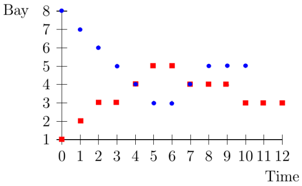
Request No.	Origin (Service Time)	Destination (Service Time)	Dispatch	Route Position
1	3 (1)	5 (1)	Outer Crane	1
2	3 (1)	5 (2)	Inner Crane	1
3	4 (2)	3 (2)	Outer Crane	2

Solution Approach: Example

Example: Storage area with 8 bays, outer crane starts at bay 1, inner crane starts at bay 8

Table of Requests:

Request No.	Origin (Service Time)	Destination (Service Time)	Dispatch	Route Position
1	3 (1)	5 (1)	Outer Crane	1
2	3 (1)	5 (2)	Inner Crane	1
3	4 (2)	3 (2)	Outer Crane	2

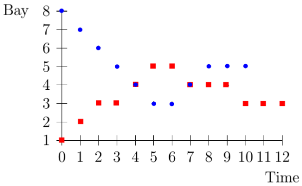


Solution Approach: Example

Example: Storage area with 8 bays, outer crane starts at bay 1, inner crane starts at bay 8

Table of Requests:

Request No.	Origin (Service Time)	Destination (Service Time)	Dispatch	Route Position
1	3 (1)	5 (1)	Outer Crane	1
2	3 (1)	5 (2)	Inner Crane	1
3	4 (2)	3 (2)	Outer Crane	2



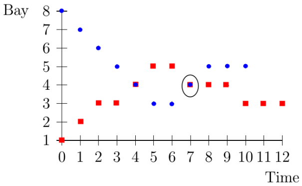
Are their any crane interferences?

Solution Approach: Example

Example: Storage area with 8 bays, outer crane starts at bay 1, inner crane starts at bay 8

Table of Requests:

Request No.	Origin (Service Time)	Destination (Service Time)	Dispatch	Route Position
1	3 (1)	5 (1)	Outer Crane	1
2	3 (1)	5 (2)	Inner Crane	1
3	4 (2)	3 (2)	Outer Crane	2



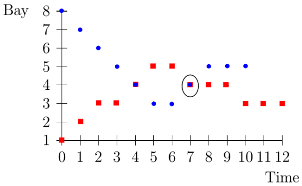
Are their any crane interferences?

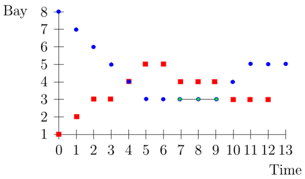
Solution Approach: Example

Example: Storage area with 8 bays, outer crane starts at bay 1, inner crane starts at bay 8

Table of Requests:

Request No.	Origin (Service Time)	Destination (Service Time)	Dispatch	Route Position
1	3 (1)	5 (1)	Outer Crane	1
2	3 (1)	5 (2)	Inner Crane	1
3	4 (2)	3 (2)	Outer Crane	2





Solution Approach: Sketch of Algorithm

Master Problem: Dispatch & Routing

$$
\begin{aligned}
&\min\ W && \\
&\sum_{i \in R \cup \{n+1\}} y^k_{0_k,i} = 1 && \forall k \in K \\
&\sum_{k \in K} \sum_{\substack{i \in R \cup \{0_k\} \\ i \neq j}} y^k_{i,j} = 1 && \forall j \in R \\
&\sum_{\substack{j \in R \cup \{n+1\} \\ i \neq j}} y^k_{i,j} - \sum_{\substack{j \in R \cup \{0_k\} \\ i \neq j}} y^k_{j,i} = 0 && \forall i \in R; k \in K \\
&\sum_{\substack{i,j \in S \\ i \neq j}} y^k_{i,j} \leq |S| - 1 && \forall S \subseteq R \cup \{0_k\}; k \in K \\
&\sum_{\substack{i \in R \cup \{0_k\}, j \in R \\ i \neq j}} y^k_{i,j} \cdot (t_{i,i} + t_{i,j} + s^O_i + s^D_i) \leq W && \forall k \in K \\
&y^k_{i,j} \in \{0,1\} && \forall i \in R \cup \{0_k\}; j \in R \cup \{n+1\}; i \neq j; k \in K \\
&W \in \mathbb{R}^+_0 && \forall k \in K
\end{aligned}
$$

Solution Approach: Sketch of Algorithm

Master Problem: Dispatch & Routing

$$\min W$$

$$\sum_{i \in R \cup \{n+1\}} y^k_{0_k,i} = 1 \qquad \forall k \in K$$

$$\sum_{k \in K} \sum_{\substack{i \in R \cup \{0_k\} \\ i \neq j}} y^k_{i,j} = 1 \qquad \forall j \in R$$

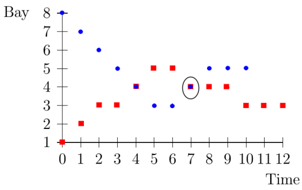
$$\sum_{\substack{j \in R \cup \{n+1\} \\ i \neq j}} y^k_{i,j} - \sum_{\substack{j \in R \cup \{0_k\} \\ i \neq j}} y^k_{j,i} = 0 \qquad \forall i \in R; k \in K$$

$$\sum_{\substack{i,j \in S \\ i \neq j}} y^k_{i,j} \leq |S| - 1 \qquad \forall S \subseteq R \cup \{0_k\}; k \in K$$

$$\sum_{\substack{i \in R \cup \{0_k\}, j \in R \\ i \neq j}} y^k_{i,j} \cdot (t_{i,i} + t_{i,j} + s^O_i + s^D_i) \leq W \qquad \forall k \in K$$

$$y^k_{i,j} \in \{0,1\} \qquad \forall i \in R \cup \{0_k\}; j \in R \cup \{n+1\}; i \neq j; k \in K$$

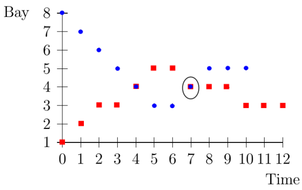
$$W \in \mathbb{R}^+_0 \qquad \forall k \in K$$



Solution Approach: Sketch of Algorithm

Master Problem: Dispatch & Routing

$$
\begin{aligned}
& \min\ W && \\
& \sum_{i\in R\cup\{n+1\}} y^k_{0_k,i} = 1 && \forall k \in K \\
& \sum_{k\in K}\ \sum_{\substack{i\in R\cup\{0_k\}\\ i\neq j}} y^k_{i,j} = 1 && \forall j \in R \\
& \sum_{\substack{j\in R\cup\{n+1\}\\ i\neq j}} y^k_{i,j} - \sum_{\substack{j\in R\cup\{0_k\}\\ i\neq j}} y^k_{j,i} = 0 && \forall i \in R; k \in K \\
& \sum_{\substack{i,j\in S\\ i\neq j}} y^k_{i,j} \le |S| - 1 && \forall S \subseteq R\cup\{0_k\}; k \in K \\
& \sum_{\substack{i\in R\cup\{0_k\}, j\in R\\ i\neq j}} y^k_{i,j}\cdot(t_{i,i} + t_{i,j} + s^O_i + s^D_i) \le W && \forall k \in K \\
& y^k_{i,j} \in \{0,1\} && \forall i \in R\cup\{0_k\}; j \in R\cup\{n+1\}; i\neq j; k \in K \\
& W \in \mathbb{R}^+_0 && \forall k \in K
\end{aligned}
$$



Subproblem: Conflict-Free Crane Schedule

Solution Approach: Sketch of Algorithm

Master Problem: Dispatch & Routing

$$\min W$$

$$\sum_{i \in R \cup \{n+1\}} y^k_{0_k,i} = 1 \qquad \forall k \in K$$

$$\sum_{k \in K} \sum_{\substack{i \in R \cup \{0_k\} \\ i \neq j}} y^k_{i,j} = 1 \qquad \forall j \in R$$

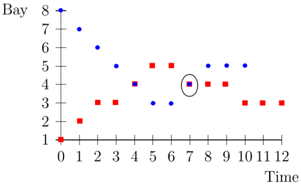
$$\sum_{\substack{j \in R \cup \{n+1\} \\ i \neq j}} y^k_{i,j} - \sum_{\substack{j \in R \cup \{0_k\} \\ i \neq j}} y^k_{j,i} = 0 \qquad \forall i \in R; k \in K$$

$$\sum_{\substack{i,j \in S \\ i \neq j}} y^k_{i,j} \leq |S| - 1 \qquad \forall S \subseteq R \cup \{0_k\}; k \in K$$

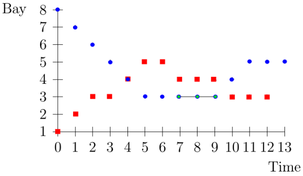
$$\sum_{\substack{i \in R \cup \{0_k\}, j \in R \\ i \neq j}} y^k_{i,j} \cdot (t_{i,i} + t_{i,j} + s^O_i + s^D_i) \leq W \qquad \forall k \in K$$

$$y^k_{i,j} \in \{0,1\} \qquad \forall i \in R \cup \{0_k\}; j \in R \cup \{n+1\}; i \neq j; k \in K$$

$$W \in \mathbb{R}^+_0 \qquad \forall k \in K$$



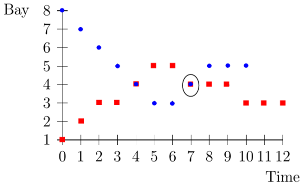
Subproblem: Conflict-Free Crane Schedule



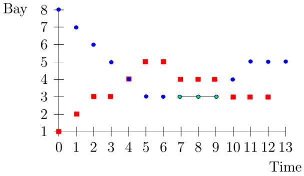
Solution Approach: Sketch of Algorithm

Master Problem: Dispatch & Routing

$$
\begin{aligned}
&\min\ W \\
&\sum_{i \in R \cup \{n+1\}} y^k_{0_k,i} = 1 && \forall k \in K \\
&\sum_{k \in K} \sum_{\substack{i \in R \cup \{0_k\} \\ i \neq j}} y^k_{i,j} = 1 && \forall j \in R \\
&\sum_{\substack{j \in R \cup \{n+1\} \\ i \neq j}} y^k_{i,j} - \sum_{\substack{j \in R \cup \{0_k\} \\ i \neq j}} y^k_{j,i} = 0 && \forall i \in R; k \in K \\
&\sum_{\substack{i,j \in S \\ i \neq j}} y^k_{i,j} \leq |S| - 1 && \forall S \subseteq R \cup \{0_k\}; k \in K \\
&\sum_{\substack{i \in R \cup \{0_k\}, j \in R \\ i \neq j}} y^k_{i,j} \cdot (t_{i,i} + t_{i,j} + s^O_i + s^D_i) \leq W && \forall k \in K \\
&y^k_{i,j} \in \{0,1\} && \forall i \in R \cup \{0_k\}; j \in R \cup \{n+1\}; i \neq j; k \in K \\
&W \in \mathbb{R}^+_0 && \forall k \in K
\end{aligned}
$$



Subproblem: Conflict-Free Crane Schedule



Logic-Based Benders Constraints

$$\hat{W}^h \Big(1 - \sum_{k \in K} \sum_{(i,j) \in J^h_k} (1 - y^k_{i,j})\Big) \leq W \qquad \forall h \in H$$

Master Problem: Dispatch & Routing

Master Problem: Dispatch & Routing

Notation

Parameter	Parameter Description
K	cranes, $K := \{1,2\}$ with 1 outer crane and 2 inner crane
Q	yard bays, $Q := \{1,\dots,l\}$
R	set of transportation requests, $R := \{1,\dots,n\}$
O_i	origin location of request $i \in R$, $O_i \in Q$
D_i	destination location of request $i \in R$, $D_i \in Q$
S_k	initial location of crane k, $S_k \in Q$
s_{O_i}	service time at origin O_i, $s_{O_i} \geq 0$
s_{D_i}	service time at destination D_i, $s_{D_i} \geq 0$
$t_{i,j}$	travel time from the destination D_i of request i to the origin O_j of request j, $t_{i,j} := \lvert D_i - O_j \rvert$
$t_{i,i}$	travel time from the origin O_i to the destination D_i of request i, $t_{i,i} := \lvert O_i - D_i \rvert$

Decision Variables

$$y_{ij}^k = \begin{cases} 1, & \text{if request } j \text{ is conducted after request } i \\ & \text{by crane } k \\ 0, & \text{otherwise} \end{cases}$$

$$W \in \mathbb{R}_0^+ = \text{makespan}$$

Master Problem: Dispatch & Routing

Master Problem: Mathematical Model

$$\min\ W$$

$$\sum_{i \in R \cup \{n+1\}} y^k_{0_k,i} = 1 \qquad \forall k \in K$$

$$\sum_{k \in K} \sum_{\substack{i \in R \cup \{0_k\} \\ i \neq j}} y^k_{i,j} = 1 \qquad \forall j \in R$$

$$\sum_{\substack{j \in R \cup \{n+1\} \\ i \neq j}} y^k_{i,j} - \sum_{\substack{j \in R \cup \{0_k\} \\ i \neq j}} y^k_{j,i} = 0 \qquad \forall i \in R; k \in K$$

$$\sum_{\substack{i,j \in S \\ i \neq j}} y^k_{i,j} \leq |S| - 1 \qquad \forall S \subseteq R \cup \{0_k\}; k \in K$$

$$\sum_{\substack{i \in R \cup \{0_k\}, j \in R \\ i \neq j}} y^k_{i,j} \cdot (t_{i,i} + t_{i,j} + s_{O_i} + s_{D_i}) \leq W \qquad \forall k \in K$$

$$y^k_{i,j} \in \{0,1\} \qquad \forall i \in R \cup \{0_k\}; j \in R \cup \{n+1\}; i \neq j; k \in K$$

$$W \in \mathbb{R}^+_0 \qquad \forall k \in K$$

Subproblem: Conflict-Free Crane Scheduling

Subproblem: Conflict-Free Crane Scheduling

- Crane interferences are resolved in the subproblem.
- For a given dispatch and route, the subproblem determines a conflict-free crane schedule with minimum makespan $\hat{W}$.
- Briskorn / Angeloudis (2016) provide a polynomial algorithm that reduces the conflict-free crane schedule problem to a shortest path problem in specially designed acyclic arc-weighted directed graph.

Subproblem: Conflict-Free Crane Scheduling

- Crane interferences are resolved in the subproblem.
- For a given dispatch and route, the subproblem determines a conflict-free crane schedule with minimum makespan $\hat{W}$.
- Briskorn / Angeloudis (2016) provide a polynomial algorithm that reduces the conflict-free crane schedule problem to a shortest path problem in specially designed acyclic arc-weighted directed graph.

Subproblem: Conflict-Free Crane Scheduling

- Crane interferences are resolved in the subproblem.
- For a given dispatch and route, the subproblem determines a conflict-free crane schedule with minimum makespan $\hat{W}$.
- Briskorn / Angeloudis (2016) provide a polynomial algorithm that reduces the conflict-free crane schedule problem to a shortest path problem in specially designed acyclic arc-weighted directed graph.

Subproblem: Conflict-Free Crane Scheduling

- Crane interferences are resolved in the subproblem.
- For a given dispatch and route, the subproblem determines a conflict-free crane schedule with minimum makespan $\hat{W}$.
- Briskorn / Angeloudis (2016) provide a polynomial algorithm that reduces the conflict-free crane schedule problem to a shortest path problem in specially designed acyclic arc-weighted directed graph.

Logic-Based Benders Constraints

$$\hat{W}^h(1 - \sum_{k \in K} \sum_{(i,j) \in J_k^h} (1 - y_{i,j}^k)) \leq W \qquad \forall h \in H$$

Computational Study

System Specifications, Data Set, Implementation

- System specifications: Intel Pentium Core 2 Duo, 2.2 GHz PC, 4GB system memory
- Mathematical model: CPLEX 12.5 Concert Technology
- Data set: Instance Generator by Briskorn / Jaehn / Wiehl 2019

Computational Study (1 Hour)

# Requests = 15						# Requests = 20					
Type	LB	Obj. Val.	CPU (in sec)	#SEC	#LBC	Type	LB	Obj. Val.	CPU (in sec)	#SEC	#LBC
6/6/3	258.00	261.00	14.09	70	6	10/9/1	414.00	415.00	3600(0.24%)	5008	6543
5/7/3	240.50	243.00	28.60	146	56	9/7/4	348.00	349.00	300.45	1042	1230
4/8/3	270.50	271.00	65.54	176	41	13/6/1	377.00	379.00	3600(0.53%)	4456	5944
7/5/3	236.50	237.00	3.00	70	7	5/10/5	345.50	346.00	3600(0.14%)	242	31
7/5/3	262.50	264.00	22.42	448	112	7/10/13	349.50	350.00	142.64	84	3
7/6/2	298.50	300.00	9.26	72	29	5/11/4	337.50	338.00	431.81	142	24
4/7/4	254.50	258.00	9.52	170	29	9/8/3	354.50	355.00	5.98	110	19
3/8/4	257.00	257.00	2.98	164	31	8/8/4	384.00	387.00	3600(0.78%)	5622	4236
8/4/3	231.50	234.00	14.39	382	311	9/8/3	321.00	323.00	3600(0.62%)	2298	5104
4/8/3	263.00	263.00	6.00	156	31	6/10/4	315.50	317.00	393.15	712	1378
6/7/2	215.50	217.00	2.33	58	24	9/8/3	315.50	316.00	30.03	138	22
6/6/3	246.00	247.00	68.79	434	383	8/8/4	298.50	299.00	265.28	352	45
9/4/2	286.00	287.00	471.50	882	2594	10/8/2	355.50	356.00	113.28	186	50
6/5/4	241.00	241.00	1.41	44	6	8/9/3	328.00	328.00	2.45	158	30
6/5/4	235.50	237.00	3.93	182	46	9/10/1	379.50	380.00	327.38	200	21
5/7/3	248.00	249.00	0.42	34	4	9/10/1	378.00	379.00	3600(0.26%)	2030	9179
6/5/4	235.50	237.00	1.32	148	20	10/4/6	322.00	325.00	3600(0.92%)	3780	5098
6/6/3	260.00	260.00	3.37	236	62	9/7/4	355.00	355.00	8.49	106	22
6/6/3	282.00	283.00	297.10	308	214	9/7/4	365.00	366.00	0.27	1738	8635
7/6/2	284.50	285.00	2.88	38	4	9/8/3	298.50	299.00	5.76	100	21
3/9/3	279.50	281.00	7.46	48	1	8/10/12	365.50	367.00	3600(0.41%)	3822	2925
3/9/3	322.50	323.00	489.85	64	5	10/8/2	353.50	355.00	23.01	274	47
3/9/3	241.00	241.00	0.41	30	4	11/6/3	355.50	356.00	9.81	266	100
6/6/3	234.00	236.00	0.96	64	10	7/10/3	366.50	367.00	8.72	372	99
5/8/2	265.50	266.00	3.85	164	20	9/9/2	321.50	322.00	357.71	260	29
9/4/2	266.00	266.00	5.63	152	42	10/6/4	413.00	413.00	2.32	276	31
3/8/4	254.50	256.00	67.97	254	89	9/9/2	329.00	329.00	2.63	228	16
7/8/0	284.00	287.00	7.79	40	2	7/10/3	373.00	373.00	0.96	72	5
7/4/4	300.50	302.00	19.67	188	61	8/7/5	381.00	382.00	3600(0.26%)	3186	3611
5/7/3	247.50	248.00	1.14	34	1	4/10/6	373.50	374.00	914.51	510	106
Avg.	260.05	261.23	54.45	175.20	141.50		352.47	353.33	1191.55	1259.00	1820.13

Computational Study (10 Seconds)

# Requests = 30						# Requests = 40					
Type	LB	Obj. Val.	Gap (in %)	#SEC	#LBC	Type	LB	Obj. Val.	Gap (in %)	#SEC	#LBC
12/14/4	610.00	618.00	1.29	110	13	20/13/7	695.00	702.00	1.00	200	27
10/16/4	531.50	538.00	1.21	98	14	12/19/9	720.00	732.00	1.64	260	22
9/14/7	483.00	498.00	3.01	150	13	15/17/8	621.00	628.00	1.11	224	28
12/14/4	472.50	480.00	1.56	164	17	20/15/5	702.50	710.00	1.06	216	31
15/10/5	524.00	529.00	0.95	212	22	15/17/8	663.50	671.00	1.21	176	35
17/9/4	500.00	504.00	0.79	234	9	14/15/11	703.00	713.00	1.40	308	28
12/12/6	505.00	506.00	0.20	342	53	15/13/12	699.50	705.00	0.78	244	19
13/14/3	517.50	518.00	0.10	286	24	19/14/7	675.50	690.00	2.10	258	30
12/11/7	576.00	584.00	1.37	402	53	12/21/7	706.50	714.00	1.05	232	24
18/5/7	495.00	500.00	1.00	568	81	16/15/9	645.00	655.00	1.53	284	20
14/14/2	550.00	560.00	1.79	464	59	15/16/9	688.00	702.00	1.99	210	28
10/12/8	525.50	559.00	1.16	524	84	12/19/9	603.50	611.00	1.23	214	16
8/16/6	496.50	498.00	0.30	472	71	16/16/8	695.50	713.00	2.45	238	20
11/14/5	473.50	484.00	2.17	496	68	11/19/10	664.50	670.00	0.82	112	18
13/14/3	541.50	542.00	0.09	278	38	18/15/7	672.50	682.00	1.39	236	23
11/15/4	493.00	496.00	0.60	404	102	19/15/6	719.00	730.00	1.51	144	14
12/12/6	540.50	549.00	1.55	488	66	18/14/8	708.00	726.00	2.48	196	11
10/16/4	482.50	483.00	0.10	158	22	15/17/8	624.00	629.00	0.79	192	16
11/16/3	543.50	547.00	0.64	478	73	19/13/8	675.00	682.00	1.03	216	25
19/8/3	491.50	494.00	0.51	444	33	19/14/7	717.50	726.00	1.17	272	17
14/10/6	493.50	501.00	1.50	374	49	20/10/10	683.00	687.00	0.58	188	17
14/11/5	517.50	526.00	1.62	398	55	20/14/6	693.50	702.00	1.21	154	25
10/14/6	482.00	492.00	2.03	446	32	14/18/8	706.00	719.00	1.81	234	36
15/11/4	535.00	540.00	0.93	362	51	15/17/8	639.50	643.00	0.54	172	31
11/9/10	547.00	554.00	1.26	188	17	12/18/10	701.50	707.00	0.78	212	18
9/14/7	517.50	521.00	0.67	172	20	16/16/8	668.00	671.00	0.45	123	16
11/12/7	526.50	533.00	1.22	136	22	16/17/7	577.50	582.00	0.77	238	31
15/11/4	482.50	486.00	0.72	126	7	20/12/8	713.00	720.00	0.97	182	12
12/16/2	560.50	567.00	1.15	172	11	16/16/8	709.50	723.00	1.87	238	27
13/10/7	600.00	606.00	0.99	146	15	14/18/8	673.50	678.00	0.66	206	26
Avg.	520.48	527.10	1.08	309.73	39.80		678.80	687.43	1.25	212.63	23.03

Computational Study (60 Seconds)

# Requests = 30						# Requests = 40					
Type	LB	Obj. Val.	Gap (in %)	#SEC	#LBC	Type	LB	Obj. Val.	Gap (in %)	#SEC	#LBC
12/14/4	610.00	613.00	0.49	594	99	20/13/7	695.00	701.00	0.86	676	171
10/16/4	531.50	536.00	0.84	488	80	12/19/9	720.00	722.00	0.28	432	39
9/14/7	483.00	490.00	1.43	686	107	15/17/8	621.00	628.00	1.11	500	82
12/14/4	472.50	473.00	0.11	304	35	20/15/5	702.50	708.00	0.78	882	290
15/10/5	524.00	528.00	0.76	1842	304	15/17/8	663.50	667.00	0.52	1110	201
17/9/4	500.00	500.00	0.00	650	53	14/15/11	703.00	705.00	0.28	972	154
12/12/6	505.00	506.00	0.20	858	616	15/13/12	699.50	705.00	0.78	482	62
13/14/3	517.50	518.00	0.10	286	24	19/14/7	675.50	687.00	1.67	378	48
12/11/7	576.00	580.00	0.69	574	83	12/21/7	706.50	713.00	0.91	370	56
18/5/7	495.00	500.00	1.00	646	94	16/15/9	645.00	646.00	0.15	894	224
14/14/2	550.00	557.00	1.26	772	137	15/16/9	688.00	689.00	0.15	1022	238
10/12/8	525.50	558.00	0.99	1874	298	12/19/9	603.50	605.00	0.25	684	82
8/16/6	496.50	497.00	0.10	872	169	16/16/8	695.50	699.00	0.50	464	45
11/14/5	473.50	475.00	0.32	1042	313	11/19/10	664.50	670.00	0.82	320	47
13/14/3	541.50	542.00	0.09	278	38	18/15/7	672.50	682.00	1.39	718	106
11/15/4	493.00	496.00	0.60	472	132	19/15/6	719.00	726.00	0.96	984	260
12/12/6	540.50	546.00	1.01	844	145	18/14/8	708.00	716.00	1.12	1016	157
10/16/4	482.50	483.00	0.10	158	22	15/17/8	624.00	624.00	0.00	318	46
11/16/3	543.50	545.00	0.28	1212	235	19/13/8	675.00	682.00	1.03	472	63
19/8/3	491.50	493.00	0.30	1280	213	19/14/7	717.50	726.00	1.17	384	31
14/10/6	493.50	494.00	0.10	1138	212	20/10/10	683.00	687.00	0.58	566	89
14/11/5	517.50	519.00	0.29	1474	466	20/14/6	693.50	701.00	1.07	850	180
10/14/6	482.00	484.00	0.41	1082	81	14/18/8	706.00	714.00	1.21	1096	166
15/11/4	535.00	538.00	0.56	1220	234	15/17/8	639.50	641.00	0.23	670	140
11/9/10	547.00	550.00	0.55	770	126	12/18/10	701.50	707.00	0.78	328	28
9/14/7	517.50	519.00	0.29	442	49	16/16/8	668.00	671.00	0.45	138	17
11/12/7	526.50	527.00	0.09	292	60	16/17/7	577.50	581.00	0.60	346	47
15/11/4	482.50	483.00	0.10	190	25	20/12/8	713.00	719.00	0.83	620	70
12/16/2	560.50	561.00	0.09	796	88	16/16/8	709.50	717.00	1.05	1136	174
13/10/7	600.00	604.00	0.66	1244	288	14/18/8	673.50	678.00	0.66	986	155
Avg.	520.48	523.83	0.46	812.67	160.87		678.80	683.90	0.74	660.47	115.60

Computational Study

Comparison with Simulated Annealing

- Simulated Annealing (SA) by Vis / Roodbergen 2010
- SA: The basic idea is to randomly assign requests to either crane and to solve for each crane a single-cane routing problem to optimality by a solution approach presented by Vis / Roodbergen 2010.
- We implemented two variants of the SA (SA1 and SA2).

Computational Study (60 Seconds, Simulated Annealing)

# Requests = 20					# Requests = 30				
Type	LB	Obj. Val. BD	Obj. Val. SA1	Obj. Val. SA2	Type	LB	Obj. Val. BD	Obj. Val. SA1	Obj. Val. SA2
9/11	384.50	387.00 (0.65%)	388.00 (0.90%)	388.00 (0.90%)	15/15	574.00	576.00 (0.35%)	578.00 (0.69%)	579.00 (0.86%)
11/9	358.00	358.00 (0.00%)	362.00 (1.10%)	363.00 (1.38%)	19/11	604.50	605.00 (0.08%)	606.00 (0.25%)	607.00 (0.41%)
8/12	405.00	405.00 (0.00%)	405.00 (0.00%)	406.00 (0.25%)	13/17	506.00	509.00 (0.59%)	513.00 (1.36%)	510.00 (0.78%)
13/7	353.50	354.00 (0.14%)	357.00 (0.98%)	356.00 (0.70%)	15/15	503.00	503.00 (0.00%)	507.00 (0.79%)	508.00 (0.98%)
9/11	353.00	356.00 (0.84%)	358.00 (1.40%)	360.00 (1.94%)	15/15	511.00	515.00 (0.00%)	517.00 (1.16%)	517.00 (1.16%)
10/10	452.00	452.00 (0.00%)	452.00 (0.00%)	452.00 (0.00%)	17/13	498.00	498.00 (0.00%)	501.00 (0.60%)	504.00 (1.19%)
9/11	356.00	356.00 (0.00%)	360.00 (1.11%)	357.00 (0.28%)	18/12	551.00	551.00 (0.00%)	555.00 (0.72%)	554.00 (0.54%)
10/10	362.50	363.00 (0.14%)	364.00 (0.41%)	363.00 (0.14%)	13/17	513.00	518.00 (0.97%)	518.00 (0.97%)	519.00 (1.16%)
10/10	368.00	372.00 (1.08%)	374.00 (1.60%)	372.00 (1.08%)	15/15	543.50	544.00 (0.09%)	551.00 (1.36%)	546.00 (0.46%)
10/10	360.50	362.00 (0.41%)	364.00 (0.96%)	363.00 (0.69%)	15/15	483.00	483.00 (0.00%)	488.00 (1.02%)	487.00 (0.82%)
9/11	299.00	299.00 (0.00%)	302.00 (0.99%)	303.00 (1.32%)	15/15	583.00	587.00 (0.68%)	589.00 (1.02%)	587.00 (0.68%)
8/12	313.50	318.00 (1.42%)	320.00 (2.03%)	318.00 (1.42%)	16/14	525.00	529.00 (0.76%)	530.00 (0.94%)	532.00 (1.32%)
9/11	364.00	365.00 (0.27%)	367.00 (0.82%)	366.00 (0.55%)	16/14	560.50	568.00 (1.32%)	569.00 (1.49%)	569.00 (1.49%)
11/9	352.00	353.00 (0.28%)	356.00 (1.12%)	356.00 (1.12%)	15/15	463.00	467.00 (0.86%)	469.00 (1.28%)	469.00 (1.28%)
11/9	366.50	368.00 (0.41%)	368.00 (0.41%)	370.00 (0.95%)	13/17	556.50	564.00 (1.33%)	561.00 (0.80%)	562.00 (0.98%)
11/9	397.50	398.00 (0.13%)	399.00 (0.38%)	399.00 (0.38%)	15/15	567.00	569.00 (0.35%)	571.00 (0.70%)	569.00 (0.35%)
10/10	360.00	361.00 (0.28%)	361.00 (0.28%)	361.00 (0.28%)	13/17	549.00	553.00 (0.72%)	556.00 (1.26%)	555.00 (1.08%)
10/10	382.50	383.00 (0.13%)	384.00 (0.39%)	386.00 (0.91%)	15/15	562.00	562.00 (0.00%)	564.00 (0.35%)	565.00 (0.53%)
12/8	382.00	384.00 (0.52%)	386.00 (1.04%)	386.00 (1.04%)	17/13	584.50	585.00 (0.09%)	588.00 (0.60%)	587.00 (0.43%)
10/10	353.00	354.00 (0.28%)	355.00 (0.56%)	354.00 (0.28%)	12/18	573.00	573.00 (0.00%)	573.00 (0.00%)	574.00 (0.17%)
10/10	331.00	331.00 (0.00%)	337.00 (1.78%)	334.00 (0.90%)	17/13	502.50	503.00 (0.10%)	510.00 (1.47%)	509.00 (1.28%)
10/10	396.50	401.00 (1.12%)	404.00 (1.86%)	402.00 (1.37%)	15/15	534.00	537.00 (0.56%)	540.00 (1.11%)	540.00 (1.11%)
8/12	375.00	375.00 (0.00%)	378.00 (0.79%)	376.00 (0.27%)	13/17	504.50	507.00 (0.49%)	507.00 (0.49%)	511.00 (1.27%)
7/13	367.00	371.00 (1.08%)	372.00 (1.34%)	372.00 (1.34%)	15/15	491.50	492.00 (0.10%)	495.00 (0.71%)	494.00 (0.51%)
10/10	320.00	320.00 (0.00%)	322.00 (0.62%)	322.00 (0.62%)	13/17	570.50	576.00 (0.95%)	575.00 (0.78%)	577.00 (1.13%)
9/11	367.50	368.00 (0.14%)	368.00 (0.14%)	368.00 (0.14%)	17/13	611.00	612.00 (0.16%)	613.00 (0.33%)	611.00 (0.00%)
10/10	363.00	365.00 (0.55%)	367.00 (1.09%)	367.00 (1.09%)	14/16	550.00	553.00 (0.54%)	556.00 (1.08%)	556.00 (1.08%)
12/8	382.50	384.00 (0.39%)	386.00 (0.91%)	388.00 (1.42%)	11/19	483.00	483.00 (0.00%)	483.00 (0.00%)	484.00 (0.21%)
9/11	369.00	371.00 (0.54%)	372.00 (0.81%)	371.00 (0.54%)	19/11	548.00	553.00 (0.90%)	553.00 (0.90%)	553.00 (0.90%)
11/9	377.00	377.00 (0.00%)	380.00 (0.79%)	380.00 (0.79%)	17/13	539.00	540.00 (0.19%)	543.00 (0.74%)	541.00 (0.37%)
Avg.	365.72	367.03 (0.36%)	368.93 (0.89%)	368.63 (0.80%)		538.15	540.50 (0.43%)	542.63 (0.83%)	542.53 (0.82%)

Crane Scheduling in Container Terminals at Seaports

Erwin Pesch

University of Siegen, Department of Management Information Science, Germany

IARIA BUSTECH 2023, June 26-30, 2023

Universität Siegen