Efficient Propagators for Global Constraints

Claude-Guy Quimper

Supervisor: Alejandro López-Ortiz

University of Waterloo

Outline

- My first contact with constraint programming
- The all-different constraint
- The global cardinality constraint
- The inter-distance constraint
- Post-doctoral work

My first contact with constraint programming

- I took Peter's course in Constraint Programming
- The field requires efficient algorithms that are executed gazillions of times.
- Project: To implement Thiel and Mehlhorn's alldiff propagator.

Peter van Beek

The All-Different Constraint

$$\textsc{All-Different}(X_1, \ldots, X_n) \iff X_i \neq X_j$$

- Scheduling: We want execution times to be all different.
- Encoding permutations.
- Sometimes, one simply wants things to be different!

The All-Different Constraint

$$\textsc{All-Different}(X_1, \ldots, X_n) \iff X_i \neq X_j$$

The All-Different Constraint

$$\textsc{All-Different}(X_1, \ldots, X_n) \iff X_i \neq X_j$$

Régin '94	Domain	$O(n^{1.5}d)$

The All-Different Constraint

AL… X_j

Rég… …d)

Domain Consistency (GAC)

$$
\begin{aligned}
X_1 &\in \{\ \ 2\ \ \ \ 4\ \ 5\} \\
X_2 &\in \{\ \ \ \ \ 3\ \ \ \ 5\} \\
X_3 &\in \{1\ \ \ 3\ \ \ \ \ \ \} \\
X_4 &\in \{\ \ 2\ \ 3\ \ \ \ \ \ \}
\end{aligned}
$$

The All-Different Constraint

AL… X_j

Rég… …d)

Domain Consistency (GAC)

$$
\begin{aligned}
X_1 &\in \{\quad 2 \quad\quad 4 \quad 5\} \\
X_2 &\in \{\quad\quad 3 \quad\quad 5\} \\
X_3 &\in \{1 \quad\quad 3 \quad\quad\quad\} \\
X_4 &\in \{\quad 2 \quad 3 \quad\quad\quad\}
\end{aligned}
$$

Remove all inconsistent values

The All-Different Constraint

ALL… X_j

Rég… …d)

Domain Consistency (GAC)

$$
\begin{aligned}
X_1 &\in \{\quad 2 \qquad 4 \quad\} \\
X_2 &\in \{\qquad\qquad\quad 5\} \\
X_3 &\in \{1 \qquad 3 \qquad\} \\
X_4 &\in \{\quad 2 \qquad\qquad\}
\end{aligned}
$$

Remove all inconsistent values

The All-Different Constraint

$$\textsc{All-Different}(X_1, \ldots, X_n) \iff X_i \neq X_j$$

Régin '94	Domain	$O(n^{1.5}d)$

The All-Different Constraint

$$\textsc{All-Different}(X_1, \ldots, X_n) \iff X_i \neq X_j$$

Régin '94	Domain	$O(n^{1.5}d)$
Leconte '96	Range	$O(n^2)$
Puget '98	Bounds	$O(n \log n)$
Mehlhorn & Thiel	Bounds	$O(n)$
López-Ortiz, Quimper, Tromp, & van Beek	Bounds	$O(n)$

The All-Different Constraint

Range Consistency

$$X_1 \in \{\ \ 2\ \ \ \ 4\ \ 5\}$$
$$X_2 \in \{\ \ \ \ 3\ \ \ \ 5\}$$
$$X_3 \in \{1\ \ \ \ 3\ \ \ \ \}$$
$$X_4 \in \{\ \ 2\ \ 3\ \ \ \ \}$$

1) Make domains intervals

2) Remove all inconsistent values

van Beek

The All-Different Constraint

Range Consistency

$$
\begin{aligned}
X_1 &\in \{\quad 2 \quad 3 \quad 4 \quad 5\} \\
X_2 &\in \{\qquad\quad 3 \quad 4 \quad 5\} \\
X_3 &\in \{1 \quad 2 \quad 3 \qquad\qquad\} \\
X_4 &\in \{\quad 2 \quad 3 \qquad\qquad\}
\end{aligned}
$$

1) Make domains intervals

2) Remove all inconsistent values

R		
Le		
P		
Mehl		
Lóp Quim van Beek		

The All-Different Constraint

Range Consistency

$$
\begin{aligned}
X_1 &\in \{\quad 2 \quad 3 \quad 4 \quad\} \\
X_2 &\in \{\quad\quad 3 \quad\quad 5\} \\
X_3 &\in \{1 \quad\quad\quad\quad\} \\
X_4 &\in \{\quad 2 \quad 3 \quad\quad\}
\end{aligned}
$$

1) Make domains intervals

2) Remove all inconsistent values

A

R		
Le		
P		
Mehl		
Lóp Quim van Beek		

The All-Different Constraint

$$\text{All-Different}(X_1, \ldots, X_n) \iff X_i \neq X_j$$

Régin '94	Domain	$O(n^{1.5}d)$
Leconte '96	Range	$O(n^2)$
Puget '98	Bounds	$O(n \log n)$

The All-Different Constraint

Bounds Consistency

$$
\begin{array}{l}
X_1 \in \{\ \ \ 2 \ \ \ \ \ \ 4\ \ 5\} \\
X_2 \in \{\ \ \ \ \ \ 3 \ \ \ \ \ \ 5\} \\
X_3 \in \{1 \ \ \ \ 3 \ \ \ \ \ \ \ \ \} \\
X_4 \in \{\ \ \ 2\ \ 3 \ \ \ \ \ \ \ \ \}
\end{array}
$$

1) Make domains intervals

2) Shrink intervals

The All-Different Constraint

R
Le
P
Mehl
Lóp Quim V

Bounds Consistency

$$
\begin{array}{llllll}
X_1 \in \{ & & 2 & 3 & 4 & 5\} \\
X_2 \in \{ & & & 3 & 4 & 5\} \\
X_3 \in \{1 & & 2 & 3 & & \} \\
X_4 \in \{ & & 2 & 3 & & \}
\end{array}
$$

1) Make domains intervals

2) Shrink intervals

The All-Different Constraint

Bounds Consistency

$$
\begin{array}{l}
X_1 \in \{\quad 2 \quad 3 \qquad\qquad \} \\
X_2 \in \{\qquad\qquad\quad 4 \quad 5\} \\
X_3 \in \{1 \quad 2 \qquad\qquad\quad \} \\
X_4 \in \{\quad 2 \quad 3 \qquad\qquad \}
\end{array}
$$

1) Make domains intervals

2) Shrink intervals

The All-Different Constraint

$$\textsc{All-Different}(X_1, \ldots, X_n) \iff X_i \neq X_j$$

Régin '94	Domain	$O(n^{1.5}d)$
Leconte '96	Range	$O(n^2)$
Puget '98	Bounds	$O(n \log n)$
Mehlhorn & Thiel	Bounds	$O(n)$

The All-Different Constraint

$$\text{All-Different}(X_1, \ldots, X_n) \iff X_i \neq X_j$$

Régin '94	Domain	$O(n^{1.5}d)$
Leconte '96	Range	$O(n^2)$
Puget '98	Bounds	$O(n \log n)$
Mehlhorn & Thiel	Bounds	$O(n)$
López-Ortiz, Quimper, Tromp, & van Beek	Bounds	$O(n)$

Hall's Marriage Theorem

$$\begin{aligned} \text{dom}(X_1) &= [3,4] \\ \text{dom}(X_2) &= [3,4] \\ \text{dom}(X_3) &= [1,3] \end{aligned}$$

Hall's Marriage Theorem

$$\begin{aligned} \mathrm{dom}(X_1) &= [3,4] \\ \mathrm{dom}(X_2) &= [3,4] \\ \mathrm{dom}(X_3) &= [1,3] \end{aligned}$$

- A Hall interval is an interval of k values that contains the domains of k variables.

Hall's Marriage Theorem

$$\begin{aligned} \text{dom}(X_1) &= [3,4] \\ \text{dom}(X_2) &= [3,4] \\ \text{dom}(X_3) &= [1,3] \end{aligned}$$

} Hall interval

- A Hall interval is an interval of k values that contains the domains of k variables.

Hall's Marriage Theorem

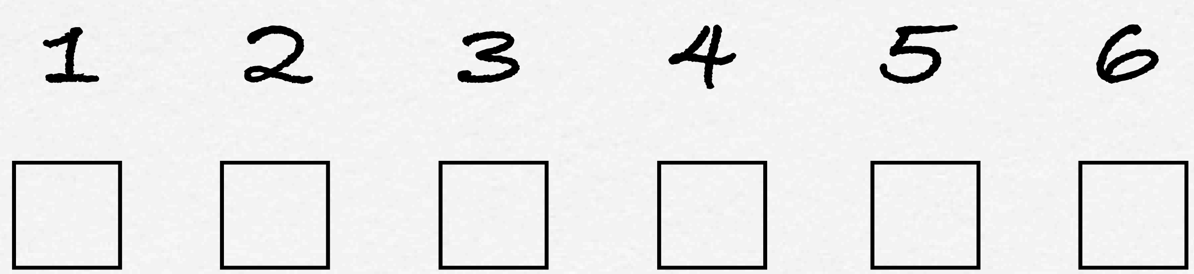
$$\begin{aligned} \text{dom}(X_1) &= [3,4] \\ \text{dom}(X_2) &= [3,4] \\ \text{dom}(X_3) &= [1,2] \end{aligned}$$

} Hall interval (annotating the first two lines)

- A Hall interval is an interval of k values that contains the domains of k variables.

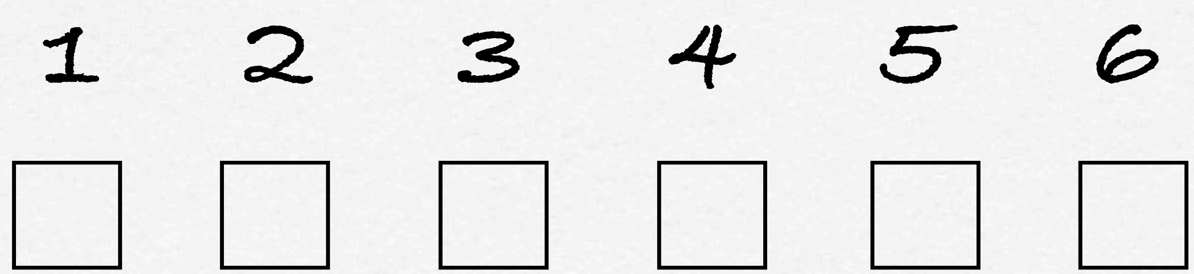
A Propagator for the Bounds Consistency

$$
\begin{aligned}
\mathrm{dom}(X_1) &= [2, 3] \\
\mathrm{dom}(X_2) &= [2, 3] \\
\mathrm{dom}(X_3) &= [3, 4] \\
\mathrm{dom}(X_4) &= [2, 6]
\end{aligned}
$$

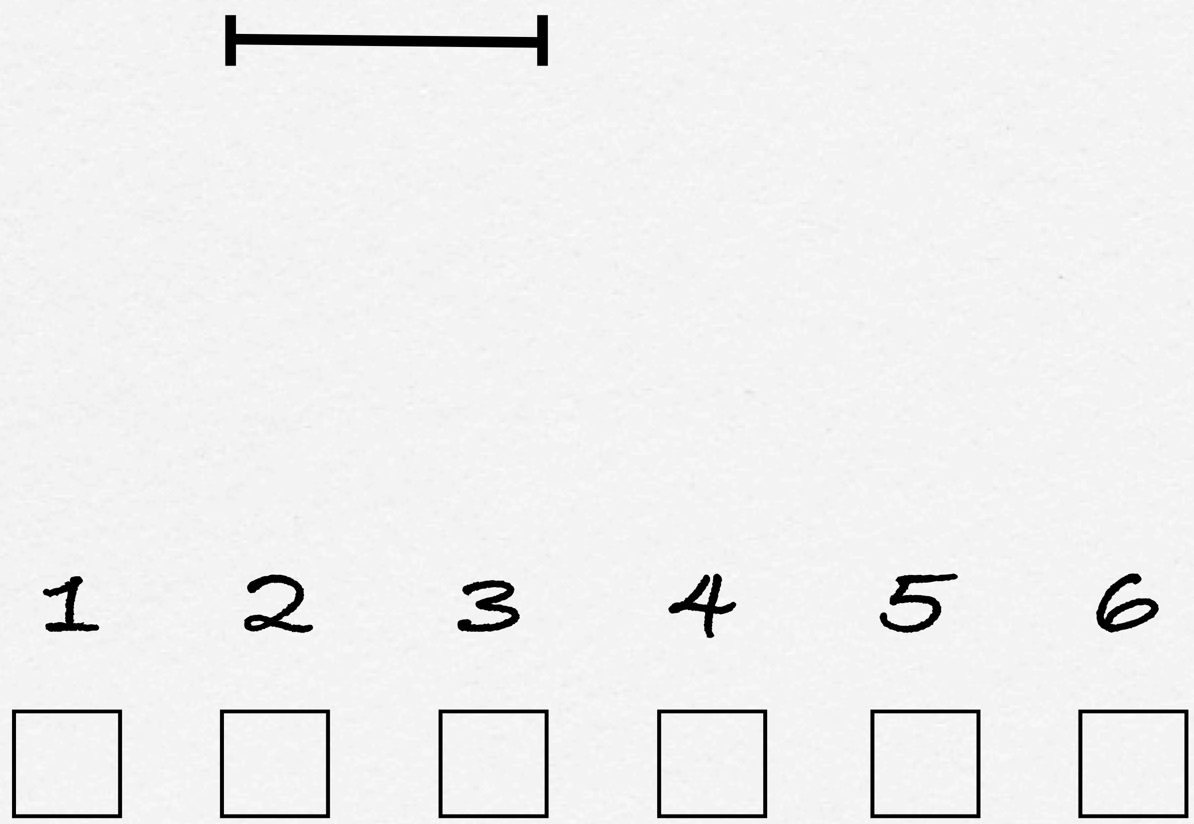


A Propagator for the Bounds Consistency

$$\begin{aligned} \text{dom}(X_1) &= [2, 3] \\ \text{dom}(X_2) &= [2, 3] \\ \text{dom}(X_3) &= [3, 4] \\ \text{dom}(X_4) &= [2, 6] \end{aligned}$$

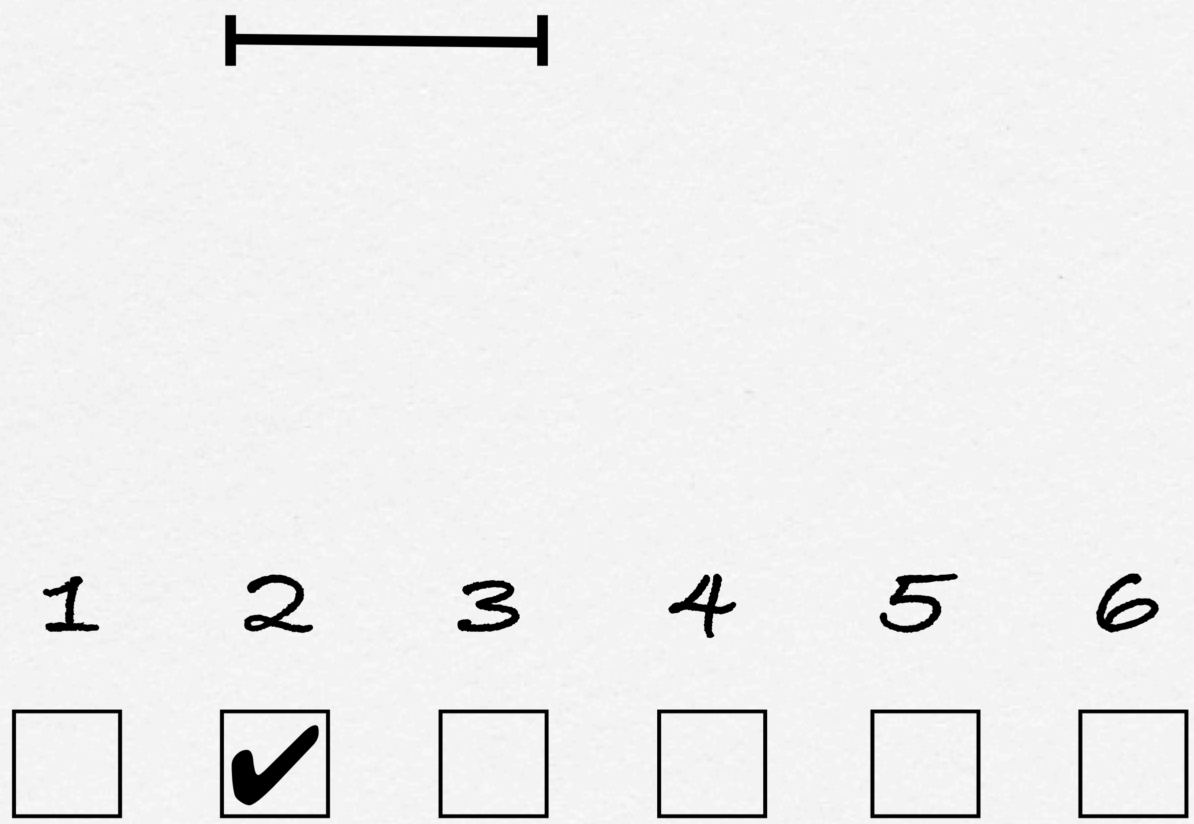


A Propagator for the Bounds Consistency



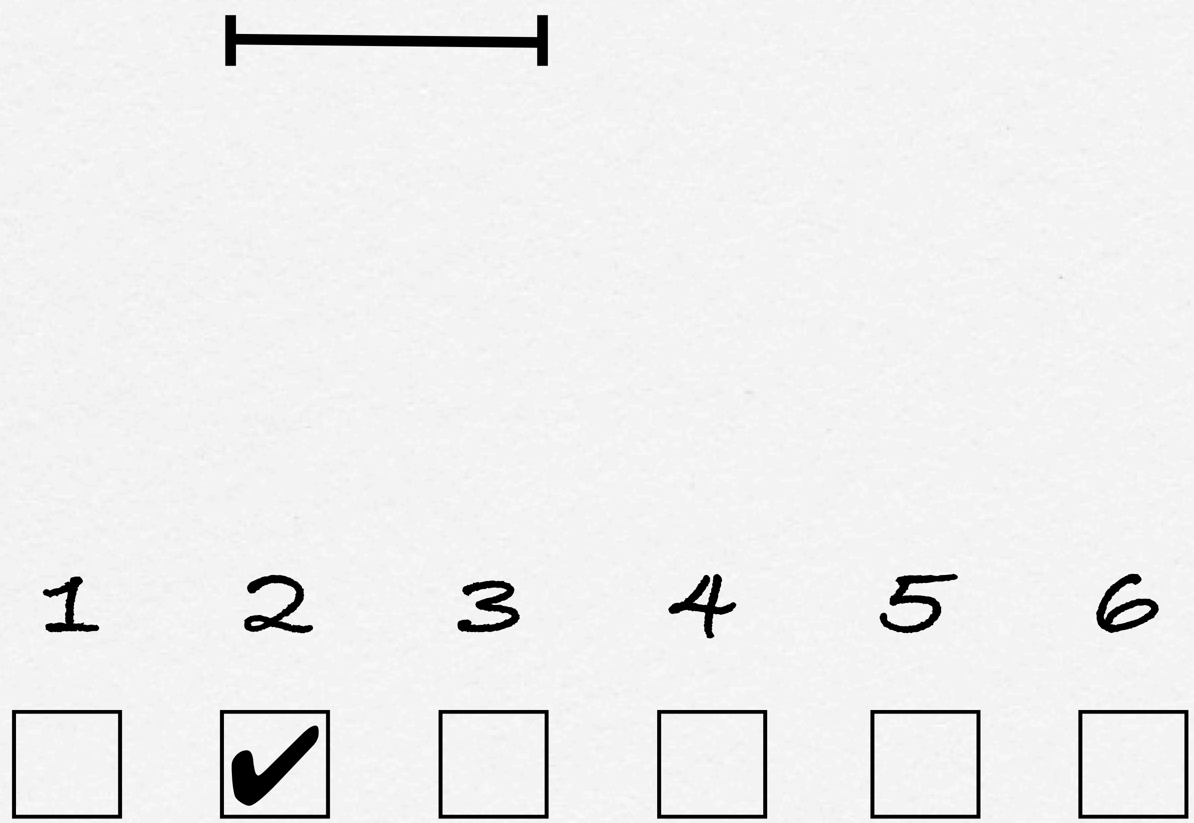
$$
\begin{aligned}
\text{dom}(X_1) &= [2, 3] \\
\text{dom}(X_2) &= [2, 3] \\
\text{dom}(X_3) &= [3, 4] \\
\text{dom}(X_4) &= [2, 6]
\end{aligned}
$$

A Propagator for the Bounds Consistency



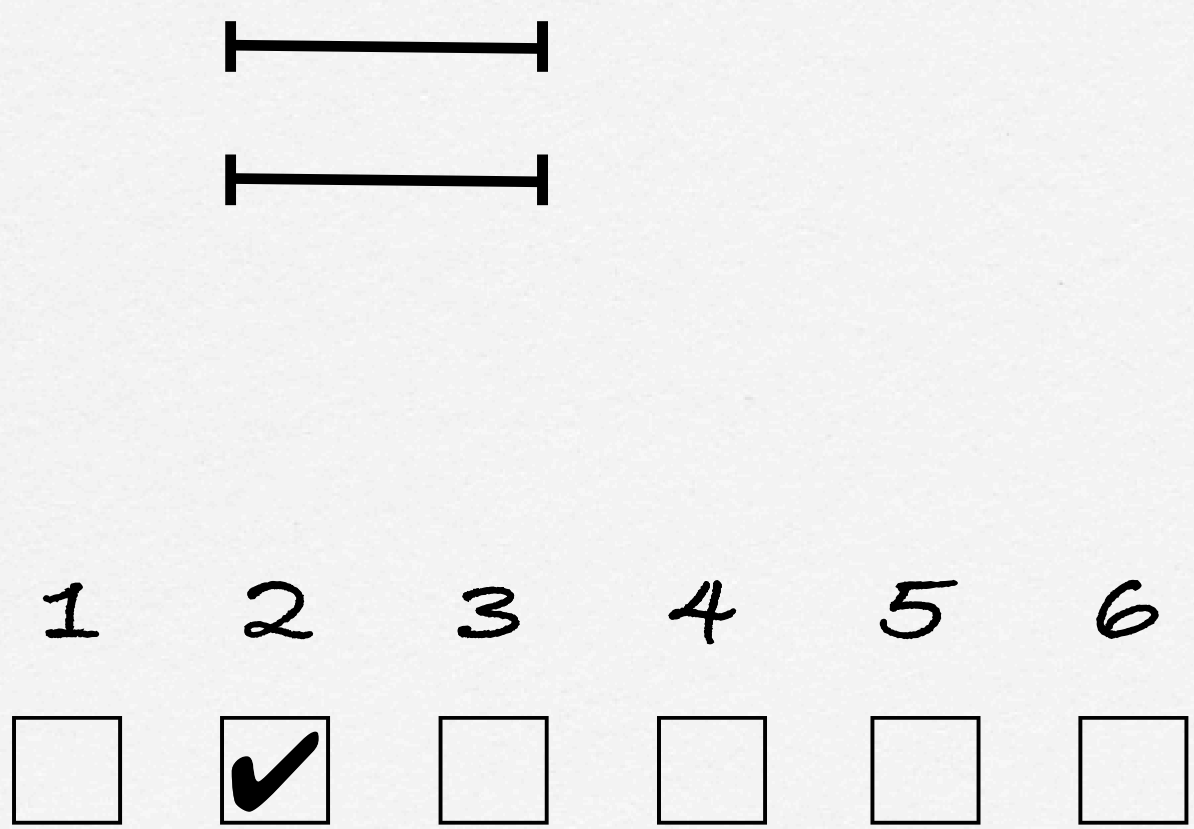
$$\begin{aligned}
\text{dom}(X_1) &= [2, 3] \\
\text{dom}(X_2) &= [2, 3] \\
\text{dom}(X_3) &= [3, 4] \\
\text{dom}(X_4) &= [2, 6]
\end{aligned}$$

A Propagator for the Bounds Consistency



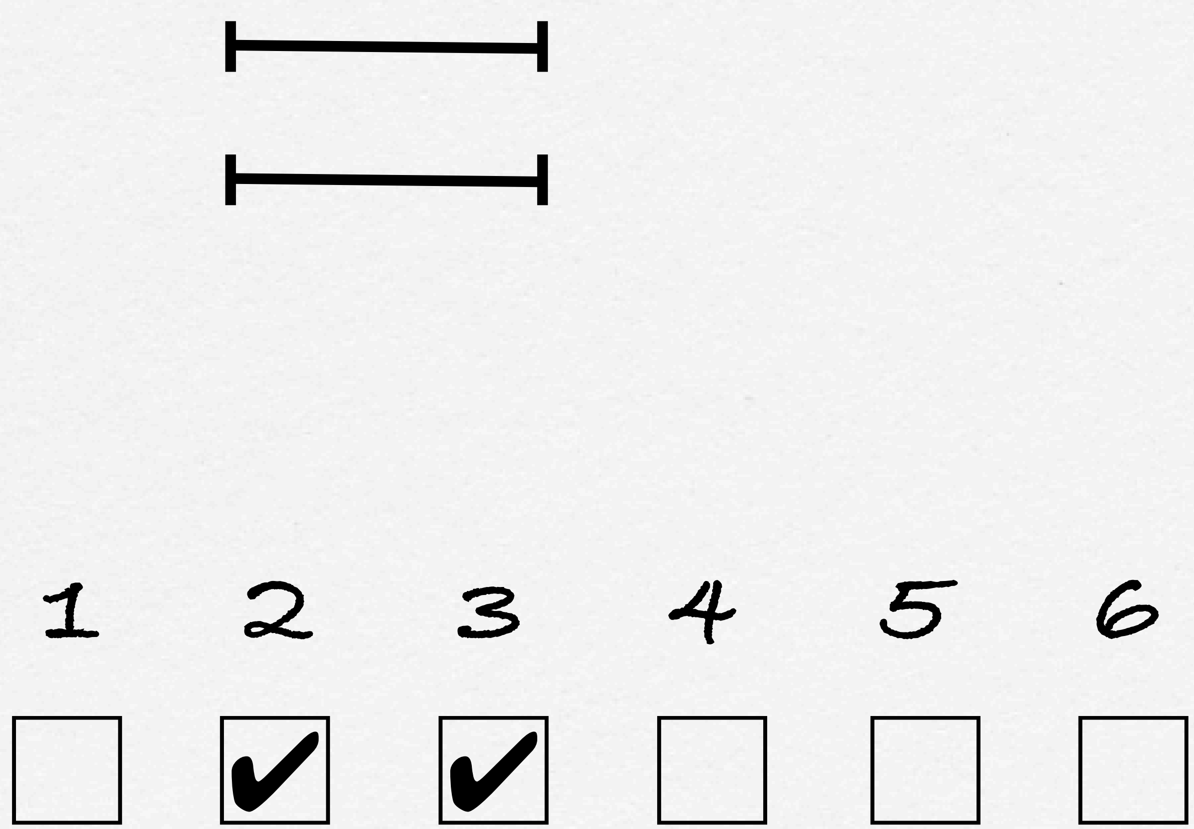
$$\begin{aligned} \text{dom}(X_1) &= [2, 3] \\ \text{dom}(X_2) &= [2, 3] \\ \text{dom}(X_3) &= [3, 4] \\ \text{dom}(X_4) &= [2, 6] \end{aligned}$$

A Propagator for the Bounds Consistency



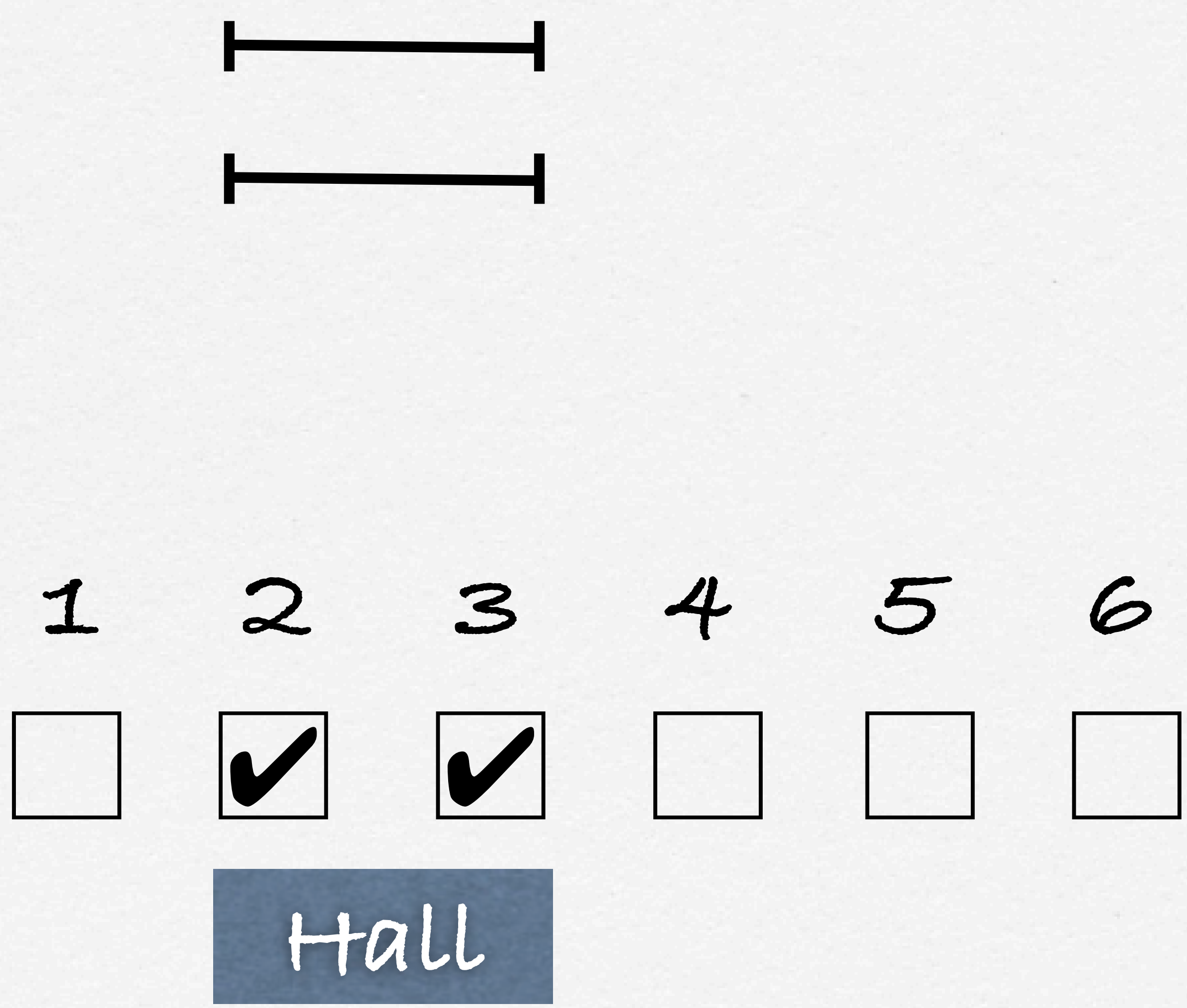
$$
\begin{aligned}
\text{dom}(X_1) &= [2,3] \\
\text{dom}(X_2) &= [2,3] \\
\text{dom}(X_3) &= [3,4] \\
\text{dom}(X_4) &= [2,6]
\end{aligned}
$$

A Propagator for the Bounds Consistency



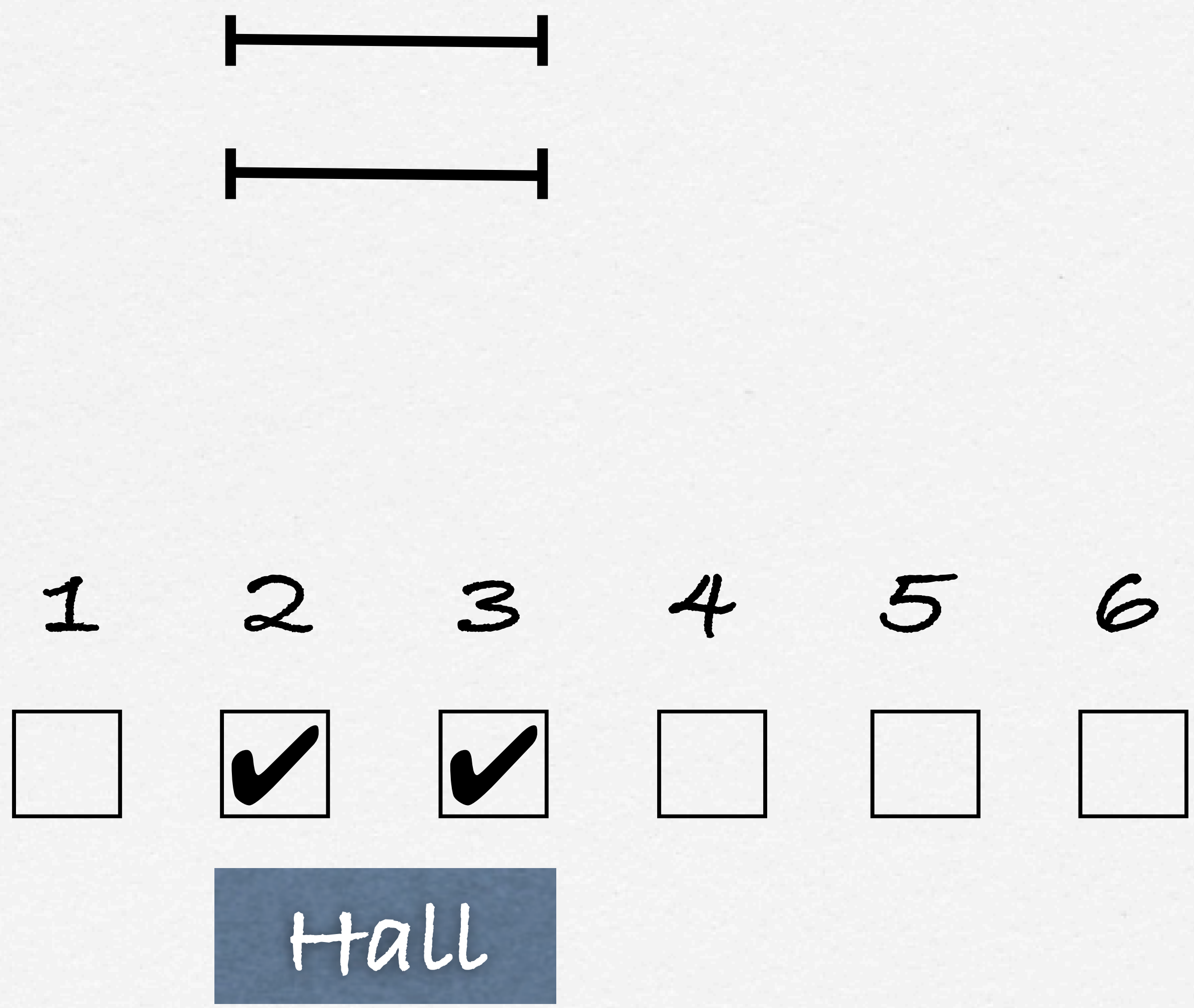
$$
\begin{aligned}
\mathrm{dom}(X_1) &= [2,3] \\
\mathrm{dom}(X_2) &= [2,3] \\
\mathrm{dom}(X_3) &= [3,4] \\
\mathrm{dom}(X_4) &= [2,6]
\end{aligned}
$$

A Propagator for the Bounds Consistency



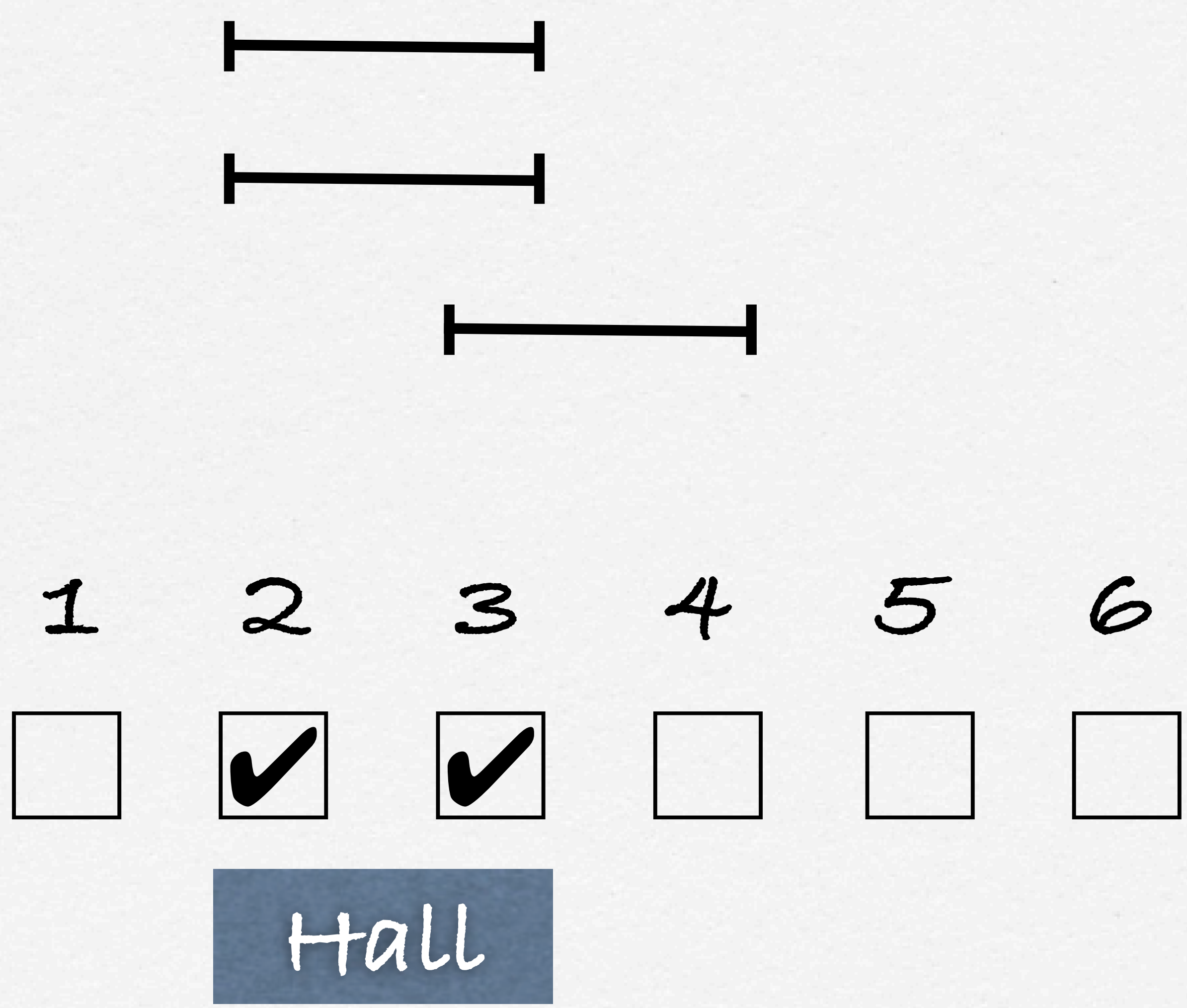
$$
\begin{aligned}
\text{dom}(X_1) &= [2,3] \\
\text{dom}(X_2) &= [2,3] \\
\text{dom}(X_3) &= [3,4] \\
\text{dom}(X_4) &= [2,6]
\end{aligned}
$$

A Propagator for the Bounds Consistency



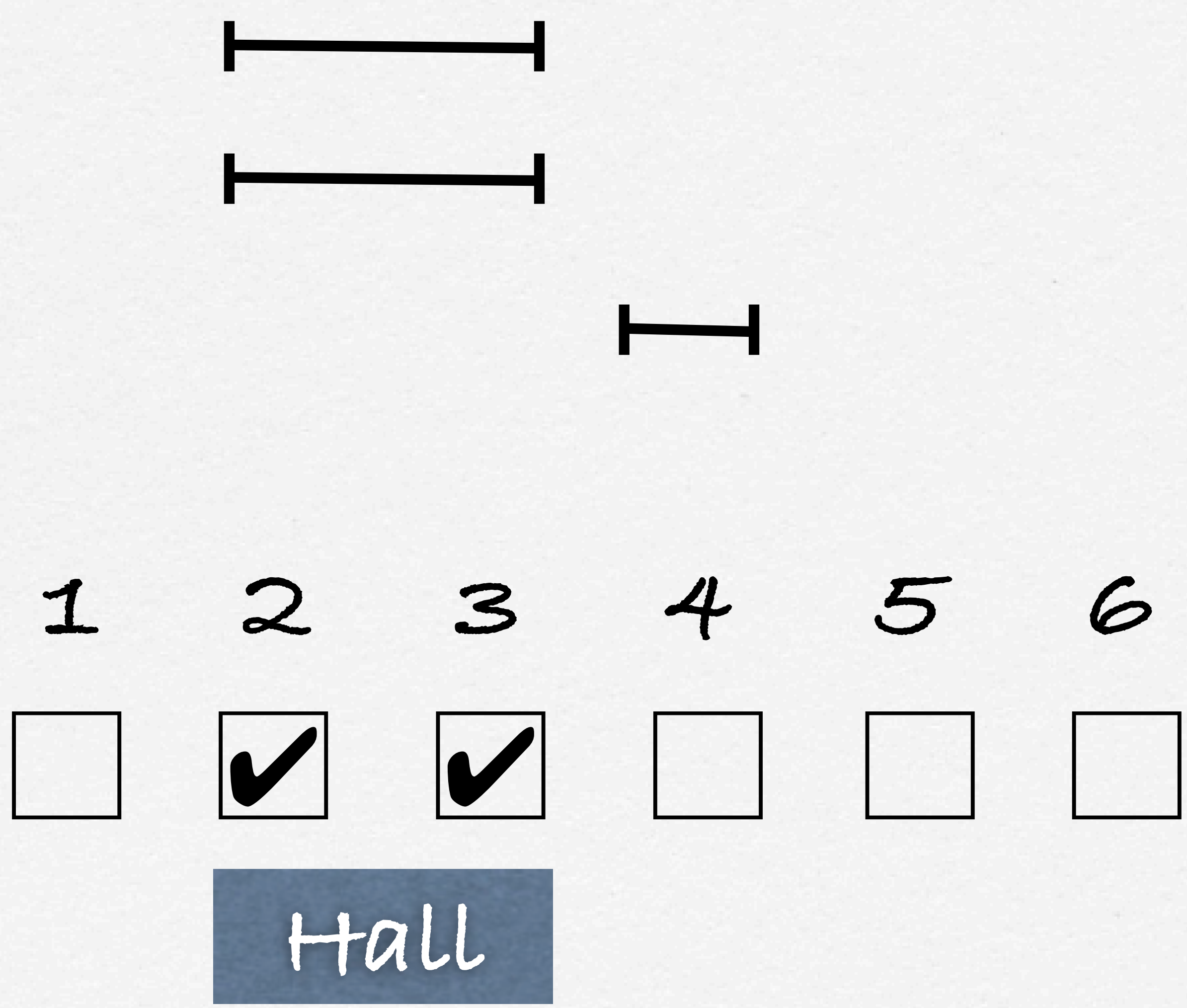
$$\begin{aligned}
\mathrm{dom}(X_1) &= [2,3] \\
\mathrm{dom}(X_2) &= [2,3] \\
\mathrm{dom}(X_3) &= [3,4] \\
\mathrm{dom}(X_4) &= [2,6]
\end{aligned}$$

A Propagator for the Bounds Consistency



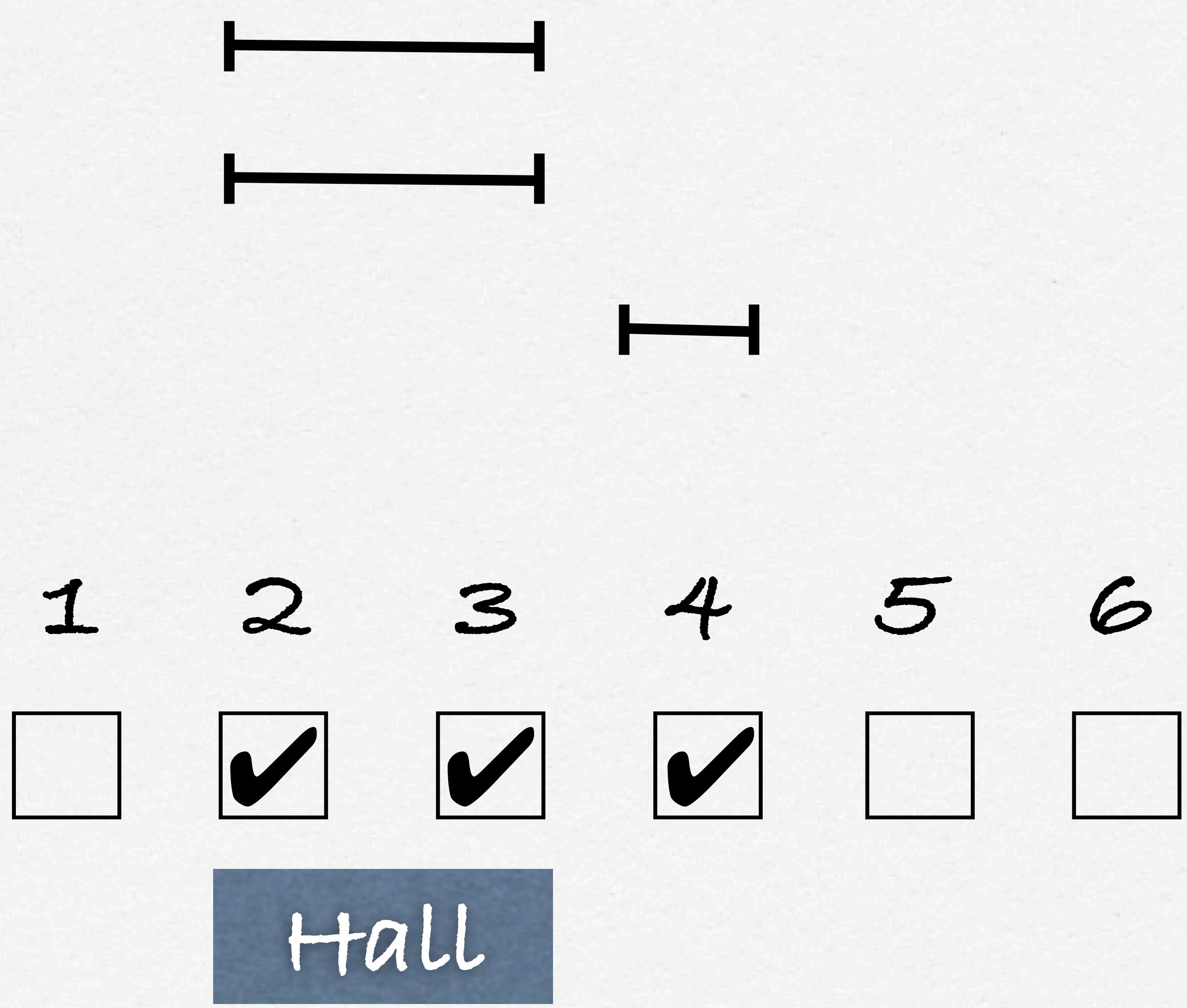
$$\begin{aligned}
\mathrm{dom}(X_1) &= [2,3] \\
\mathrm{dom}(X_2) &= [2,3] \\
\rightarrow \mathrm{dom}(X_3) &= [3,4] \\
\mathrm{dom}(X_4) &= [2,6]
\end{aligned}$$

A Propagator for the Bounds Consistency



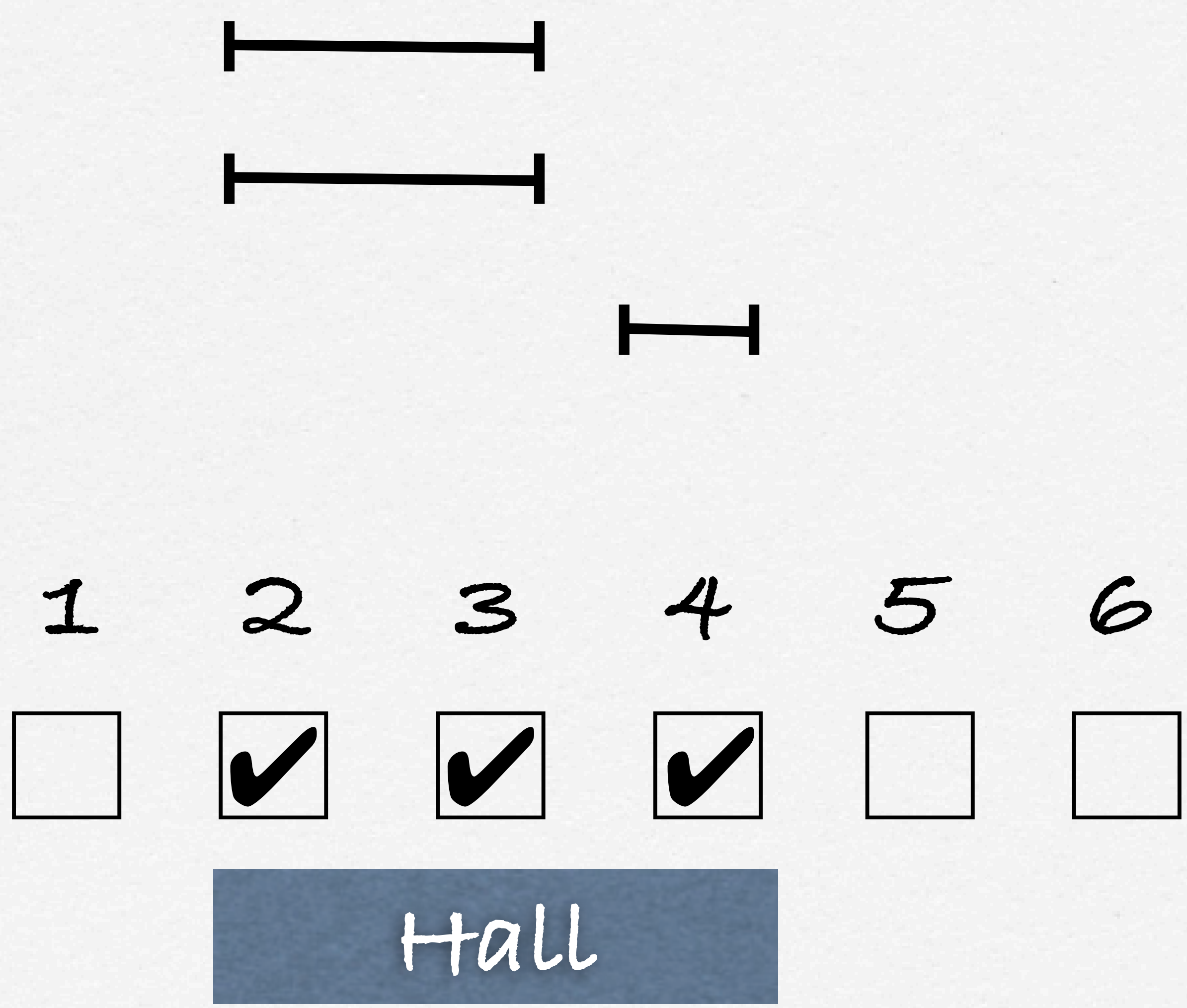
$$\begin{aligned} \text{dom}(X_1) &= [2,3] \\ \text{dom}(X_2) &= [2,3] \\ \text{dom}(X_3) &= [4,4] \\ \text{dom}(X_4) &= [2,6] \end{aligned}$$

A Propagator for the Bounds Consistency



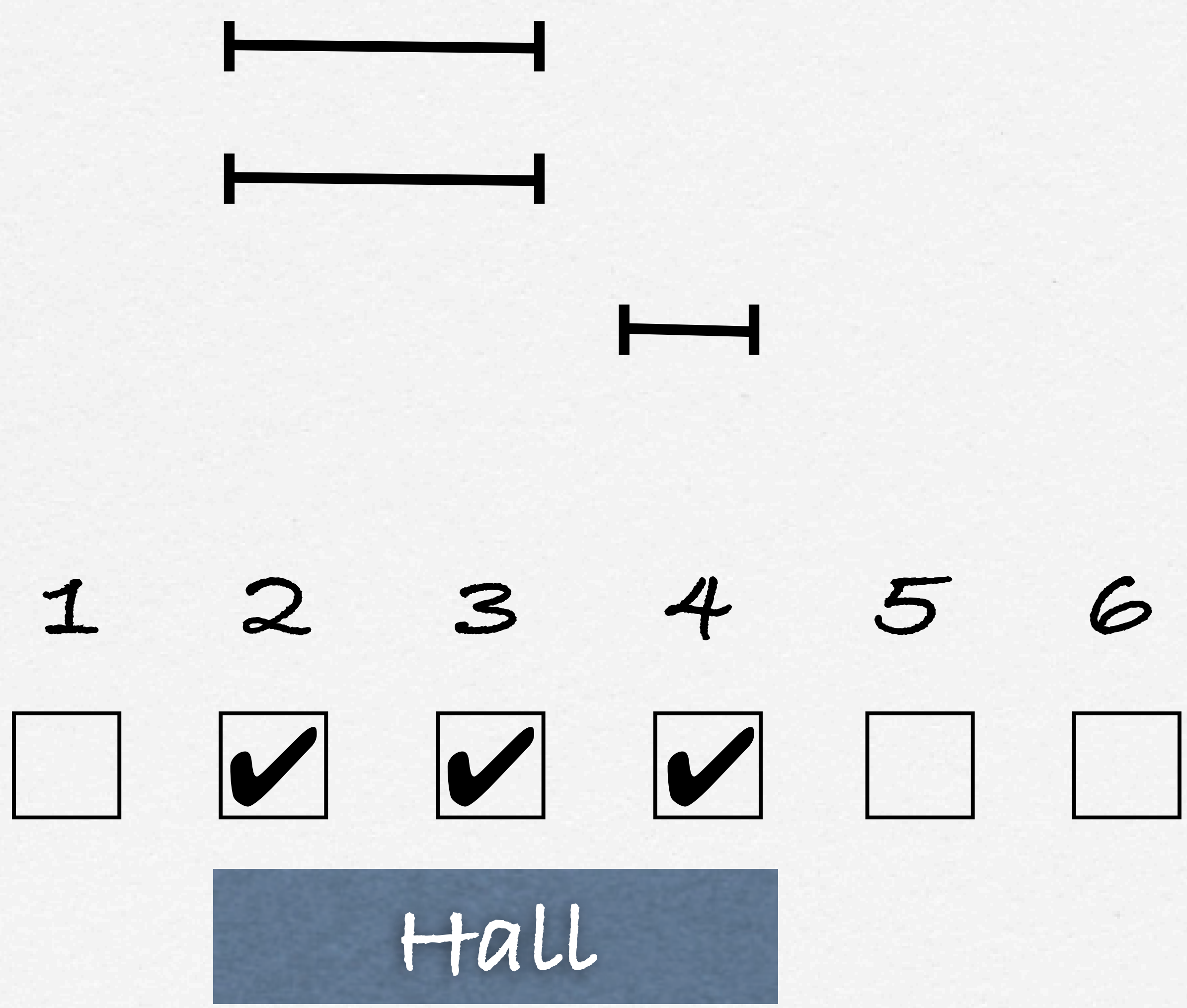
$$\begin{aligned} \text{dom}(X_1) &= [2,3] \\ \text{dom}(X_2) &= [2,3] \\ \text{dom}(X_3) &= [4,4] \\ \text{dom}(X_4) &= [2,6] \end{aligned}$$

A Propagator for the Bounds Consistency



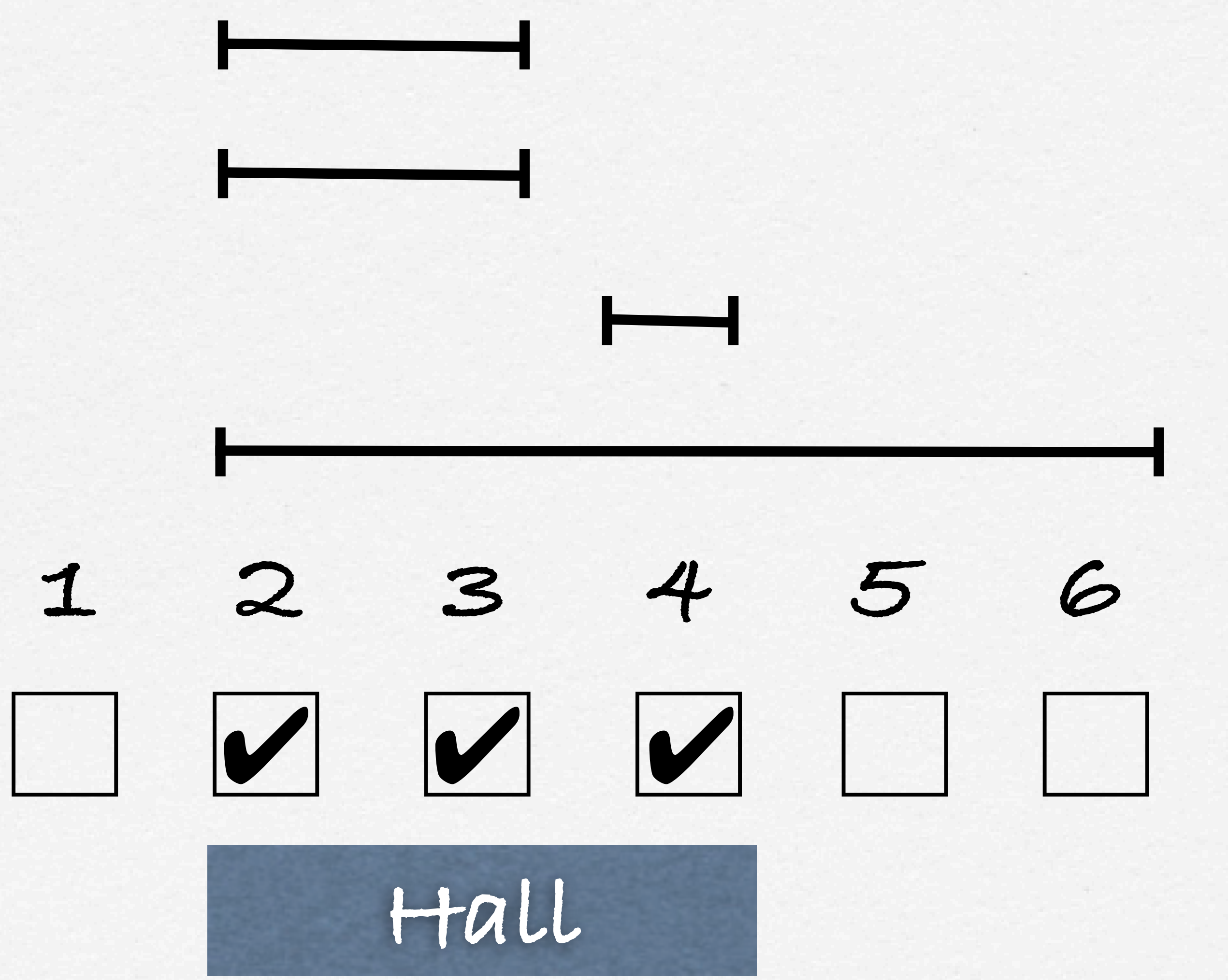
$$
\begin{aligned}
\mathrm{dom}(X_1) &= [2,3] \\
\mathrm{dom}(X_2) &= [2,3] \\
\mathrm{dom}(X_3) &= [4,4] \\
\mathrm{dom}(X_4) &= [2,6]
\end{aligned}
$$

A Propagator for the Bounds Consistency



$$\begin{aligned}
\mathrm{dom}(X_1) &= [2, 3]\\
\mathrm{dom}(X_2) &= [2, 3]\\
\mathrm{dom}(X_3) &= [4, 4]\\
\mathrm{dom}(X_4) &= [2, 6]
\end{aligned}$$

A Propagator for the Bounds Consistency



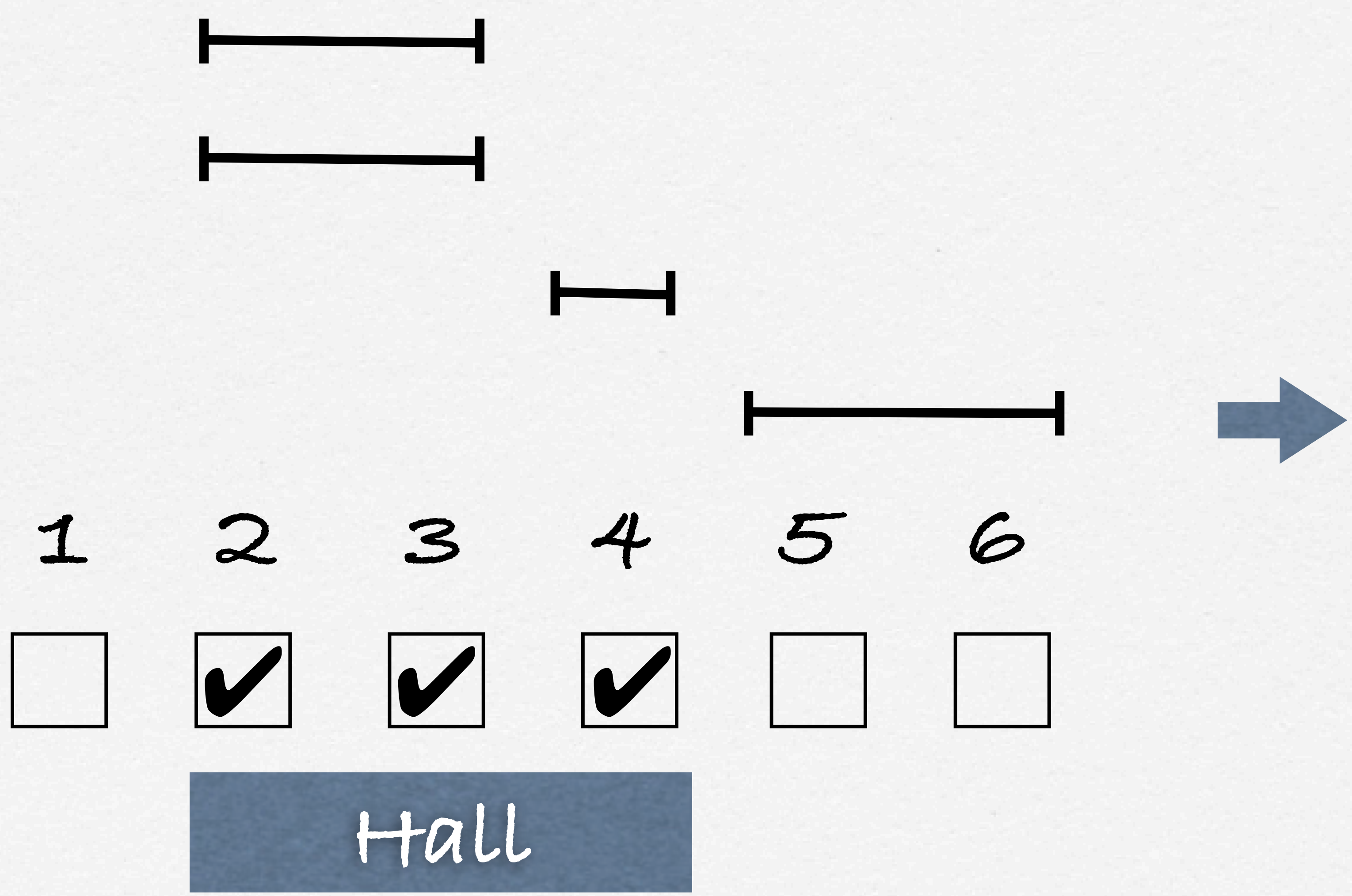
$$\mathrm{dom}(X_1) = [2, 3]$$

$$\mathrm{dom}(X_2) = [2, 3]$$

$$\mathrm{dom}(X_3) = [4, 4]$$

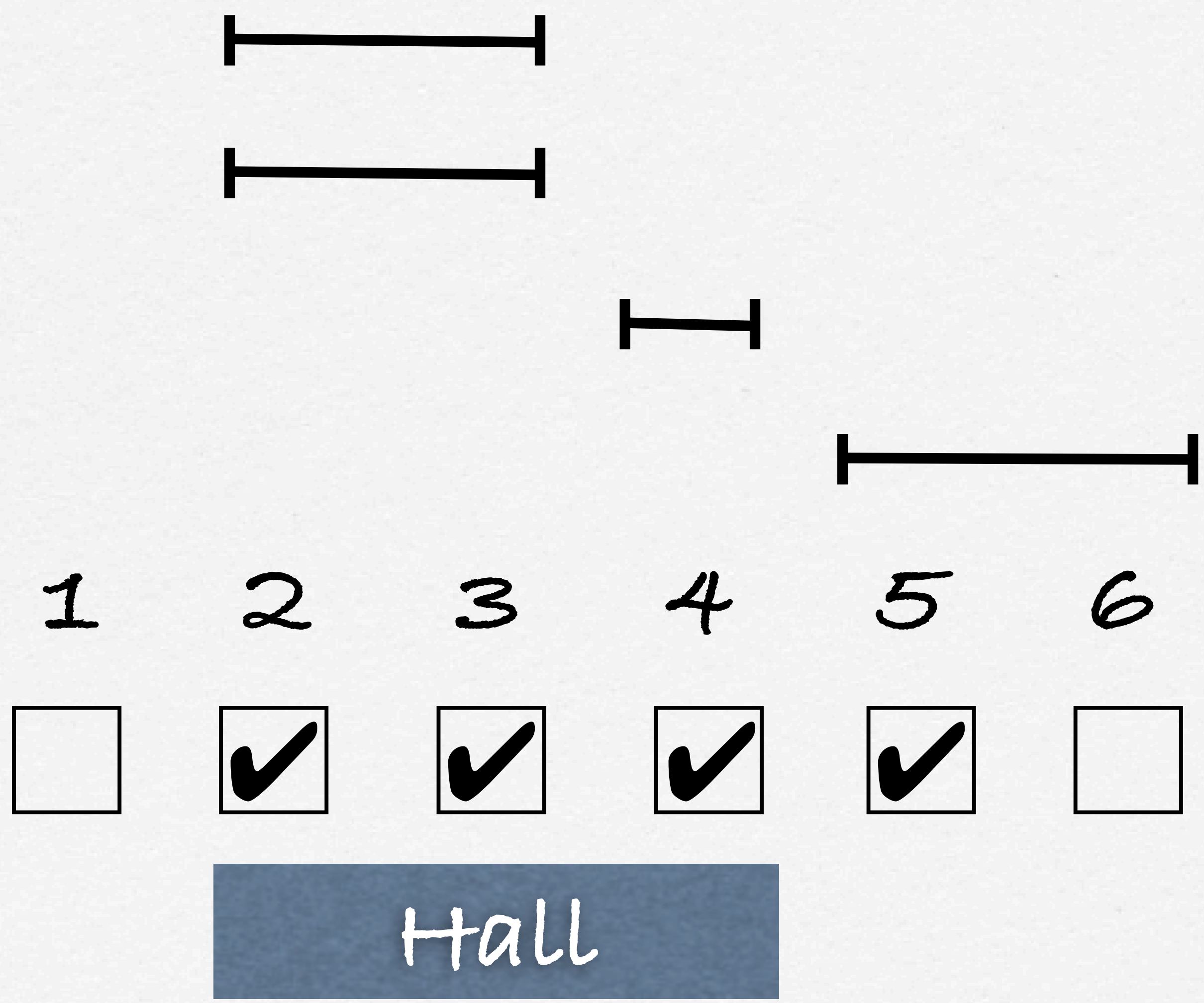
$$\mathrm{dom}(X_4) = [2, 6]$$

A Propagator for the Bounds Consistency



$$\begin{aligned}
\mathrm{dom}(X_1) &= [2, 3] \\
\mathrm{dom}(X_2) &= [2, 3] \\
\mathrm{dom}(X_3) &= [4, 4] \\
\mathrm{dom}(X_4) &= [5, 6]
\end{aligned}$$

A Propagator for the Bounds Consistency



$$\begin{aligned}
\mathrm{dom}(X_1) &= [2,3] \\
\mathrm{dom}(X_2) &= [2,3] \\
\mathrm{dom}(X_3) &= [4,4] \\
\mathrm{dom}(X_4) &= [5,6]
\end{aligned}$$

Analysis of the Algorithm

Version	Complexity	Note
First version	$O(n^2 + m)$	6 lines of C code!

Analysis of the Algorithm

Version	Complexity	Note
First version	$O(n^2 + m)$	6 lines of C code!
Union-find data structure	$O(n \log n)$	The fastest in practice

Analysis of the Algorithm

Version	Complexity	Note
First version	$O(n^2 + m)$	6 lines of C code!
Union-find data structure	$O(n \log n)$	The fastest in practice
Balanced union-find data structure	$O(n\,\alpha(n))$	Slightly slower than the previous version

Analysis of the Algorithm

Version	Complexity	Note
First version	$O(n^2 + m)$	6 lines of C code!
Union-find data structure	$O(n \log n)$	The fastest in practice
Balanced union-find data structure	$O(n\,\alpha(n))$	Slightly slower than the previous version
Gabow and Tarjan's data structure	$O(n)$	Slower and pages of code

The Global Cardinality Constraint

$$\mathrm{GCC}([X_1, \ldots, X_n], l, u) \iff \forall v \;\; l_v \leq |\{i \mid X_i = v\}| \leq u_v$$

- A value v must be taken at least l_v times and at most u_v times.

The Global Cardinality Constraint

$$\mathrm{GCC}([X_1,\ldots,X_n],l,u) \iff \forall v \;\; l_v \leq |\{i \mid X_i = v\}| \leq u_v$$

- A value v must be taken at least l_v times and at most u_v times.
- Scheduling: No more than 2 tasks can be executed at a given time.

The Global Cardinality Constraint

$$\mathrm{GCC}([X_1, \ldots, X_n], l, u) \iff \forall v \;\; l_v \leq |\{i \mid X_i = v\}| \leq u_v$$

- A value v must be taken at least l_v times and at most u_v times.
- Scheduling: No more than 2 tasks can be executed at a given time.
- Sequencing: We want to restrict the number of occurrences of an event in a sequence.

The Global Cardinality Constraint

$$\mathrm{GCC}([X_1, \ldots, X_n], l, u) \iff \forall v \;\; l_v \leq |\{i \mid X_i = v\}| \leq u_v$$

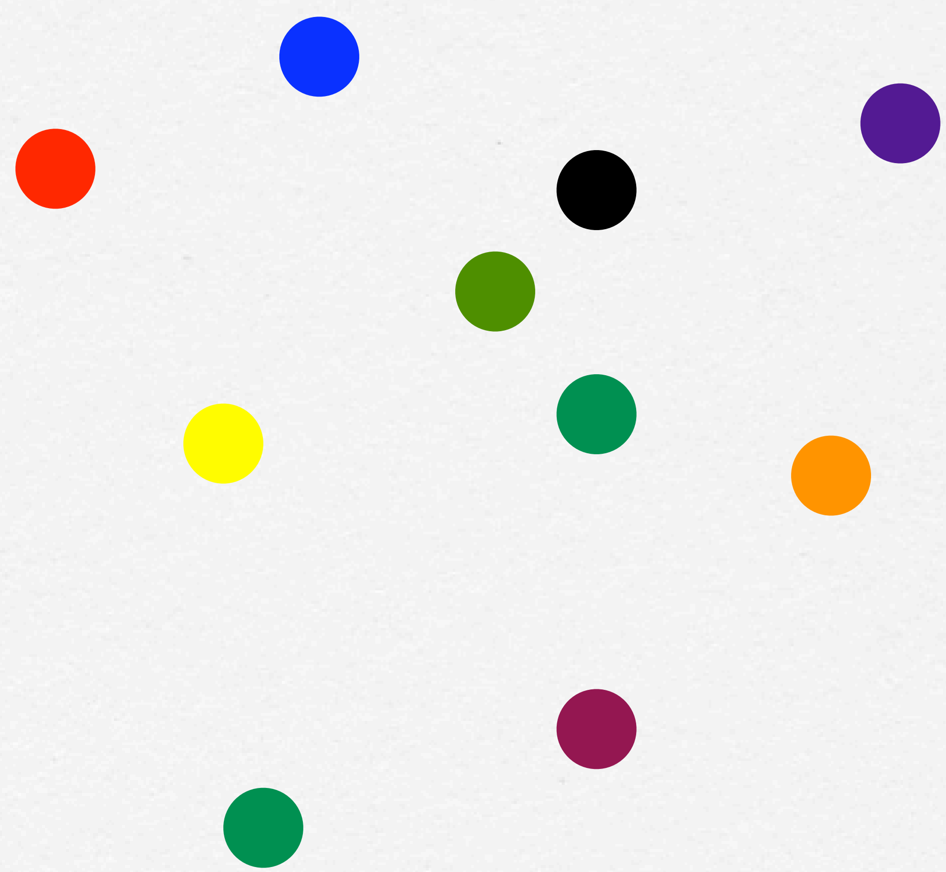
- [Régin '96] gives a propagator achieving domain consistency.

The Global Cardinality Constraint

$$\mathrm{GCC}([X_1, \ldots, X_n], l, u) \iff \forall v \;\; l_v \leq |\{i \mid X_i = v\}| \leq u_v$$

- [Régin '96] gives a propagator achieving domain consistency.
- There were no propagators for bounds consistency.

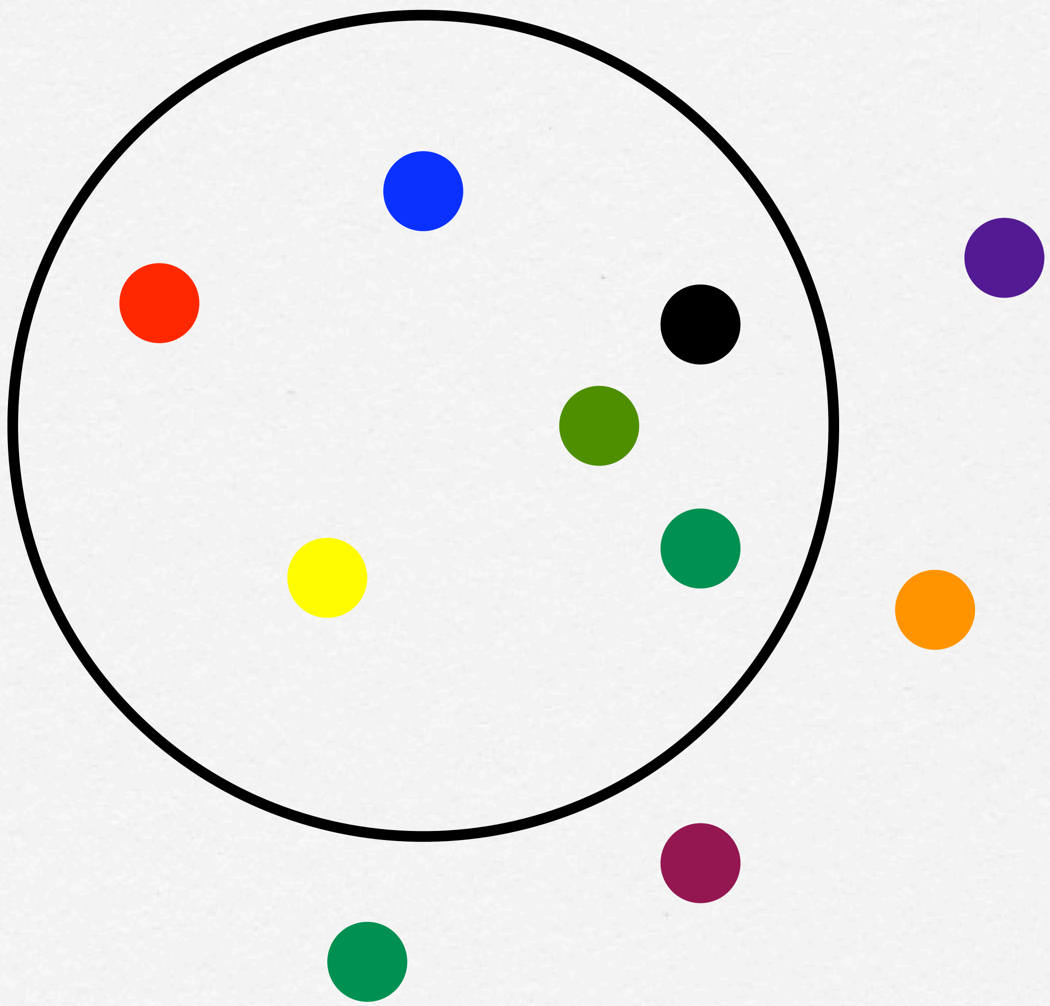
Decomposing the GCC



Decomposing the GCC

The upper bound constraint (ubc)

Each value is assigned to at most 2 variables.



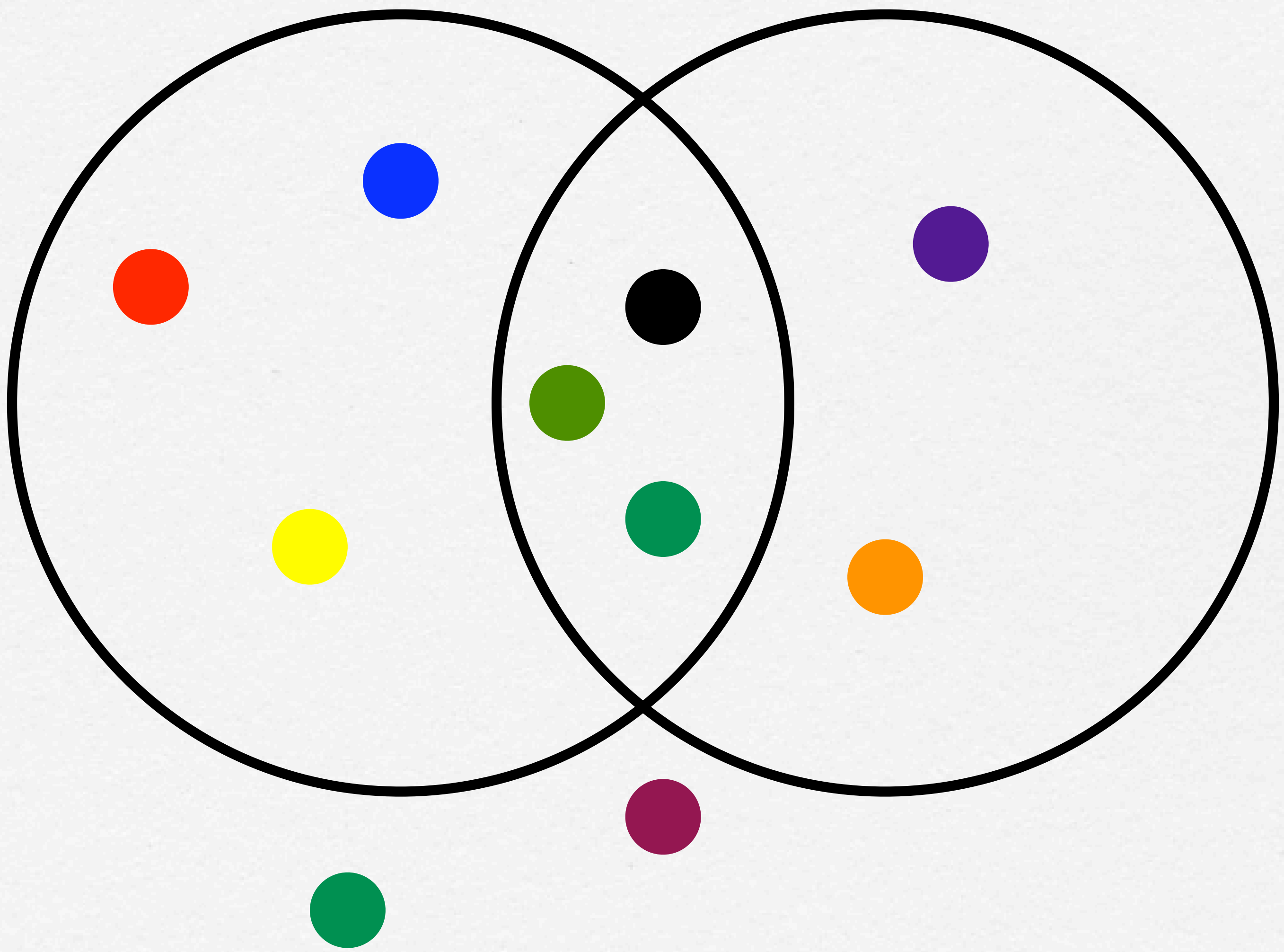
Decomposing the GCC

The upper bound constraint (ubc)

Each value is assigned to at most 2 variables.

The lower bound constraint (lbc)

Each value is assigned to at least 1 variable.



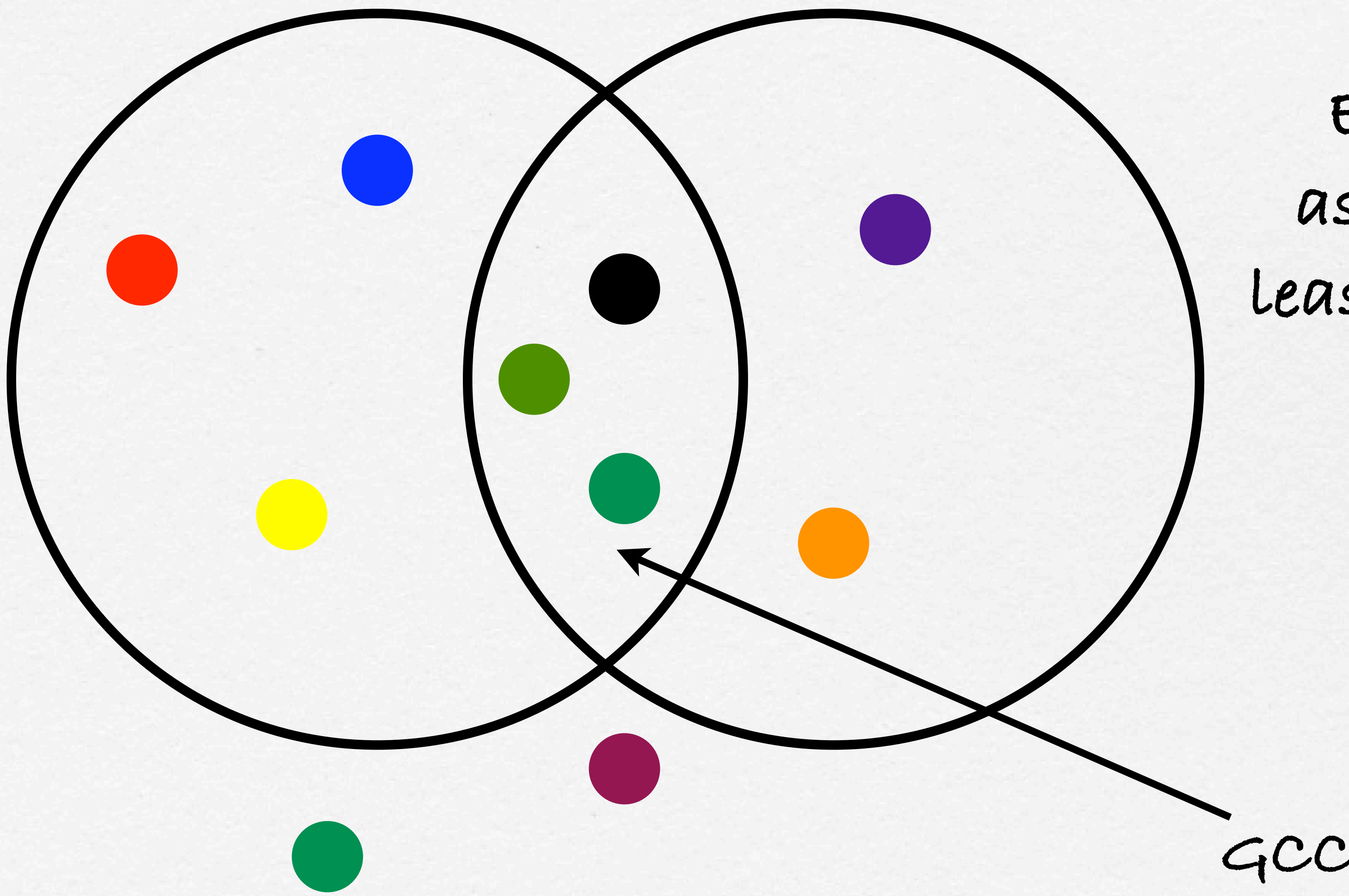
Decomposing the GCC

The upper bound constraint (ubc)

The lower bound constraint (lbc)

Each value is assigned to at most 2 variables.

Each value is assigned to at least 1 variable.



The Upper Bound Constraint

- All values must be assigned to at most 2 variables.

$$
\begin{aligned}
X_1 &: \{1 \quad 2 \quad \} \\
X_2 &: \{1 \quad \quad \} \\
X_3 &: \{1 \quad 2 \quad \} \\
X_4 &: \{ \quad 2 \quad \} \\
X_5 &: \{1 \quad 2 \quad 3\}
\end{aligned}
$$

The Upper Bound Constraint

- All values must be assigned to at most 2 variables.

$$
\begin{aligned}
X_1 &: \{1 \quad 2 \quad \} \\
X_2 &: \{1 \quad \quad \} \\
X_3 &: \{1 \quad 2 \quad \} \\
X_4 &: \{ \quad 2 \quad \} \\
X_5 &: \{\underbrace{1 \quad 2}_{} \quad 3\}
\end{aligned}
$$

$$S = \{1, 2\}$$

The Upper Bound Constraint

- All values must be assigned to at most 2 variables.

$$
\begin{aligned}
X_1 &: \{1 \quad 2 \quad\quad \} \\
X_2 &: \{1 \quad\quad\quad\quad \} \\
X_3 &: \{1 \quad 2 \quad\quad \} \\
X_4 &: \{\quad\; 2 \quad\quad \} \\
X_5 &: \{1 \quad 2 \quad 3\}
\end{aligned}
$$

$$S = \underbrace{\{1, 2\}}$$

Upper capacity: $\lceil S \rceil = 2 + 2 = 4$

The Upper Bound Constraint

- All values must be assigned to at most 2 variables.

$$
\begin{aligned}
X_1 &: \{1 \quad 2 \quad\quad \} \subseteq S \\
X_2 &: \{1 \quad\quad\quad\quad \} \subseteq S \\
X_3 &: \{1 \quad 2 \quad\quad \} \subseteq S \\
X_4 &: \{\quad\quad 2 \quad\quad \} \subseteq S \\
X_5 &: \{1 \quad 2 \quad 3\}
\end{aligned}
$$

$$S = \{1, 2\}$$

Upper capacity: $\lceil S \rceil = 2 + 2 = 4$

The Upper Bound Constraint

- All values must be assigned to at most 2 variables.

$$
\begin{array}{l}
X_1 : \{1 \quad 2 \quad\quad\} \subseteq S \\
X_2 : \{1 \quad\quad\quad\quad\} \subseteq S \\
X_3 : \{1 \quad 2 \quad\quad\} \subseteq S \\
X_4 : \{\quad\quad 2 \quad\quad\} \subseteq S \\
X_5 : \{\quad\quad\quad\quad 3\}
\end{array}
$$

$$S = \{1, 2\}$$

Upper capacity: $\lceil S \rceil = 2 + 2 = 4$

The Upper Bound Constraint

- All values must be assigned to at most 2 variables.

$$
\begin{aligned}
X_1 &: \{1 \quad 2 \quad\quad \} \subseteq S \\
X_2 &: \{1 \quad\quad\quad\quad \} \subseteq S \\
X_3 &: \{1 \quad 2 \quad\quad \} \subseteq S \\
X_4 &: \{\quad\quad 2 \quad\quad \} \subseteq S \\
X_5 &: \{\quad\quad\quad\quad 3\}
\end{aligned}
$$

$$S = \{1, 2\}$$

Upper capacity: $\lceil S \rceil = 2 + 2 = 4$

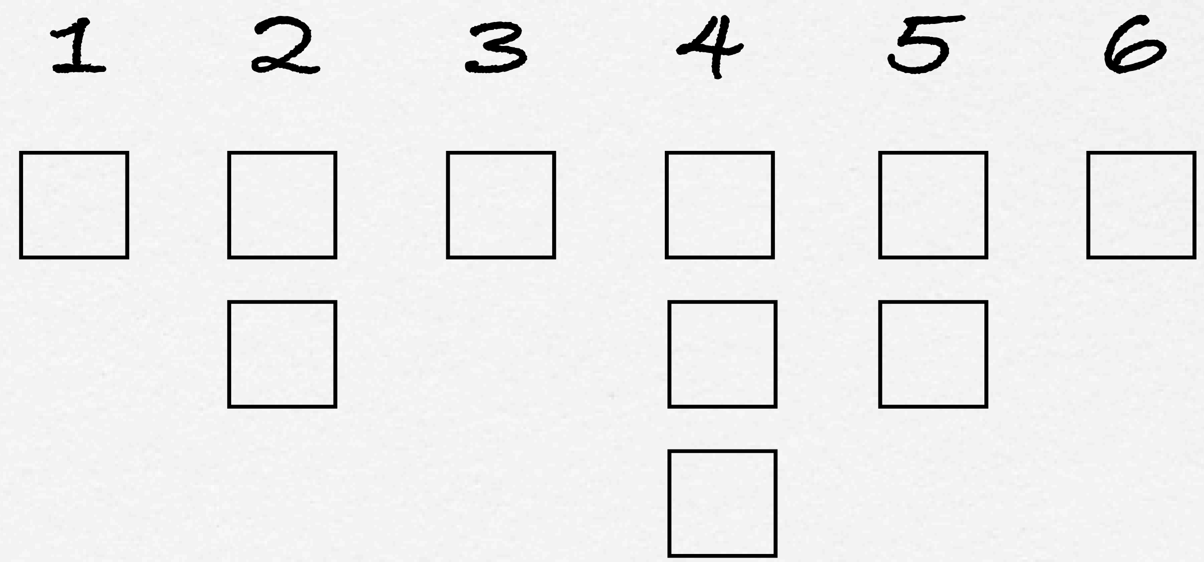
Hall Interval

An interval containing as many domains as its upper capacity.

$$\lceil S \rceil = |\{i \mid \mathrm{dom}(X_i) \subseteq S\}|$$

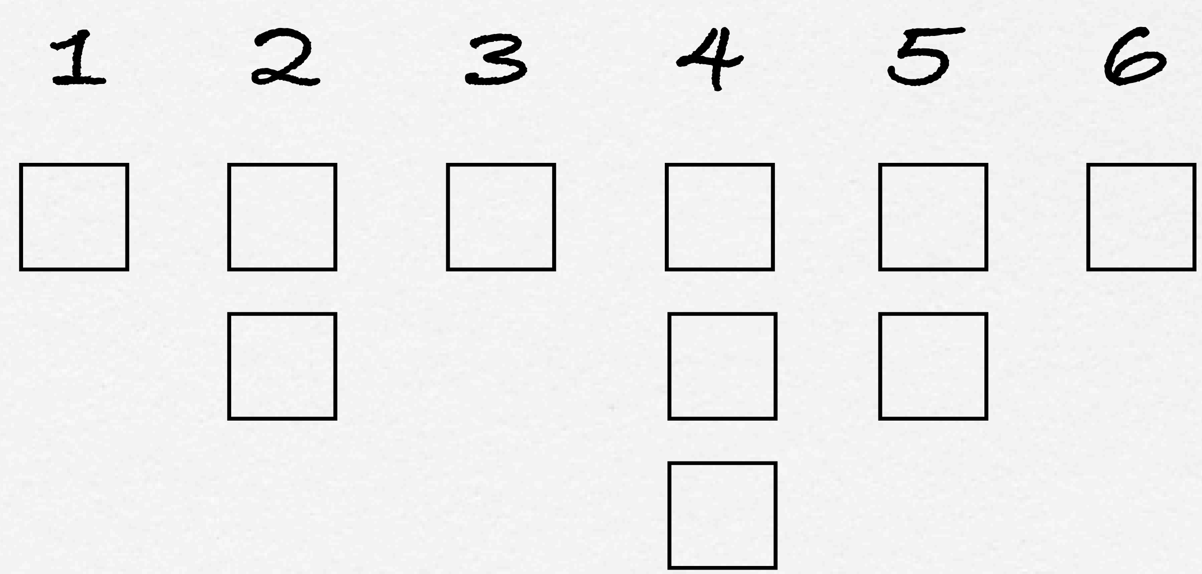
A Propagator for the UBC

- Similar to the one for the all-different Constraint.



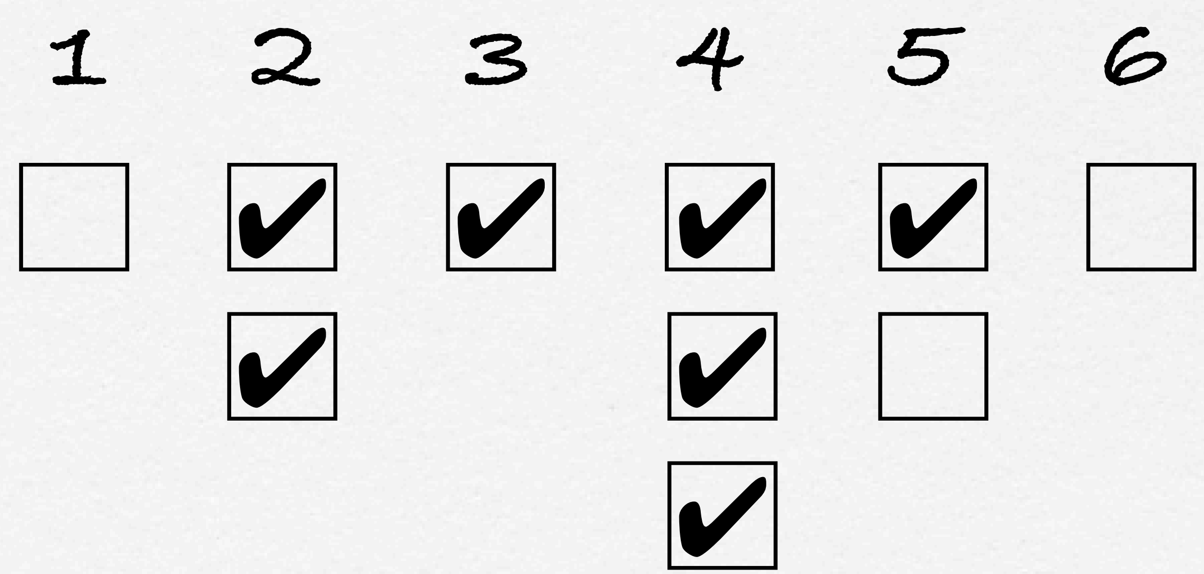
A Propagator for the UBC

- Similar to the one for the all-different constraint.
- Values can have more than one bucket.



A Propagator for the UBC

- Similar to the one for the all-different Constraint.
- Values can have more than one bucket.



The Lower Bound Constraint

- All values must be assigned to at least 1 variable.

$$
\begin{array}{llllll}
X_1 : & \{1 & & & & \} \\
X_2 : & \{ & & & 4 & \} \\
X_3 : & \{1 & & & 4 & \} \\
X_4 : & \{1 & 2 & 3 & & \} \\
X_5 : & \{ & 2 & 3 & 4 & \}
\end{array}
$$

The Lower Bound Constraint

- All values must be assigned to at least 1 variable.

$$
\begin{array}{llcccl}
X_1 : & \{1 & & & & \} \\
X_2 : & \{ & & & 4 & \} \\
X_3 : & \{1 & & & 4 & \} \\
X_4 : & \{1 & 2 & 3 & & \} \\
X_5 : & \{ & 2 & 3 & 4 & \}
\end{array}
$$

$$S = \{2, 3\}$$

The Lower Bound Constraint

- All values must be assigned to at least 1 variable.

$$
\begin{array}{lllll}
X_1 : \{1 & & & \} \\
X_2 : \{ & & & 4\} \\
X_3 : \{1 & & & 4\} \\
X_4 : \{1 & 2 & 3 & \} \\
X_5 : \{ & 2 & 3 & 4\}
\end{array}
$$

$$S = \{2, 3\}$$

Lower capacity: $\lfloor S \rfloor = 1 + 1 = 2$

The Lower Bound Constraint

- All values must be assigned to at least 1 variable.

$$
\begin{array}{ll}
X_1: & \{1 \quad\quad\quad\quad\quad\quad \} \\
X_2: & \{\quad\quad\quad\quad\quad\quad 4\} \\
X_3: & \{1 \quad\quad\quad\quad\quad 4\} \\
X_4: & \{1 \quad 2 \quad 3 \quad\quad \} \cap S \neq \emptyset \\
X_5: & \{\quad\quad 2 \quad 3 \quad 4\} \cap S \neq \emptyset
\end{array}
$$

$$S = \{2, 3\}$$

Lower capacity: $\lfloor S \rfloor = 1 + 1 = 2$

The Lower Bound Constraint

- All values must be assigned to at least 1 variable.

$$
\begin{array}{llllll}
X_1 : & \{1 & & & & \} \\
X_2 : & \{ & & & 4 & \} \\
X_3 : & \{1 & & & 4 & \} \\
X_4 : & \{ & 2 & 3 & & \} \cap S \neq \emptyset \\
X_5 : & \{ & 2 & 3 & & \} \cap S \neq \emptyset
\end{array}
$$

$$S = \{2, 3\}$$

Lower capacity: $\lfloor S \rfloor = 1 + 1 = 2$

The Lower Bound Constraint

- All values must be assigned to at least 1 variable.

$$\begin{aligned}
X_1 &: \{1 \quad\quad\quad\quad\quad\quad\quad\quad \} \\
X_2 &: \{ \quad\quad\quad\quad\quad\quad\quad\quad 4\} \\
X_3 &: \{1 \quad\quad\quad\quad\quad\quad\quad 4\} \\
X_4 &: \{ \quad\quad 2 \quad 3 \quad\quad \} \cap S \neq \emptyset \\
X_5 &: \{ \quad\quad 2 \quad 3 \quad\quad \} \cap S \neq \emptyset
\end{aligned}$$

$$S = \{2, 3\}$$

Lower capacity: $\lfloor S \rfloor = 1 + 1 = 2$

Unstable Set

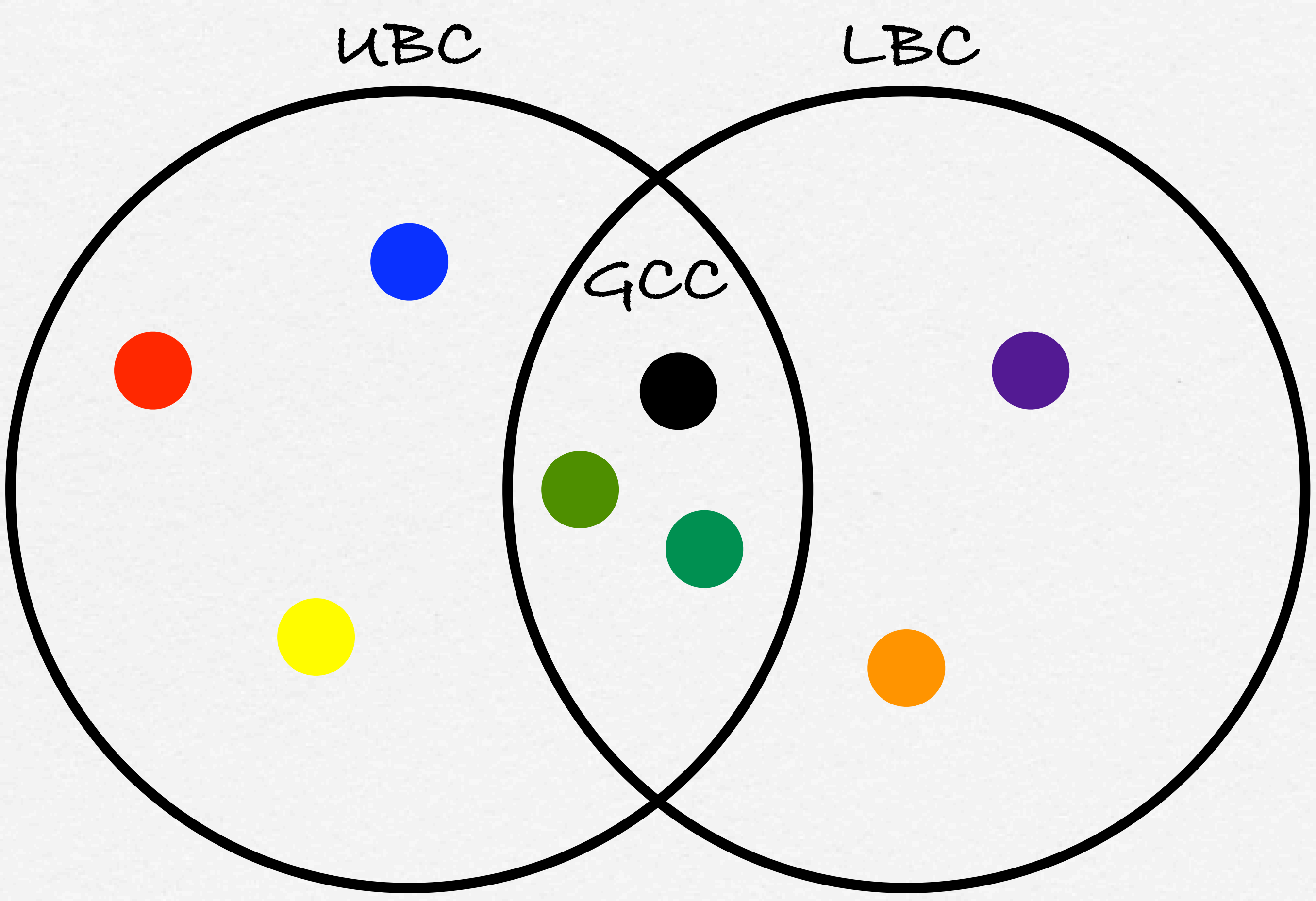
A set intersecting as many domains as its lower capacity.

$$\lfloor S \rfloor = |\{i \mid \mathrm{dom}(X_i) \cap S \neq \emptyset\}|$$

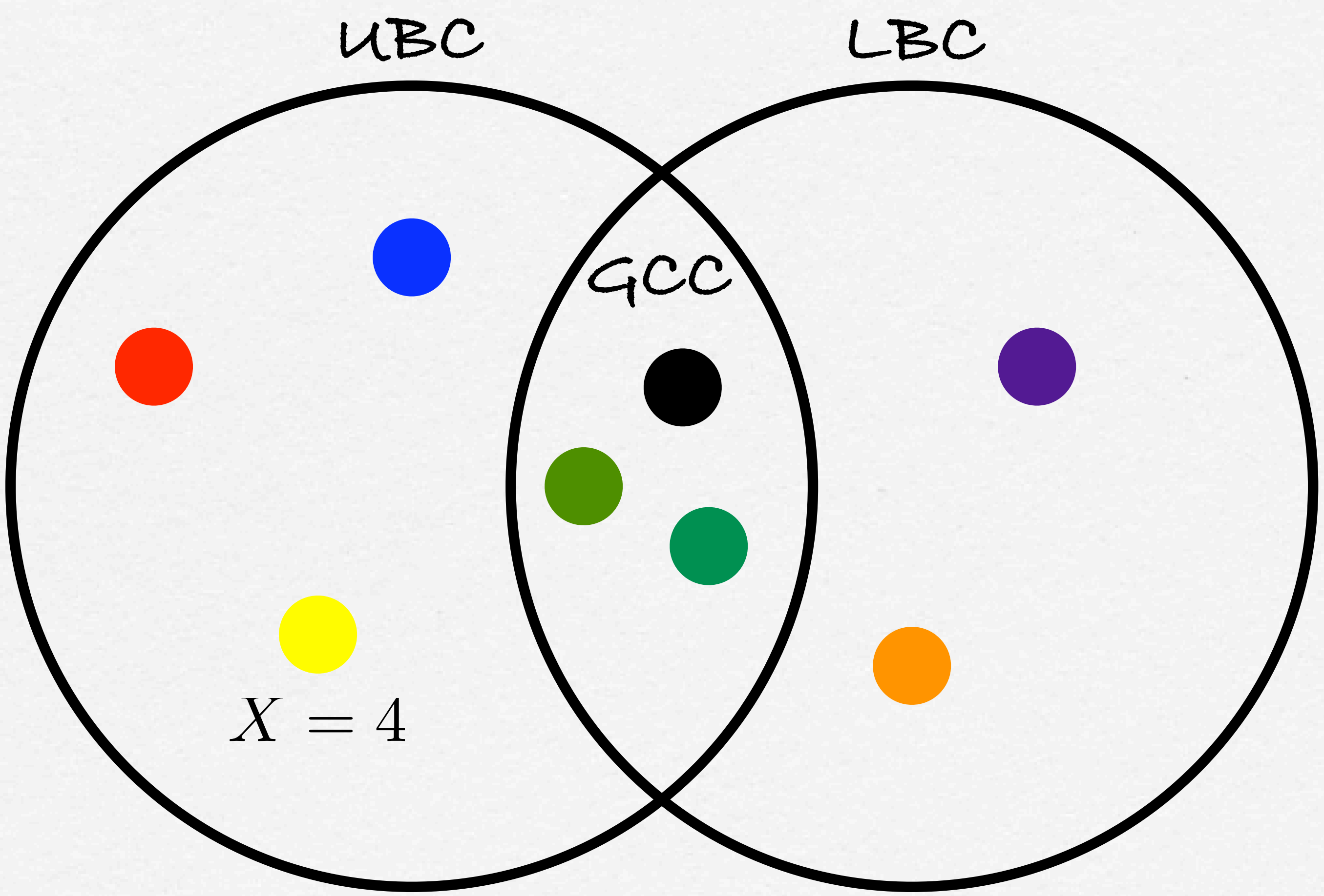
A Propagator for the LBC

- We adapted the algorithm for the All-different constraint
  - Detects unstable sets rather than Hall intervals.
- Time complexity: O(n)

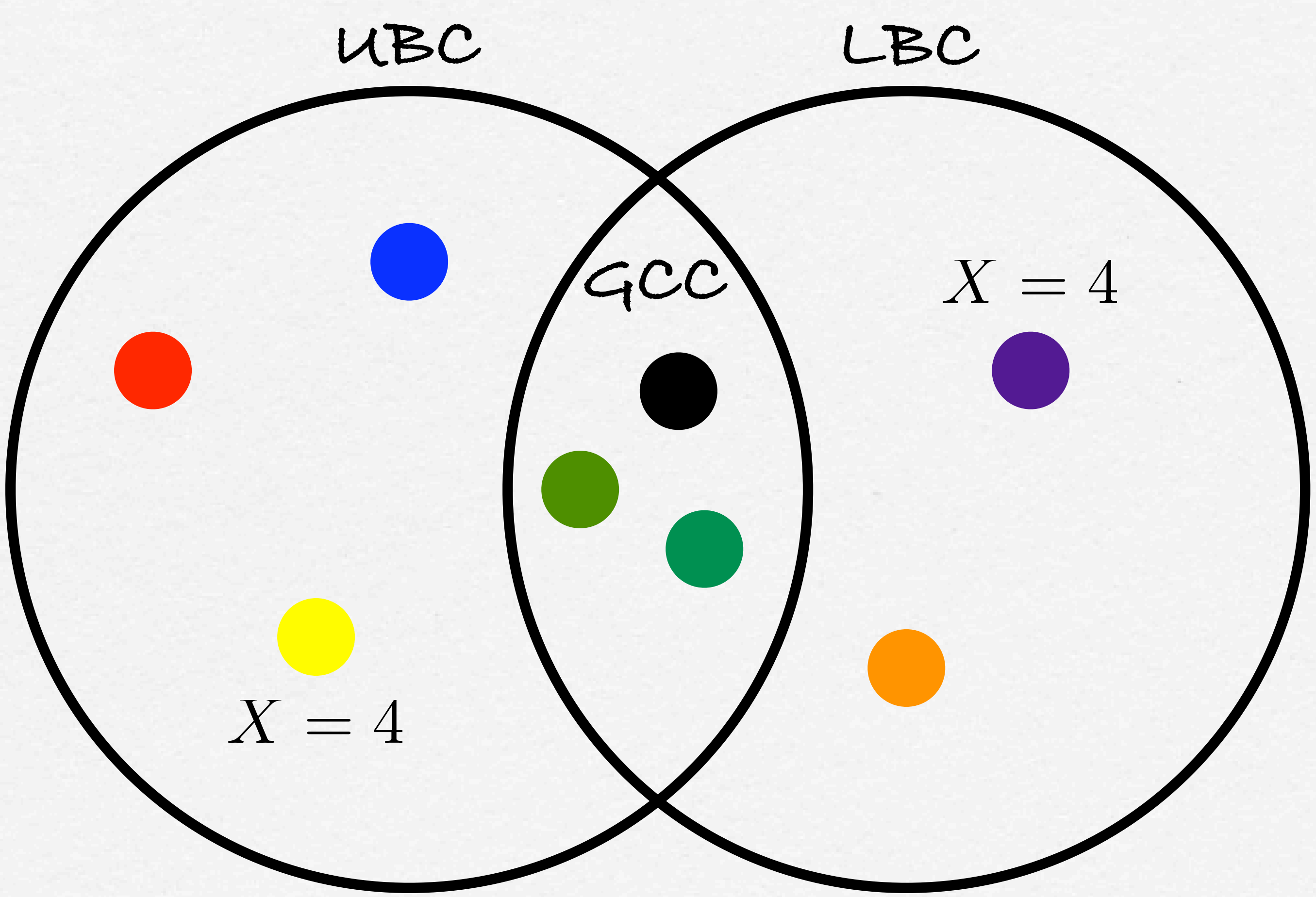
The Global Cardinality Constraint



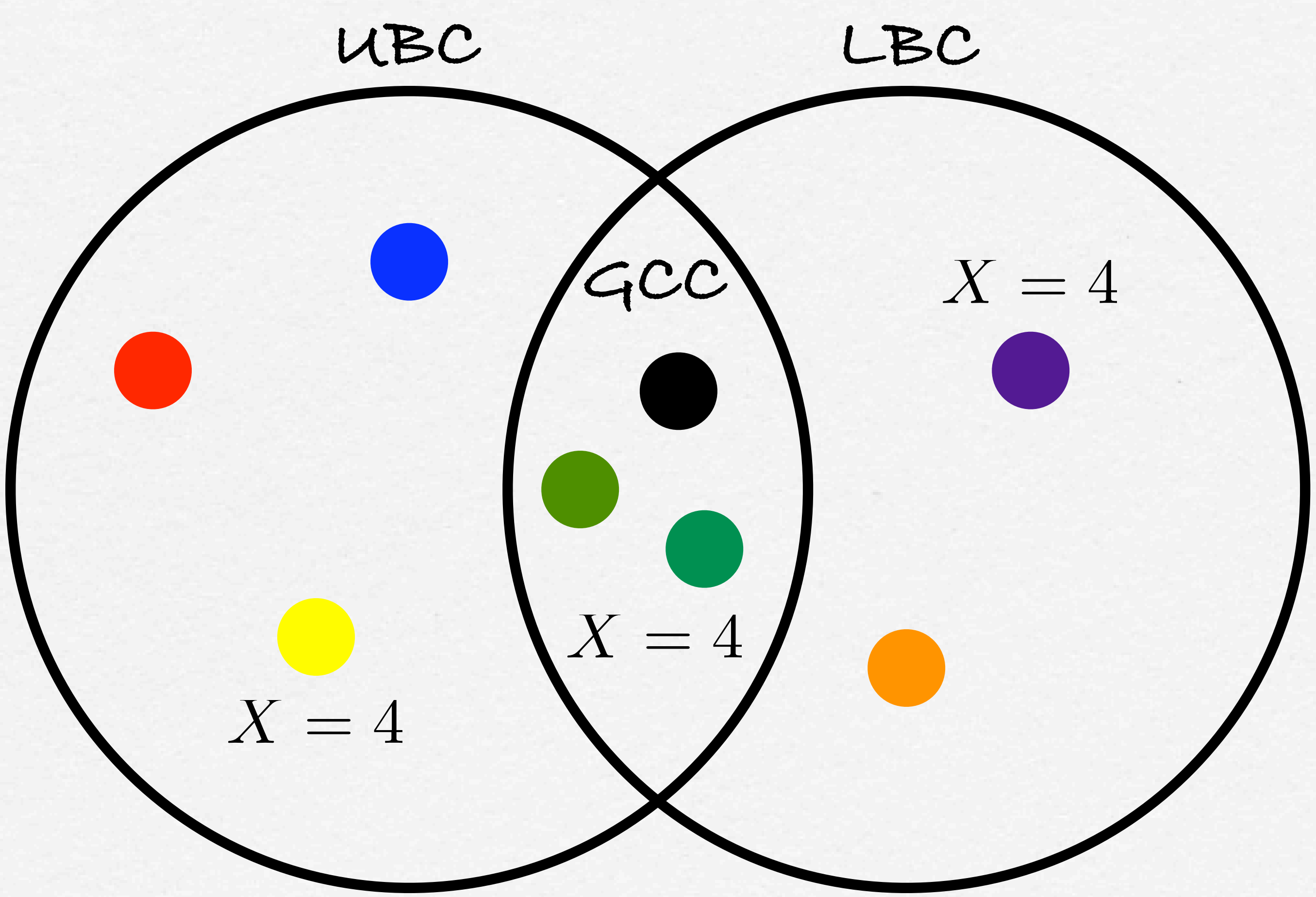
The Global Cardinality Constraint



The Global Cardinality Constraint



The Global Cardinality Constraint



Theorem:

A value has a support in the GCC iff it has a support in the UBC and the LBC.

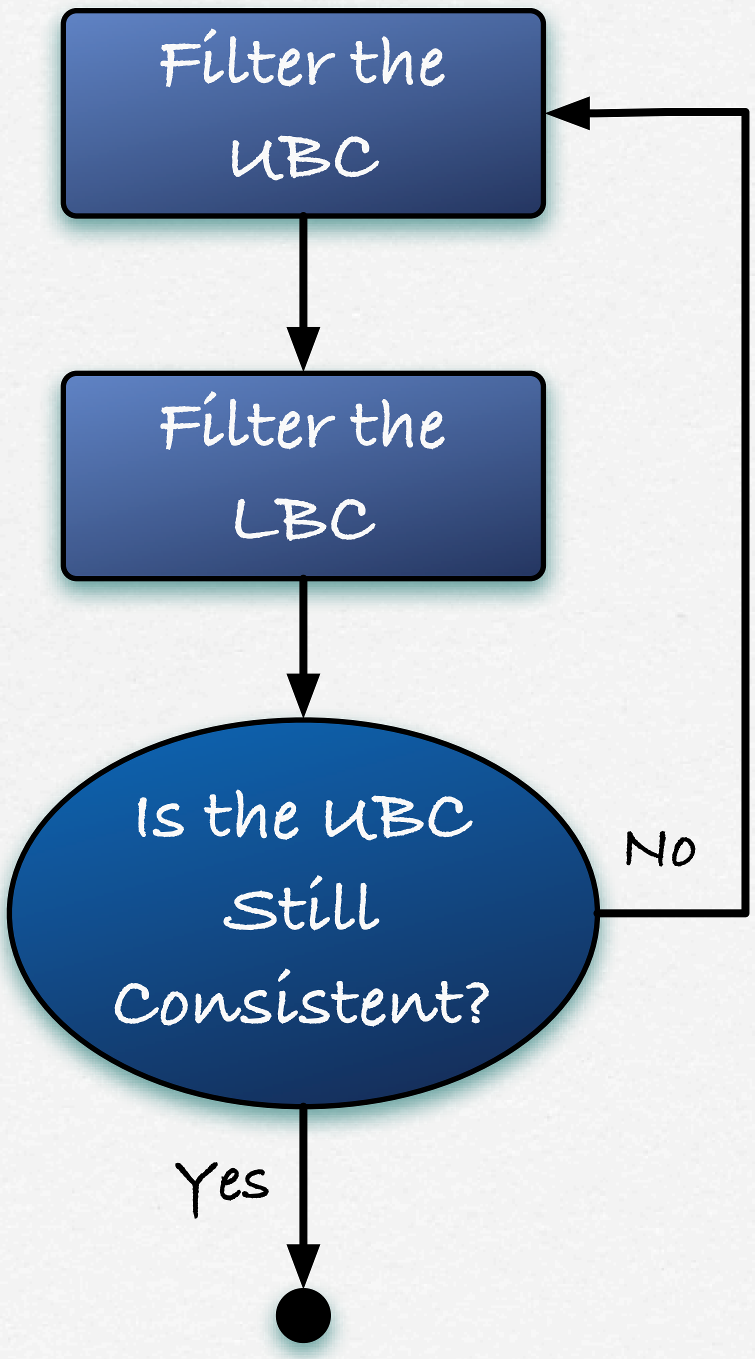
Proof:

Based on the relationship between Hall sets and unstable sets.

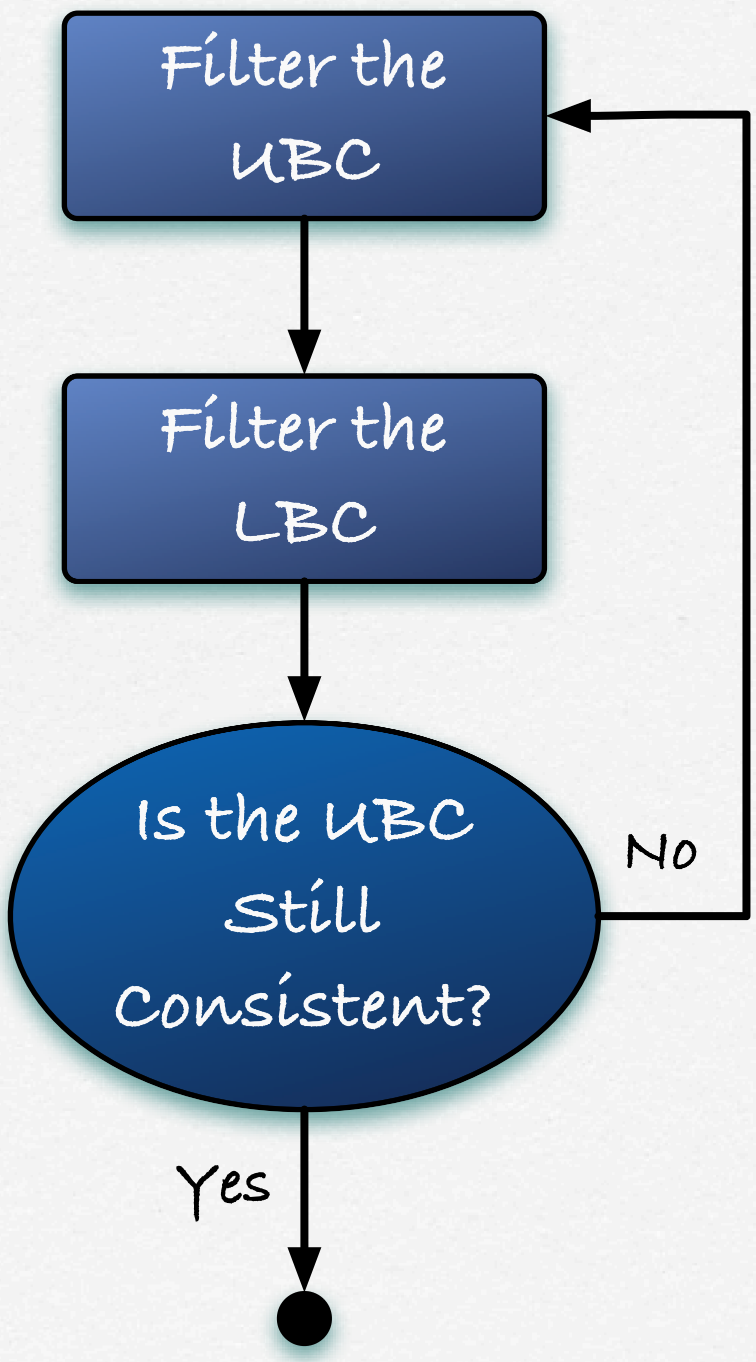
Note:

Holds for domain, range, and bounds consistency

A Propagator for the GCC



A Propagator for the GCC



Theorem:

This algorithm never loops!

Proof:

Based on the relationship between Hall sets and unstable sets.

Note:

Holds for domain, range, and bounds consistency

Extended GCC

- $\mathrm{EGCC}([X_1, \ldots, X_n], [C_1, \ldots, C_m])$ is satisfied when v is taken c_v times.

Extended GCC

- $\mathrm{EGCC}([X_1, \ldots, X_n], [C_1, \ldots, C_m])$ is satisfied when v is taken c_v times.

Theorem

When domains are sets, testing the satisfiability of EGCC is NP-Hard.

Extended GCC

- $\mathrm{EGCC}([X_1, \ldots, X_n], [C_1, \ldots, C_m])$ is satisfied when v is taken c_v times.

Theorem

When domains are sets, testing the satisfiability of EGCC is NP-Hard.

Theorem

When domains are intervals, filtering EGCC takes linear time.

Katriel & Thiel

Beyond Integer Domains

$$\textsc{All-Different}(X_1, \ldots, X_n) \iff X_i \neq X_j$$

Beyond Integer Domains

$$\textsc{All-Different}(X_1, \ldots, X_n) \iff X_i \neq X_j$$

- variables could be sets, multi-sets, or tuples.

Beyond Integer Domains

$$\textsc{All-Different}(X_1, \ldots, X_n) \iff X_i \neq X_j$$

- Variables could be sets, multi-sets, or tuples.
- Sets, multi-set, and tuple variables often have large domains.

Beyond Integer Domains

$$\textsc{All-Different}(X_1, \ldots, X_n) \iff X_i \neq X_j$$

- Variables could be sets, multi-sets, or tuples.
- Sets, multi-set, and tuple variables often have large domains.
- $\{\} \subseteq X \subseteq \{1, \ldots, u\} \Rightarrow |X| = 2^u$

Beyond Integer Domains

$$\textsc{All-Different}(X_1, \ldots, X_n) \iff X_i \neq X_j$$

- Variables could be sets, multi-sets, or tuples.
- Sets, multi-set, and tuple variables often have large domains.
- $\{\} \subseteq X \subseteq \{1, \ldots, u\} \Rightarrow |X| = 2^u$
- We adapted the propagator to obtain a polynomial complexity: $O(n^{2.5} + n^2 u)$

The Inter-Distance Constraint

$$\textsc{Inter-Distance}([X_1, \ldots, X_n], p) \iff |X_i - X_j| \geq p$$

- There must be a gap of p between each variable.

The Inter-Distance Constraint

$$\textsc{Inter-Distance}([X_1, \ldots, X_n], p) \iff |X_i - X_j| \geq p$$

- There must be a gap of p between each variable.
- When p = 1, we obtain the All-Different Constraint.

The Inter-Distance Constraint

$$\textsc{Inter-Distance}([X_1, \ldots, X_n], p) \iff |X_i - X_j| \geq p$$

- There must be a gap of p between each variable.
- When p = 1, we obtain the All-Different Constraint.
- Scheduling: Execution times must be p units of time apart.

The Inter-Distance Constraint

$$\textsc{Inter-Distance}([X_1, \ldots, X_n], p) \iff |X_i - X_j| \geq p$$

- There must be a gap of p between each variable.
- When p = 1, we obtain the All-Different Constraint.
- Scheduling: Execution times must be p units of time apart.
- Radio frequency allocation problem.

The Inter-Distance Constraint

- [Régin '97] introduces the global minimum distance constraint.

The Inter-Distance Constraint

- [Régin '97] introduces the global minimum distance constraint.
- [Artiouchine & Baptiste '05]

The Inter-Distance Constraint

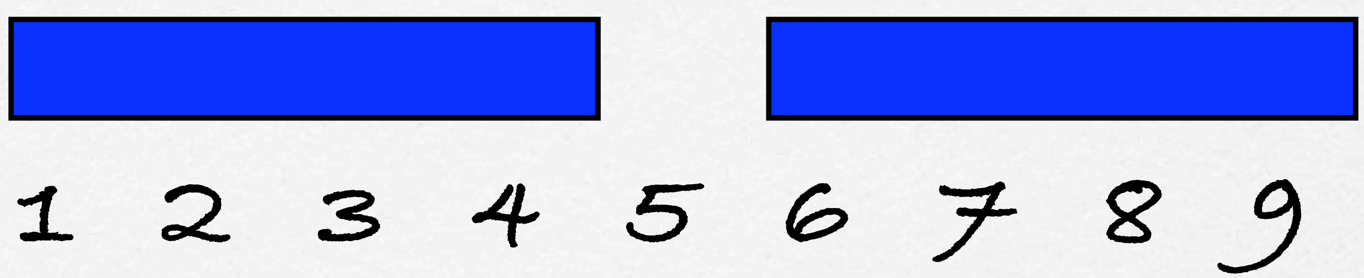
- [Régin '97] introduces the global minimum distance constraint.
- [Artiouchine & Baptiste '05]
  - prove the constraint is NP-Hard when variables are sets.

The Inter-Distance Constraint

- [Régin '97] introduces the global minimum distance constraint.
- [Artiouchine & Baptiste '05]
  - prove the constraint is NP-Hard when variables are sets.
  - achieve bounds consistency in cubic time.

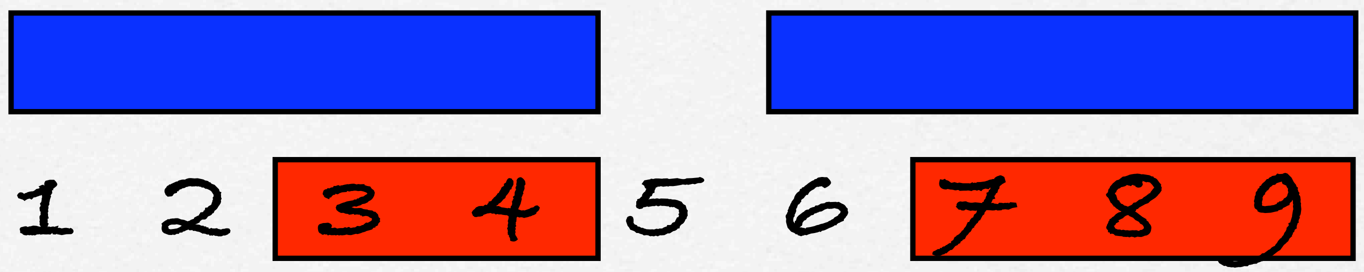
Block Placement

- Place two blocks of size 4 on the axis without overlapping them.



Block Placement

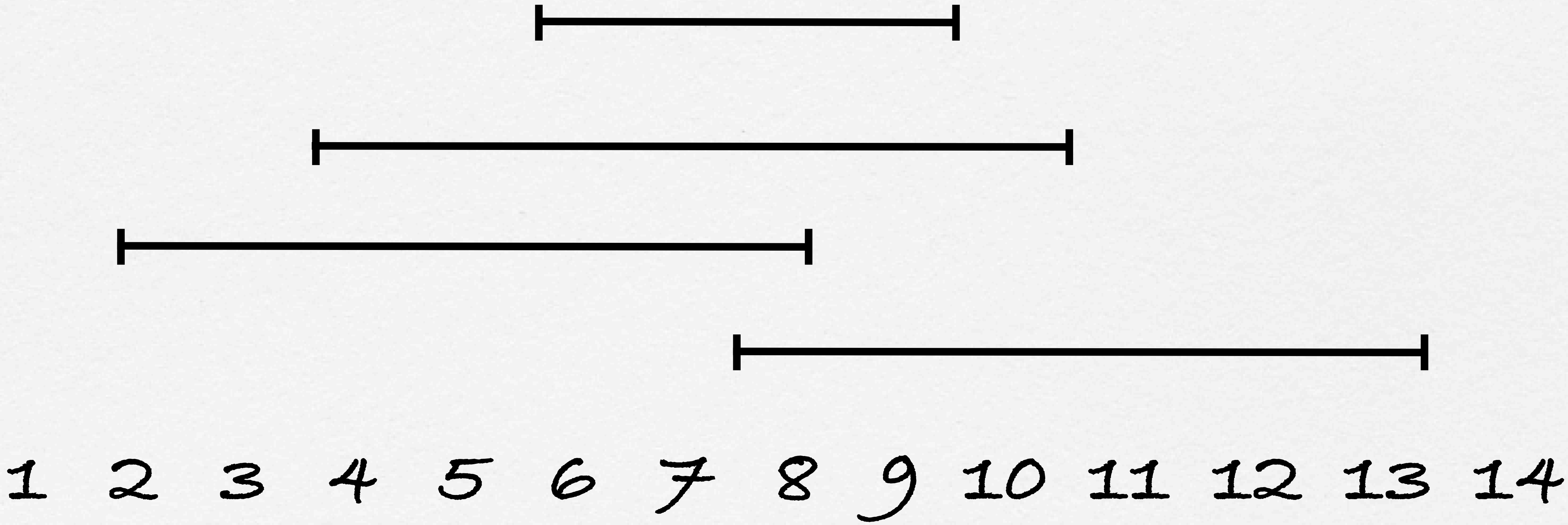
- Place two blocks of size 4 on the axis without overlapping them.



- No block can have its left end inside a red zone.

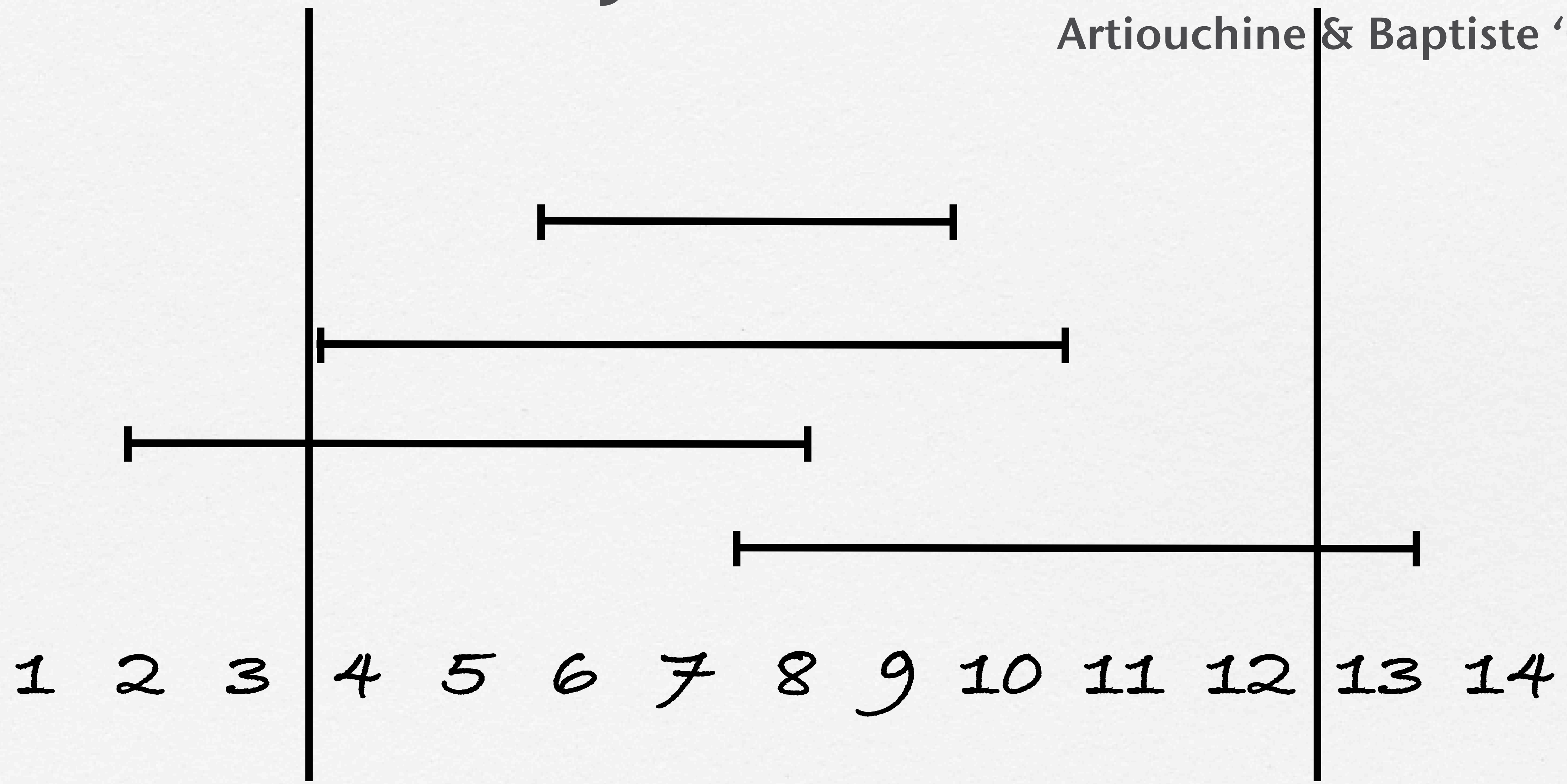
Internal Adjustment Intervals

Artiouchine & Baptiste '05



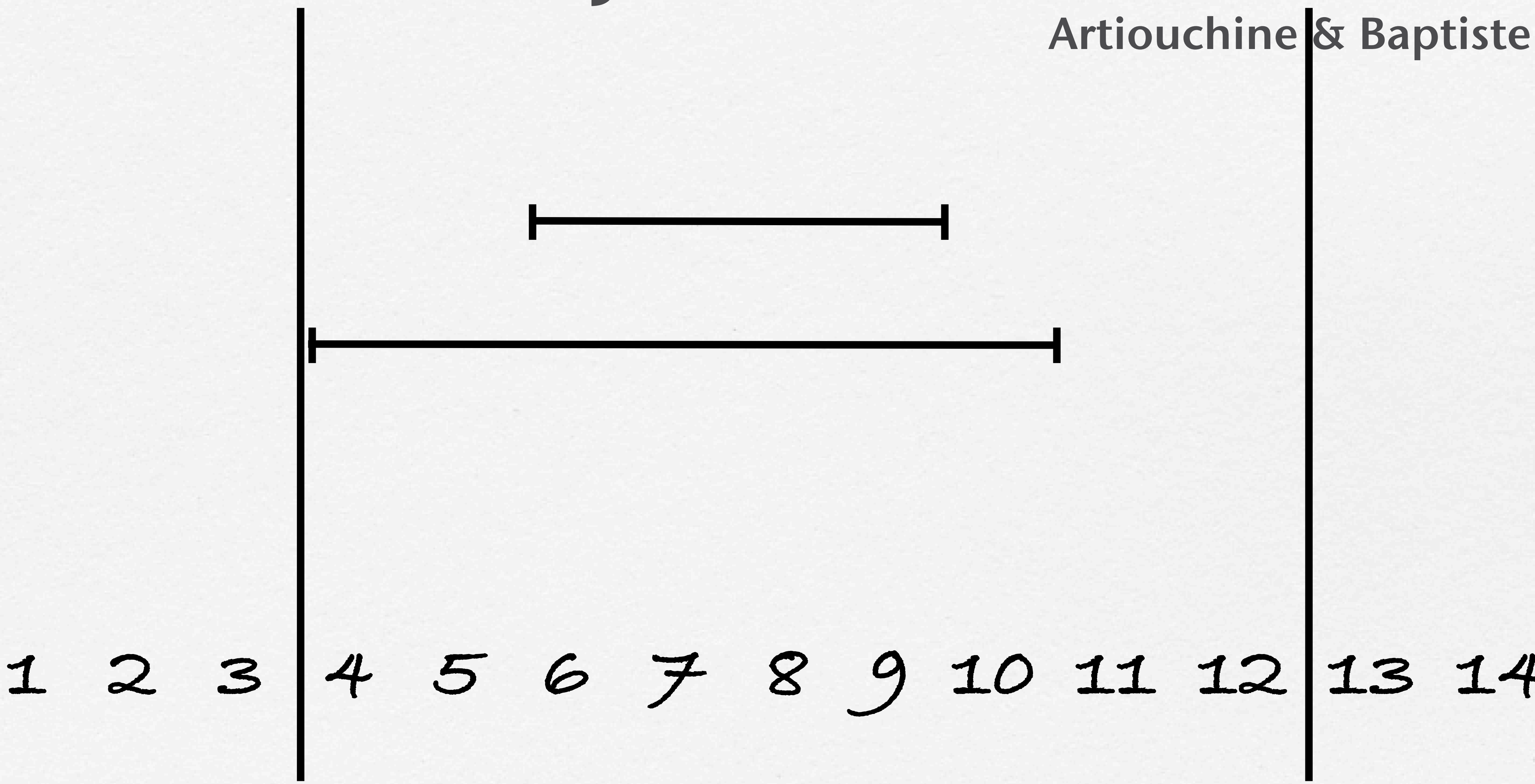
Internal Adjustment Intervals

Artiouchine & Baptiste '05



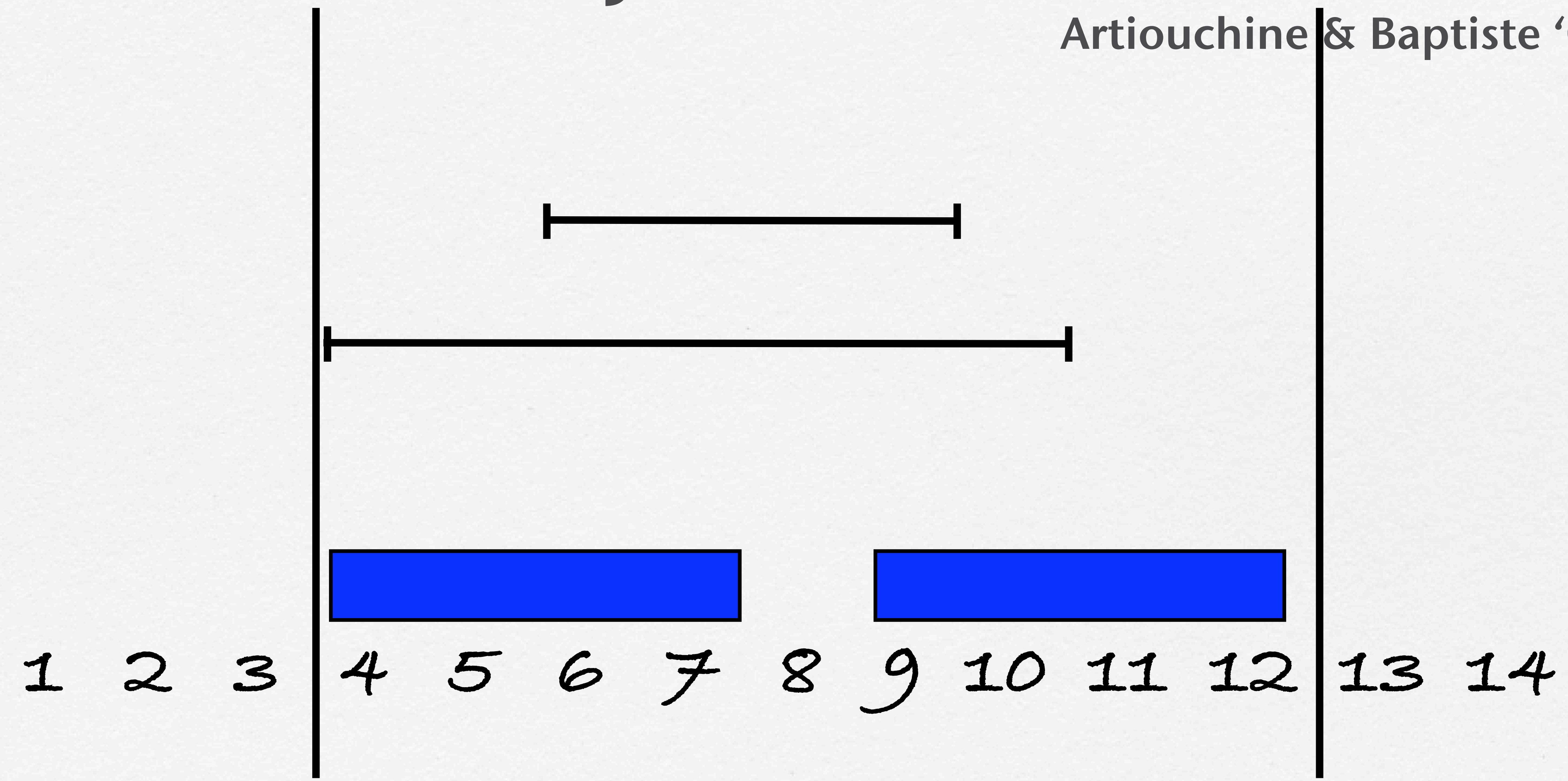
Internal Adjustment Intervals

Artiouchine & Baptiste '05



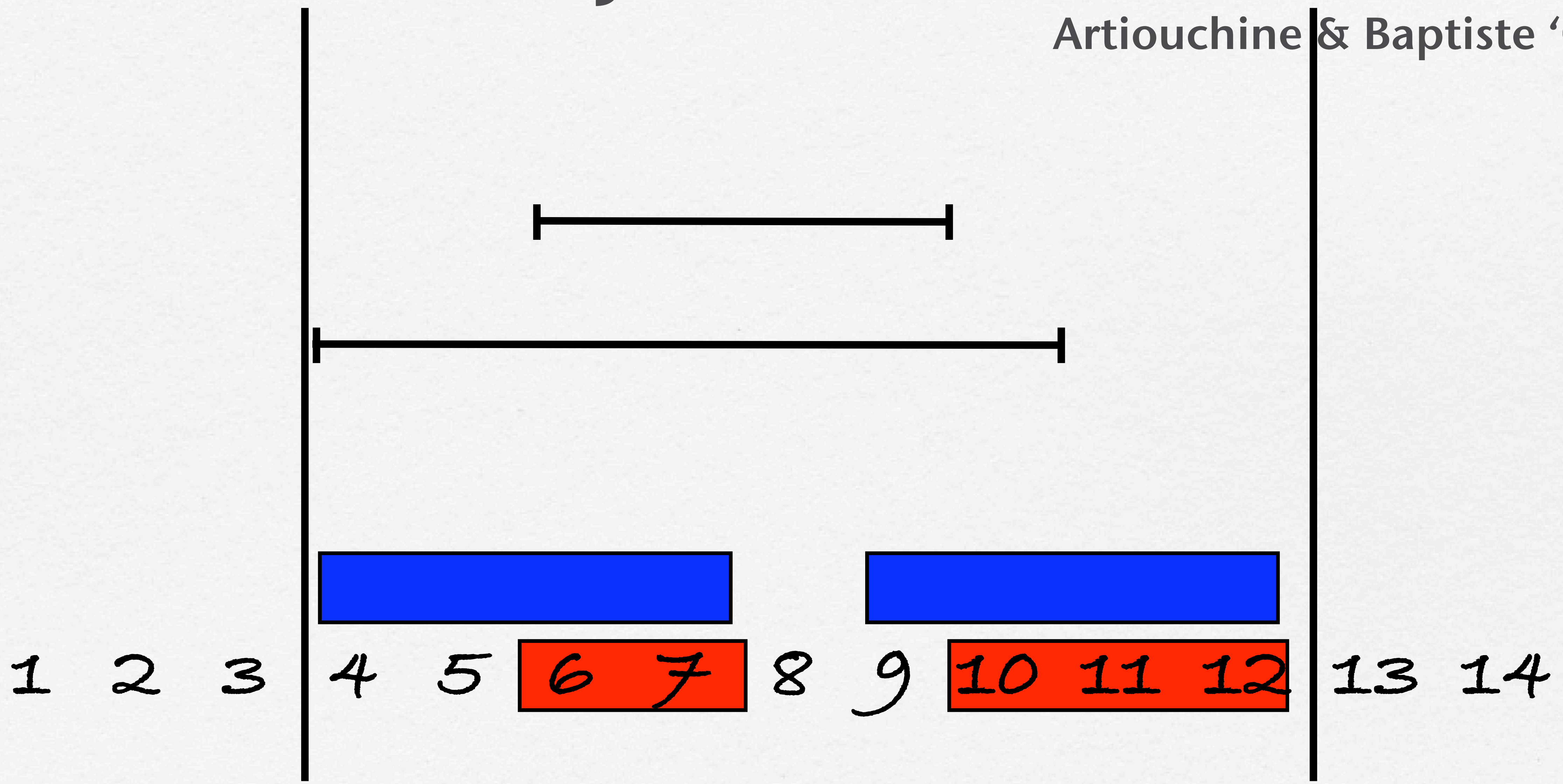
Internal Adjustment Intervals

Artiouchine & Baptiste '05



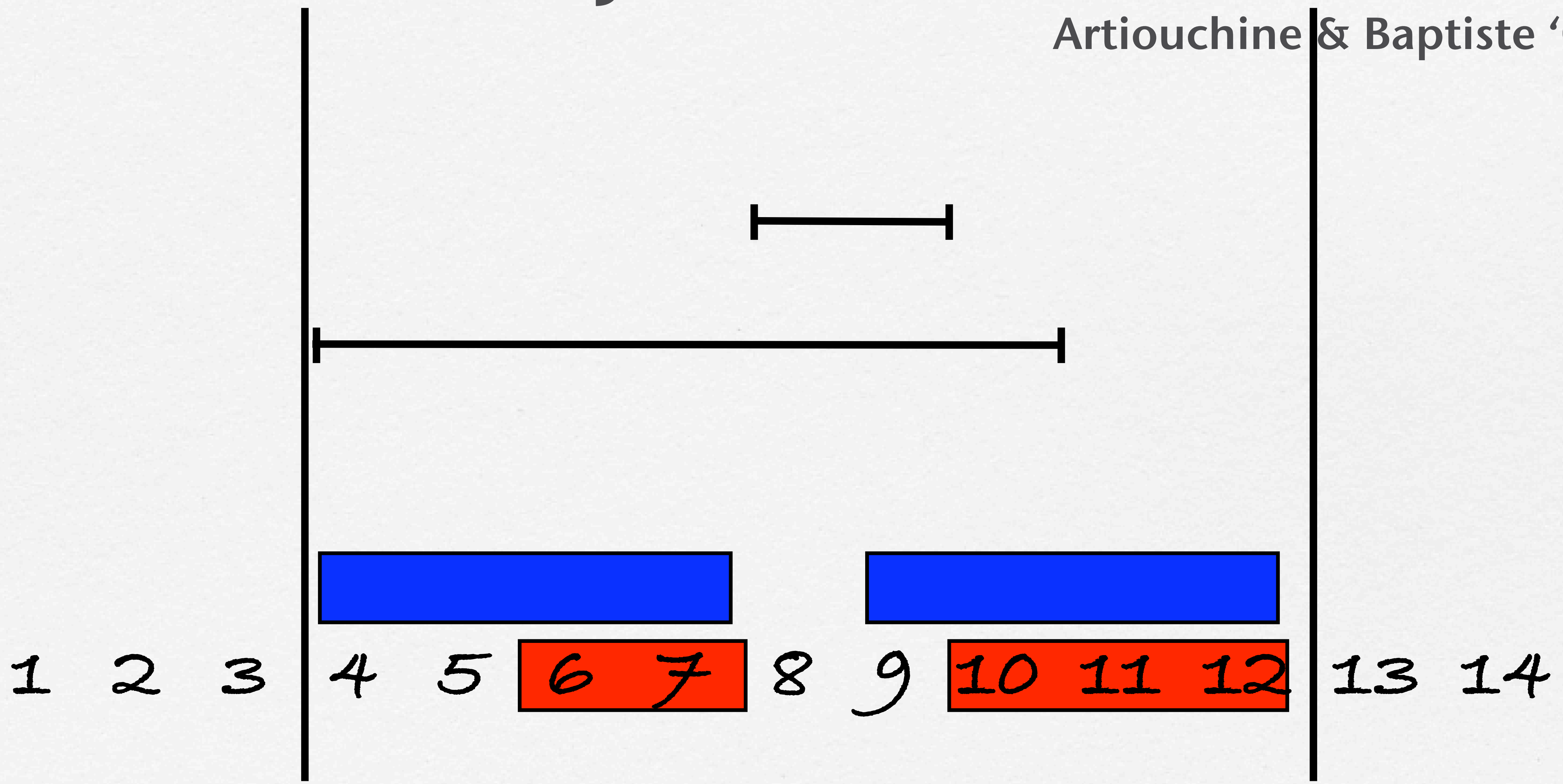
Internal Adjustment Intervals

Artiouchine & Baptiste '05



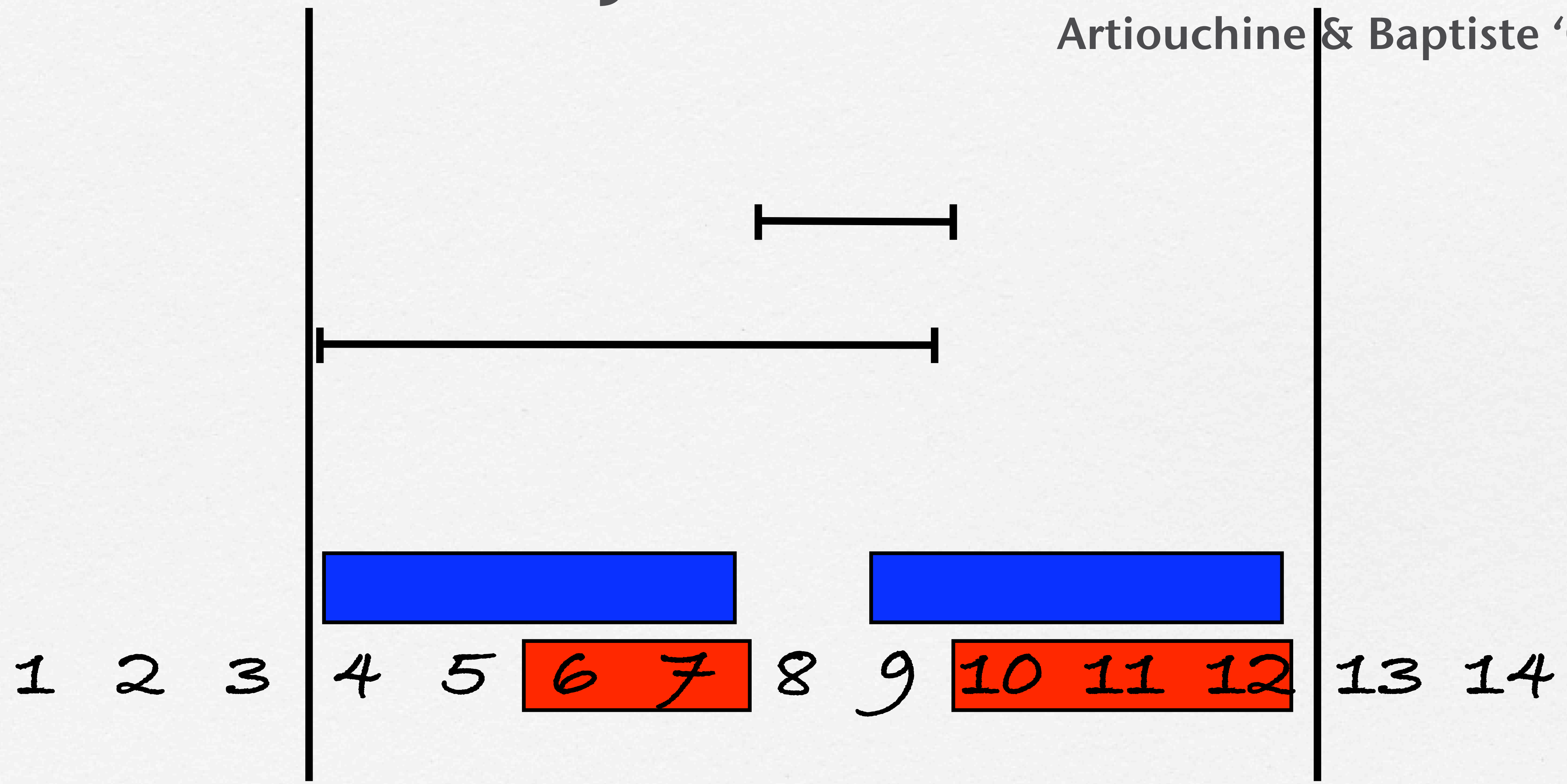
Internal Adjustment Intervals

Artiouchine & Baptiste '05

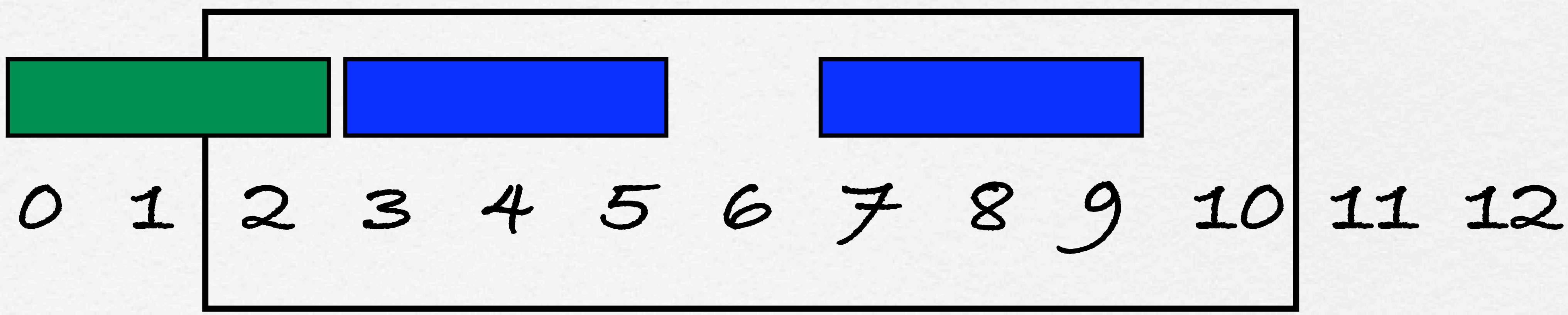


Internal Adjustment Intervals

Artiouchine & Baptiste '05

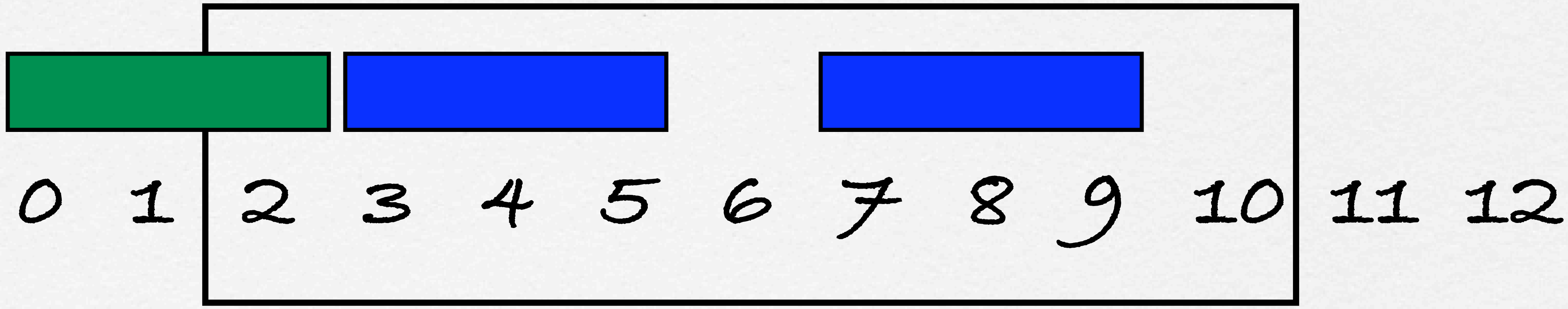


Block Placement



Block Placement

- Place the 3 blocks on the axis such that the blue blocks are in the box.

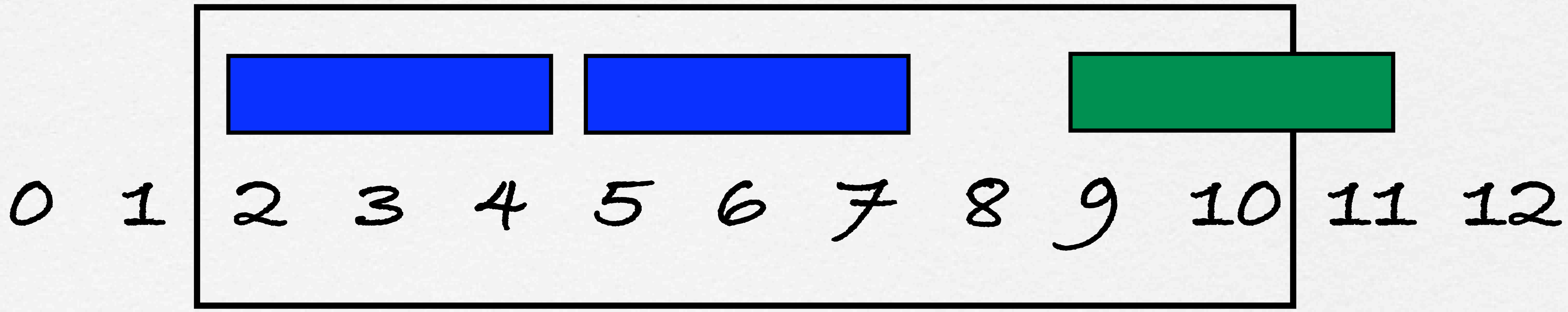


Block Placement

- Place the 3 blocks on the axis such that the blue blocks are in the box.

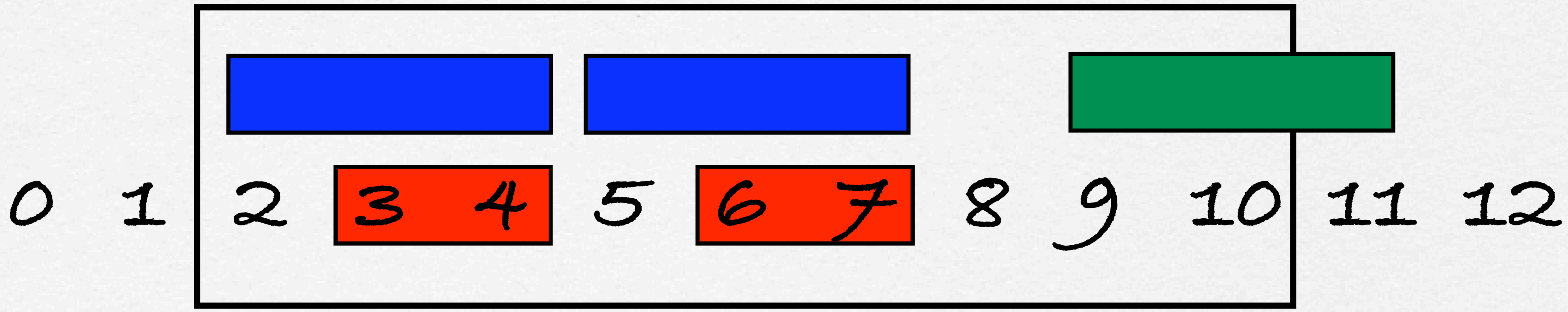
Block Placement

- Place the 3 blocks on the axis such that the blue blocks are in the box.



Block Placement

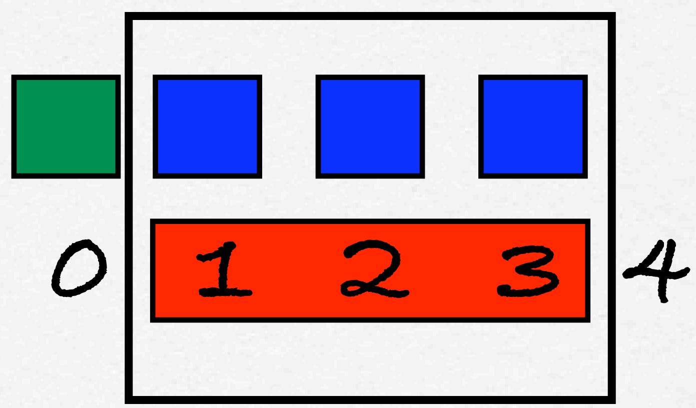
- Place the 3 blocks on the axis such that the blue blocks are in the box.



- The green box cannot have its left end inside a red zone.

Parenthesis

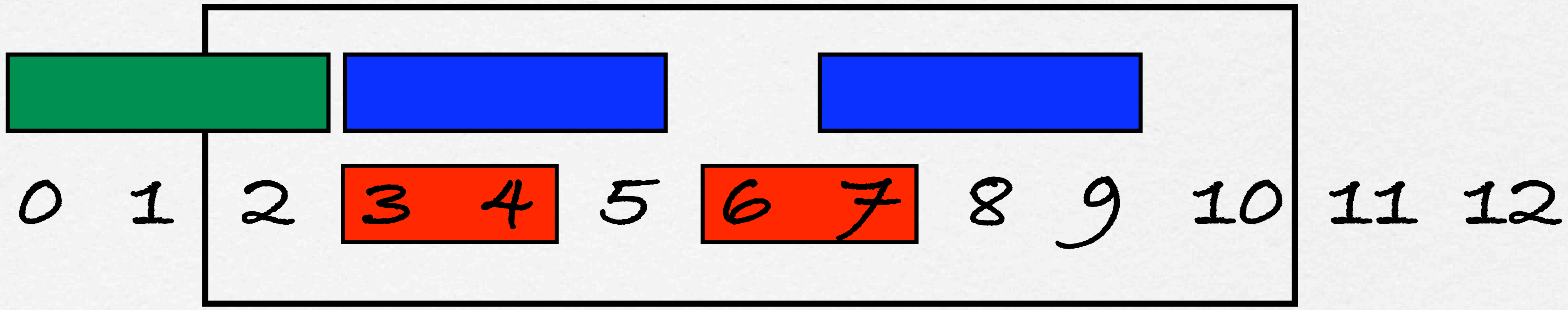
- If you place n blue blocks of size one inside a box of size n, you obtain a red zone of n elements.



- This is a Hall interval!

Block Placement

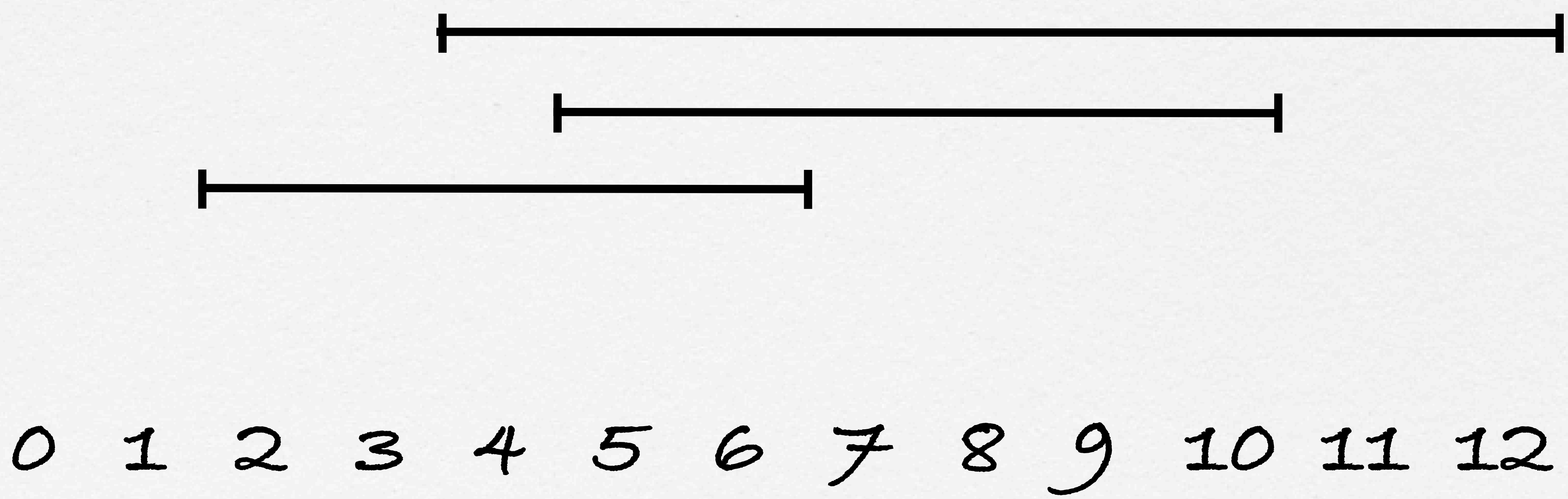
- Place the 3 blocks on the axis such that the blue blocks are in the box.



- The green box cannot have its left end inside a red zone

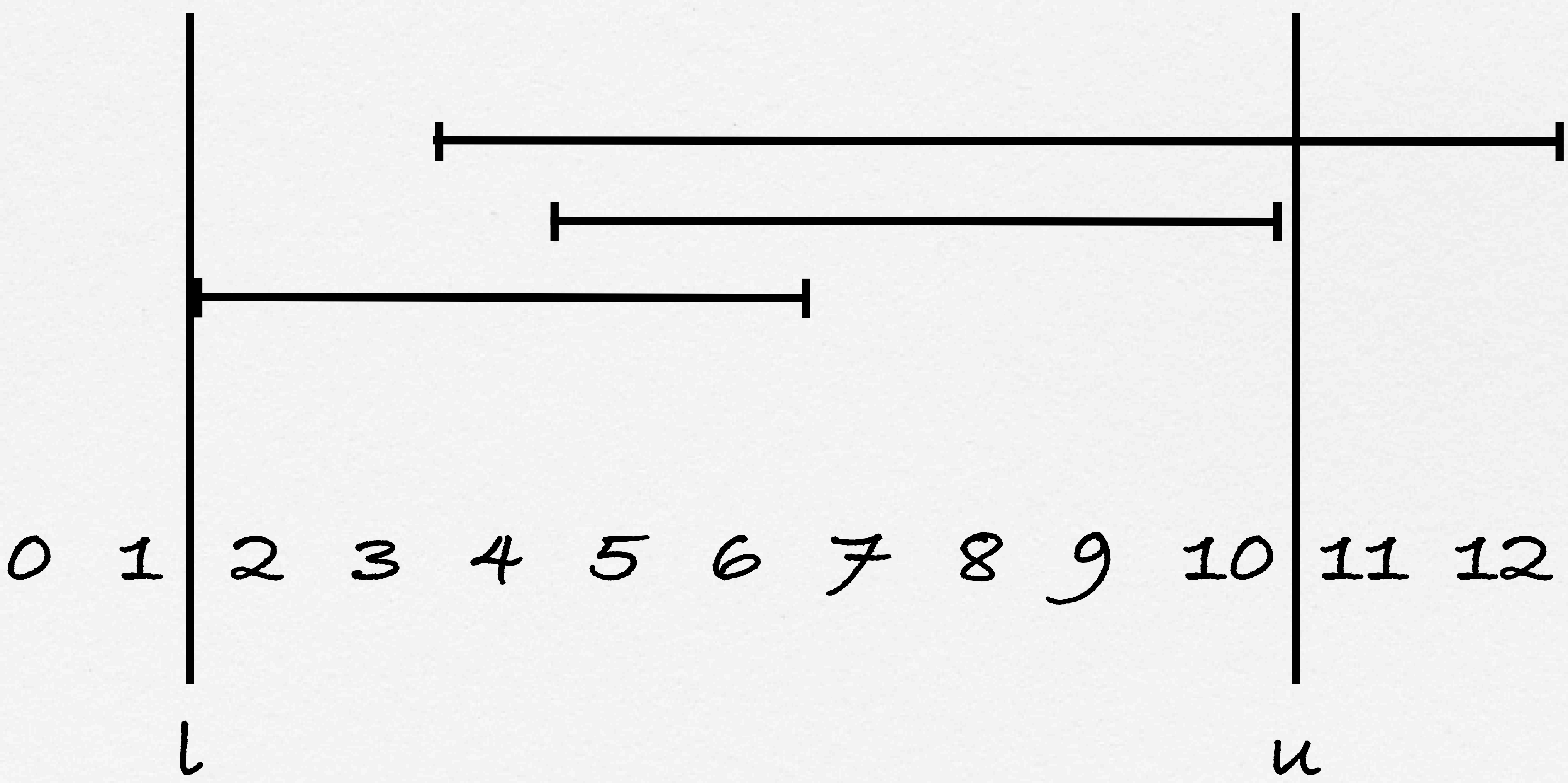
External Adjustment Intervals

Artiouchine & Baptiste '05



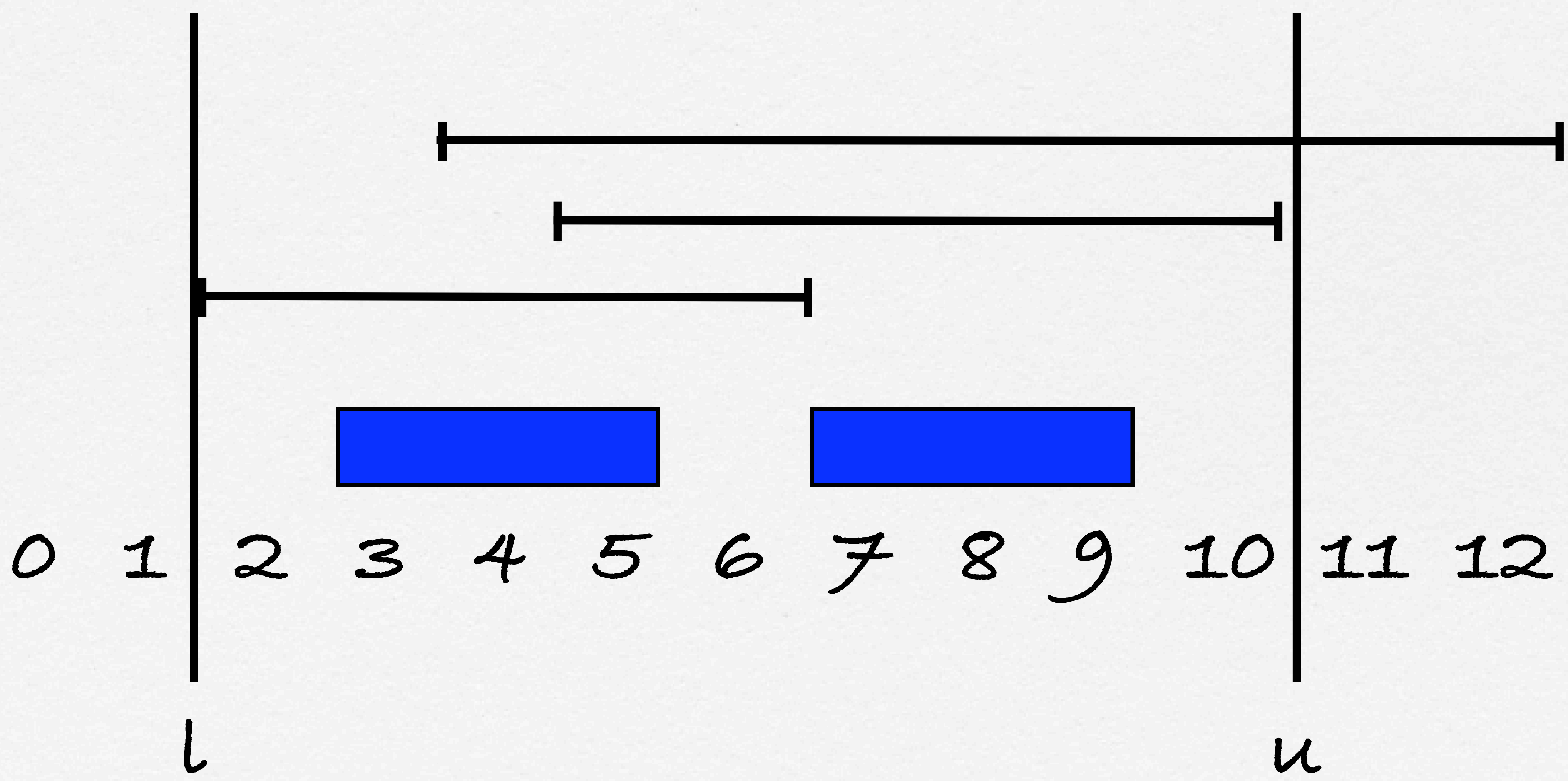
External Adjustment Intervals

Artiouchine & Baptiste '05



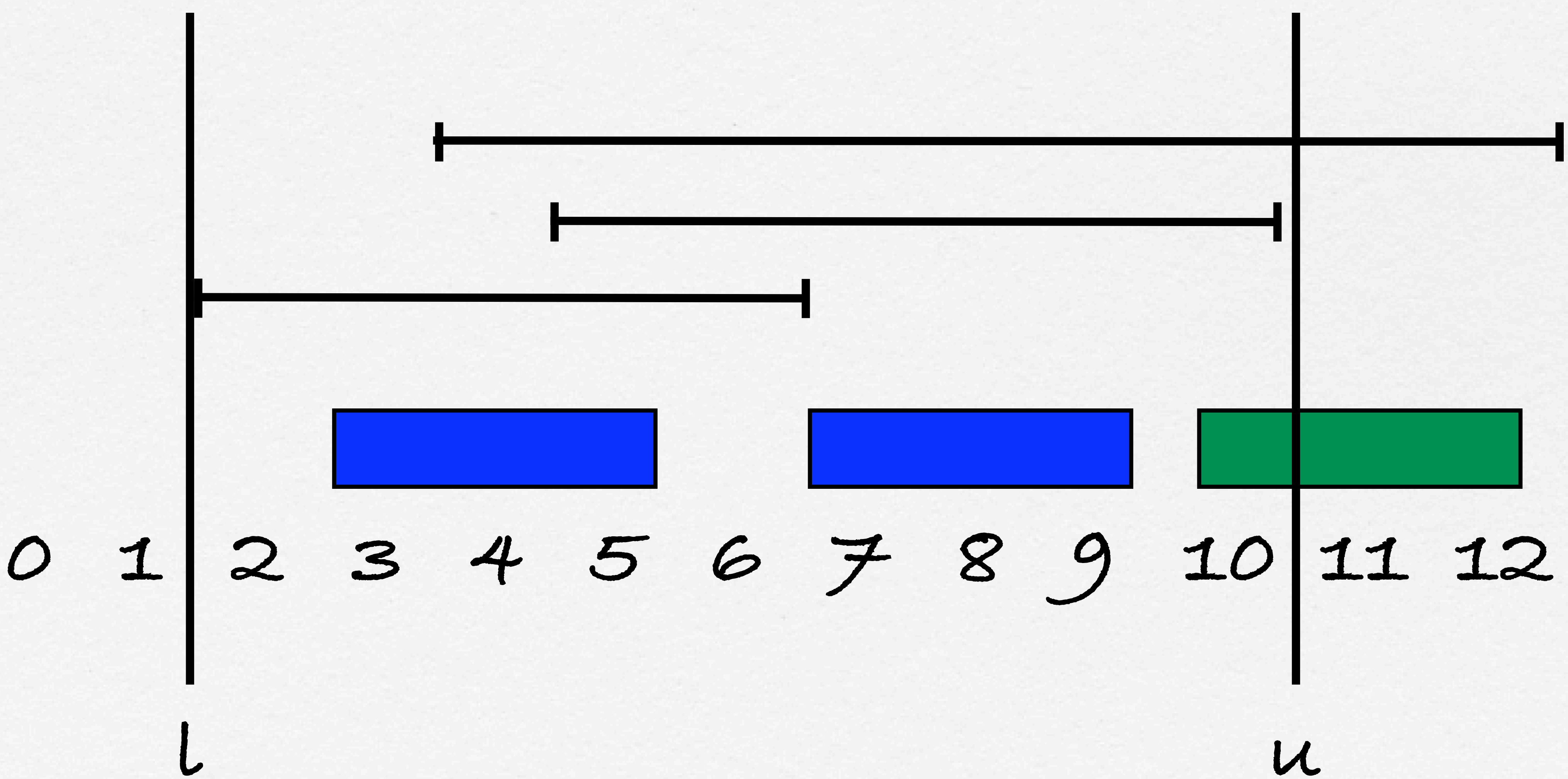
External Adjustment Intervals

Artiouchine & Baptiste '05



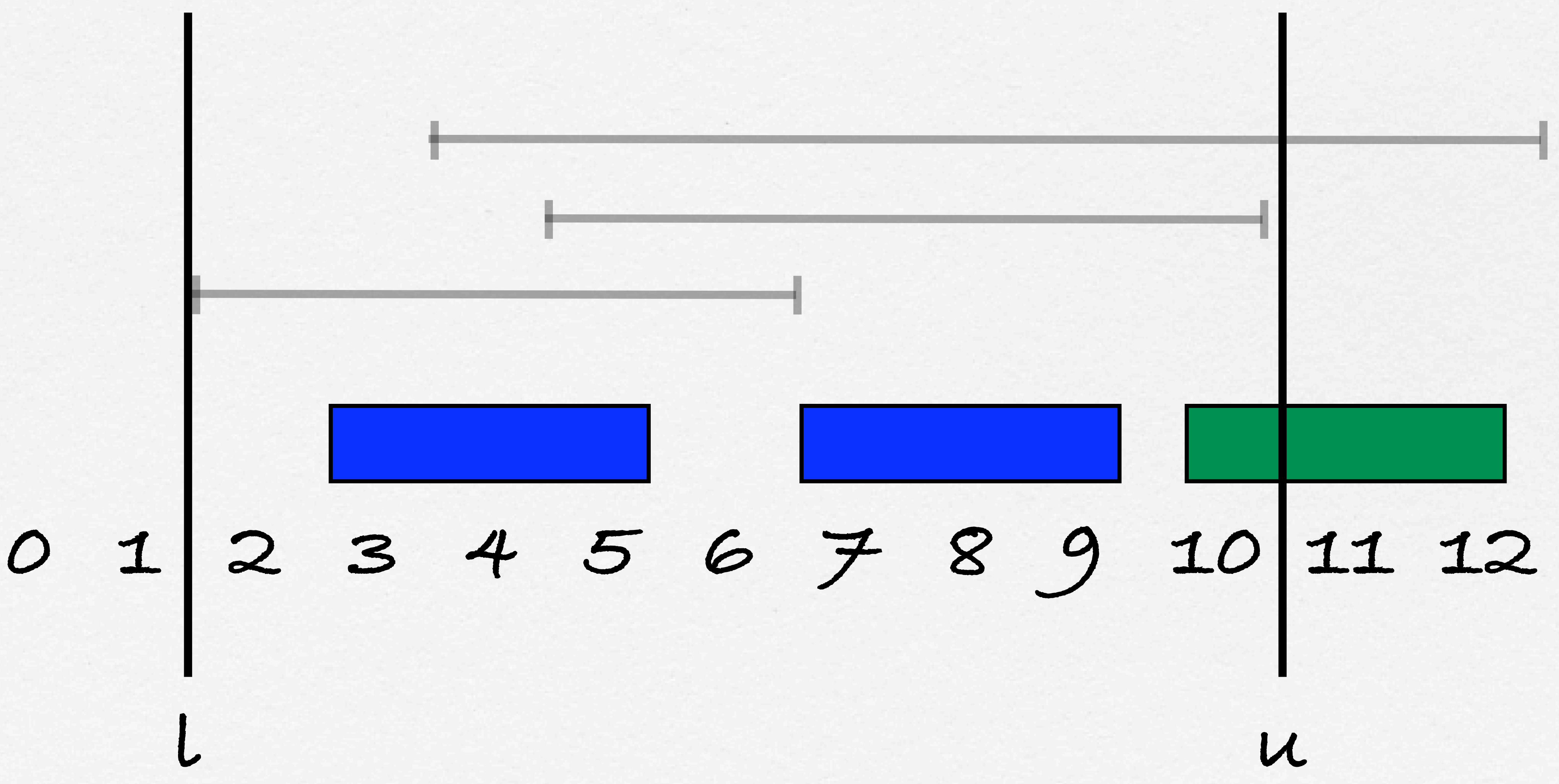
External Adjustment Intervals

Artiouchine & Baptiste '05



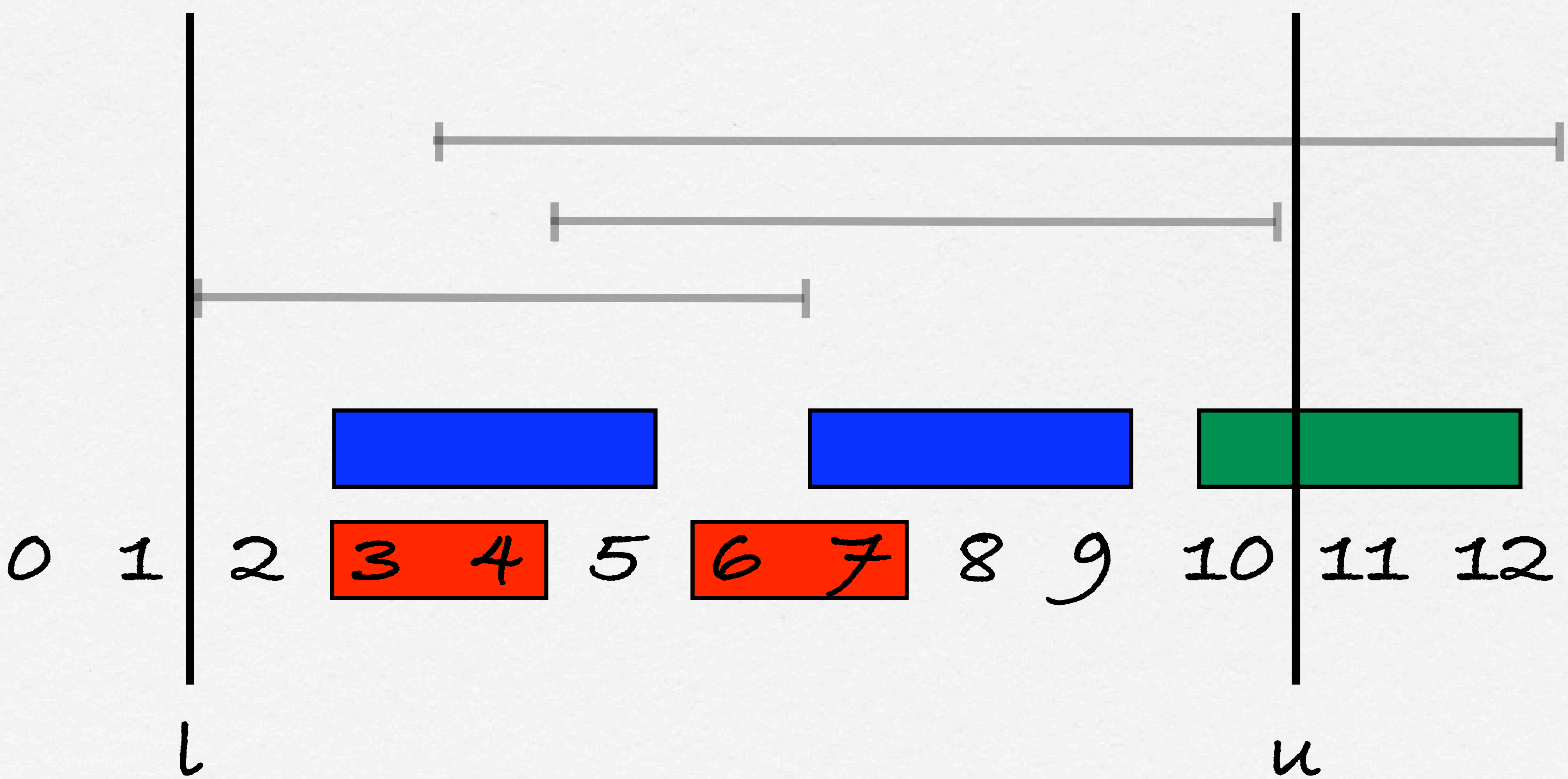
External Adjustment Intervals

Artiouchine & Baptiste '05



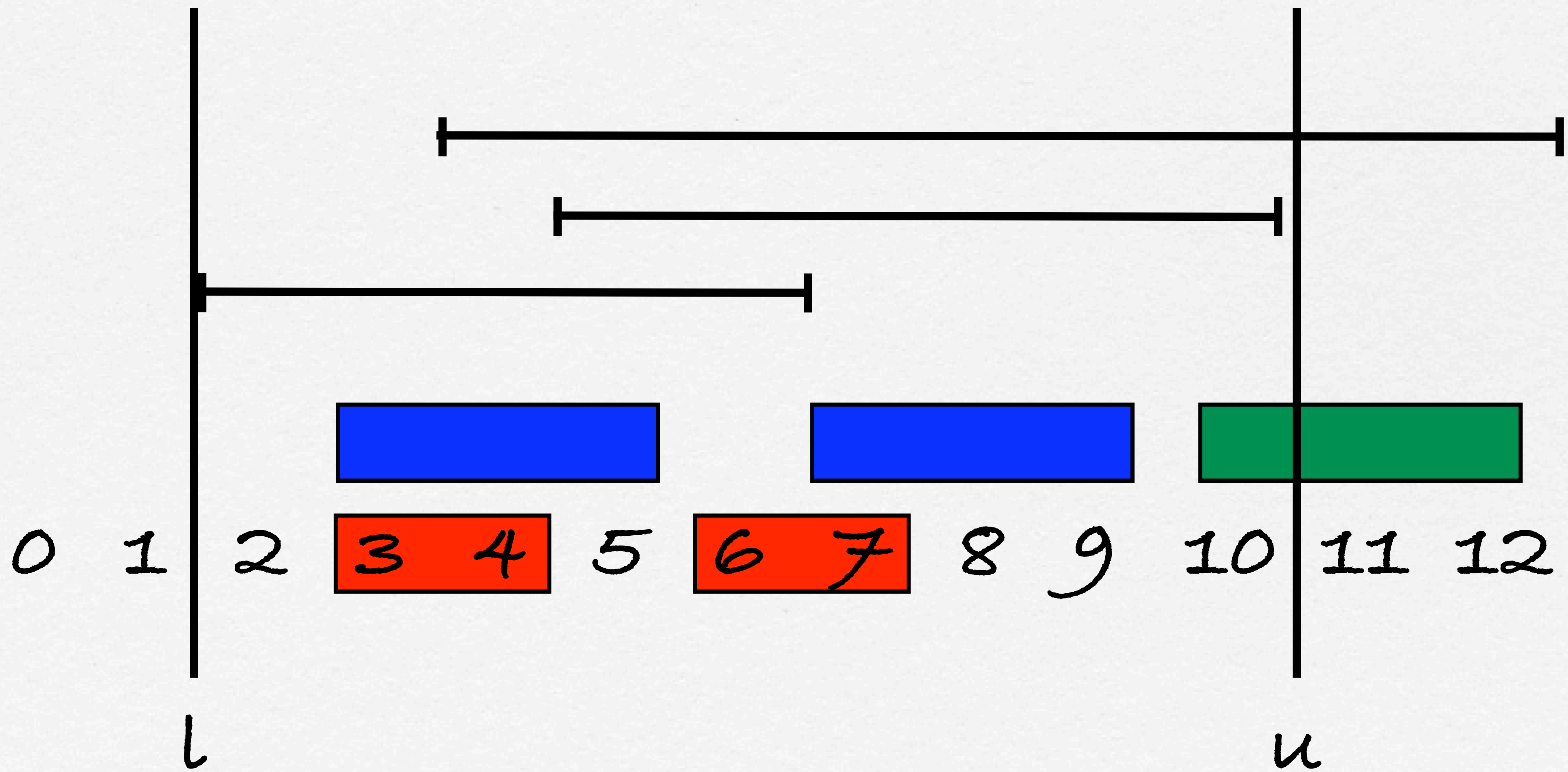
External Adjustment Intervals

Artiouchine & Baptiste '05



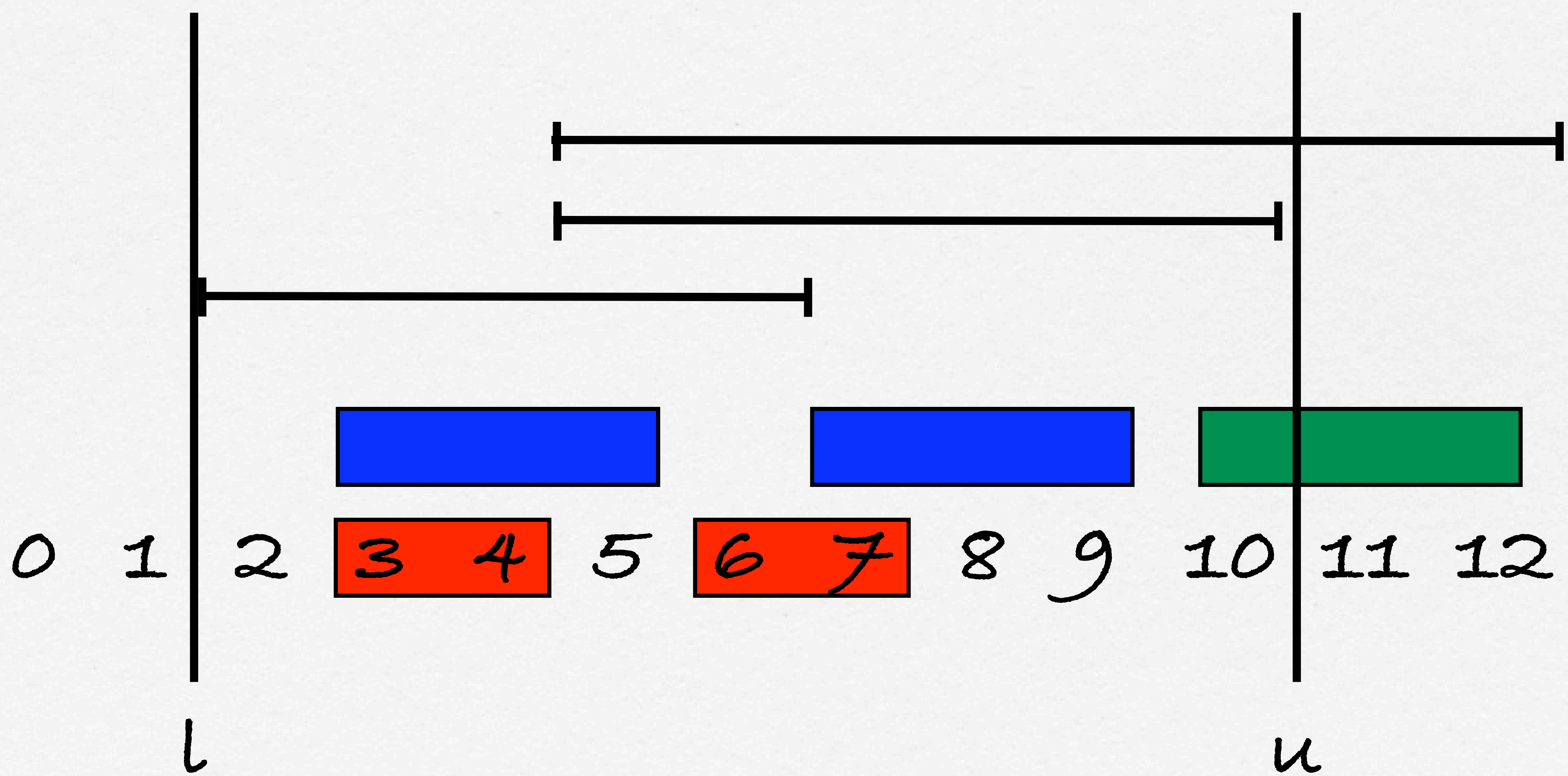
External Adjustment Intervals

Artiouchine & Baptiste '05



External Adjustment Intervals

Artiouchine & Baptiste '05



Number of Adjustment Intervals

$$O(n^2) \times O(n) = O(n^3)$$

Number of Adjustment Intervals

$$O(n^2) \times O(n) = O(n^3)$$

Number of intervals [l, u]

Number of Adjustment Intervals

Number of red zones produced per interval

$$O(n^2) \times O(n) = O(n^3)$$

Number of intervals [l, u]

Number of Adjustment Intervals

Number of red zones produced per interval

$$O(n^2) \times O(n) = O(n^3)$$

Number of intervals [l, u]

Total number of red zones

Number of Adjustment Intervals

Number of red zones produced per interval

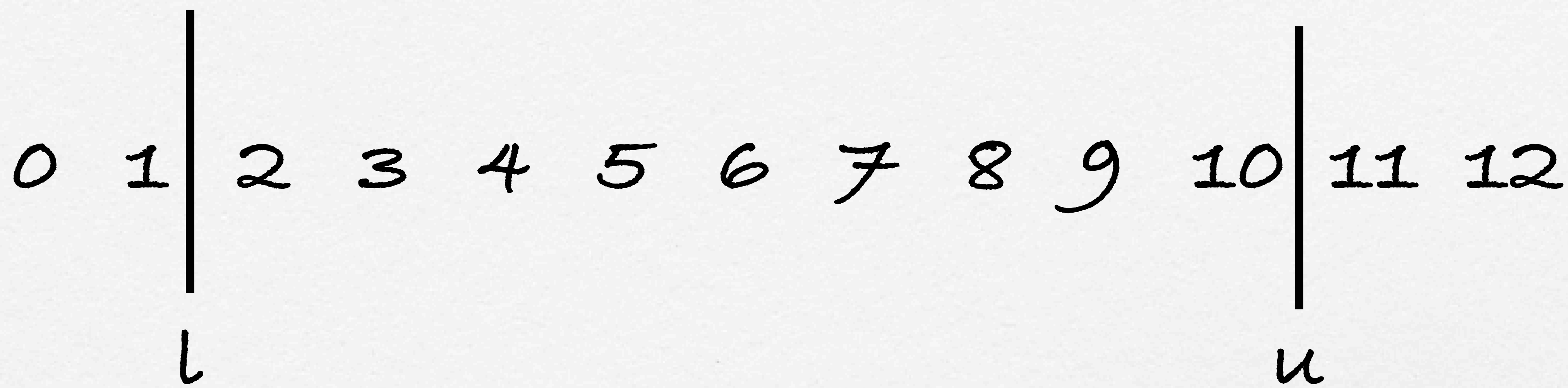
Complexity of Artiouchine & Baptiste's propagator

$$O(n^2) \times O(n) = O(n^3)$$

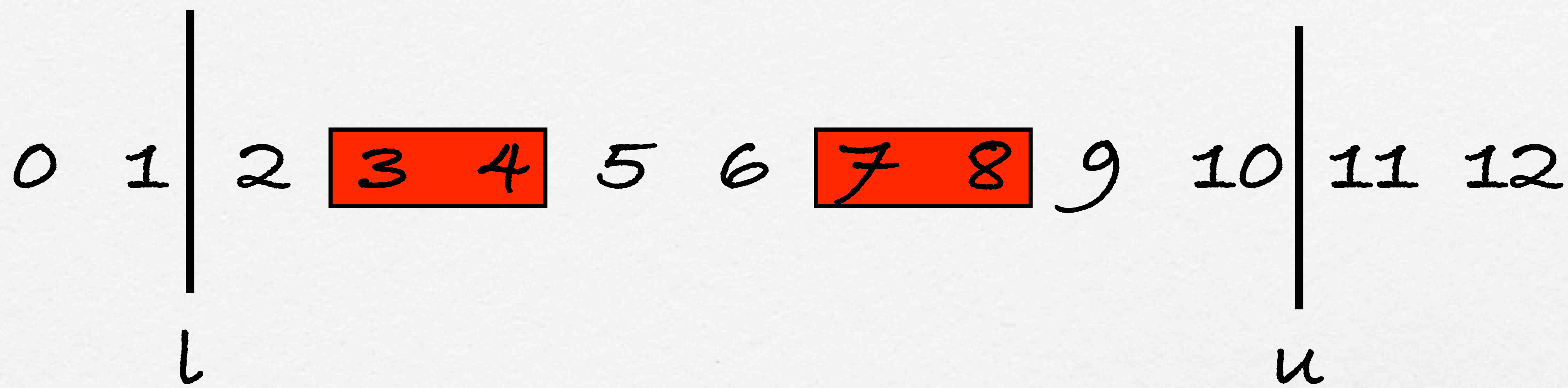
Number of intervals [l, u]

Total number of red zones

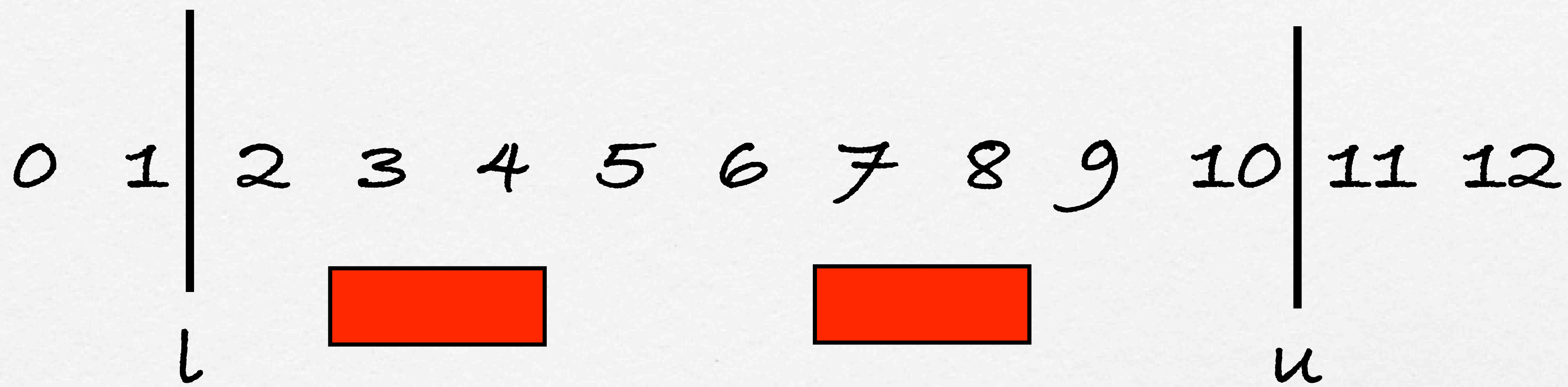
Dominance



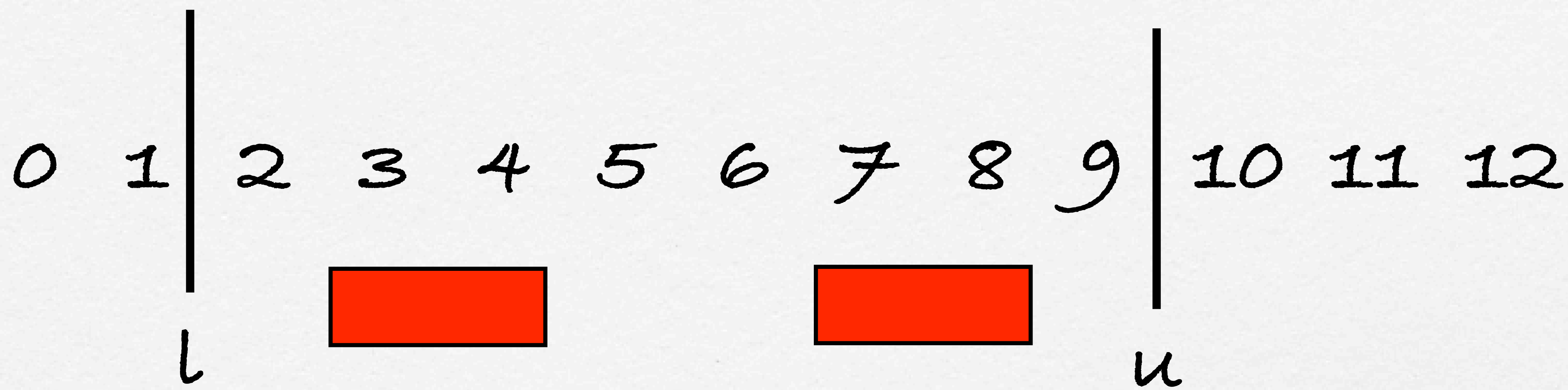
Dominance



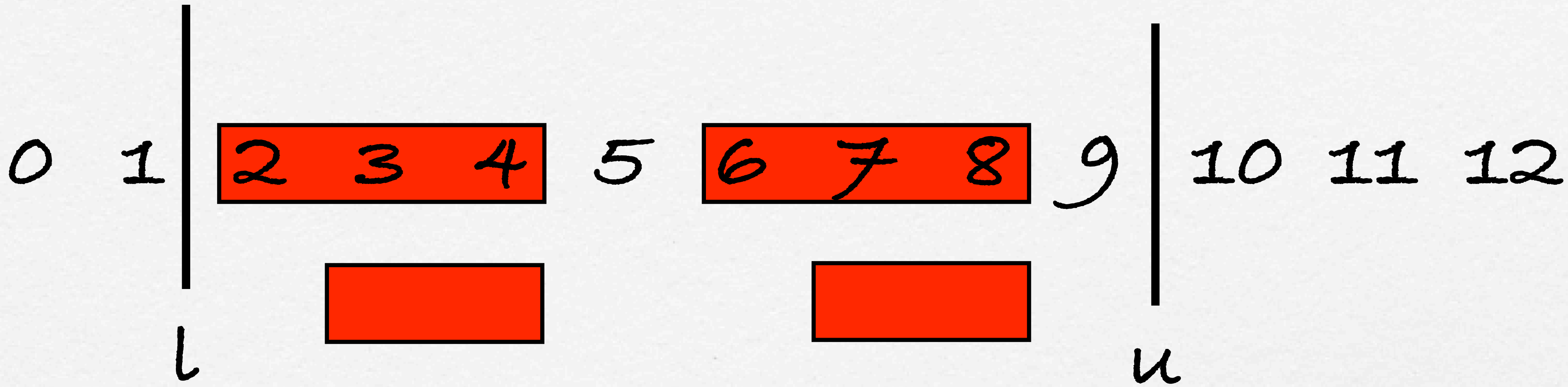
Dominance



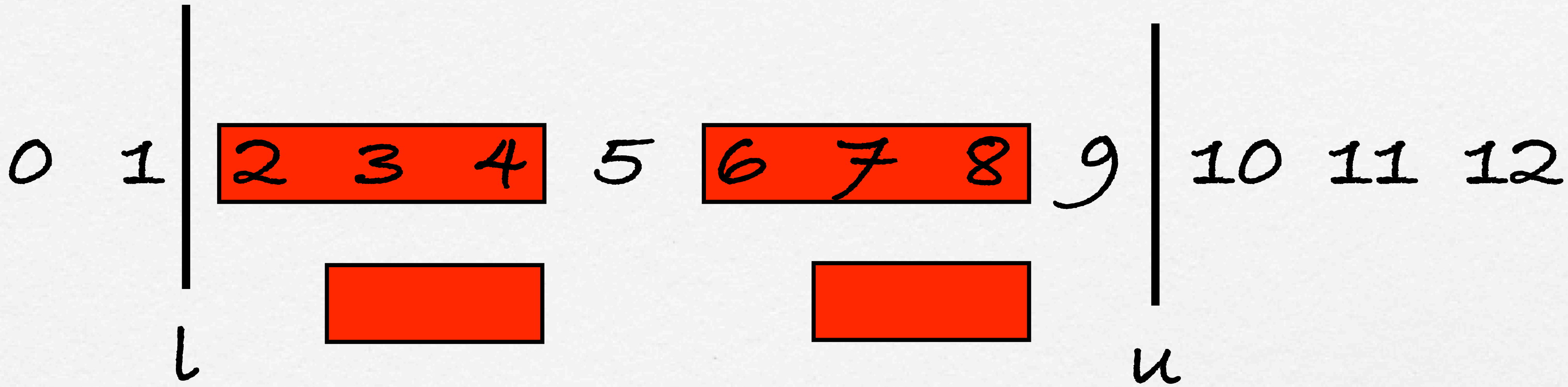
Dominance



Dominance



Dominance



<u>Theorem</u>

Only $O(n^2)$ red zones needs to be computed to achieve bounds consistency.

Propagator

- Uses a special data structure to store the adjustment intervals

Propagator

- Uses a special data structure to store the adjustment intervals
- Time complexity: $O(n^2)$

Summary

- Bounds consistency for the All-Different Constraint.

Summary

- Bounds consistency for the All-Different Constraint.
- Generalization of Hall's marriage theorem for the GCC.

Summary

- Bounds consistency for the All-Different Constraint.
- Generalization of Hall's marriage theorem for the GCC.
- Extension to non-integer domains

Summary

- Bounds consistency for the All-Different Constraint.
- Generalization of Hall's marriage theorem for the GCC.
- Extension to non-integer domains
- Quadratic propagator for the Inter-Distance.

Life after the PhD

Life after the PhD

NICTA

Microsoft® Research

Life after the PhD

Life after the PhD

NICTA

Microsoft® Research

Ωmega optimisation

UT TENSIO SIC VIS
ÉCOLE POLYTECHNIQUE MONTRÉAL

Special Thanks to a Special Supervisor

Alex López-Ortiz

Special Thanks to Special Collaborators

Peter van Beek

Toby Walsh

Thanks to my Thesis Committee





Thanks to the ACP