

**Integer vs. constraint programming**

*Practical Problem Solving*

- Model building: Language
- Model solving: Algorithms

**IP vs. CP: Language**

<table>
<thead>
<tr>
<th></th>
<th>IP</th>
<th>CP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Variables</td>
<td>(mostly) 0-1</td>
<td>Finite domain</td>
</tr>
<tr>
<td>Constraints</td>
<td>Linear equations and inequalities</td>
<td>Arithmetic constraints Symbolic/global constraints</td>
</tr>
</tbody>
</table>

**Example**

- Variables: $x_1, \ldots, x_n \in \{0, \ldots, m - 1\}$
- Constraint: Pairwise different values

**Example**

- Integer programming: Only linear equations and inequalities
  \[ x_i \neq x_j \iff x_i < x_j \lor x_i > x_j \]
  \[ \iff x_i \leq x_j - 1 \lor x_i \geq x_j + 1 \]

- Eliminating disjunction
  \[ x_i - x_j + 1 \leq my_1, \quad x_j - x_i + 1 \leq my_2, \quad y_1 + y_2 = 1, \]
  \[ y_1, y_2 \in \{0, 1\}, \quad 0 \leq x_i, x_j \leq m - 1, \]

- New variables: $z_{ik} = 1$ iff $x_i = k$, $i = 1, \ldots, n$, $k = 0, \ldots, m - 1$
  \[ z_{i0} + \cdots + z_{im} = 1, \quad z_{1k} + \cdots + z_{nk} \leq 1, \]

- Constraint programming $\rightarrow$ **symbolic constraint**
  \[ \text{alldifferent}(x_1, \ldots, x_n) \]
Symbolic/global constraints

- \textit{alldifferent}\{x_1, \ldots, x_n\}
- \textit{cumulative}\{s_1, \ldots, s_n, d_1, \ldots, d_n, r_1, \ldots, r_n, c, e\}.

- \(n\) tasks: starting time \(s_i\), duration \(d_i\), resource demand \(r_i\)
- resource capacity \(c\), completion time \(e\)

\[
\text{cumulative}([1,2,4], [4,2,2], [1,2,2], 3) \\
\text{cumulative}([1,2,2], [1,1,1], [2,1,2], 3) \\
\text{cumulative}([1,3,5], [2,1,1], [1,1,1], 1)
\]

Difn Constraint
Beldiceanu/Contejean’94

- Nonoverlapping of \(n\)-dimensional rectangles \([O_{1i}, \ldots, O_{ni}, L_{1i}, \ldots, L_{ni}]\), where \(O_i\) (resp. \(L_i\)) denotes the origin (resp. length) in dimension \(i\)

- \textit{difn}([[O_{11}, \ldots, O_{1n}, L_{11}, \ldots, L_{1n}], \ldots, [O_{mn}, \ldots, O_{mn}, L_{mn}, \ldots, L_{mn}]])

- General form: \textit{difn}(Rectangles, Min_Vol, Max_Vol, End, Distances, Regions)

IP vs. CP: Algorithms

<table>
<thead>
<tr>
<th></th>
<th>IP</th>
<th>CP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inference</td>
<td>Linear programming</td>
<td>Domain filtering</td>
</tr>
<tr>
<td></td>
<td>Cutting planes</td>
<td>Constraint propagation</td>
</tr>
<tr>
<td>Search</td>
<td>Branch-and-relax</td>
<td>Branch-and-bound</td>
</tr>
<tr>
<td></td>
<td>Branch-and-cut</td>
<td></td>
</tr>
<tr>
<td>Bounds on the object</td>
<td>Two-sided</td>
<td>One-sided</td>
</tr>
<tr>
<td>function</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Local vs. global reasoning

Linear arithmetic constraints

\[ 3x + y \leq 7, \]
\[ 3y + x \leq 7, \]
\[ x + y = z, \]
\[ x, y \in \{0, \ldots, 3\} \]

CP \( x, y \leq 2, z \leq 4 \)
LP \( x, y \leq 2, z \leq 3.5 \)
IP \( x, y \leq 2, z \leq 3 \)

Global reasoning in CP ? \(\leadsto\) global constraints!

Global reasoning in CP

Example

- \(x_1, x_2, x_3 \in \{0, 1\}\)
- pairwise different values
- **Local** consistency, 3 disequalities: \(x_1 \neq x_2, x_1 \neq x_3, x_2 \neq x_3\)
  \(\leadsto x_1, x_2, x_3 \in \{0, 1\}\), i.e., no domain reduction is possible
- **Global** constraint: \(\text{alldifferent}(x_1, x_2, x_3)\)
  \(\leadsto\) detects infeasibility (uses bipartite matching)

Global reasoning in CP: inside global constraints

Summary

<table>
<thead>
<tr>
<th>General</th>
<th>ILP</th>
<th>CP(FD)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Language</strong></td>
<td>Linear arithmetic</td>
<td>Arithmetic constraints</td>
</tr>
<tr>
<td><strong>Algorithms</strong></td>
<td>Global consistency (LP)</td>
<td>Local consistency</td>
</tr>
<tr>
<td></td>
<td>Cutting planes</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Branch-and-bound</td>
<td>User-defined enumeration</td>
</tr>
<tr>
<td></td>
<td>Branch-and-cut</td>
<td></td>
</tr>
</tbody>
</table>

- Symbolic constraints \(\leadsto\) more expressivity + more efficiency
- Unifying framework for CP and IP: *Branch-and-infer*
  (Bockmayr/Kasper 98), \ldots, SCIP
Discrete Tomography

- Binary matrix with $m$ rows and $n$ columns
  - Horizontal projection numbers ($h_1, \ldots, h_m$)
  - Vertical projection numbers ($v_1, \ldots, v_n$)

- Properties
  - Horizontal convexity ($h$)
  - Vertical convexity ($v$)
  - Connectivity (polyomino) ($p$)

- Complexity (Woeginger’01)
  - Polynomial: ($\_\_\_$), ($p,v,h$)
  - NP-complete: ($p,v$), ($p,h$), ($v,h$), ($v$), ($h$), ($p$)

**IP Model**

- **Variables** $x_{ij} = \begin{cases} 0 & \text{cell}(i,j) \text{ is labeled white} \\ 1 & \text{cell}(i,j) \text{ is labeled black} \end{cases}$

- **Constraints I**: Projections
  $$\sum_{j=1}^{n} x_{ij} = h_i, \quad \sum_{i=1}^{m} x_{ij} = v_j$$

- **Constraints II**: Convexity
  $$h_i x_{ik} + \sum_{l=k+h_i}^{n} x_{il} \leq h_i, \quad v_j x_{kj} + \sum_{l=k+v_j}^{m} x_{lj} \leq v_j,$$

**IP Model (contd)**

- **Constraints III**: Connectivity
  $$\sum_{k=j}^{j+h_i-1} x_{ik} - \sum_{k=j}^{j+h_i-1} x_{i+1,k} \leq h_i - 1$$

- Various linear arithmetic models possible, e.g. convexity
- Enormous differences in size and running time, e.g. 1 day vs. < 1 sec
- Large number of constraints ($\sim 3mn$ in the above model)
Finite Domain Model

• Variables
  - $x_i$ start of horizontal convex block in row $i$, for $1 \leq i \leq m$
  - $y_j$ start of vertical convex block in column $j$, for $1 \leq j \leq n$

```
   2 1 1 3 3 3
 H
  2 1
  3 2
  3 2
```

• Domain
  - $x_i \in [1,\ldots,n-h_i+1]$, for $1 \leq i \leq m$
  - $y_j \in [1,\ldots,m-v_j+1]$, for $1 \leq j \leq n$

Conditional Propagation

• Projection/Convexity modelled by FD variables
• Compatibility of $x_i$ and $y_j$
  \[
  x_i \leq j < x_i + h_i \iff y_j \leq i < y_j + v_j
  \]

  for $1 \leq i \leq m$ and $1 \leq j \leq n$

```
   row i
```

• Conditional propagation
  
  if $x_i \leq j$ then (if $j < x_i + h_i$ then $(y_j \leq i$, $i < y_j + v_j)$)

Finite Domain Model (contd)

• Connectivity

```
   row i
   row i+1
```

• Block $i$ must start before the end of block $i+1$
  \[
  x_i \leq x_{i+1} + h_{i+1} - 1, \text{ for } 1 \leq i \leq m - 1
  \]

• Block $i+1$ must start before the end of block $i$
  \[
  x_{i+1} \leq x_i + h_i - 1, \text{ for } 1 \leq i \leq m - 1
  \]
Cumulative

2d and 3d Diffn Model