Automatic Scheduling of Hypermedia Documents with Elastic Times

Celso C. Ribeiro

Joint work with M.T. Medina and L.F.G. Soares

Computer Science Department
Catholic University of Rio de Janeiro, Brazil

Workshop on
New Trends in Scheduling Parallel and Distributed Systems
Luminy, October 1-5, 2001
Contents:

- Introduction
- Problem formulation
- Primal heuristic
- Lagrangean relaxation
- Dual heuristic
- Concluding remarks
Hypermedia systems should have some desirable features, such as:

- supporting a rich set of capabilities for each media segment (video, audio, text, pictures, animation, ...):
  - alternatives for different presentations
  - presentation duration flexibility
  - behavior changes during presentation

- allowing the explicit definition of temporal relationships among media segments with non-deterministic times

- supporting a temporal formatting algorithm that
  - considers non-determinism
  - allows on-the-fly corrections whenever unpredictable events (network delays, user interactions) occur
A presentation event is a media segment presentation

- $d^{min}$: minimum allowed presentation time
- $d^{max}$: maximum allowed presentation time
- $d^{ideal}$: duration with best presentation quality
- $c$: cost of shrinking or stretching the presentation duration
Scheduling algorithm: determination of starting times and optimal durations $d^{opt}$ (which will be tried by the formatter) so as to ensure spatial and temporal consistency

- at compile time: build an execution plan to guide the scheduling of the object presentation

- at run time: online adjustments to handle unpredictable events (network delays, user interactions)

Difficult of performing fast rescheduling in run time led the systems that make time adjustments to make them only at compile time.

Other systems: Firefly, Isis, Madeus (linear programming, PERT, constraints, ... )
Problem: find (at compile time) the optimal starting times and durations of objects to be presented, to ensure spatial and temporal consistency of the presentation while respecting limits on shrinking and stretching the ideal duration of each object.

Objectives:

1. minimize the total cost of shrinking and stretching objects;

2. minimize the number of presentation events that will have their durations different than the ideal ones;

3. spread the changes in object duration proportionally to their duration; and

4. find the shortest and longest presentation durations.

Objectives (1), (3), and (4) can be dealt with at compile time by linear programming models.
Although conflicting, objectives

(1) minimize the total cost of shrinking and stretching objects and
(2) minimize the number of presentation events that will have their durations different than the ideal ones

are the most natural.

Possible strategy to handle both of them: select the best solution with respect to (2) among all those which optimize (1). The combinatorial nature of the minimization of the number of objects whose ideal duration is modified makes objective (2) the most difficult to be tackled by combinatorial algorithms.

- Extendable mixed 0-1 LIP formulation using objective (2)
- Fast and effective primal heuristic based on variable fixations
- Lagrangean relaxation and dual heuristic
Problem formulation

Hypermedia document represented by a directed graph $G = [N, A]$:

- set $N_s = \{1, \ldots, n\}$ of synchronization points where presentation events either start or stop

- set $N = \{0\} \cup \{1, \ldots, n\} \cup \{n + 1\}$ of nodes, where $0$ and $n + 1$ denote respectively the initial and the final nodes

- set $A$ of arcs:
  - presentation events are arcs $(i, j)$ with both extremities $i, j \in \{1, \ldots, n\}$ and durations $d_{ij}^{\text{min}}$, $d_{ij}^{\text{ideal}}$, and $d_{ij}^{\text{max}}$
  - dummy initial arcs $(0, j)$ with $j \in \{1, \ldots, n\}$ and durations $d_{0j}^{\text{min}} = 0$, $d_{0j}^{\text{ideal}} = 0$, and $d_{0j}^{\text{max}} = \infty$
  - dummy final arcs $(i, n + 1)$ with $i \in \{1, \ldots, n\}$ and durations $d_{i,n+1}^{\text{min}} = 0$, $d_{i,n+1}^{\text{ideal}} = 0$, and $d_{i,n+1}^{\text{max}} = \infty$
Variables:

- continuous variable $t_i \geq 0$, for every $i \in N$
  - time in which synchronization takes place at $i \in N_s$
  - presentation starting time: $t_0 = 0$
  - end of presentation: $t_{n+1}$

- continuous variable $x_{ij} \in [d_{ij}^{min}, d_{ij}^{max}]$, for every $(i, j) \in A$
  - duration of each presentation (or dummy) event $(i, j) \in A$

- binary variable $y_{ij} \in \{0, 1\}$, for every event $(i, j) \in A$
  - $y_{ij} = 1$, if the ideal duration of event $(i, j)$ is modified
  - $y_{ij} = 0$, otherwise

- artificial costs $c_{ij}$ associated to every arc $(i, j) \in A$
  - $c_{ij} = 1$, if $(i, j)$ is a presentation event
  - $c_{ij} = 0$, otherwise (i.e., $(i, j)$ is a dummy event)
**Formulation**: mixed 0-1 integer programming problem ASHD

\[ z^* = \min \sum_{(i,j) \in A} c_{ij} \cdot y_{ij} \]

with:

\[ t_i - t_j + x_{ij} = 0 \quad \forall (i, j) \in A \quad (1) \]

\[ x_{ij} - b_{ij} \cdot y_{ij} \leq d_{ij}^{\text{ideal}} \quad \forall (i, j) \in A \quad (2) \]

\[ -x_{ij} + a_{ij} \cdot y_{ij} \leq -d_{ij}^{\text{ideal}} \quad \forall (i, j) \in A \quad (3) \]

\[ y_{ij} \in \{0, 1\} \quad \forall (i, j) \in A \quad (4) \]

\[ t_i \geq 0 \quad \forall i \in N \setminus \{0\} \]

\[ t_0 = 0 \]

\[ \begin{cases} a_{ij} = d_{ij}^{\text{min}} - d_{ij}^{\text{ideal}} \\ b_{ij} = d_{ij}^{\text{max}} - d_{ij}^{\text{ideal}} \end{cases} \]
Formulation: mixed 0-1 integer programming problem ASHD

\[
  z^* = \min \sum_{(i,j) \in A} c_{ij} \cdot y_{ij}
\]

with:

\[
  t_i - t_j + x_{ij} = 0 \quad \forall (i,j) \in A \quad (1)
\]

\[
  x_{ij} - b_{ij} \cdot y_{ij} \leq d_{ij}^{ideal} \quad \forall (i,j) \in A \quad (2)
\]

\[
  -x_{ij} + a_{ij} \cdot y_{ij} \leq -d_{ij}^{ideal} \quad \forall (i,j) \in A \quad (3)
\]

\[
  y_{ij} \in \{0, 1\} \quad \forall (i,j) \in A \quad (4)
\]

\[
  t_i \geq 0 \quad \forall i \in N \setminus \{0\}
\]

\[
  t_0 = 0
\]

\[
  \begin{cases}
    a_{ij} = d_{ij}^{min} - d_{ij}^{ideal} \\
    b_{ij} = d_{ij}^{max} - d_{ij}^{ideal}
  \end{cases}
\]

Open question: complexity of ASHD?
Primal heuristic PH for problem ASHD:

procedure PH
1    Solve ASHD and let \((\overline{y}, \overline{x}, \overline{t})\) be its optimal solution;
2    while \(\exists (i, j) \in A : 0 < y_{ij} < 1\) do
3        Fix to one in ASHD all variables \(y_{ij}\) such that \(\overline{y}_{ij} = 1\);
4        Fix to zero in ASHD all variables \(y_{ij}\) such that \(\overline{y}_{ij} = 0\);
5        Variable_selection: select and fix at either zero or
          one in ASHD a variable \(y_{st}\) with a fractional
          value \(0 < \overline{y}_{st} < 1\) in ASHD;
6    Solve ASHD and let \((\overline{y}, \overline{x}, \overline{t})\) be its optimal solution;
7    end_while;
8    return \(\overline{y}\);
end PH;
procedure Variable_selection
1   if \( \exists (i, j) \in A, i \neq 0, j \neq n + 1 : 0 < \bar{y}_{ij} \leq \alpha \) then do
2       \((s, t) \leftarrow \arg\min_{(i,j) \in A,i \neq 0,j \neq n+1} \{ \bar{y}_{ij} : 0 < \bar{y}_{ij} \leq \alpha \};\)
3       Fix \( y_{st} \) to zero in ASHD;
4       if ASHD is not feasible then fix \( y_{st} \) to one in ASHD;
5     else if \( \exists (i, j) \in A, i \neq 0, j \neq n + 1 : 1 - \alpha \leq \bar{y}_{ij} < 1 \) then do
6       \((s, t) \leftarrow \arg\max_{(i,j) \in A,i \neq 0,j \neq n+1} \{ \bar{y}_{ij} : 1 - \alpha \leq \bar{y}_{ij} < 1 \};\)
7       Fix \( y_{st} \) to one in ASHD;
8     else do
9       \((s, t) \leftarrow \arg\min_{(i,j) \in A,i \neq 0,j \neq n+1} \min\{ \bar{y}_{ij}, 1 - \bar{y}_{ij} \};\)
10      if \( y_{st} \leq 0.5 \) then do
11         Fix \( y_{st} \) to zero in ASHD;
12         if ASHD is not feasible then fix \( y_{st} \) to one in ASHD;
13     else fix \( y_{st} \) to one in ASHD;
14   return \((s, t)\);  
end Variable_selection;
Primal heuristic

UltraSPARC station, CPLEX 5.0, 35 instances, \( n = 20 \rightarrow 1000 \)

<table>
<thead>
<tr>
<th>Instance</th>
<th>Nodes</th>
<th>Arcs</th>
<th>Best value</th>
<th>Time (s)</th>
<th>Duality gap (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>PH</td>
<td>Exact</td>
<td>PH</td>
</tr>
<tr>
<td>a1</td>
<td>20</td>
<td>40</td>
<td>22</td>
<td>20</td>
<td>0.09</td>
</tr>
<tr>
<td>a2</td>
<td>20</td>
<td>80</td>
<td>62</td>
<td>62</td>
<td>0.29</td>
</tr>
<tr>
<td>a3</td>
<td>20</td>
<td>120</td>
<td>103</td>
<td>102</td>
<td>0.76</td>
</tr>
<tr>
<td>c1</td>
<td>40</td>
<td>80</td>
<td>45</td>
<td>( \leq 42 )</td>
<td>0.28</td>
</tr>
<tr>
<td>c2</td>
<td>40</td>
<td>160</td>
<td>125</td>
<td>( \leq 124 )</td>
<td>1.05</td>
</tr>
<tr>
<td>d1</td>
<td>50</td>
<td>100</td>
<td>53</td>
<td>53</td>
<td>0.33</td>
</tr>
<tr>
<td>d2</td>
<td>50</td>
<td>200</td>
<td>156</td>
<td>( \leq 154 )</td>
<td>1.54</td>
</tr>
<tr>
<td>e1</td>
<td>100</td>
<td>200</td>
<td>103</td>
<td>( \leq 102 )</td>
<td>1.29</td>
</tr>
<tr>
<td>e2</td>
<td>100</td>
<td>400</td>
<td>303</td>
<td>( \leq 304 )</td>
<td>6.74</td>
</tr>
</tbody>
</table>

* maximum size of the enumeration tree exceeded
Conclusions:

- PH found good solutions within small computation times for real-size instances.

- “Reasonable” computation times, except for very large instances (500 or more nodes).

- PH found the optimal solutions for two of seven instances for which the exact optimal is known and was off by exactly one unit for two others.

- Average and maximum gaps are respectively 54.5% and 59.6%.

- Very large duality gaps even for small instances for which PH found the optimal solution:
Conclusions:

- PH found good solutions within small computation times for real-size instances.
- “Reasonable” computation times, except for very large instances (500 or more nodes).
- PH found the optimal solutions for two of seven instances for which the exact optimal is known and was off by exactly one unit for two others.
- Average and maximum gaps are 54.5% and 59.6%.
- Very large duality gaps even for small instances for which PH found the optimal solution:
  - conjecture: solutions are quite good, but bounds are poor
  - search for polyhedral cuts or valid inequalities should be pursued if one wants to develop efficient exact methods
Lagrangean dual of problem ASHD:

- Associate non-negative dual multipliers $\lambda^+_{ij}$ and $\lambda^-_{ij}$ to constraints (2) and (3) for each arc $(i, j) \in A$:

- Lagrangean function: $L(x, y, t, \lambda^+, \lambda^-) = 
\sum_{(i, j) \in A} c_{ij} \cdot y_{ij} + \lambda^+_{ij} (x_{ij} - b_{ij} \cdot y_{ij} - d^{ideal}_{ij}) + \lambda^-_{ij} (-x_{ij} + a_{ij} \cdot y_{ij} + d^{ideal}_{ij})$

- Dual function:
\[
\begin{align*}
\text{w}(\lambda^+, \lambda^-) &= \min \quad L(x, y, t, \lambda^+, \lambda^-) \\
\text{with:} \\
| & \quad t_i - t_j + x_{ij} = 0 \quad \forall (i, j) \in A \quad (1) \\
| & \quad y_{i,j} \in \{0, 1\} \quad \forall (i, j) \in A \quad (4) \\
| & \quad d^{min}_{ij} \leq x_{ij} \leq d^{max}_{ij} \quad \forall (i, j) \in A \quad (5) \\
& \quad t_i \geq 0 \quad \forall i \in N \setminus \{0\} \\
& \quad t_0 = 0.
\end{align*}
\]
Dual heuristic DH:

- Combines the primal heuristic and the solution of the dual problem by subgradient optimization.
- Uses the multipliers obtained along the solution of the dual problem to periodically generate reduced costs $c'_{ij}$ associated with the 0-1 variables $y_{ij}$.
- Periodically applies the primal heuristic PH.
- Reduced costs $c'_{ij}$ replacing the original costs $c_{ij}$ in the primal heuristic are computed according with two different schemes:
  
  $c'_{ij} = c_{ij} + a_{ij} \cdot \lambda^{-}_{ij} - b_{ij} \cdot \lambda^{+}_{ij}$ (Lagrangean multipliers)
  
  $c'_{ij} = (1 - \hat{y}_{ij}) \cdot c_{ij}$ (primal approximation)

- Stopping criteria: total number of iterations of the subgradient algorithm equals 1200 or the relative difference between two consecutive dual function values becomes less than $10^{-4}$. 
Dual heuristic DH for problem ASHD:

\[ \begin{align*} 
\text{procedure } & \text{DH} \\
1 & \text{Apply PH to compute a feasible solution } \overline{y}; \\
2 & \text{Set the upper bound } z_{best} \leftarrow \sum_{(i,j) \in A} c_{ij} \cdot \overline{y}_{ij} \text{ and } y_{best} \leftarrow \overline{y}; \\
3 & \text{while stopping criteria not attained do} \\
4 & \quad \text{Perform a certain number of subgradient iterations;} \\
5 & \quad \text{Recompute reduced costs } c' \text{ using the current dual multipliers } (\lambda^+, \lambda^-); \\
6 & \quad \text{Apply PH using } c' \text{ to compute a new feasible solution } \overline{y}; \\
7 & \quad \text{if } \sum_{(i,j) \in A} c_{ij} \cdot \overline{y}_{ij} < z_{best} \\
8 & \quad \text{then set } z_{best} \leftarrow \sum_{(i,j) \in A} c_{ij} \cdot \overline{y}_{ij} \text{ and } y_{best} \leftarrow \overline{y}; \\
9 & \quad \text{end;} \\
10 & \text{return } \overline{y}; \\
\end{align*} \]

end DH;
### Dual heuristic (1/2)

<table>
<thead>
<tr>
<th>Instance</th>
<th>Nodes</th>
<th>Arcs</th>
<th>$z_{best}$</th>
<th>$s$</th>
<th>$z_{best}$</th>
<th>$s$</th>
<th>$z^*$</th>
</tr>
</thead>
<tbody>
<tr>
<td>a4</td>
<td>20</td>
<td>160</td>
<td>142</td>
<td>18.23</td>
<td>144</td>
<td>1.05</td>
<td>141</td>
</tr>
<tr>
<td>a5</td>
<td>20</td>
<td>200</td>
<td>178</td>
<td>24.88</td>
<td>180</td>
<td>1.72</td>
<td>176</td>
</tr>
<tr>
<td>b1</td>
<td>30</td>
<td>60</td>
<td>31</td>
<td>8.35</td>
<td>32</td>
<td>0.16</td>
<td>31</td>
</tr>
<tr>
<td>b3</td>
<td>30</td>
<td>180</td>
<td>154</td>
<td>21.87</td>
<td>155</td>
<td>1.46</td>
<td>$\leq$ 151</td>
</tr>
<tr>
<td>b4</td>
<td>30</td>
<td>240</td>
<td>209</td>
<td>31.88</td>
<td>210</td>
<td>2.64</td>
<td>$\leq$ 208</td>
</tr>
<tr>
<td>c3</td>
<td>40</td>
<td>240</td>
<td>202</td>
<td>32.83</td>
<td>203</td>
<td>2.64</td>
<td>$\leq$ 205</td>
</tr>
<tr>
<td>c4</td>
<td>40</td>
<td>320</td>
<td>280</td>
<td>49.20</td>
<td>283</td>
<td>4.71</td>
<td>$\leq$ 282</td>
</tr>
<tr>
<td>c5</td>
<td>40</td>
<td>400</td>
<td>362</td>
<td>67.19</td>
<td>363</td>
<td>7.54</td>
<td>$\leq$ 367</td>
</tr>
<tr>
<td>d2</td>
<td>50</td>
<td>200</td>
<td>154</td>
<td>28.63</td>
<td>156</td>
<td>1.54</td>
<td>$\leq$ 154</td>
</tr>
<tr>
<td>d4</td>
<td>50</td>
<td>400</td>
<td>352</td>
<td>69.73</td>
<td>354</td>
<td>7.07</td>
<td>$\leq$ 358</td>
</tr>
<tr>
<td>d5</td>
<td>50</td>
<td>500</td>
<td>451</td>
<td>95.40</td>
<td>453</td>
<td>11.62</td>
<td>$\leq$ 452</td>
</tr>
</tbody>
</table>
# Dual heuristic (2/2)

<table>
<thead>
<tr>
<th>Instance</th>
<th>Nodes</th>
<th>Arcs</th>
<th>$z_{best}$</th>
<th>s</th>
<th>$z_{best}$</th>
<th>s</th>
<th>$z^*$</th>
</tr>
</thead>
<tbody>
<tr>
<td>e3</td>
<td>100</td>
<td>600</td>
<td>505</td>
<td>142.63</td>
<td>506</td>
<td>16.69</td>
<td>≤ 520</td>
</tr>
<tr>
<td>e4</td>
<td>100</td>
<td>800</td>
<td>699</td>
<td>219.34</td>
<td>702</td>
<td>30.47</td>
<td>≤ 709</td>
</tr>
<tr>
<td>f1</td>
<td>500</td>
<td>1000</td>
<td>510</td>
<td>395.25</td>
<td>515</td>
<td>42.01</td>
<td>≤ 520</td>
</tr>
<tr>
<td>f3</td>
<td>500</td>
<td>3000</td>
<td>2539</td>
<td>3602.95</td>
<td>2541</td>
<td>782.47</td>
<td>≤ 2551</td>
</tr>
<tr>
<td>f5</td>
<td>500</td>
<td>5000</td>
<td>4514</td>
<td>5678.42</td>
<td>4519</td>
<td>1659.88</td>
<td>≤ 4553</td>
</tr>
<tr>
<td>g1</td>
<td>1000</td>
<td>2000</td>
<td>1043</td>
<td>786.30</td>
<td>1045</td>
<td>208.05</td>
<td>≤ 1177</td>
</tr>
</tbody>
</table>
Conclusions:

- The dual heuristic DH consistently improves upon the solutions found by the primal heuristic PH, finding better solutions for 17 out of the 35 test problems.

- However, the computational times observed for the dual heuristic DH are much larger.

- Need for better/faster construction algorithms and local search procedures (work-in-progress) based on variable complementation.
Concluding remarks and extensions:

- Primal heuristic: effective in finding good solutions in small processing times (real-size instances); quite promising not only in algorithmic terms, but also concerning its integration within existing document formatting systems.
Concluding remarks and extensions:

- Primal heuristic: effective in finding good solutions in small processing times (real-size instances); quite promising not only in algorithmic terms, but also concerning its integration within existing document formatting systems.

- Dual heuristic: better solutions, but at the cost of much larger computation times.
Concluding remarks and extensions:

- Primal heuristic: effective in finding good solutions in small processing times (real-size instances); quite promising not only in algorithmic terms, but also concerning its integration within existing document formatting systems.

- Dual heuristic: better solutions, but at the cost of much larger computation times.

- Establish the computational complexity of problem ASHD.
Concluding remarks and extensions:

- Primal heuristic: effective in finding good solutions in small processing times (real-size instances); quite promising not only in algorithmic terms, but also concerning its integration within existing document formatting systems.

- Dual heuristic: better solutions, but at the cost of much larger computation times.

- Establish the computational complexity of problem ASHD.

- Replace CPLEX by a customized LP algorithm to speedup the solution of the linear relaxation ASHD.
Concluding remarks and extensions:

- Primal heuristic: effective in finding good solutions in small processing times (real-size instances); quite promising not only in algorithmic terms, but also concerning its integration within existing document formatting systems.

- Dual heuristic: better solutions, but at the cost of much larger computation times.

- Establish the computational complexity of problem ASHD.

- Replace CPLEX by a customized LP algorithm to speedup the solution of the linear relaxation ASHD.

- Improved construction procedures and local search strategies.
Concluding remarks and extensions:

- Primal heuristic: effective in finding good solutions in small processing times (real-size instances); quite promising not only in algorithmic terms, but also concerning its integration within existing document formatting systems.

- Dual heuristic: better solutions, but at the cost of much larger computation times.

- Establish the computational complexity of problem ASHD.

- Replace CPLEX by a customized LP algorithm to speedup the solution of the linear relaxation ASHD.

- Improved construction procedures and local search strategies.

- Integration of such heuristics and the bicriteria optimization approach within an existing document formatter to perform compile time scheduling and even run time adjustments.
Automatic Scheduling of Hypermedia Documents with Elastic Times

Maira T. Medina
Celso C. Ribeiro
Luiz Fernando G. Soares

Computer Science Department
Catholic University of Rio de Janeiro