Solution to Exercises 8: Integer Linear Programming & Iterative Algorithms

Discussion Date: 31.05.2017

Task 1: Integer Linear Programming

Given the sequence graph \( G_S = (V_S, E_S) \) in Fig. 1.

![Sequence graph](image)

Figure 1: Sequence graph.

For the execution times of the operations assume: A multiplication operation (MULT) takes 2 time units and all other (ALU) operations take 1 time unit each. Two units of the resource type \( r_1 \) (multiplier) and two units of the resource type \( r_2 \) (ALU) are allocated.

a) Apply the ASAP and ALAP algorithms to compute the earliest \( l_i \) and the latest \( h_i \) starting time of all operations \( v_i \in V_S, i \in \{1, \ldots, 11\} \). For ALAP, assume the maximum latency \( T = 7 \). Fill in the starting times in Table 1.
Table 1: Earliest and latest starting times (Task 1a)

<table>
<thead>
<tr>
<th>$v_i$</th>
<th>$l_i$ (ASAP)</th>
<th>$h_i$ (ALAP)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$v_1$</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>$v_2$</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>$v_3$</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>$v_4$</td>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>$v_5$</td>
<td>6</td>
<td>7</td>
</tr>
<tr>
<td>$v_6$</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>$v_7$</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>$v_8$</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>$v_9$</td>
<td>3</td>
<td>7</td>
</tr>
<tr>
<td>$v_{10}$</td>
<td>1</td>
<td>6</td>
</tr>
<tr>
<td>$v_{11}$</td>
<td>2</td>
<td>7</td>
</tr>
</tbody>
</table>

b) Formulate the problem of latency minimization with restricted resources as an integer linear program (ILP). For this, you should introduce the binary variables $x_{i,t} \in \{0,1\} \ \forall v_i \in V_S$ and $\forall t \in \{t \in \mathbb{Z} \mid l_i \leq t \leq h_i\}$. $\tau(v_i)$ is used to denote the starting time of operation $v_i \in V_S$ and $\alpha(r_i)$ with $r_i \in V_R = \{\text{MULT, ALU}\}$ denotes the number of allocated resource instances. Given the above notations, write down the following equations/inequations without using the $\sum$ symbol.

i) Express the objective function of the ILP

ii) Define $\tau(v_i) \ \forall i \in \{1, \ldots, 11\}$ as a function of $x_{i,t}$, where $l_1 \leq t \leq h_1$

iii) Express all data dependencies

iv) Express all resource limitations

c) In an analogous manner try to formulate an ILP that solves the problem of cost minimization with latency limitation. Hint: We assume that the cost of a realization is the sum of the costs $c$ of the multipliers with $c(r_1) = 2$ per allocated unit, and of the ALUs with $c(r_2) = 1$ per allocated unit. For the latency bound, we choose $\bar{L} = 6$.

a) The starting times are listed in Table 1. The corresponding ASAP/ALAP schedules are depicted in Figure 2.

![Figure 2: Schedule with ASAP and ALAP](image)

b) i) Objective function:

$$\min \ L = \tau(v_n) - \tau(v_0)$$
ii) Introduction of binary variables:

\[
\begin{align*}
\tau(v_3) - \tau(v_1) &\geq 2 \\
\tau(v_4) - \tau(v_3) &\geq 2 \\
\tau(v_7) - \tau(v_6) &\geq 2 \\
\tau(v_9) - \tau(v_8) &\geq 2 \\
\tau(v_n) - \tau(v_5) &\geq 1 \\
\tau(v_n) - \tau(v_{11}) &\geq 1
\end{align*}
\]

\[
\tau(v_1), \tau(v_2), \tau(v_6), \tau(v_8), \tau(v_{10}) \geq \tau(v_0) \geq 1
\]

iv) Resource limitations:

\( t = 1: \)

\[
\begin{align*}
x_{1,1} + x_{2,1} + x_{6,1} + x_{8,1} &\leq 2 \\
x_{10,1} &\leq 2
\end{align*}
\]

\( t = 2: \)

\[
\begin{align*}
x_{1,2} + x_{2,1} + x_{2,2} + x_{6,1} + x_{6,2} + x_{8,1} + x_{8,2} &\leq 2 \\
x_{10,2} + x_{11,2} &\leq 2
\end{align*}
\]

\( t = 3: \)

\[
\begin{align*}
x_{1,2} + x_{2,2} + x_{6,2} + x_{6,3} + x_{8,2} + x_{8,3} + x_{3,3} + x_{7,3} &\leq 2 \\
x_{10,3} + x_{11,3} + x_{9,3} &\leq 2
\end{align*}
\]

\( t = 4: \)

\[
\begin{align*}
x_{6,3} + x_{8,3} + x_{8,4} + x_{3,3} + x_{3,4} + x_{7,3} + x_{7,4} &\leq 2 \\
x_{10,4} + x_{11,4} + x_{9,4} &\leq 2
\end{align*}
\]

\( t = 5: \)

\[
\begin{align*}
x_{8,4} + x_{8,5} + x_{3,4} + x_{7,4} + x_{7,5} &\leq 2 \\
x_{10,5} + x_{11,5} + x_{9,5} + x_{4,5} &\leq 2
\end{align*}
\]
\[ t = 6: \]
\[
    x_{8,5} + x_{7,5} \leq 2
\]
\[
    x_{10,6} + x_{11,6} + x_{9,6} + x_{4,6} + x_{5,6} \leq 2
\]

\[ t = 7: \]
\[
    (0 \leq 2)
\]
\[
    x_{11,7} + x_{9,7} + x_{5,7} \leq 2
\]

c) Restating the resource limitations, and introducing additional variables:
\[ t = 1: \]
\[
    x_{1,1} + x_{2,1} + x_{6,1} + x_{8,1} - \alpha(r_1) \leq 0
\]
\[
    x_{10,1} - \alpha(r_2) \leq 0
\]

Latency limitations:
\[ L = \tau(v_n) - \tau(v_0) \leq \bar{L} = 6 \]

New objective function:
\[
    \min \quad C = \alpha(r_1) \cdot c(r_1) + \alpha(r_2) \cdot c(r_2) = 2 \cdot \alpha(r_1) + \alpha(r_2)
\]

**Task 2: Iterative Algorithms**

Please answer the following questions considering the given video codec application specified as a marked graph in Figure 3.

![Diagram of Video Codec Application](attachment:image.png)

**Figure 3: Video codec marked graph representation**

**Table 2: Execution time of each function**

<table>
<thead>
<tr>
<th>( w(v_i) )</th>
<th>( \nu_1 )</th>
<th>( \nu_2 )</th>
<th>( \nu_3 )</th>
<th>( \nu_4 )</th>
<th>( \nu_5 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>10</td>
<td>10</td>
<td>5</td>
<td>5</td>
<td></td>
</tr>
</tbody>
</table>

a) Formulate all existing dependencies in Figure 3 from \( \nu_i \) to \( \nu_j \) in the form of
\[ \tau(\nu_j) - \tau(\nu_i) \geq w(\nu_i) - d_{ij} \cdot P, \]
where \( P \) is the minimum iteration interval. The execution time of each function is listed in Table 2.

b) Assuming unlimited resources and only one token on the edge between \( \nu_5 \) and \( \nu_1 \), determine the minimum iteration interval \( P \) and the latency \( L \). To justify your answer, draw the scheduling on the timeline given in Figure 4 with the dependency from \( \nu_5 \) to \( \nu_1 \) highlighted.
c) The motion estimation function ($\nu_1$) uses the result of the previous frame (See the dependency between $\nu_1$ and $\nu_5$). Let us now suppose that any arbitrary number of tokens can be inserted to reduce $P$ using functional pipelining. Then, determine the minimum number of tokens that should be added on the edge $\nu_5 \rightarrow \nu_1$ to achieve $P = 10$? To justify your answer, draw the pipelined scheduling on the timeline given in Figure 5 with the dependency from $\nu_5$ to $\nu_1$ highlighted and calculate the latency $L$ of the schedule.
a) Dependencies:
\[
\begin{align*}
\tau(\nu_2) - \tau(\nu_1) & \geq 10 \\
\tau(\nu_3) - \tau(\nu_2) & \geq 10 \\
\tau(\nu_4) - \tau(\nu_2) & \geq 10 \\
\tau(\nu_5) - \tau(\nu_4) & \geq 5 \\
\tau(\nu_1) - \tau(\nu_5) & \geq 5 - 1 \cdot P
\end{align*}
\]

b) We solve the system of inequalities of 2a) for \( P \).
\[ \Rightarrow P_{\text{min}} = 30 \]
\[ L = 30 \]

![Figure 6: Scheduling result of the video codec](image)

Figure 6: Scheduling result of the video codec

c) Now the iteration interval \( P \) is given \( (P = 10) \) and we are looking for the number of tokens \( n \). Therefore, we replace the last inequation in 2a) by \( \tau(\nu_1) - \tau(\nu_5) \geq 5 - n \cdot 10 \) and solve the new set of inequations for \( n \).
\[ \Rightarrow n_{\text{min}} = 3 \]
We have to add at least 2 tokens on the edge between \( \nu_5 \) and \( \nu_1 \).
\[ L = 30 \]

![Figure 7: Pipelined scheduling result of the video codec](image)

Figure 7: Pipelined scheduling result of the video codec