Aufgabe 1: Integer Linear Programming

Given the sequence graph in Fig. 1.

For the execution times of the operations assume: \( D_\times = 2, D_+ = 1 \). Two units of the resource type \( r_1 \) (multiplier) and two units of the resource type \( r_2 \) (ALU) are allocated.

a) Formulate the problem of latency minimization with restricted resources as an integer linear program (ILP). Hint: Compute the earliest and the latest start times of the operations using the ASAP and the ALAP algorithm, respectively.
b) Give reasons for the latency bound that you used with the ALAP algorithm. How can it be ensured in general that the latency bound is chosen such that the stated optimization model finds a valid solution to the problem of latency minimization with restricted resources?

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 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 \( L = 6 \).

---

a) Determining earliest and latest start times (Fig. 2):

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

Introduction of binary variables:

\[
\begin{align*}
x_{1,1} + x_{1,2} &= 1 & 1 \cdot x_{1,1} + 2 \cdot x_{1,2} &= \tau(v_1) \\
x_{2,1} + x_{2,2} &= 1 & 1 \cdot x_{2,1} + 2 \cdot x_{2,2} &= \tau(v_2) \\
x_{3,3} + x_{3,4} &= 1 & 3 \cdot x_{3,3} + 4 \cdot x_{3,4} &= \tau(v_3) \\
x_{4,5} + x_{4,6} &= 1 & 5 \cdot x_{4,5} + 6 \cdot x_{4,6} &= \tau(v_4) \\
x_{5,6} + x_{5,7} &= 1 & 6 \cdot x_{5,6} + 7 \cdot x_{5,7} &= \tau(v_5) \\
x_{6,1} + x_{6,2} + x_{6,3} &= 1 & 1 \cdot x_{6,1} + 2 \cdot x_{6,2} + 3 \cdot x_{6,3} &= \tau(v_6) \\
x_{7,3} + x_{7,4} + x_{7,5} &= 1 & 3 \cdot x_{7,3} + 4 \cdot x_{7,4} + 5 \cdot x_{7,5} &= \tau(v_7) \\
x_{8,1} + \ldots + x_{8,5} &= 1 & 1 \cdot x_{8,1} + \ldots + 5 \cdot x_{8,5} &= \tau(v_8) \\
x_{9,3} + \ldots + x_{9,7} &= 1 & 3 \cdot x_{9,3} + \ldots + 7 \cdot x_{9,7} &= \tau(v_9) \\
x_{10,1} + \ldots + x_{10,6} &= 1 & 1 \cdot x_{10,1} + \ldots + 6 \cdot x_{10,6} &= \tau(v_{10}) \\
x_{11,2} + \ldots + x_{11,7} &= 1 & 2 \cdot x_{11,2} + \ldots + 7 \cdot x_{11,7} &= \tau(v_{11}) \\
\end{align*}
\]

Data dependencies:

\[
\begin{align*}
\tau(v_3) - \tau(v_1) &\geq 2 & \tau(v_3) - \tau(v_2) &\geq 2 \\
\tau(v_4) - \tau(v_3) &\geq 2 & \tau(v_5) - \tau(v_4) &\geq 1 \\
\tau(v_7) - \tau(v_6) &\geq 2 & \tau(v_5) - \tau(v_7) &\geq 2 \\
\tau(v_9) - \tau(v_8) &\geq 2 & \tau(v_11) - \tau(v_{10}) &\geq 1 \\
\tau(v_n) - \tau(v_5) &\geq 1 & \tau(v_n) - \tau(v_9) &\geq 1 \\
\tau(v_n) - \tau(v_{11}) &\geq 1 & 
\end{align*}
\]

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

\( t = 1 \):
\[
\begin{align*}
  x_{1,1} + x_{2,1} + x_{6,1} + x_{8,1} & \leq 2 \\
  x_{10,1} & \leq 2
\end{align*}
\]

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

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

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

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

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

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

Objective function:
\[
\min \quad L = \tau(v_n) - \tau(v_0)
\]

b) Deadline \( \bar{L} = 7 \): Looking at the ASAP schedule, it is clear that there is no solution with \( L = 6 \) with only two multipliers.

However, in principle you need to use an algorithm which considers resource constraints (e.g. List Scheduling) and find \( \bar{L} \). It will guarantee you that there exists at least one solution for this \( \bar{L} \) even if not optimal.

c) Restating the resource limitations, and introducing additional variables:

\( t = 1 \):
\[
\begin{align*}
  x_{1,1} + x_{2,1} + x_{6,1} + x_{8,1} - \alpha(r_1) & \leq 0 \\
  x_{10,1} - \alpha(r_2) & \leq 0
\end{align*}
\]

[\ldots]

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

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

Please answer the following questions considering the given video codec application specified as a marked graph:

![Graph representation](image)

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

**Table 1: Execution time of each function**

<table>
<thead>
<tr>
<th>Function</th>
<th>$ν_1$</th>
<th>$ν_2$</th>
<th>$ν_3$</th>
<th>$ν_4$</th>
<th>$ν_5$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$w(ν_i)$</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>5</td>
<td>5</td>
</tr>
</tbody>
</table>

(a) Formulate all existing dependencies in Figure 3 from $ν_i$ to $ν_j$ in the form of

$$\tau(ν_j) - \tau(ν_i) \geq w(ν_i) - d_{ij} \times P,$$

where $P$ is the minimum iteration interval.

(b) Assuming unlimited resource, determine the minimum iteration interval $P$ and the latency $L$ without pipelining. To justify your answer, draw the scheduling on the timeline given in Figure 4 with the dependency from $ν_5$ to $ν_1$ highlighted.

![Timeline](image)

**Figure 4: Scheduling result of the video codec**

(c) The motion estimation function ($ν_1$) uses the result of the previous frame (See the dependency between $ν_1$ and $ν_5$). Let us now suppose that any arbitrary number of tokens can be inserted to reduce $P$ using functional pipelining. Then, how many tokens should be added on the dependency $ν_5 → ν_1$ to achieve $P = 10$? To justify your answer, draw the pipelined scheduling on the timeline given in Figure 5 with the dependency from $ν_5$ to $ν_1$ highlighted and calculate the latency $L$ of the schedule.
a) Dependencies:
\[
\tau(\nu_2) - \tau(\nu_1) \geq 10 \\
\tau(\nu_3) - \tau(\nu_2) \geq 10 \\
\tau(\nu_4) - \tau(\nu_2) \geq 10 \\
\tau(\nu_5) - \tau(\nu_4) \geq 5 \\
\tau(\nu_1) - \tau(\nu_5) \geq 5 - 1 \times P
\]

b) \(P=L=30\).

c) Two more tokens should be added and \(L=30\).