Design Technologies for Integrated Systems – EPFL

Exercise 2 — Solutions for Problem 3 and Problem 5
October 4th, 2018

Problem 3

Given the sequencing graph in Fig.1:

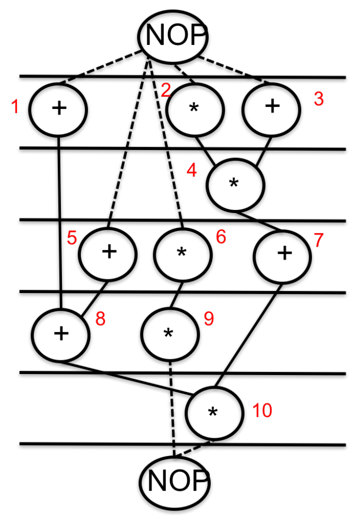

Figure 1: Sequencing graph.

Write the ILP (Integer Linear Program) equations that minimize resource usage given a latency bound. Assume that all operations complete in one unit of time, the multiplier takes 3 units of area and the adder one unit of area. The latency bound is 5 cycles.

Solution

For the vertexes, we use the labels (red) from Fig. 1. The first step consists of scheduling the sequencing graph with ASAP (Fig. 2) and ALAP (Fig. 3). We use a latency bound of 5 for ALAP (as this is the constraint). ASAP and ALAP are

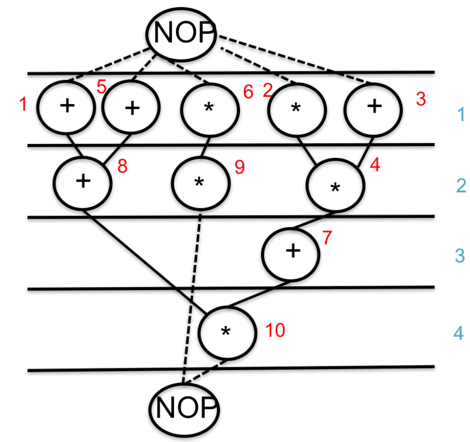

Figure 2: Scheduled sequencing graph with ASAP.

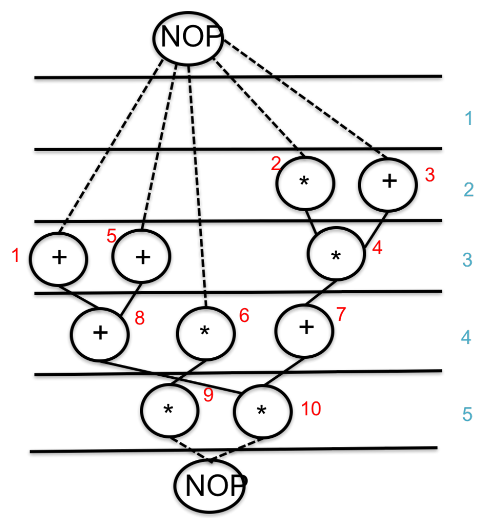

Figure 3: Scheduled sequencing graph with ALAP, $\lambda = 5$.

needed in order to evaluate the time frames for each single node. According to the results in Fig. 2 and 3, we report the time frames of each vertex in Table 1.

The goal of this exercise is the minimization of the resources usage, given a latency bound = 5. The multiplier takes 3 units of area and the adder one unit of area, so the **objective function to minimize is**:
$c^T \cdot \mathbf{a} = 3 \cdot a_1 + 1 \cdot a_2$.

Table 1: Time frames obtained from ASAP and ALAP (latency bound = 5)

Vtx	t_S	t_L	L1	L2	L3	L4	L5
1	1	3	✓	✓	✓	×	×
2	1	2	✓	✓	×	×	×
3	1	2	✓	✓	×	×	×
4	2	3	×	✓	✓	×	×
5	1	3	✓	✓	✓	×	×
6	1	4	✓	✓	✓	✓	×
7	3	4	×	×	✓	✓	×
8	2	4	×	✓	✓	✓	×
9	2	5	×	✓	✓	✓	✓
10	4	5	×	×	×	✓	✓

Once the minimization objective is stated, we need to add the constraints. We can write the sets of constraints for ILP:

1. Constraints on start time: (one for each vertex)

$x_{0,1} = 1;$
$x_{1,1} + x_{1,2} + x_{1,3} = 1;$
$x_{2,1} + x_{2,2} = 1;$
$x_{3,1} + x_{3,2} = 1;$
$x_{4,2} + x_{4,3} = 1;$
$x_{5,1} + x_{5,2} + x_{5,3} = 1;$
$x_{6,1} + x_{6,2} + x_{6,3} + x_{6,4} = 1;$
$x_{7,3} + x_{7,4} = 1;$
$x_{8,2} + x_{8,3} + x_{8,4} = 1;$
$x_{9,2} + x_{9,3} + x_{9,4} + x_{9,5} = 1;$
$x_{10,4} + x_{10,5} = 1;$
$x_{n,5} + x_{n,6} = 1;$

2. Constraints on sequencing: (one for each edge)

v_1:
$x_{1,1} + 2x_{1,2} + 3x_{1,3} \geq 1;$
$2x_{8,2} + 3x_{8,3} + 4x_{8,4} \geq x_{1,1} + 2x_{1,2} + 3x_{1,3} + 1;$

v_2:

$x_{2,1} + 2x_{2,2} \geq 1;$
$2x_{4,2} + 3x_{4,3} \geq x_{2,1} + 2x_{2,2} + 1;$

v_3:
$x_{3,1} + 2x_{3,2} \geq 1;$
$2x_{4,2} + 3x_{4,3} \geq x_{3,1} + 2x_{3,2} + 1;$

v_4:
$3x_{7,3} + 4x_{7,4} \geq 2x_{4,2} + 3x_{4,3} + 1;$

v_5:
$x_{5,1} + 2x_{5,2} + 3x_{5,3} \geq 1;$
$2x_{8,2} + 3x_{8,3} + 4x_{8,4} \geq x_{5,1} + 2x_{5,2} + 3x_{5,3} + 1;$

v_6:
$x_{6,1} + 2x_{6,2} + 3x_{6,3} + 4x_{6,4} \geq 1;$
$2x_{9,2} + 3x_{9,3} + 4x_{9,4} + 5x_{9,5} \geq x_{6,1} + 2x_{6,2} + 3x_{6,3} + 4x_{6,4} + 1;$

v_7:
$4x_{10,4} + 5x_{10,5} \geq 3x_{7,3} + 4x_{7,4} + 1;$

v_8:
$4x_{10,4} + 5x_{10,5} \geq 2x_{8,2} + 3x_{8,3} + 4x_{8,4} + 1;$

v_9:
$5x_{n,5} + 6x_{n,6} \geq 2x_{9,2} + 3x_{9,3} + 4x_{9,4} + 5x_{9,5} + 1;$

v_{10}:
$5x_{n,5} + 6x_{n,6} \geq 4x_{10,4} + 5x_{10,5} + 1;$

3. Constraints on resource bound (assume a_1 is the number of multipliers while a_2 is the number of adders):

L1:
$x_{2,1} + x_{6,1} \leq a_1$; [multipliers]
$x_{1,1} + x_{3,1} + x_{5,1} \leq a_2$; [adders]

L2:
$x_{2,2} + x_{4,2} + x_{6,2} + x_{9,2} \leq a_1$; [multipliers]
$x_{1,2} + x_{3,2} + x_{5,2} + x_{8,2} \leq a_2$; [adders]

L3:
$x_{4,3} + x_{6,3} + x_{9,3} \leq a_1$; [multipliers]
$x_{1,3} + x_{5,3} + x_{7,3} + x_{8,3} \leq a_2$; [adders]

L4:
$x_{6,4} + x_{9,4} + x_{10,4} \leq a_1$; [multipliers]
$x_{7,4} + x_{8,4} \leq a_2$; [adders]

L5:
$x_{9,5} + x_{10,5} \leq a_1$; [multipliers]

4. Latency constraint:

$5x_{n,5} + 6x_{n,6} \leq 6$;

Problem 5

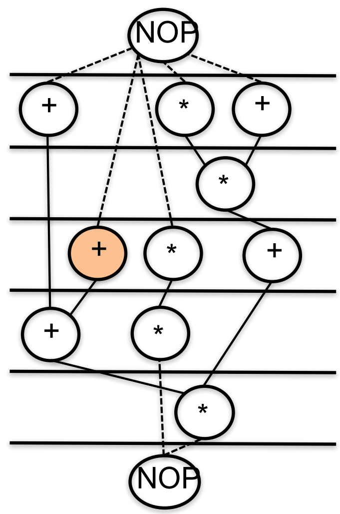

Figure 4: Sequencing graph.

Given the sequencing graph in Fig. 4, use *force directed scheduling* to select the time step at which the colored operation should be scheduled to reduce concurrency. Assume all operations have unit delays and consider an upper bound on the latency = 5.

Table 2: Time frames obtained from ASAP and ALAP (latency bound = 5), mobility μ and probabilities $p_i(l)$

Vtx	t_S	t_L	L1	L2	L3	L4	L5	μ	$p_i(1)$	$p_i(2)$	$p_i(3)$	$p_i(4)$	$p_i(5)$	type
1	1	3	✓	✓	✓	×	×	2	1/3	1/3	1/3	0	0	adder
2	1	2	✓	✓	×	×	×	1	1/2	1/2	0	0	0	multiplier
3	1	2	✓	✓	×	×	×	1	1/2	1/2	0	0	0	adder
4	2	3	×	✓	✓	×	×	1	0	1/2	1/2	0	0	multiplier
5	1	3	✓	✓	✓	×	×	2	1/3	1/3	1/3	0	0	adder
6	1	4	✓	✓	✓	✓	×	3	1/4	1/4	1/4	1/4	0	multiplier
7	3	4	×	×	✓	✓	×	1	0	0	1/2	1/2	0	adder
8	2	4	×	✓	✓	✓	×	2	0	1/3	1/3	1/3	0	adder
9	2	5	×	✓	✓	✓	✓	3	0	1/4	1/4	1/4	1/4	multiplier
10	4	5	×	×	×	✓	✓	1	0	0	0	1/2	1/2	multiplier

The first thing we need to do when solving the force directed scheduling is to evaluate time frames, mobility, probability and type distribution from the ASAP and ALAP graphs (latency bound of 5 for ALAP). The ASAP and ALAP graphs are reported in Fig 2 and Fig 3, respectively. Table 2 shows times frames, probabilities, and mobilities. The notation is the same as the ones used at page 211 of the book. We use here the same labeling of the vertexes from previous Problem.

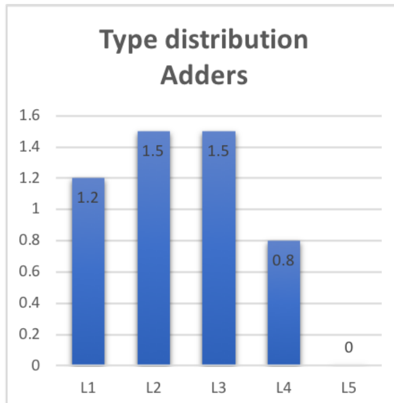

Figure 5: Type distribution for adders

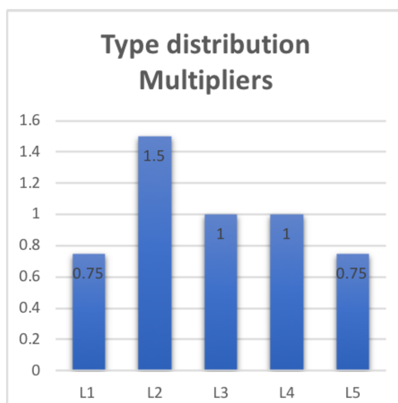

Figure 6: Type distribution for multipliers

The type distributions are reported in Fig. 5 and Fig. 6. They need to be calculated both for adders and multipliers. For each specific type of resources, they are evaluated as the sum of the probabilities of operations that can be implemented at a given time level. For instance, $q_{adder}(1) = p_1(1) + p_3(1) + p_5(1) = 1/3 + 1/2 + 1/3 = 1.2$. More details at page 212.

We need to evaluate the force of node 5. The force needs to be evaluated for each level that belongs to vertex 5's time frame. Thus: L_1, L_2, L_3. For each level, both *self-force* and *predecessor/successor* force need to be considered.

1. Self-force

- L_1: $self\_force(5,1) = q_{adder}(1)(1-p_5(1)) + q_{adder}(2)(0-p_5(2)) + q_{adder}(3)(0-p_5(3)) = 1.2(1-1/3) + 1.5(0-1/3) + 1.5(0-1/3) = -0.2$
- L_2: $self\_force(5,2) = q_{adder}(1)(0-p_5(1)) + q_{adder}(2)(1-p_5(2)) + q_{adder}(3)(0-p_5(3)) = 0.27$
- L_3: $self\_force(5,3) = q_{adder}(1)(0-p_5(1)) + q_{adder}(2)(0-p_5(2)) + q_{adder}(3)(1-p_5(3)) = 0.27$

2. Predecessor/successor-force (PS-force)

- L_1: It is not influencing other nodes, thus:
  $PS\_force(5,1) = 0$
- L_2: It is influencing node 8 time frame:
  $PS\_force(5,2) = 1/2(q_{adder}(3)+q_{adder}(4))-1/3(q_{adder}(2)+q_{adder}(3)+q_{adder}(4)) = 1/2(1.5+0.8) - 1/3(1.5+1.5+0.8) = -0.12$
- L_3: It is influencing both node 8 and node 10 time frame ($PS\_force(5,3)$ is the sum of these two components):
  $PS\_force(5,3) = 1(q_{adder}(4))-1/3(q_{adder}(2)+q_{adder}(3)+q_{adder}(4))+1(q_{mult}(5))-1/2(q_{mult}(4)+q_{mult}(5)) = 1(0.8) - 1/3(1.5+1.5+0.8) + 0.75 - 1/2(1+0.75) = -0.47 - 0.125 = -0.6$

The total force of node 5 in each level is given by the sum of its self and PS force.
Total force:

- L_1: $total\_force(5,1) = self\_force(5,1) + PS\_force(5,1) = -0.2 + 0 = -0.2$
- L_2: $total\_force(5,2) = self\_force(5,2) + PS\_force(5,2) = 0.27 - 0.12 = 0.15$

- L_3: $total\_force(5,3) = self\_force(5,3) + PS\_force(5,3) = 0.27 - 0.6 = -0.33$

The smallest force is when node 5 is scheduled at step 3, thus we can conclude that scheduling node 5 at step 3 will reduce the concurrency.