

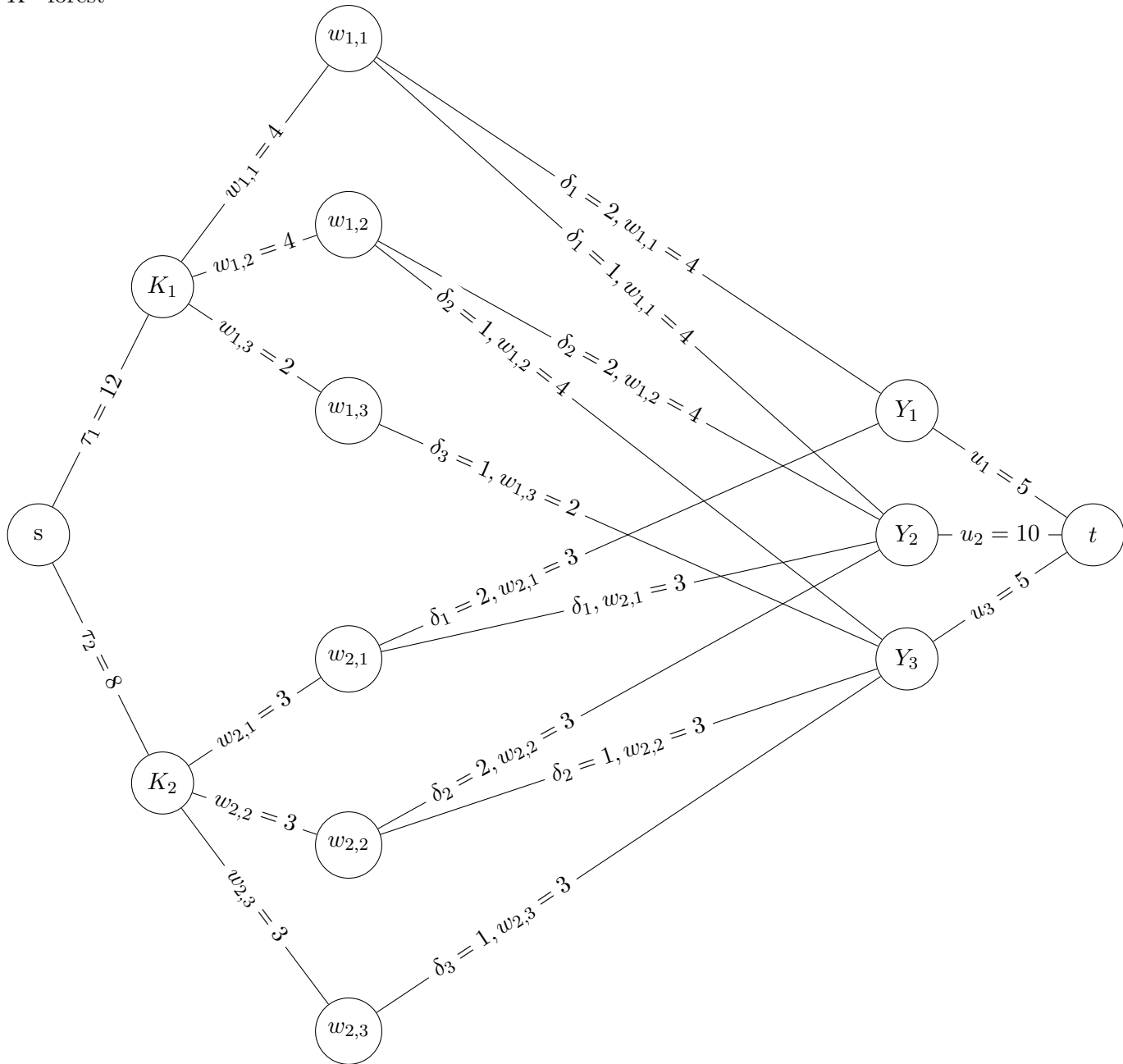
# COMP2007 - Assignment 4

Matthew Watson  
SID: 440267858

1. Consider the case when  $Y = 3, k = 2, \delta_1 = \delta_2 = 2$  and  $\delta_3 = 1$

- (a) Formulate the problem of determining a schedule with maximum number of Christmas trees sold as a network flow problem.

$Y$ =year  
 $K$ =forest



(b) **Argue why your algorithm is correct.**

In this case, we have several constraints which will contribute to the design of our flow network, these are:

- $k$ : the number of forests, in this case, 2.
- $Y$ : the number of years, in this case, 3.
- $\tau_i$  (added as per overall specification): the maximum number of trees that can be harvested from forest  $K_i$  over the entire  $Y$  years. In this case, we say  $\tau_1 = 12$  and  $\tau_2 = 8$ , meaning that we are restricted over the entire period to harvesting 12 trees from forest  $K_1$  and 8 trees from forest  $K_2$ .
- $w_{i,j}$  (added as per overall specification): the maximum number of trees that can be harvested from forest  $K_i$  in year  $Y_j$ . In this case we have  $w_{1,1} = w_{1,2} = 4, w_{1,3} = 2$  and  $w_{2,1} = w_{2,2} = w_{2,3} = 3$ .
- $\delta_j$ : the lifetime of a tree harvested in year  $Y_j$ , such that  $\delta_1 = 3$  means a tree harvested in year 1 can only be sold in year 1,2 or 3. In this case, we have  $\delta_1 = \delta_2 = 2$  and  $\delta_3 = 1$
- $u_i$  (added as per overall specification): The maximum number of trees that can be sold in year  $Y_i$ . This does not necessarily restrict the number of trees harvested, however, given for that year the  $\delta_i$  value is greater than 1. In this case, we have  $u_1 = u_2 = 5$  and  $u_3 = 10$ .

Given the above, we can construct this as a network flow problem in the following order from source  $s$  to sink  $t$ .

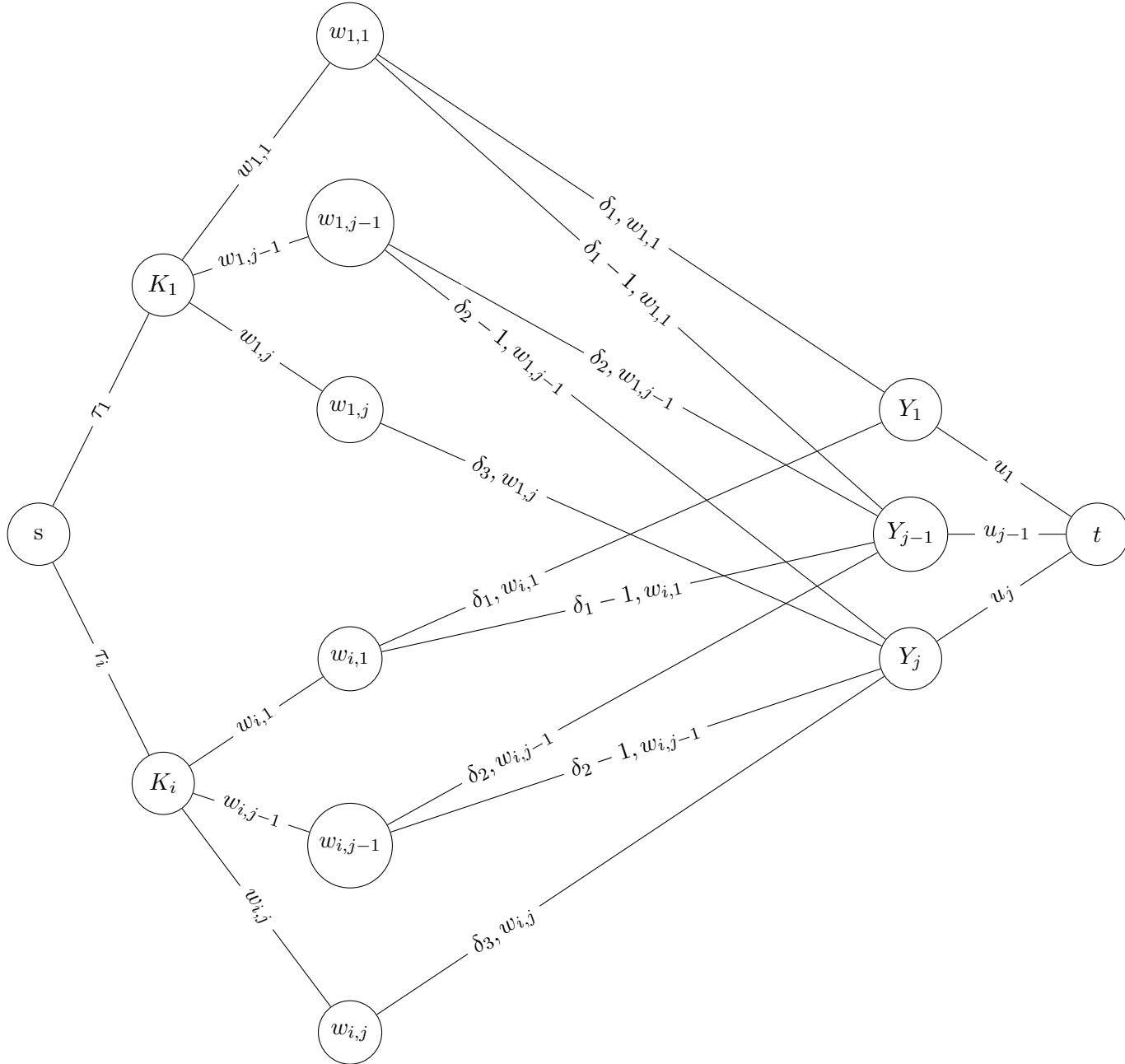
- i. From our source, for every forest  $i$ , add an edge to a respective  $k_1, \dots, k_i$  with weight  $\tau_i$ . This enforces the maximum overall harvest for each forest represented by  $\tau_k$ . Given this is a restraint on our entire harvest across all years, we place this as the first constraint from the source to ensure no more than this amount of trees can enter our network flow.
- ii. From each  $k_i$  node (each  $i$  forest), add an edge to a respective  $w_{i,j}$  node for each year  $j$  with edge weight  $w_{i,j}$ : the number of Christmas trees maturing in forest  $i$ , year  $j$ . This step ensures that we are bounded by the maximum tree harvest for each forest in each year whilst not exceeding for any forest, the maximum amount of harvest across all years.
- iii. From each  $w_{i,j}$  node, add an edge to a respective  $y_i$  node with edge weight  $w_{i,j}$ . This concentrates our maximum yield from a given field and year to their respective years.
- iv. From each  $w_{i,j}$  node, for its given  $\delta_i$  value  $z$ , add an edge to its respective  $y_{i+(z-1)}$  node with edge weight  $w_{i,j}$ . That is, for a given  $w_{i,j}$  node with  $\delta_i$  value 3, add an edge to  $y_2$  and  $y_3$  with edge weight  $w_{i,j}$ . This allows any excess harvest from year  $i$  that was held off market to prevent flooding it, to be used in following years bounded by their expected lifetime given by  $\delta_i$  and the following final constraint:.
- v. For each  $y_i$  node, add an edge to  $t$  with edge weight  $u_i$ . This bound ensures the final constraint before going to market that no more than  $u_i$  trees are sold in any given year  $y_i$  to prevent flooding the market. Given the previous step, however, any excess can be redistributed into later years to achieve a maximum number of trees sold schedule.

With the above network flow complete, running Ford-Fulkerson's algorithm to find the max flow will return the maximum flow of this network and thus the value of an optimal cutting schedule with maximum trees sold over the entire  $Y$  years.

2. Generalise your solution to  $k$  forests,  $Y$  years and variable tree lifespans.

- (a) Formulate the problem of determining a schedule with maximum profit (maximum number of Christmas trees sold) as a network flow problem for a given  $Y, k$  and  $\delta_1, \dots, \delta_Y$ .

$j$ =year number  
 $i$ =forest number  
 $Y$ =year  
 $K$ =forest



- (b) Argue why your formulation is correct.

In this case, we have several constraints which will contribute to the design of our flow network, these are:

- $i$ : the number of forests
- $j$ : the number of years
- $\tau_i$ : the maximum number of trees that can be harvested from forest  $K_i$  over the entire  $j$  years.
- $w_{i,j}$ : the maximum number of trees that can be harvested from forest  $K_i$  in year  $Y_j$ .
- $\delta_j$ : the lifetime of a tree harvested in year  $Y_j$ , such that  $\delta_1 = 3$  means a tree harvested in year 1 can only be sold in year 1,2 or 3.
- $u_i$ : The maximum number of trees that can be sold in year  $Y_i$ . This does not necessarily restrict the number of trees harvested, however, given for that year the  $\delta_i$  value is greater than 1.

Given the above, we can construct this as a network flow problem in the following order from source  $s$  to sink  $t$ .

- i. From our source, for every forest  $i$ , add an edge to a respective  $k_1, \dots, k_i$  with weight  $\tau_i$ . This enforces the maximum overall harvest for each forest represented by  $\tau_k$ . Given this is a restraint on our entire harvest across all years, we place this as the first constraint from the source to ensure no more than this amount of trees can enter our network flow.
- ii. From each  $k_i$  node (each  $i$  forest), add an edge to a respective  $w_{i,j}$  node for each year  $j$  with edge weight  $w_{i,j}$ : the number of Christmas trees maturing in forest  $i$ , year  $j$ . This step ensures that we are bounded by the maximum tree harvest for each forest in each year whilst not exceeding for any forest, the maximum amount of harvest across all years.
- iii. From each  $w_{i,j}$  node, add an edge to a respective  $y_i$  node with edge weight  $w_{i,j}$ . This concentrates our maximum yield from a given field and year to their respective years.
- iv. From each  $w_{i,j}$  node, for its given  $\delta_i$  value  $z$ , add an edge to its respective  $y_{i+(z-1)}$  node with edge weight  $w_{i,j}$ . That is, for a given  $w_{i,j}$  node with  $\delta_i$  value 3, add an edge to  $y_2$  and  $y_3$  with edge weight  $w_{i,j}$ . This allows any excess harvest from year  $i$  that was held off market to prevent flooding it, to be used in following years bounded by their expected lifetime given by  $\delta_i$  and the following final constraint:.
- v. For each  $y_i$  node, add an edge to  $t$  with edge weight  $u_i$ . This bound ensures the final constraint before going to market that no more than  $u_i$  trees are sold in any given year  $y_i$  to prevent flooding the market. Given the previous step, however, any excess can be redistributed into later years to achieve a maximum number of trees sold schedule.

With the above network flow complete, running Ford-Fulkerson's algorithm to find the max flow will return the maximum flow of this network and thus the value of an optimal cutting schedule with maximum trees sold over the entire  $Y$  years.

(c) **Prove an upper bound on the time complexity of your algorithm.**

After reading in our data, we begin the algorithm. Our first for loop runs  $k$  times. The for loop within that runs  $Y$  times. The for loop within that runs  $\delta_i$  times. Overall this results in a run time of  $O(kY\delta_i)$