

Flow Networks

CSCI 532

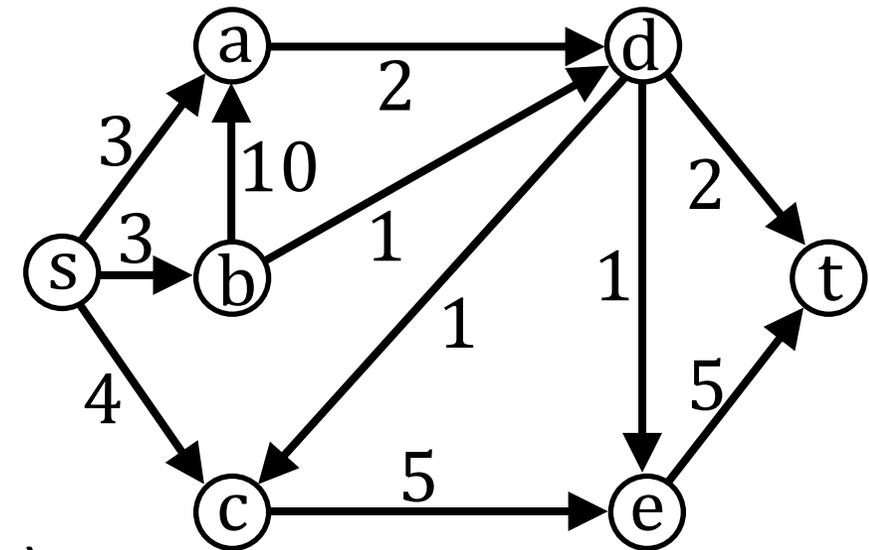
Flow Network

Maximum Flow Problem:

Given a flow network, find the maximum possible value of flow.

Flow Network:

- Directed-edge graph, $G = (V, E)$.
- Finite positive edge capacity, c_e .
- Single source, s , without input edges.
- Single sink, t , without output edges.



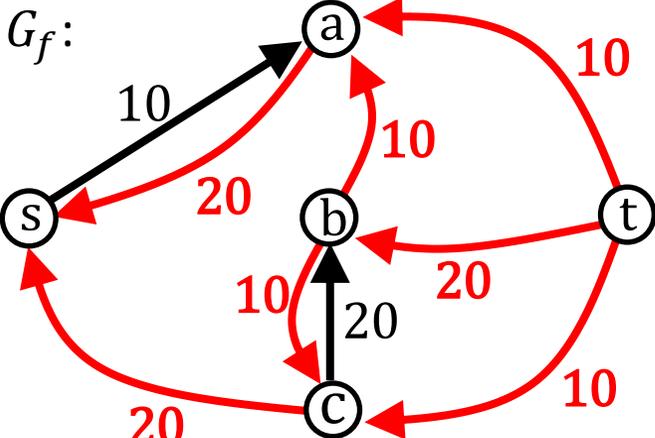
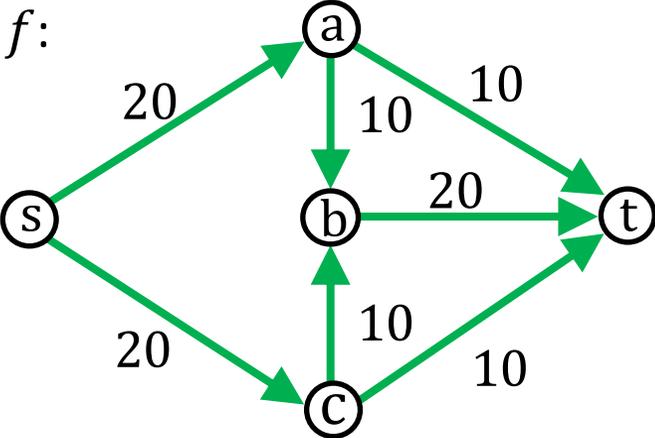
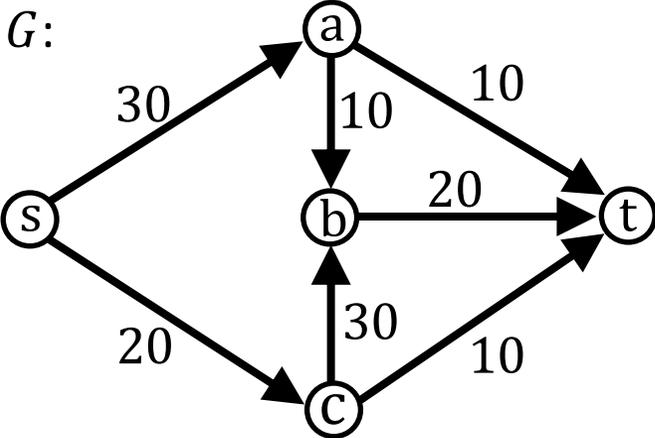
An $s - t$ flow is a function $f: E \rightarrow \mathbb{R}^+$ such that:

- $0 \leq f(e) \leq c_e, \forall e \in E$. (capacity constraint)
- $\sum_{e \in \text{input}(v)} f(e) = \sum_{e \in \text{output}(v)} f(e), \forall v \in V \setminus \{s, t\}$.

(conservation of flow constraint: “Everything that goes into a node has to come out, except for s and t ”)

- Value of flow = $val(f) = \sum_{e \in \text{output}(s)} f(e) = \sum_{e \in \text{input}(t)} f(e)$

Ford-Fulkerson



Max-Flow(G)

$f(e) = 0$ for all e in G

while s - t path in G_f exists

$P =$ simple s - t path in G_f

$f' =$ augment(f, P)

$f = f'$

$G_f = G_{f'}$

return f

augment(f, P)

$b =$ bottleneck(P, f)

for each edge (u, v) in P

if (u, v) is a back edge

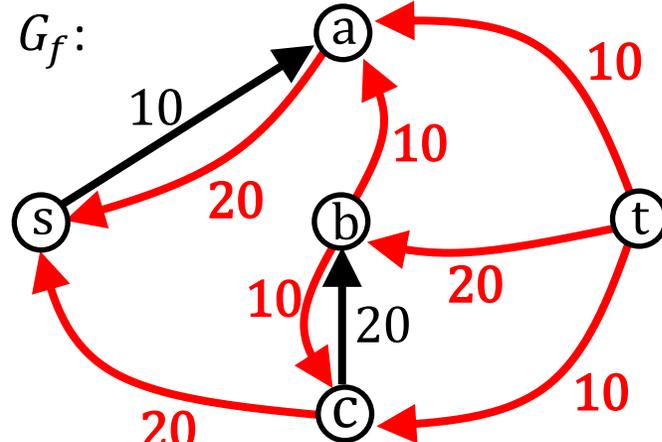
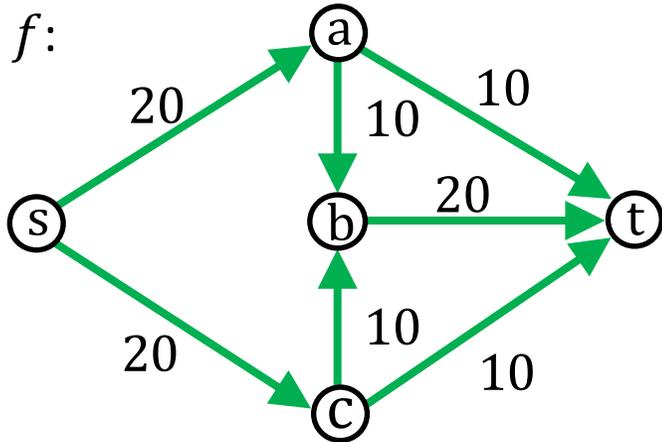
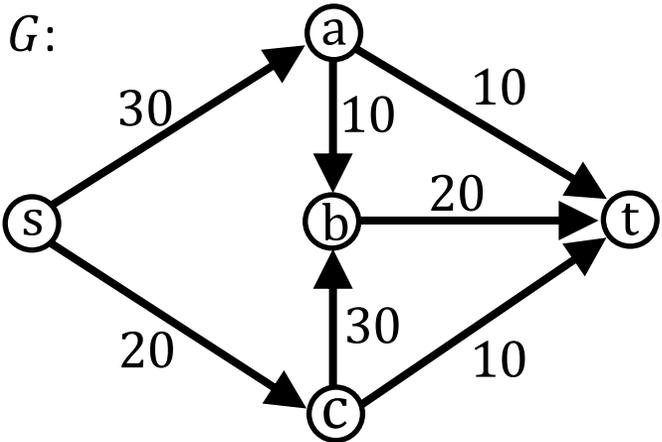
$f((v, u)) -= b$

else

$f((u, v)) += b$

return f

Ford-Fulkerson



Max-Flow(G)

```

f(e) = 0 for all e in G
while s-t path in G
    P = simple s-t path
    f' = augment(f, P)
    f = f'
    G_f = G_f'
return f
    
```

augment(f, P)

Need to show:

1. Validity.
2. Running time.
3. Finds max flow.

```

bottleneck(P, f)
for each edge (u, v) in P
    if (u, v) is a back edge
        b = min(f((v, u)), capacity(v, u))
    else
        b = min(capacity(u, v), f((u, v)))
    f((u, v)) += b
    f((v, u)) -= b
return f
    
```

Assumptions for today:

1. We can find optimal (and valid) flows for a flow network.
2. If the capacities of a flow network are integer-valued, the max flow will be integer-valued as well.

Work Scheduling

Problem: We need to make holiday schedules for our employees. Each employee has a set of holidays that they are able to work. Each employee should work at most 3 holidays. We want to maximize the number of holidays covered.

Work Scheduling

Problem: We need to make holiday schedules for our employees. Each employee has a set of holidays that they are able to work. Each employee should work at most 3 holidays. We want to maximize the number of holidays covered.



Apollo 13 Filter Problem:

“We need to fit this into the hole for this, using nothing but that”

Work Scheduling

Problem: We need to make holiday schedules for our employees. Each employee has a set of holidays that they are able to work. Each employee should work at most 3 holidays. We want to maximize the number of holidays covered.



Apollo 13 Filter Problem:

“We need to fit **this** into the ~~hole for this~~, using nothing but ~~that~~”

**work scheduling
problem**

**Max Flow
solver**

**valid flow network
components**

Work Scheduling

Problem: We need to make holiday schedules for our employees. Each employee has a set of holidays that they are able to work. Each employee should work at most 3 holidays. We want to maximize the number of holidays covered.

Algorithm:

??

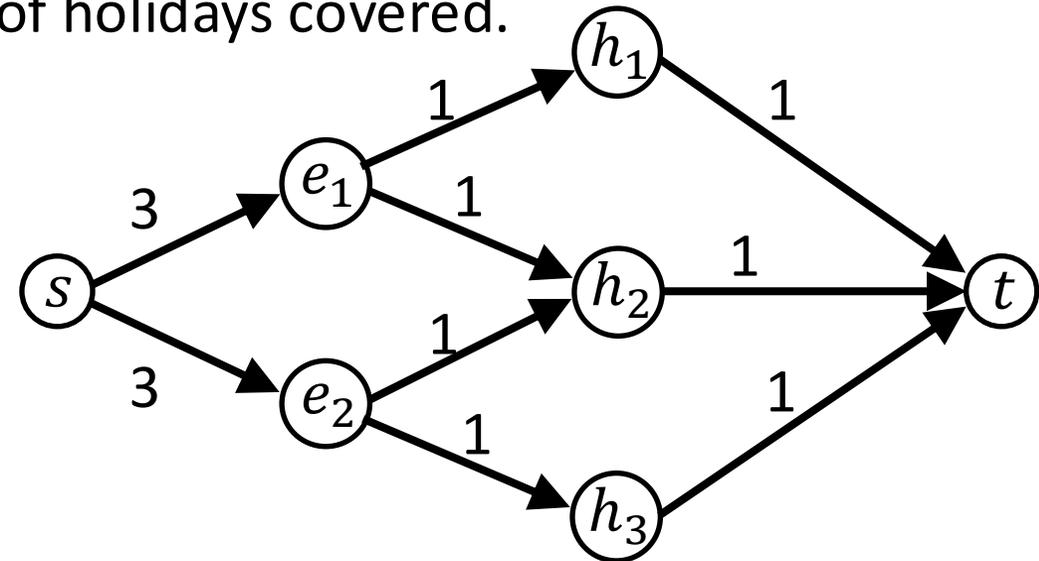
Employee	Holiday(s) they can work
e_1	h_1, h_2
e_2	h_2, h_3

Work Scheduling

Problem: We need to make holiday schedules for our employees. Each employee has a set of holidays that they are able to work. Each employee should work at most 3 holidays. We want to maximize the number of holidays covered.

Algorithm:

1. Build flow network:
 - a) Make a node for each employee, a node for each holiday, a source, and a sink.
 - b) Connect the source to each employee node with a capacity of 3.
 - c) Connect each holiday node to the sink with a capacity of 1.
 - d) If an employee is able to work a holiday, connect them with a capacity of 1.



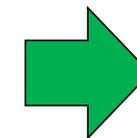
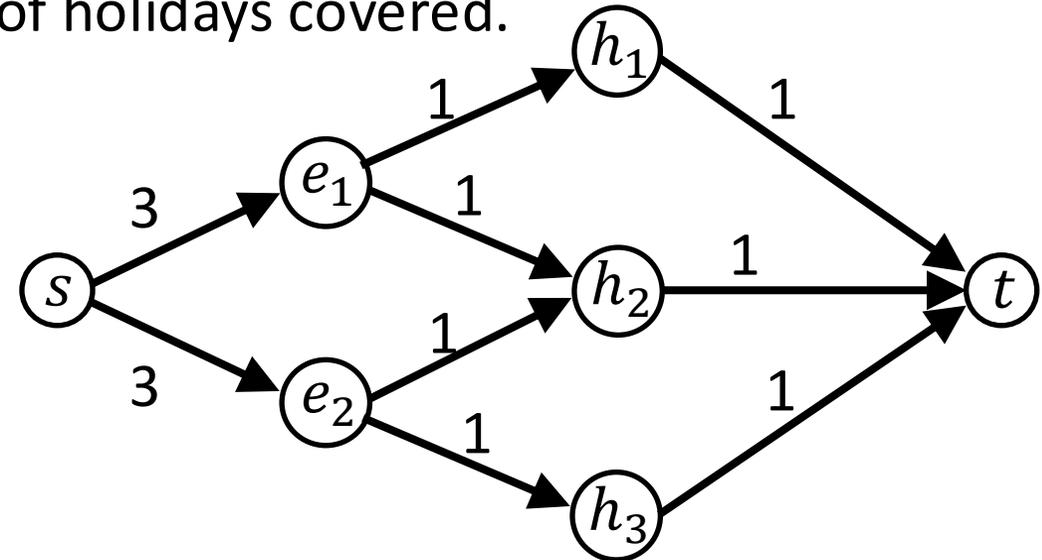
Employee	Holiday(s) they can work
e_1	h_1, h_2
e_2	h_2, h_3

Work Scheduling

Problem: We need to make holiday schedules for our employees. Each employee has a set of holidays that they are able to work. Each employee should work at most 3 holidays. We want to maximize the number of holidays covered.

Algorithm:

1. Build flow network:
 - a) Make a node for each employee, a node for each holiday, a source, and a sink.
 - b) Connect the source to each employee node with a capacity of 3.
 - c) Connect each holiday node to the sink with a capacity of 1.
 - d) If an employee is able to work a holiday, connect them with a capacity of 1.

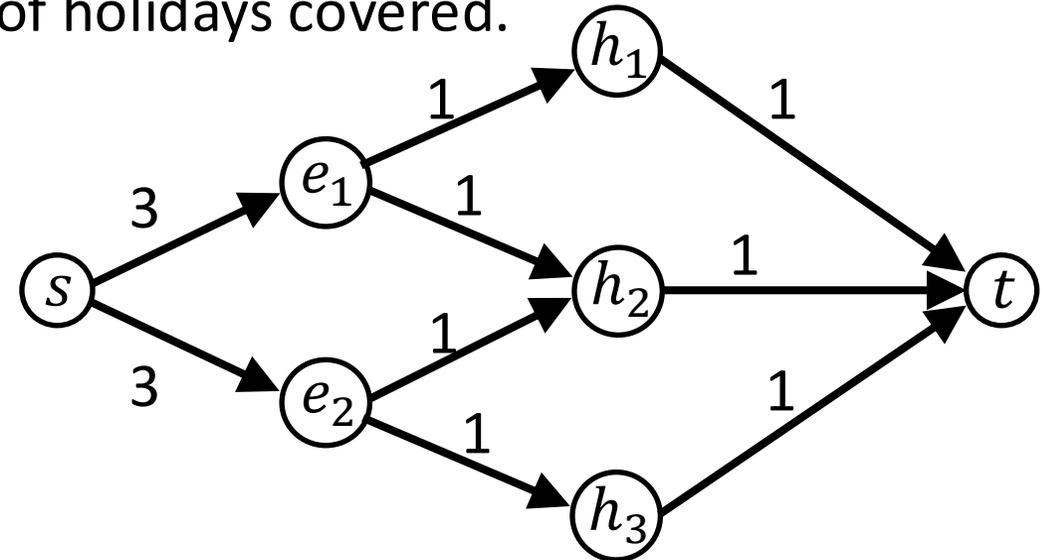


Work Scheduling

Problem: We need to make holiday schedules for our employees. Each employee has a set of holidays that they are able to work. Each employee should work at most 3 holidays. We want to maximize the number of holidays covered.

Algorithm:

1. Build flow network.
2. ??

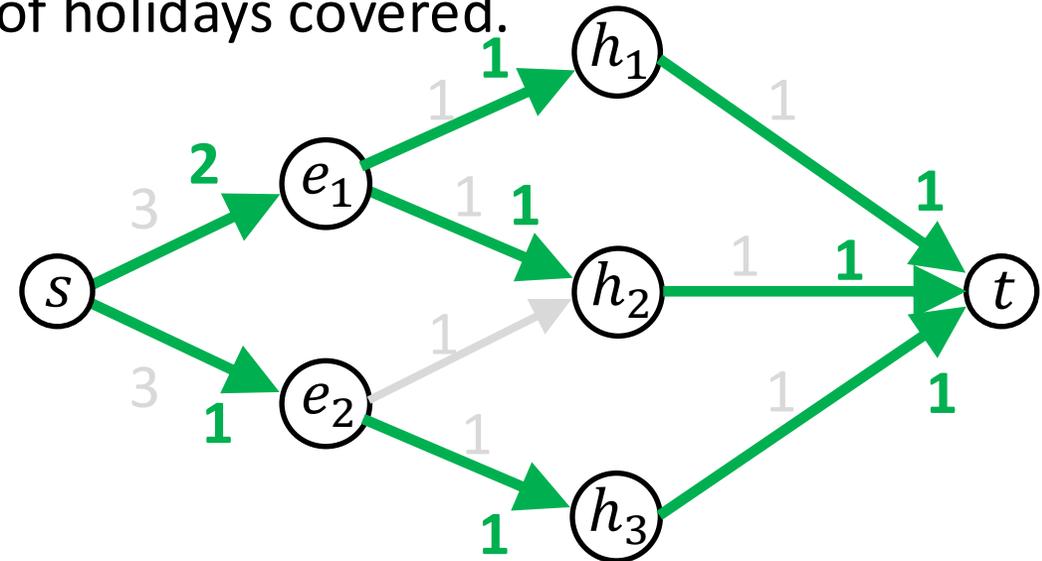


Work Scheduling

Problem: We need to make holiday schedules for our employees. Each employee has a set of holidays that they are able to work. Each employee should work at most 3 holidays. We want to maximize the number of holidays covered.

Algorithm:

1. Build flow network.
2. Find max flow.

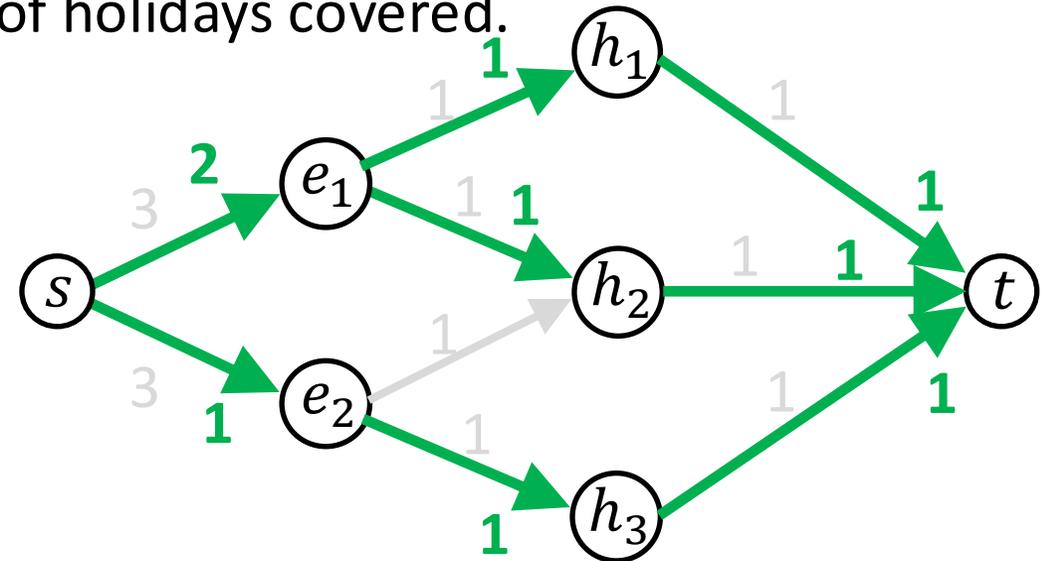


Work Scheduling

Problem: We need to make holiday schedules for our employees. Each employee has a set of holidays that they are able to work. Each employee should work at most 3 holidays. We want to maximize the number of holidays covered.

Algorithm:

1. Build flow network.
2. Find max flow.
3. ??

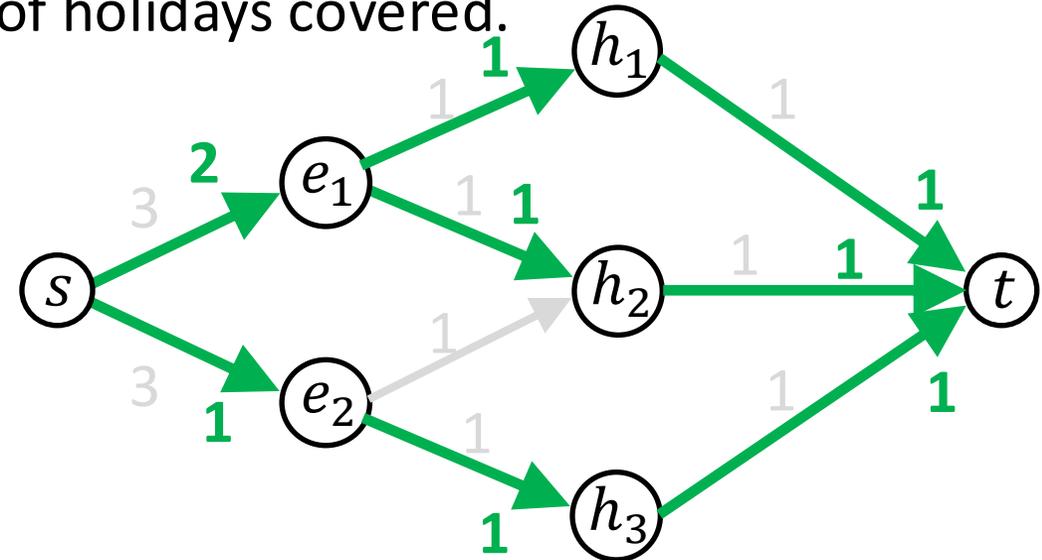


Work Scheduling

Problem: We need to make holiday schedules for our employees. Each employee has a set of holidays that they are able to work. Each employee should work at most 3 holidays. We want to maximize the number of holidays covered.

Algorithm:

1. Build flow network.
2. Find max flow.
3. <Translate flow to answer>

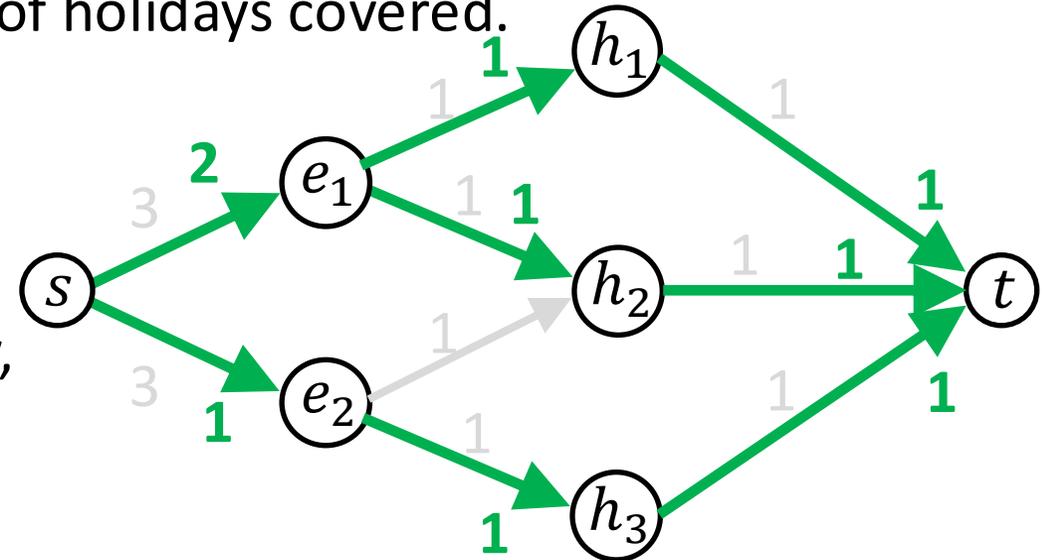


Work Scheduling

Problem: We need to make holiday schedules for our employees. Each employee has a set of holidays that they are able to work. Each employee should work at most 3 holidays. We want to maximize the number of holidays covered.

Algorithm:

1. Build flow network.
2. Find max flow.
3. If employee has outgoing edge carrying flow, assign them to work that holiday.



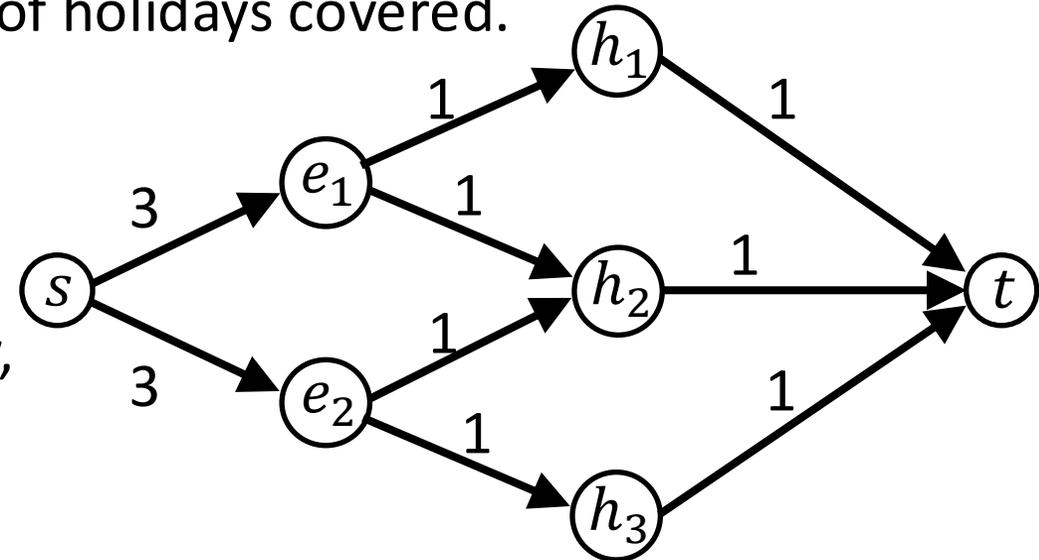
$e_1 \rightarrow h_1$
$e_1 \rightarrow h_2$
$e_2 \rightarrow h_3$

Work Scheduling

Problem: We need to make holiday schedules for our employees. Each employee has a set of holidays that they are able to work. Each employee should work at most 3 holidays. We want to maximize the number of holidays covered.

Algorithm:

1. Build flow network.
2. Find max flow.
3. If employee has outgoing edge carrying flow, assign them to work that holiday.



Running Time:

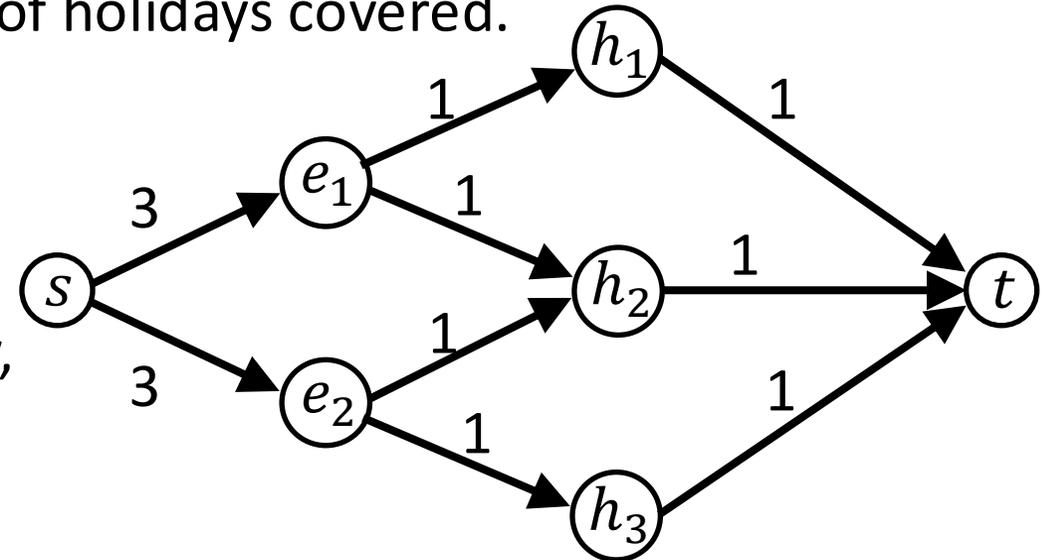
$$O(\text{time to build } G) + O(\text{Max Flow}) + O(\text{time to translate solution})$$

Work Scheduling

Problem: We need to make holiday schedules for our employees. Each employee has a set of holidays that they are able to work. Each employee should work at most 3 holidays. We want to maximize the number of holidays covered.

Algorithm:

1. Build flow network.
2. Find max flow.
3. If employee has outgoing edge carrying flow, assign them to work that holiday.



Running Time:

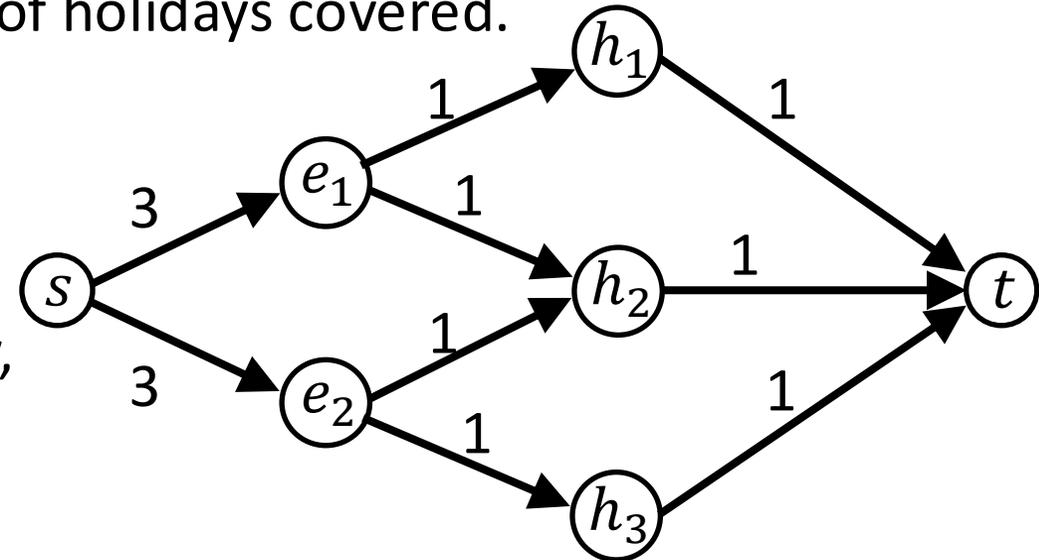
$n = \#$ employees, $m = \#$ holidays

Work Scheduling

Problem: We need to make holiday schedules for our employees. Each employee has a set of holidays that they are able to work. Each employee should work at most 3 holidays. We want to maximize the number of holidays covered.

Algorithm:

1. Build flow network.
2. Find max flow.
3. If employee has outgoing edge carrying flow, assign them to work that holiday.



Running Time:

$n = \#$ employees, $m = \#$ holidays

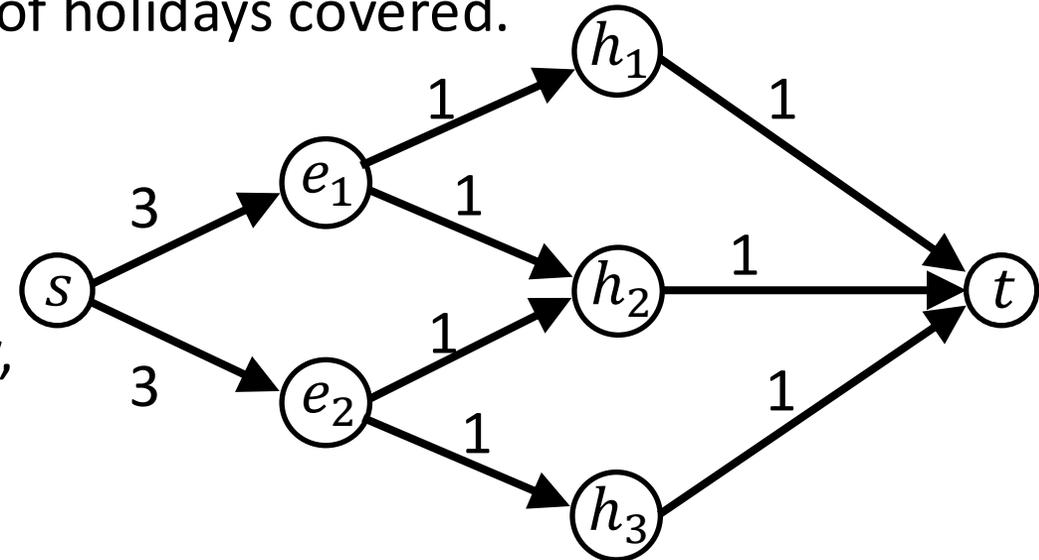
For $G = (V, E)$, $|G| = ?$

Work Scheduling

Problem: We need to make holiday schedules for our employees. Each employee has a set of holidays that they are able to work. Each employee should work at most 3 holidays. We want to maximize the number of holidays covered.

Algorithm:

1. Build flow network.
2. Find max flow.
3. If employee has outgoing edge carrying flow, assign them to work that holiday.



Running Time:

$n = \#$ employees, $m = \#$ holidays

For $G = (V, E)$, $|V| = n + m + 2$, $|E| = n + nm + m$

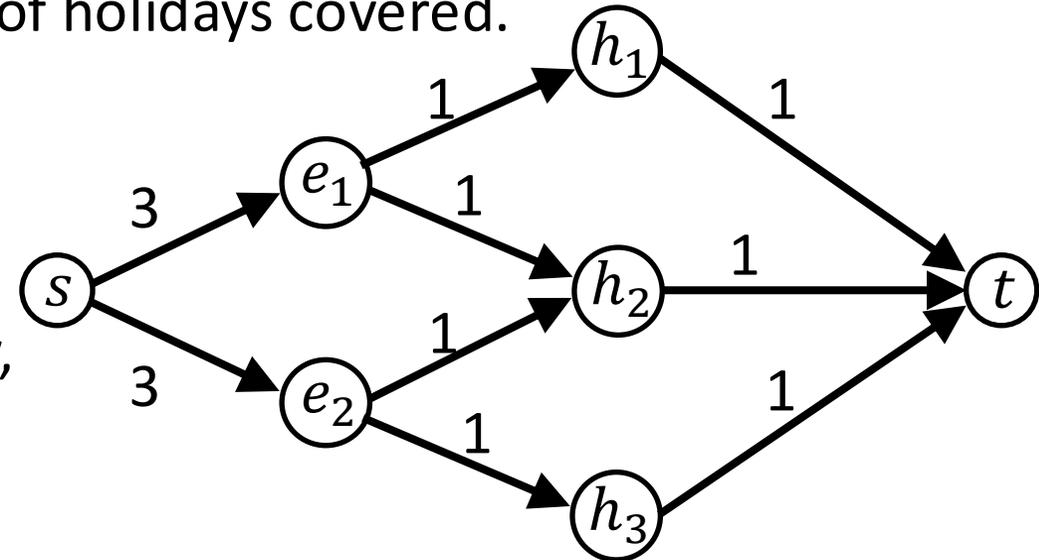
So, building $G \in O(nm)$.

Work Scheduling

Problem: We need to make holiday schedules for our employees. Each employee has a set of holidays that they are able to work. Each employee should work at most 3 holidays. We want to maximize the number of holidays covered.

Algorithm:

1. Build flow network.
2. Find max flow.
3. If employee has outgoing edge carrying flow, assign them to work that holiday.



Running Time:

$n = \#$ employees, $m = \#$ holidays

For $G = (V, E)$, $|V| = n + m + 2$, $|E| = n + nm + m$

So, building $G \in O(nm)$.

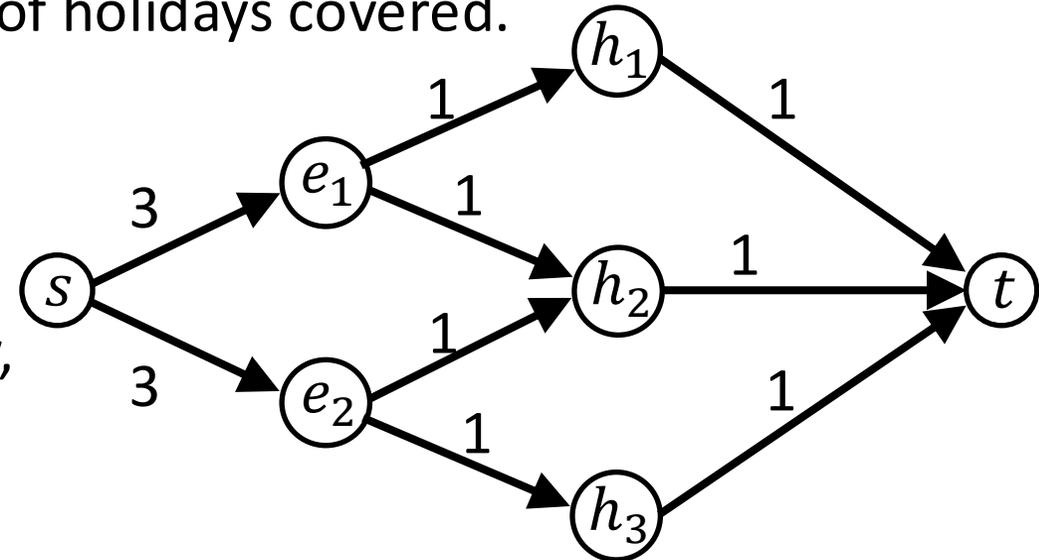
Total running time $\in ?$

Work Scheduling

Problem: We need to make holiday schedules for our employees. Each employee has a set of holidays that they are able to work. Each employee should work at most 3 holidays. We want to maximize the number of holidays covered.

Algorithm:

1. Build flow network.
2. Find max flow.
3. If employee has outgoing edge carrying flow, assign them to work that holiday.



Running Time:

$n = \#$ employees, $m = \#$ holidays

For $G = (V, E)$, $|V| = n + m + 2$, $|E| = n + nm + m$

So, building $G \in O(nm)$.

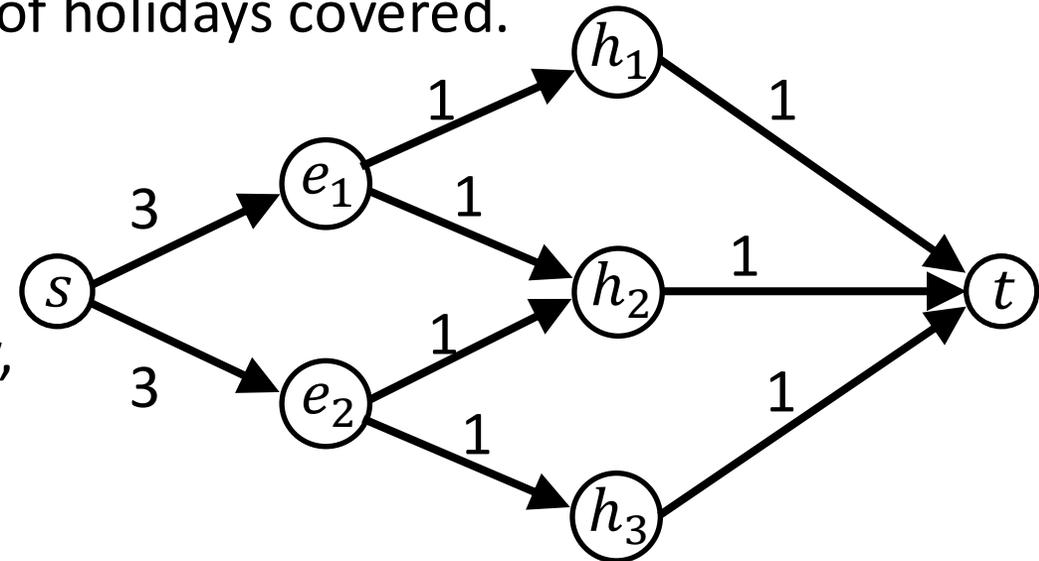
Total running time $\in O(nm) + O(\text{Max Flow}) + O(nm) \in O(nm) + O(\text{Max Flow})$

Work Scheduling

Problem: We need to make holiday schedules for our employees. Each employee has a set of holidays that they are able to work. Each employee should work at most 3 holidays. We want to maximize the number of holidays covered.

Algorithm:

1. Build flow network.
2. Find max flow.
3. If employee has outgoing edge carrying flow, assign them to work that holiday.



Validity:

??

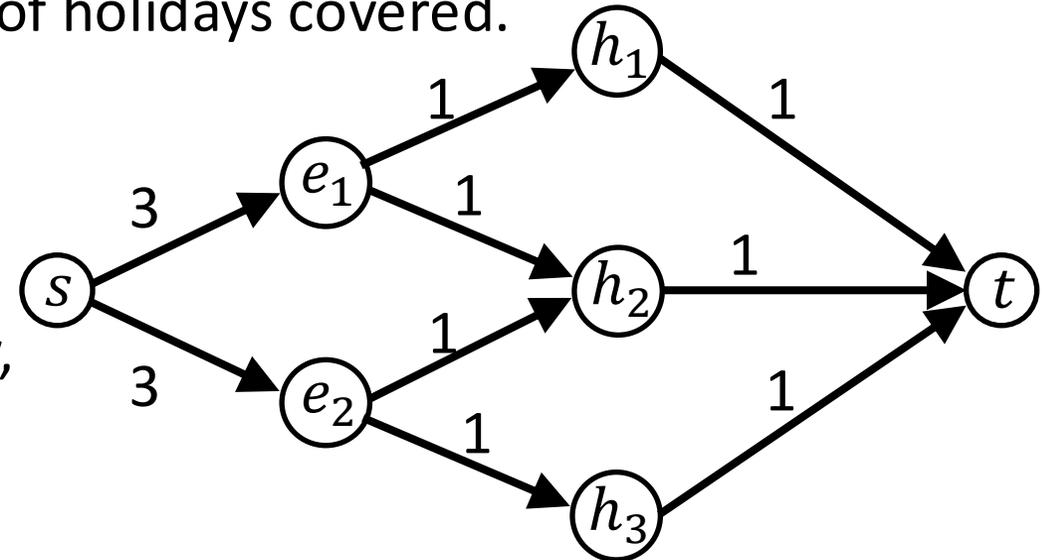
We need to show the algorithm's answer is valid for the problem, not that the max flow found is valid for the flow network we made.

Work Scheduling

Problem: We need to make holiday schedules for our employees. Each employee has a set of holidays that they are able to work. Each employee should work at most 3 holidays. We want to maximize the number of holidays covered.

Algorithm:

1. Build flow network.
2. Find max flow.
3. If employee has outgoing edge carrying flow, assign them to work that holiday.



Validity:

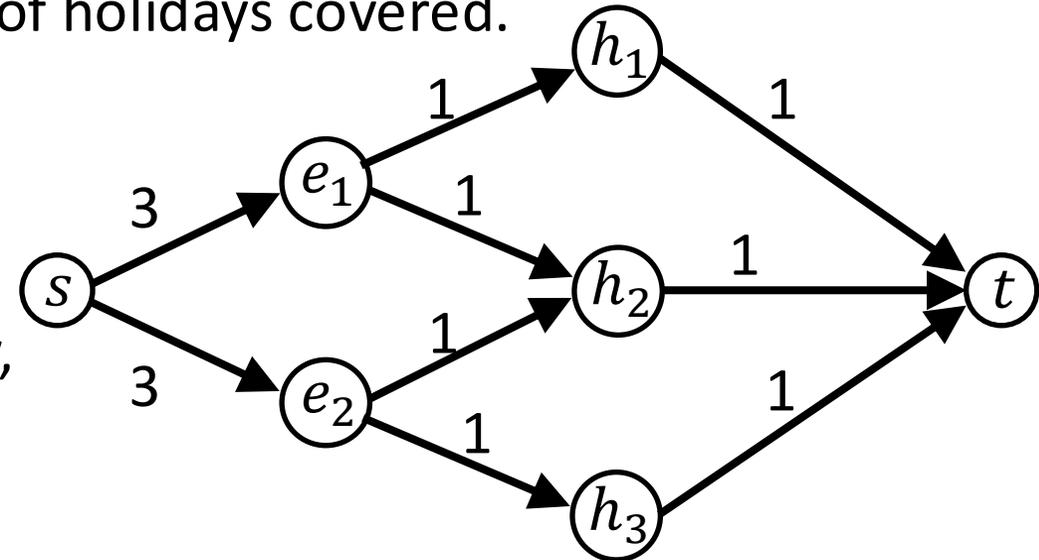
Employees cannot be assigned a holiday they are not able to work (no edges exist between incompatible pairs).

Work Scheduling

Problem: We need to make holiday schedules for our employees. Each employee has a set of holidays that they are able to work. Each employee should work at most 3 holidays. We want to maximize the number of holidays covered.

Algorithm:

1. Build flow network.
2. Find max flow.
3. If employee has outgoing edge carrying flow, assign them to work that holiday.



Validity:

Employees cannot be assigned a holiday they are not able to work (no edges exist between incompatible pairs). Employees cannot be assigned to work more than 3 holidays (Why?).

Work Scheduling

Problem: We need to make holiday schedules for our employees. Each employee has a set of holidays that they are able to work. Each employee should work at most 3 holidays. We want to maximize the number of holidays covered.

Algorithm:

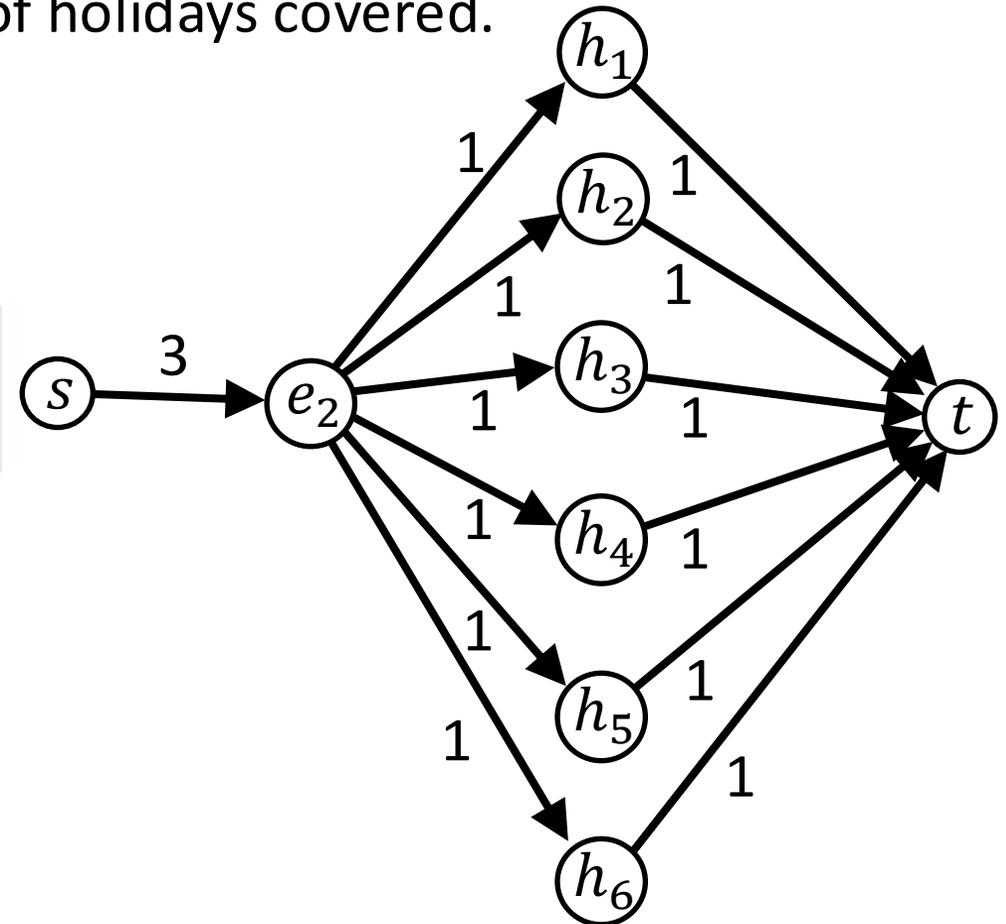
1. Build flow network.
2. Find max flow.
3. If employee has outgoing edge carrying flow, assign them to work that holiday.

Work Scheduling

Problem: We need to make holiday schedules for our employees. Each employee has a set of holidays that they are able to work. Each employee should work at most 3 holidays. We want to maximize the number of holidays covered.

Algorithm:

1. Build flow network.
2. Find max flow.
3. If employee has outgoing edge carrying flow, assign them to work that holiday.

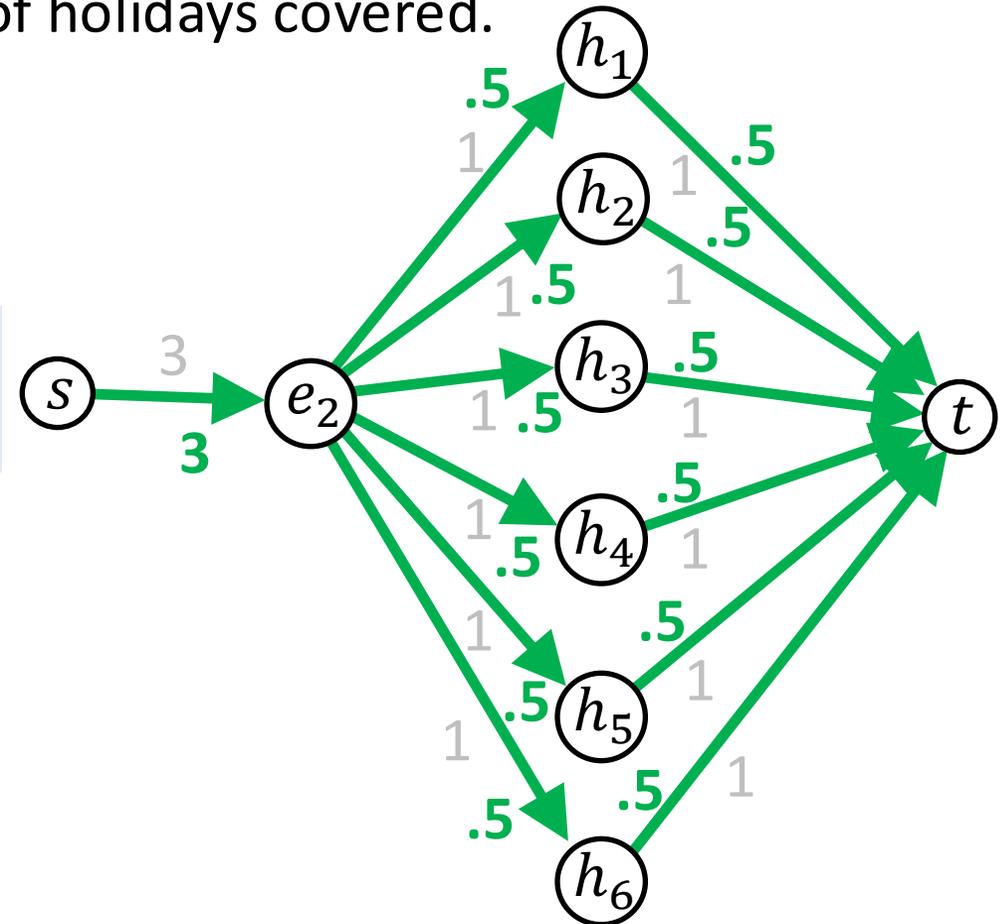


Work Scheduling

Problem: We need to make holiday schedules for our employees. Each employee has a set of holidays that they are able to work. Each employee should work at most 3 holidays. We want to maximize the number of holidays covered.

Algorithm:

1. Build flow network.
2. Find max flow.
3. If employee has outgoing edge carrying flow, assign them to work that holiday.

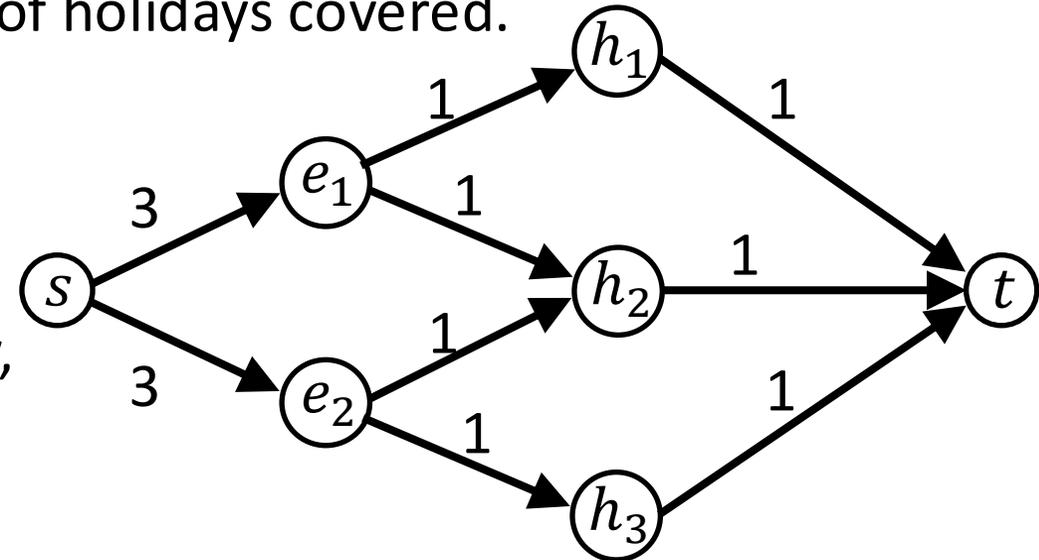


Work Scheduling

Problem: We need to make holiday schedules for our employees. Each employee has a set of holidays that they are able to work. Each employee should work at most 3 holidays. We want to maximize the number of holidays covered.

Algorithm:

1. Build flow network.
2. Find max flow.
3. If employee has outgoing edge carrying flow, assign them to work that holiday.



Validity:

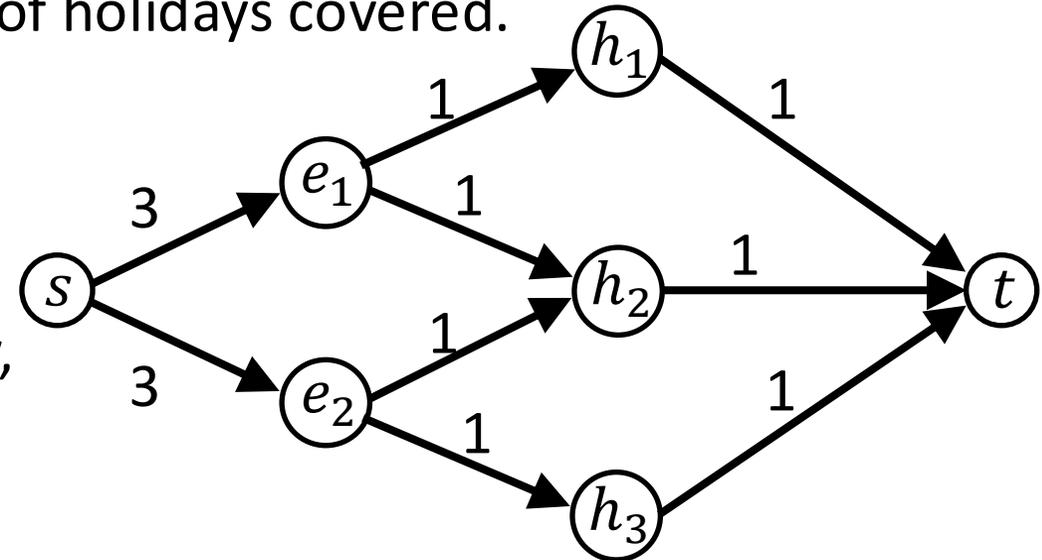
Employees cannot be assigned a holiday they are not able to work (no edges exist between incompatible pairs). Employees cannot be assigned to work more than 3 holidays (integer edge capacities yield integer flow values and total available flow is 3).

Work Scheduling

Problem: We need to make holiday schedules for our employees. Each employee has a set of holidays that they are able to work. Each employee should work at most 3 holidays. We want to maximize the number of holidays covered.

Algorithm:

1. Build flow network.
2. Find max flow.
3. If employee has outgoing edge carrying flow, assign them to work that holiday.



Optimality:

??

We need to show the algorithm's answer is optimal for the problem, not that the max flow found is optimal for the flow network we made.

Work Scheduling

Problem: We need to make holiday schedules for our employees. Each employee has a set of holidays that they are able to work. Each employee should work at most 3 holidays. We want to maximize the number of holidays covered.

Algorithm:

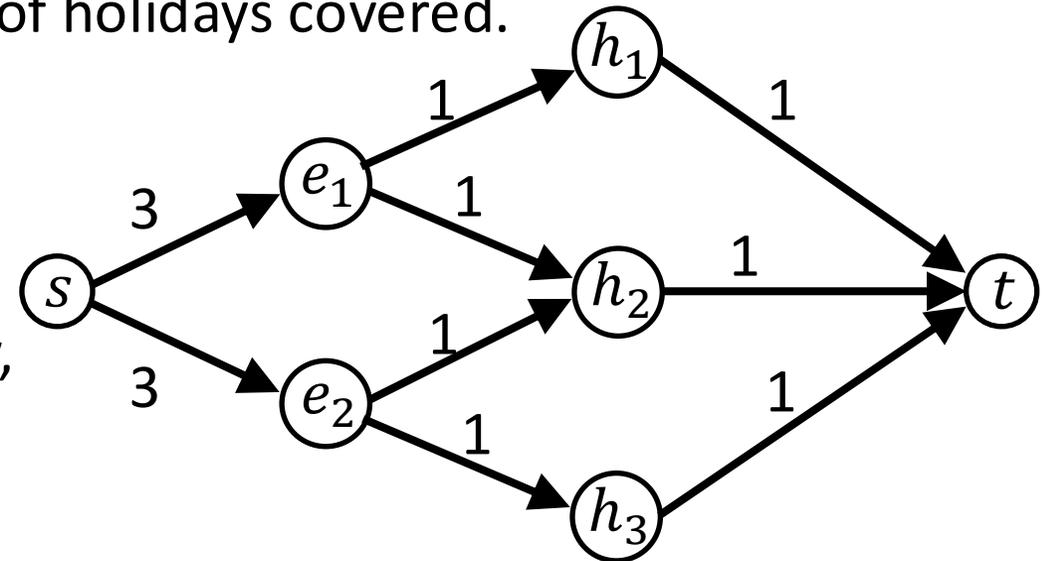
1. Build flow network.
2. Find max flow.
3. If employee has outgoing edge carrying flow, assign them to work that holiday.

Optimality:

Suppose max flow $F < OPT \leq m$.

holidays

Optimal answer
to the problem.



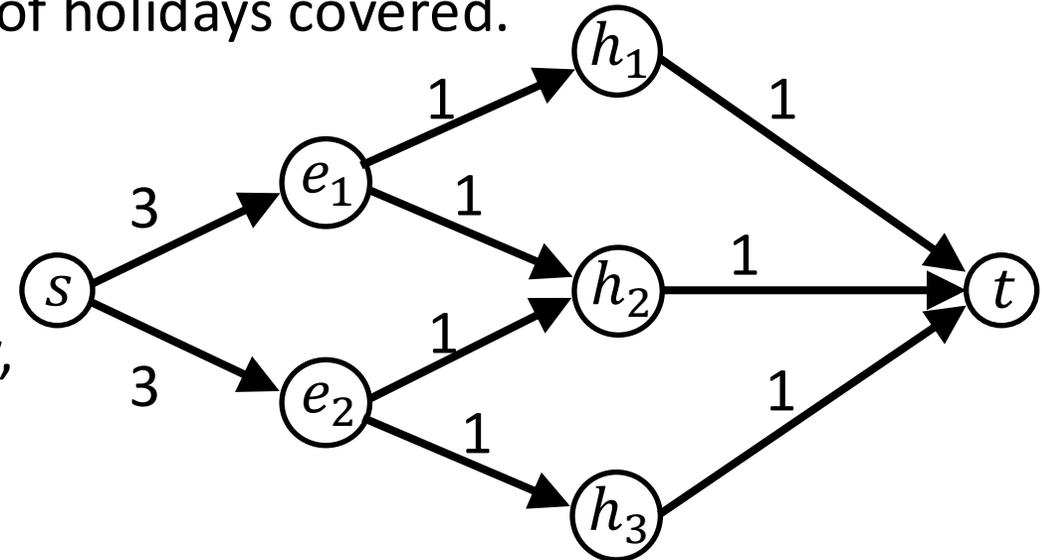
$e_1 \rightarrow h_1, e_2 \rightarrow h_2, e_2 \rightarrow h_3$

Work Scheduling

Problem: We need to make holiday schedules for our employees. Each employee has a set of holidays that they are able to work. Each employee should work at most 3 holidays. We want to maximize the number of holidays covered.

Algorithm:

1. Build flow network.
2. Find max flow.
3. If employee has outgoing edge carrying flow, assign them to work that holiday.



Optimality:

Suppose max flow $F < OPT \leq m$. That means employees can be assigned to cover OPT holidays.

Plan: Turn OPT into a valid flow with value larger than F . CONTRADICTION!!!

$e_1 \rightarrow h_1, e_2 \rightarrow h_2, e_2 \rightarrow h_3$

Work Scheduling

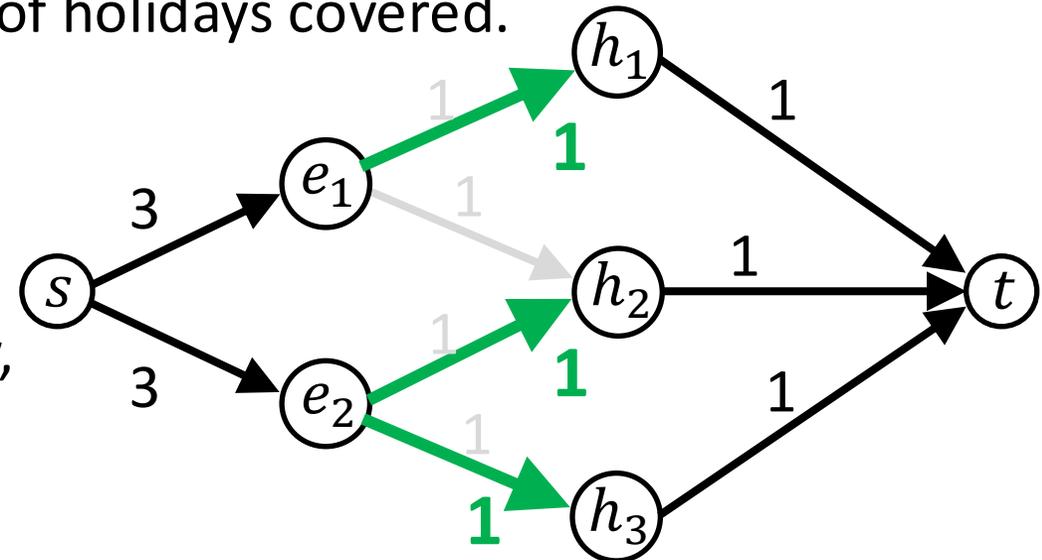
Problem: We need to make holiday schedules for our employees. Each employee has a set of holidays that they are able to work. Each employee should work at most 3 holidays. We want to maximize the number of holidays covered.

Algorithm:

1. Build flow network.
2. Find max flow.
3. If employee has outgoing edge carrying flow, assign them to work that holiday.

Optimality:

Suppose max flow $F < OPT \leq m$. That means employees can be assigned to cover OPT holidays.



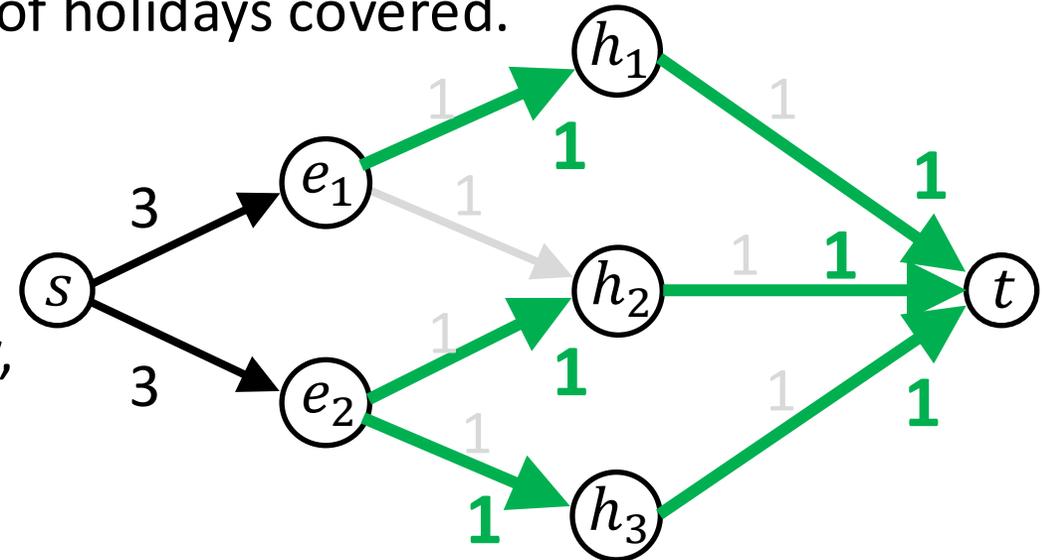
$e_1 \rightarrow h_1, e_2 \rightarrow h_2, e_2 \rightarrow h_3$

Work Scheduling

Problem: We need to make holiday schedules for our employees. Each employee has a set of holidays that they are able to work. Each employee should work at most 3 holidays. We want to maximize the number of holidays covered.

Algorithm:

1. Build flow network.
2. Find max flow.
3. If employee has outgoing edge carrying flow, assign them to work that holiday.



Optimality:

Suppose max flow $F < OPT \leq m$. That means employees can be assigned to cover OPT holidays.

Make edge for each assignment carry 1.

Make edge leaving each covered holiday carry 1.

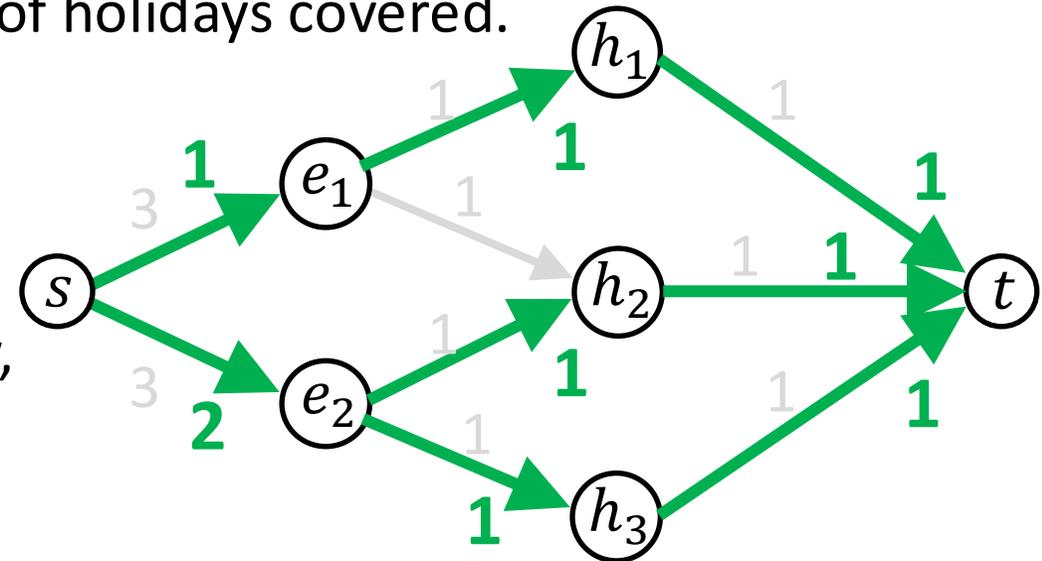
$$e_1 \rightarrow h_1, e_2 \rightarrow h_2, e_2 \rightarrow h_3$$

Work Scheduling

Problem: We need to make holiday schedules for our employees. Each employee has a set of holidays that they are able to work. Each employee should work at most 3 holidays. We want to maximize the number of holidays covered.

Algorithm:

1. Build flow network.
2. Find max flow.
3. If employee has outgoing edge carrying flow, assign them to work that holiday.



Optimality:

Suppose max flow $F < OPT \leq m$. That means employees can be assigned to cover OPT holidays.

Make edge leaving each covered holiday carry 1. Make edge into each employee carry sum of the flow leaving.

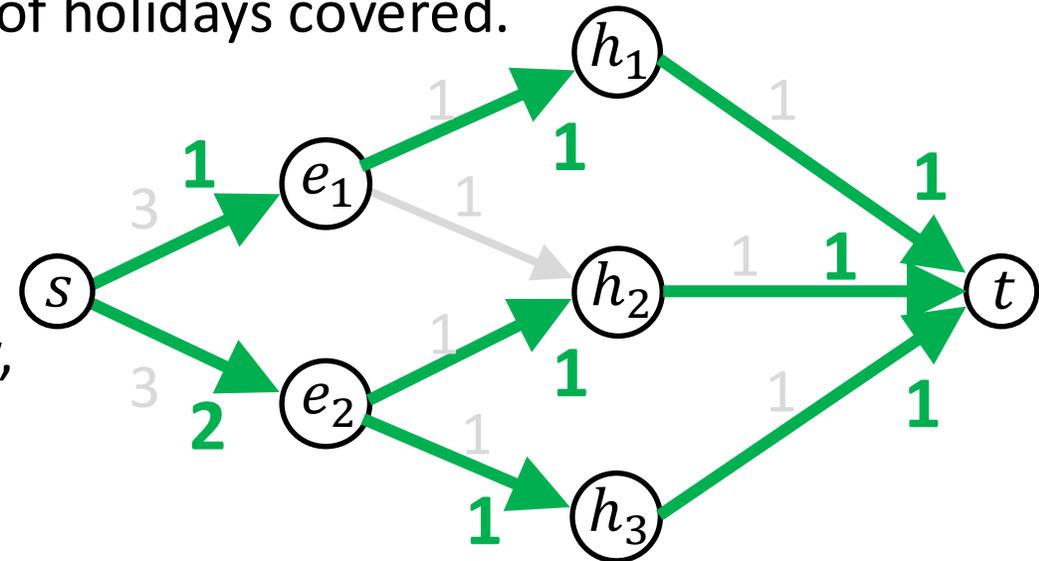
$$e_1 \rightarrow h_1, e_2 \rightarrow h_2, e_2 \rightarrow h_3$$

Work Scheduling

Problem: We need to make holiday schedules for our employees. Each employee has a set of holidays that they are able to work. Each employee should work at most 3 holidays. We want to maximize the number of holidays covered.

Algorithm:

1. Build flow network.
2. Find max flow.
3. If employee has outgoing edge carrying flow, assign them to work that holiday.



Optimality:

Suppose max flow $F < OPT \leq m$. That means employees can be assigned to cover OPT holidays.

Make edge leaving each covered holiday carry 1. Make edge into each employee carry sum of the flow leaving. Conservation of flow holds at employees because that's how we set incoming flow. Conservation at holiday nodes holds because...

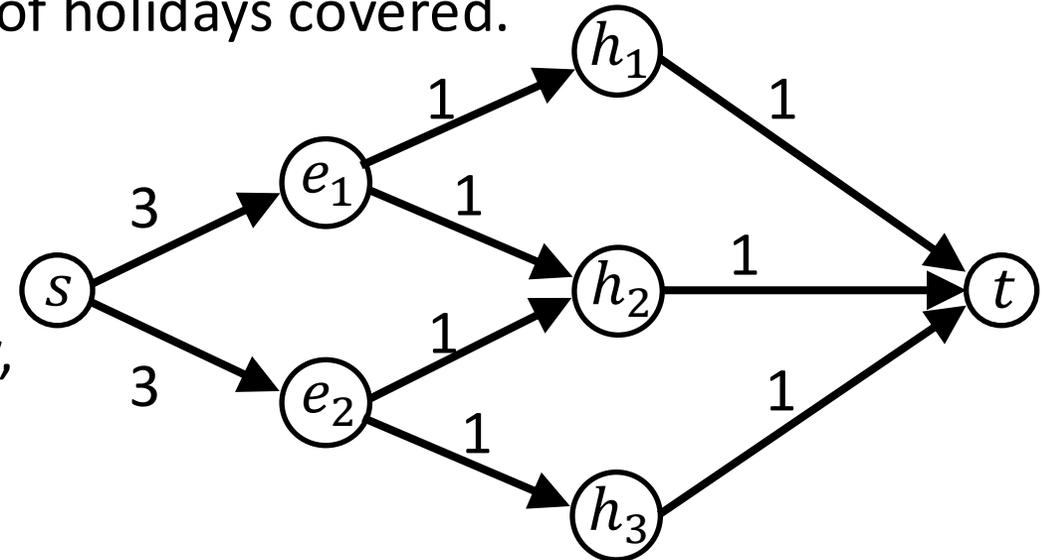
Work Scheduling

$e_1 \rightarrow h_2, e_2 \rightarrow h_2$ – Valid Schedule?

Problem: We need to make holiday schedules for our employees. Each employee has a set of holidays that they are able to work. Each employee should work at most 3 holidays. We want to maximize the number of holidays covered.

Algorithm:

1. Build flow network.
2. Find max flow.
3. If employee has outgoing edge carrying flow, assign them to work that holiday.



Optimality:

Suppose max flow $F < OPT \leq m$. That means employees can be assigned to cover OPT holidays.

Make edge for each assignment carry 1.

Make edge leaving each covered holiday carry 1. Make edge into each employee carry sum of the flow leaving. Conservation of flow holds at employees because that's how we set incoming flow. Conservation at holiday nodes holds because...

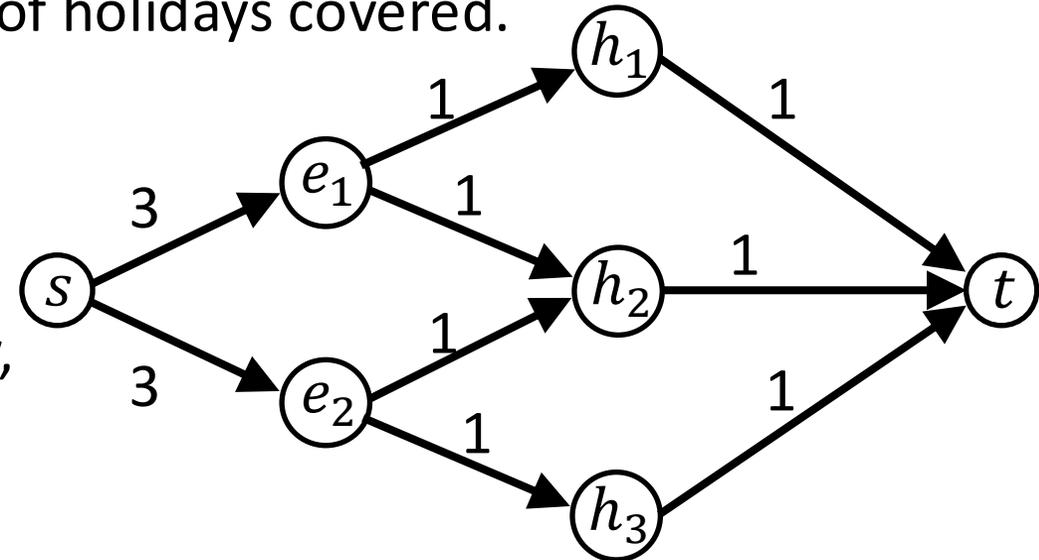
Work Scheduling

$e_1 \rightarrow h_2, e_2 \rightarrow h_2$ – Valid Schedule? ✓

Problem: We need to make holiday schedules for our employees. Each employee has a set of holidays that they are able to work. Each employee should work at most 3 holidays. We want to maximize the number of holidays covered.

Algorithm:

1. Build flow network.
2. Find max flow.
3. If employee has outgoing edge carrying flow, assign them to work that holiday.



Optimality:

Suppose max flow $F < OPT \leq m$. That means employees can be assigned to cover OPT holidays.

Make edge for each assignment carry 1.

Make edge leaving each covered holiday carry 1. Make edge into each employee carry sum of the flow leaving. Conservation of flow holds at employees because that's how we set incoming flow. Conservation at holiday nodes holds because...

Work Scheduling

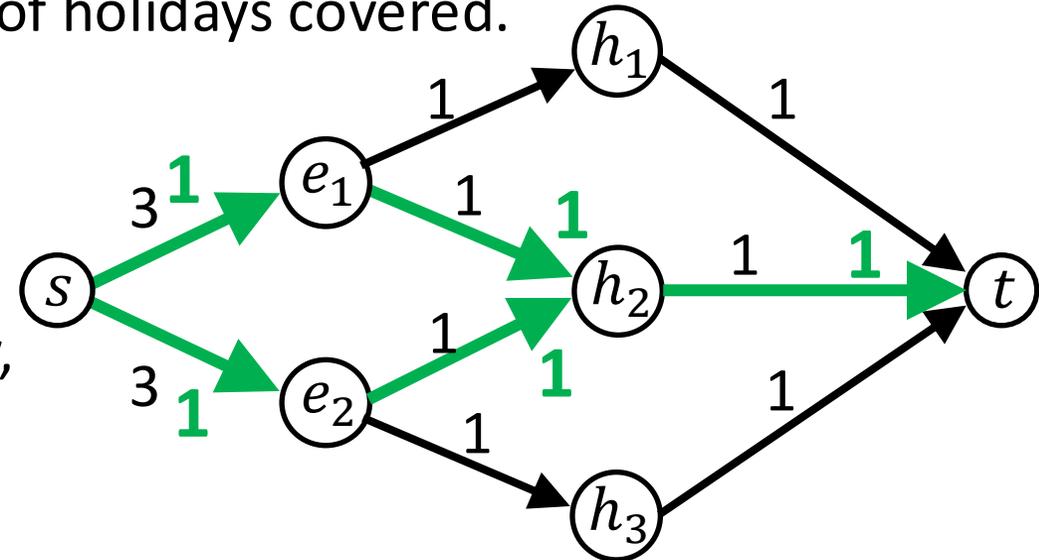
$e_1 \rightarrow h_2, e_2 \rightarrow h_2$ – Valid Schedule? ✓

Valid Flow?

Problem: We need to make holiday schedules for our employees. Each employee has a set of holidays that they are able to work. Each employee should work at most 3 holidays. We want to maximize the number of holidays covered.

Algorithm:

1. Build flow network.
2. Find max flow.
3. If employee has outgoing edge carrying flow, assign them to work that holiday.



Optimality:

Suppose max flow $F < OPT \leq m$. That means employees can be assigned to cover OPT holidays.

Make edge for each assignment carry 1.

Make edge leaving each covered holiday carry 1. Make edge into each employee carry sum of the flow leaving. Conservation of flow holds at employees because that's how we set incoming flow. Conservation at holiday nodes holds because...

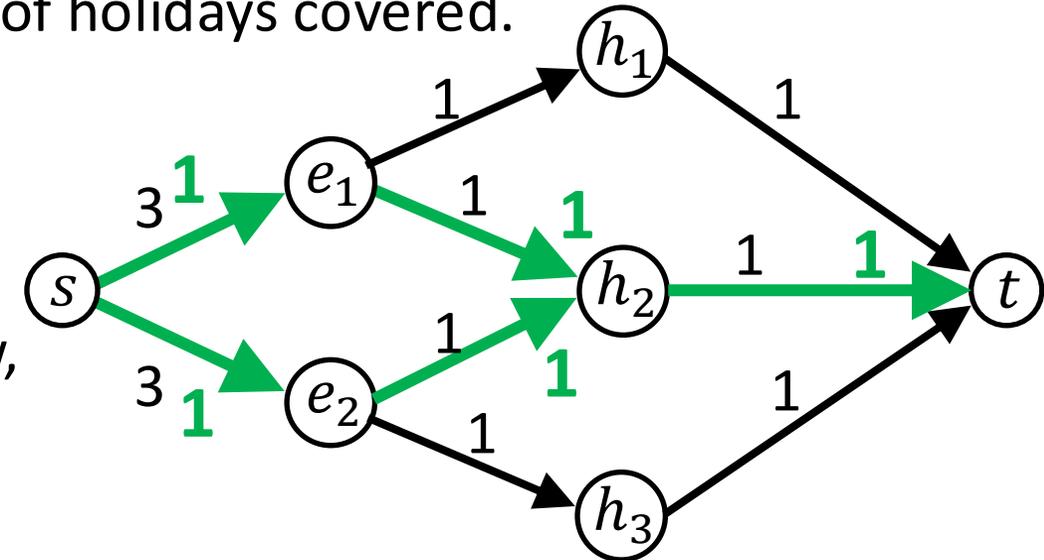
Work Scheduling

$e_1 \rightarrow h_2, e_2 \rightarrow h_2$ – Valid Schedule? ✓
Valid Flow? ✗

Problem: We need to make holiday schedules for our employees. Each employee has a set of holidays that they are able to work. Each employee should work at most 3 holidays. We want to maximize the number of holidays covered.

Algorithm:

1. Build flow network.
2. Find max flow.
3. If employee has outgoing edge carrying flow, assign them to work that holiday.



Optimality:

Suppose max flow $F < OPT \leq m$. That means employees can be assigned to cover OPT holidays.

Make edge leaving each covered holiday carry 1. Make edge into each employee carry sum of the flow leaving. Conservation of flow holds at employees because that's how we set incoming flow. Conservation at holiday nodes holds because...

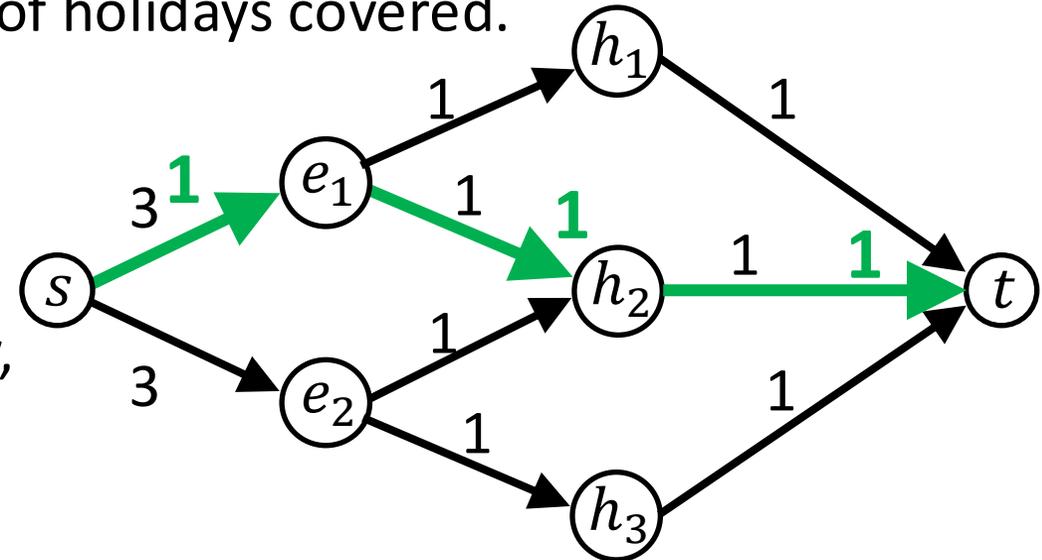
Work Scheduling

$e_1 \rightarrow h_2, e_2 \rightarrow h_2$ – Valid Schedule? ✓
Valid Flow? ✗

Problem: We need to make holiday schedules for our employees. Each employee has a set of holidays that they are able to work. Each employee should work at most 3 holidays. We want to maximize the number of holidays covered.

Algorithm:

1. Build flow network.
2. Find max flow.
3. If employee has outgoing edge carrying flow, assign them to work that holiday.



Optimality:

Suppose max flow $F < OPT \leq m$. That means employees can be assigned to cover OPT holidays. **First, remove redundant holiday coverage.** Make edge for each assignment carry 1. Make edge leaving each covered holiday carry 1. Make edge into each employee carry sum of the flow leaving. Conservation of flow holds at employees because that's how we set incoming flow. Conservation at holiday nodes holds because...

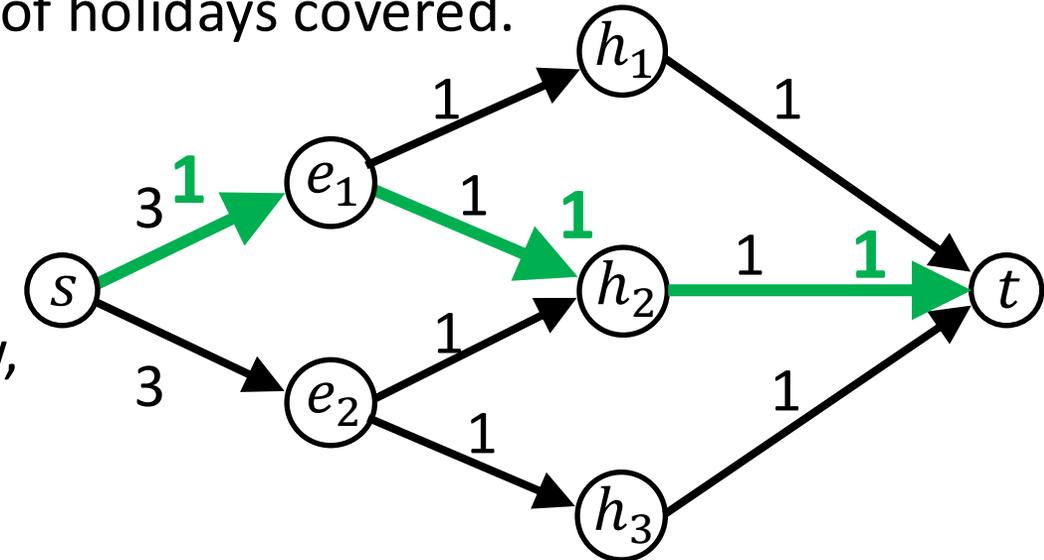
Work Scheduling

$e_1 \rightarrow h_2, e_2 \rightarrow h_2$ – Valid Schedule? ✓
Valid Flow? ✗

Problem: We need to make holiday schedules for our employees. Each employee has a set of holidays that they are able to work. Each employee should work at most 3 holidays. We want to maximize the number of holidays covered.

Algorithm:

1. Build flow network.
2. Find max flow.
3. If employee has outgoing edge carrying flow, assign them to work that holiday.



Optimality:

Suppose max flow $F < OPT \leq m$. That means employees can be assigned to cover OPT holidays. **First, remove redundant holiday coverage.** Make edge for each assignment carry 1.

Make edge leaving each covered holiday carry 1. Conservation of flow holds at employee nodes. Conservation at holiday nodes

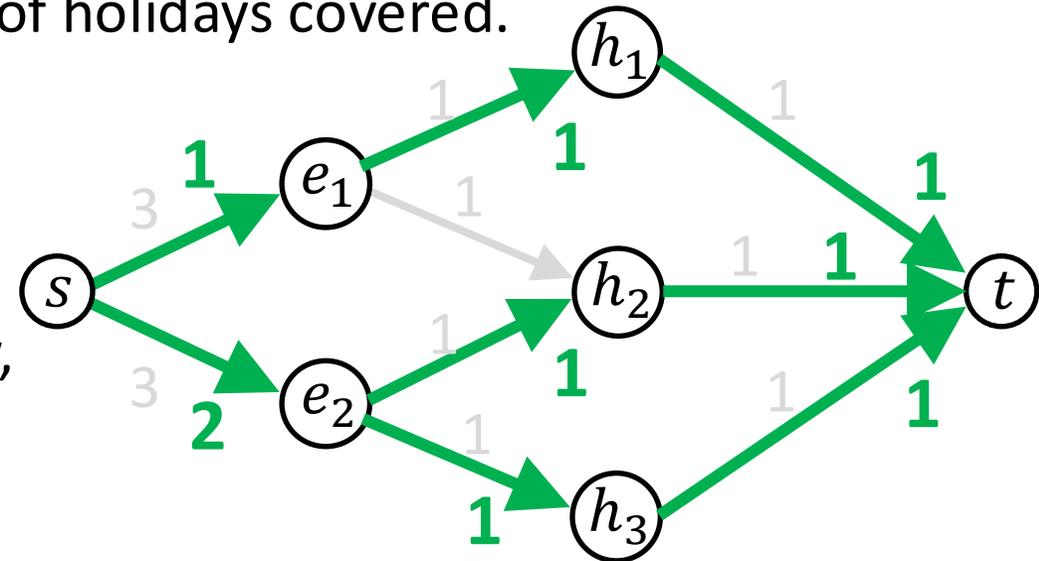
This translates a valid problem solution that is invalid on the flow network to an identical cost solution that is valid for both.

Work Scheduling

Problem: We need to make holiday schedules for our employees. Each employee has a set of holidays that they are able to work. Each employee should work at most 3 holidays. We want to maximize the number of holidays covered.

Algorithm:

1. Build flow network.
2. Find max flow.
3. If employee has outgoing edge carrying flow, assign them to work that holiday.



Optimality:

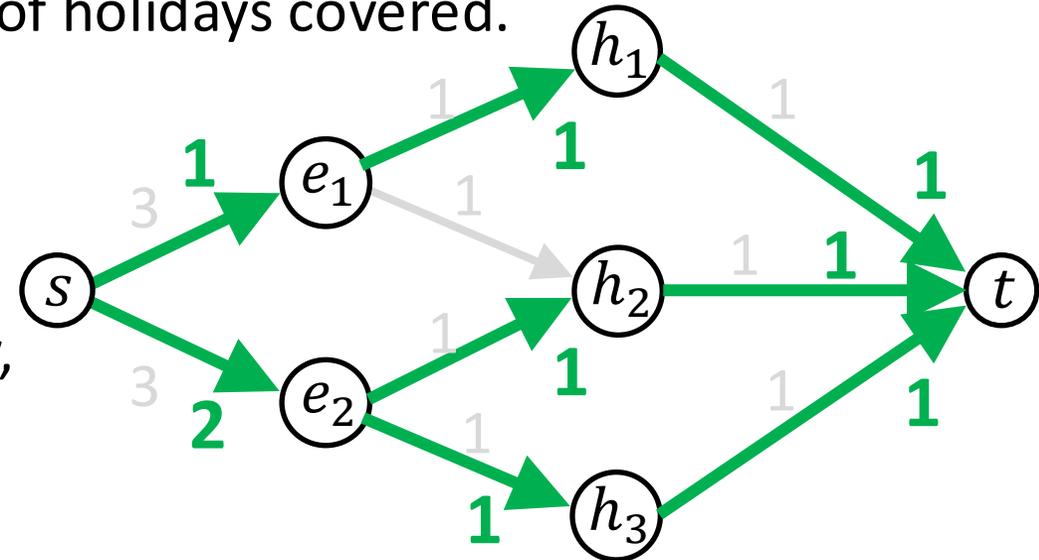
Suppose max flow $F < OPT \leq m$. That means employees can be assigned to cover OPT holidays. First, remove redundant holiday coverage. Make edge for each assignment carry 1. Make edge leaving each covered holiday carry 1. Make edge into each employee carry sum of the flow leaving. Conservation of flow holds at employees because that's how we set incoming flow. Conservation at holiday nodes holds because if a holiday is covered, it's done with a single employee.

Work Scheduling

Problem: We need to make holiday schedules for our employees. Each employee has a set of holidays that they are able to work. Each employee should work at most 3 holidays. We want to maximize the number of holidays covered.

Algorithm:

1. Build flow network.
2. Find max flow.
3. If employee has outgoing edge carrying flow, assign them to work that holiday.



Optimality:

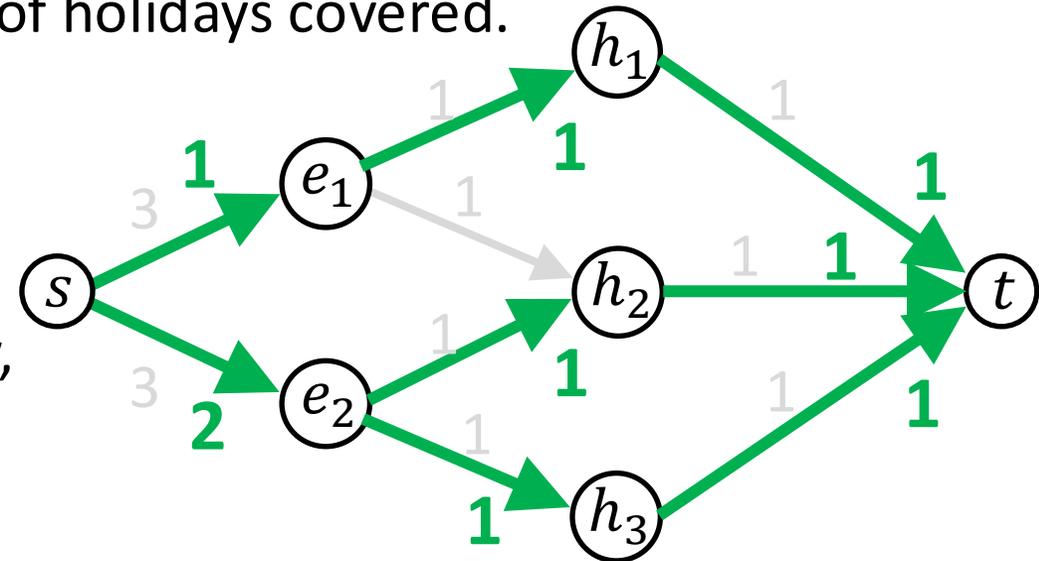
Suppose max flow $F < OPT \leq m$. That means employees can be assigned to cover OPT holidays. First, remove redundant holiday coverage. Make edge for each assignment carry 1. Make edge leaving each covered holiday carry 1. Make edge into each employee carry sum of the flow leaving. Conservation of flow holds at employees because that's how we set incoming flow. Conservation at holiday nodes holds because if a holiday is covered, it's done with a single employee. Also, no capacities are violated.

Work Scheduling

Problem: We need to make holiday schedules for our employees. Each employee has a set of holidays that they are able to work. Each employee should work at most 3 holidays. We want to maximize the number of holidays covered.

Algorithm:

1. Build flow network.
2. Find max flow.
3. If employee has outgoing edge carrying flow, assign them to work that holiday.



Optimality:

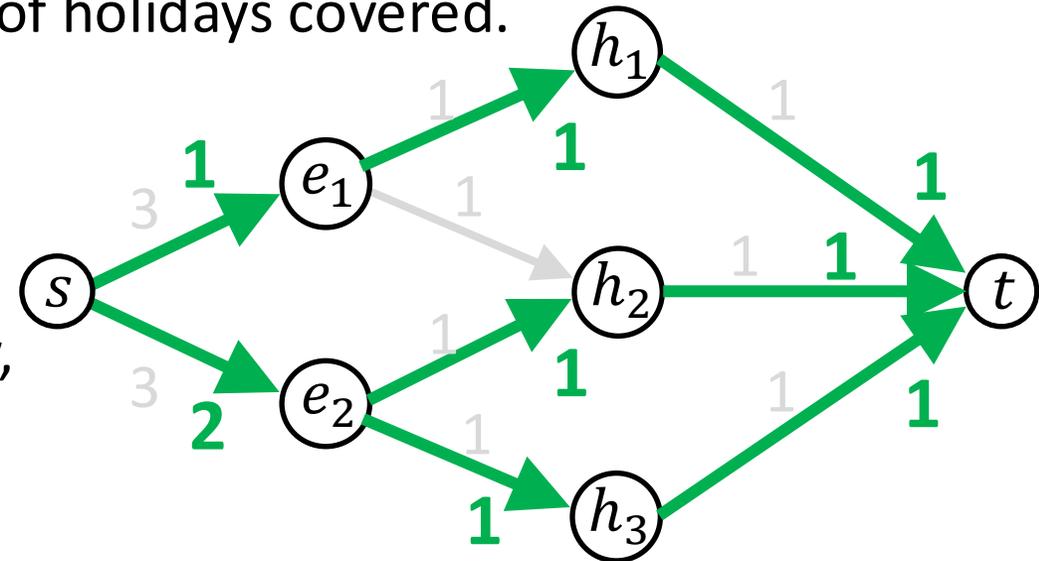
Suppose max flow $F < OPT \leq m$. That means employees can be assigned to cover OPT holidays. First, remove redundant holiday coverage. Make edge for each assignment carry 1. Make edge leaving each covered holiday carry 1. Make edge into each employee carry sum of the flow leaving. Conservation of flow holds at employees because that's how we set incoming flow. Conservation at holiday nodes holds because if a holiday is covered, it's done with a single employee. Also, no capacities are violated. \Rightarrow Valid Flow

Work Scheduling

Problem: We need to make holiday schedules for our employees. Each employee has a set of holidays that they are able to work. Each employee should work at most 3 holidays. We want to maximize the number of holidays covered.

Algorithm:

1. Build flow network.
2. Find max flow.
3. If employee has outgoing edge carrying flow, assign them to work that holiday.



Optimality:

Suppose max flow $F < OPT \leq m$. That means employees can be assigned to cover OPT holidays. First, remove redundant holiday coverage. Make edge for each assignment carry 1. Make edge leaving each covered holiday carry 1. Make edge into each employee carry sum of the flow leaving. Conservation of flow holds at employees because that's how we set incoming flow. Conservation at holiday nodes holds because if a holiday is covered, it's done with a single employee. Also, no capacities are violated. \Rightarrow Valid Flow $> F$.

