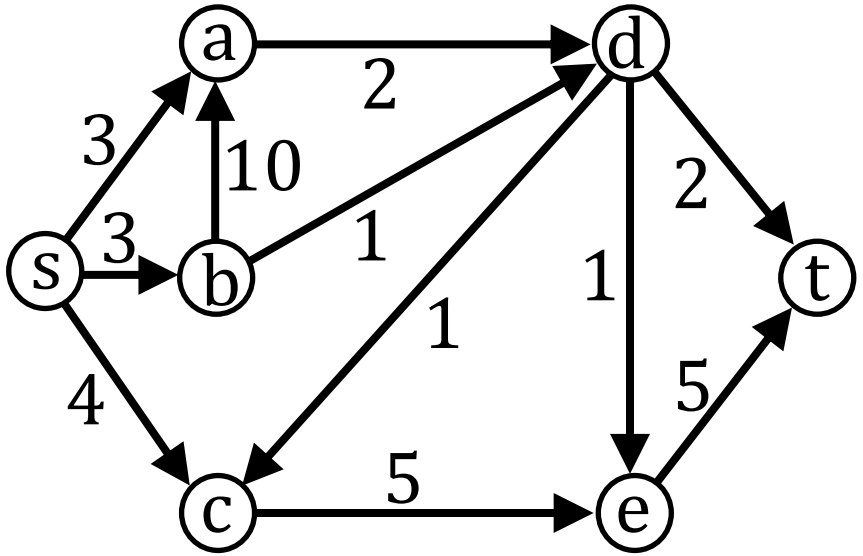
Flow Networks
CSCI 432

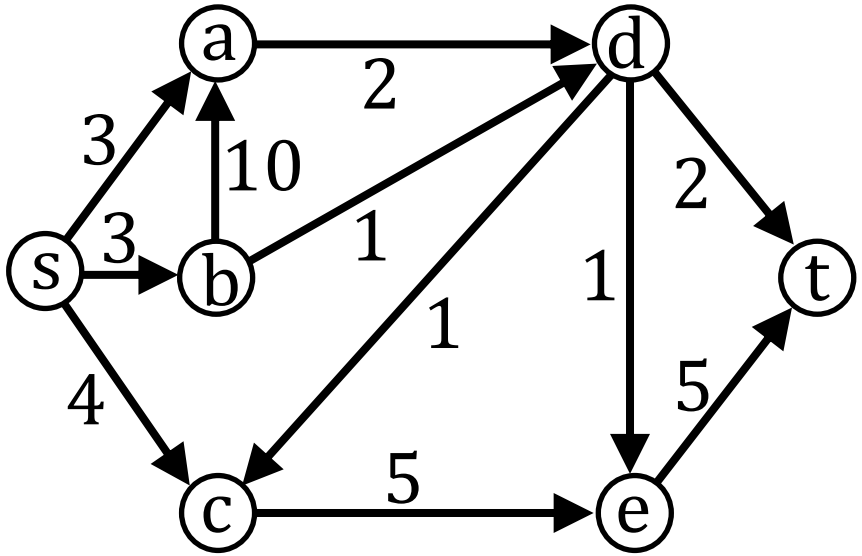
Motivation

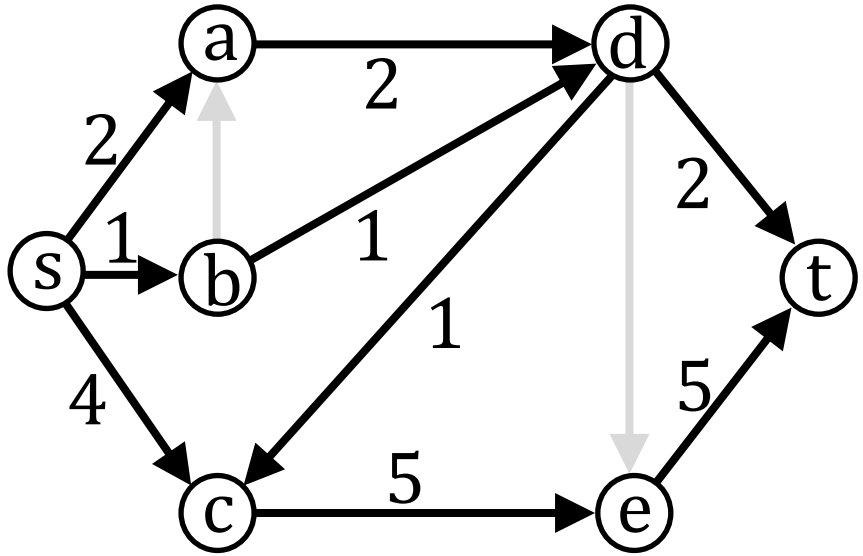
Suppose we have a directed graph that represent an oil pipeline network. Edge weight represent pipe capacity. How much oil can we transfer from source s to sink t?



Motivation

Suppose we have a directed graph that represent an oil pipeline network. Edge weight represent pipe capacity. How much oil can we transfer from source s to sink t?

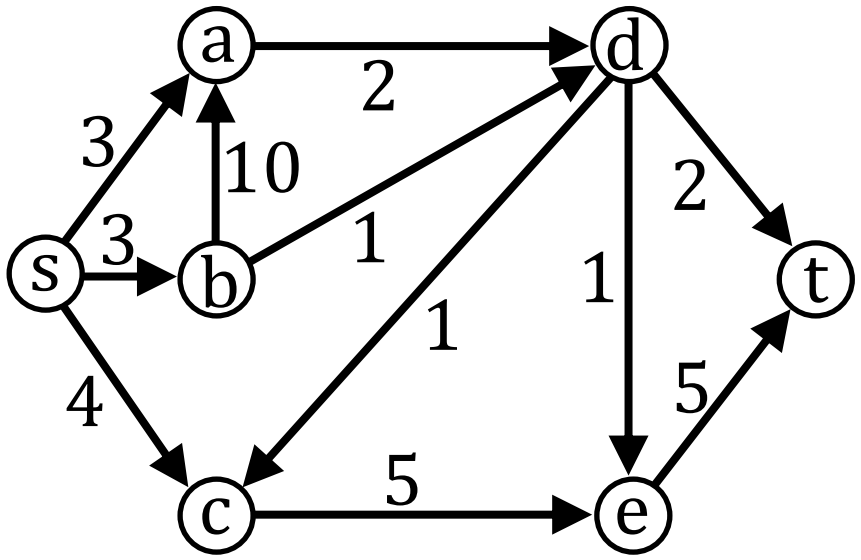




Total flow = 7

Motivation

Suppose we have a directed graph that represent an oil pipeline network. Edge weight represent pipe capacity. How much oil can we transfer from source s to sink t?



Flow networks are commonly used to model transportation networks:

- Edges carry traffic (capacity constrained).
- Nodes act as junctions between edges.

Examples:

- Internet routing.
- Road networks.
- Electricity distribution.

Flow Network

Flow Network:

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

Flow Network

Flow Network:

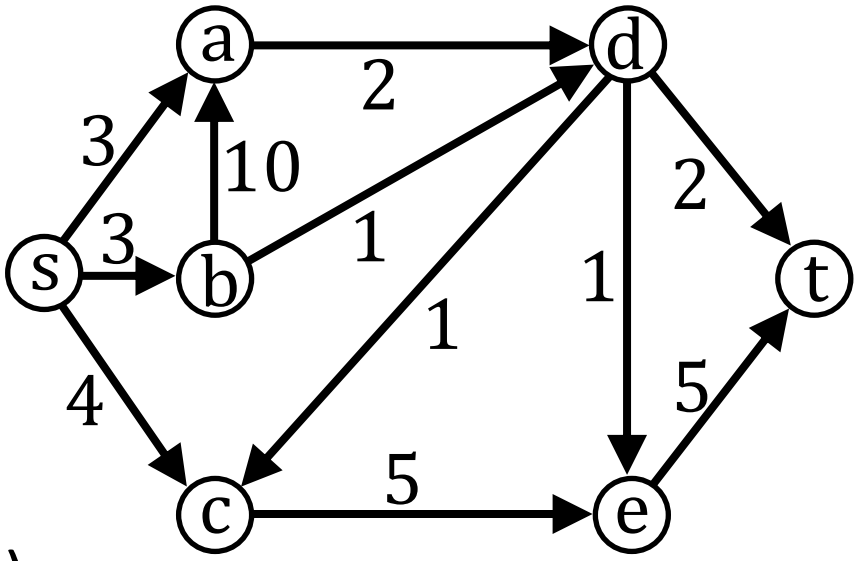
- Directed-edge graph, $G = (V, E)$.
- Non-negative edge capacity, c_e.
- Single source, s, without input edges.
- Single sink, t, without output edges.

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

Flow Network

Flow Network:

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



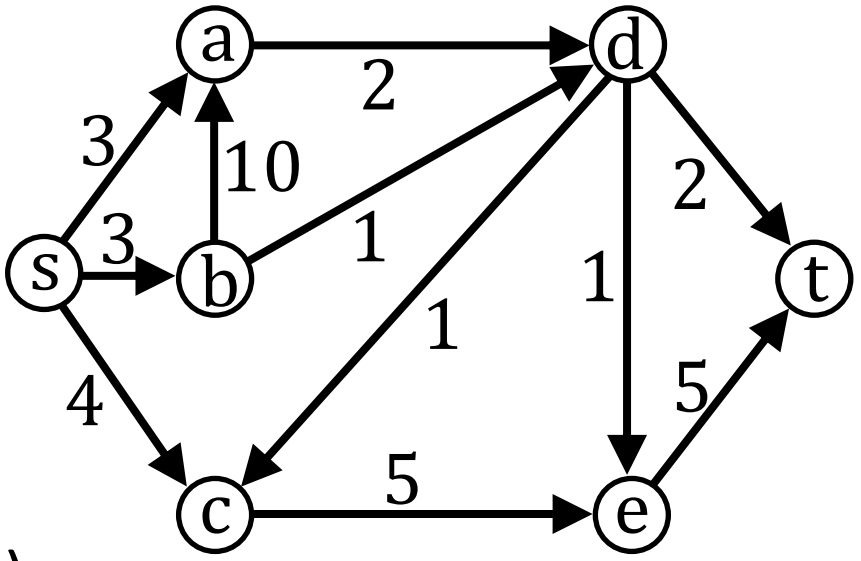
An $s - t$ flow is a function $f\colon E \to \mathbb{R}^+$ such that:

- $0 \leq f(e) \leq c_e, \forall e \in E$. (capacity constraint)

Flow Network

Flow Network:

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



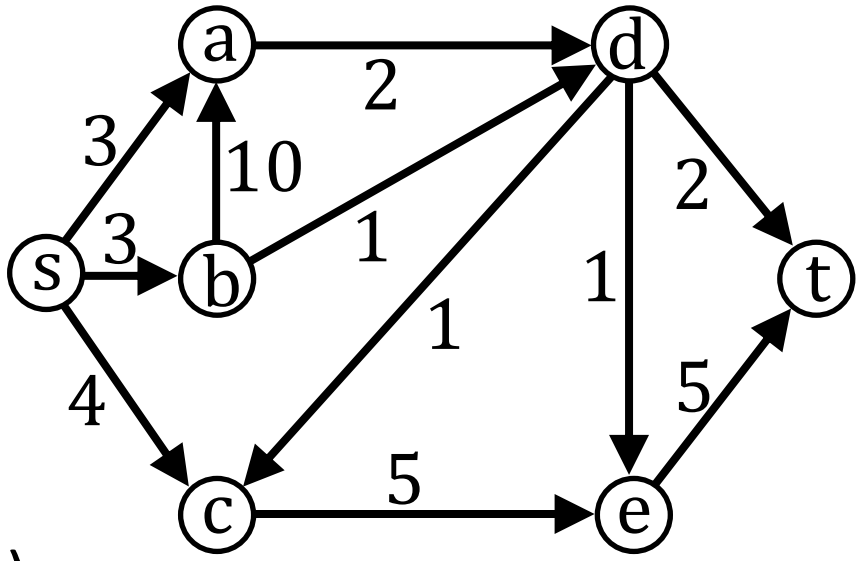
An $s - t$ flow is a function $f\colon E \to \mathbb{R}^+$ such that:

- $0 \leq f(e) \leq c_e, \forall e \in E$. (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")

Flow Network

Flow Network:

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



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

- $0 \leq f(e) \leq c_e, \forall e \in E$. (capacity constraint)
- $\sum_{e \epsilon \text{input}(v)} f(e) = \sum_{e \epsilon \text{output}(v)} f(e)\,, \forall v \in V \setminus \{s, t\}$. (conservation of flow constraint: "Everything that goes into a node has to come out, except for s and t")
- Value of flow = $val(f)$ = $\sum_{e \epsilon \text{output}(s)} f(e)$ = $\sum_{e \epsilon \text{input}(t)} f(e)$

Flow Network

Maximum Flow Problem:
Given a flow network, find the maximum possible value of flow.

Flow Network:

- Directed-edge graph, $G = (V, E)$.
- Non-negative 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)
- $\sum_{e \epsilon \text{input}(v)} f(e) = \sum_{e \epsilon \text{output}(v)} f(e)\,, \forall v \in V \setminus \{s, t\}$. (conservation of flow constraint: "Everything that goes into a node has to come out, except for s and t")
- Value of flow = $val(f)$ = $\sum_{e \epsilon \text{output}(s)} f(e)$ = $\sum_{e \epsilon \text{input}(t)} f(e)$

Flow Network

Maximum Flow Problem:
Given a flow network, find the maximum value of flow.

Flow Network:

- Direct
- Non-
-
-

How do we find max flows?

An s -

- 0
- $\sum_{e \epsilon \text{inpu}}$ (conservation of flow: … goes into a node has to come out, except … and t")
- Value of flow = $val(f) = \sum_{e \epsilon \text{output}(s)} f(e) = \sum_{e \epsilon \text{input}(t)} f(e)$

Flow Network

Maximum Flow Problem:
[illegible] a flow network, find the [illegible] value of flow.

Flow Network:

- Direct[illegible]
- Non-[illegible]
- [illegible]
- [illegible]

How do we find max flows?
We'll get to that.

An s -[illegible]

- 0[illegible]
- $\sum_{e \epsilon \text{input}}$[illegible] (conservation of flow: [illegible]oes into a node has to come out, exc[illegible]nd t")
- Value of flow = $val(f) = \sum_{e \epsilon \text{output}(s)} f(e) = \sum_{e \epsilon \text{input}(t)} f(e)$

Flow Network

Maximum Flow Problem:
Given a flow network, find the maximum value of flow.

Flow Network:

- Direct
- Non
-
-

An s -

- 0
- $\sum_{e \epsilon \text{inpu}}$ (conservation of flow: es into a node has to come out, exc and t")
- Value of flow = $val(f) = \sum_{e \epsilon \text{output}(s)} f(e) = \sum_{e \epsilon \text{input}(t)} f(e)$

How do we find max flows?

We'll get to that.

Assumptions for now:

1. We can find max flows for valid flow networks.
2. If the capacities of a flow network are integer-valued, the max flow will be too.

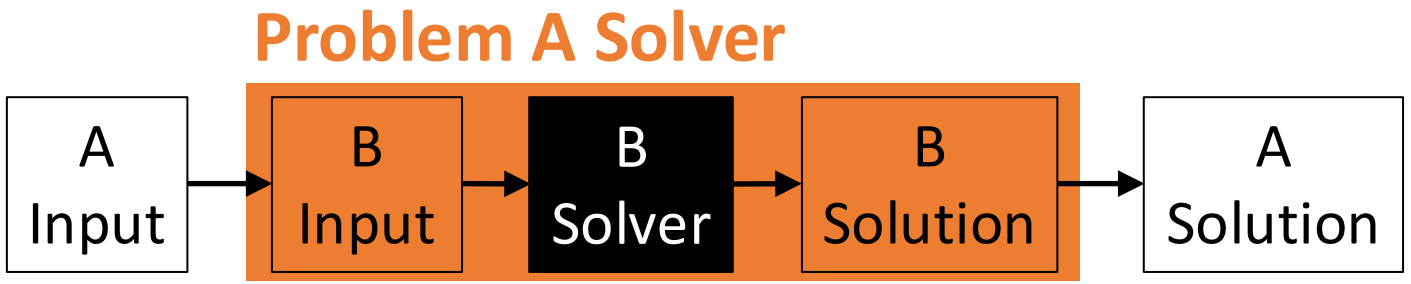
Reductions

Reductions

Reductions are a technique to solve
a problem by turning into a problem
we already have an algorithm for.

Reductions

Reductions are a technique to solve a problem by turning into a problem we already have an algorithm for.



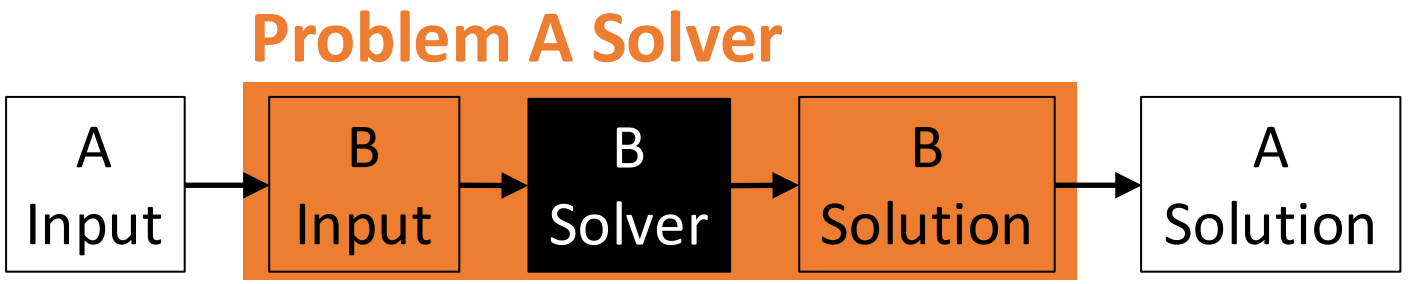
“If you can’t solve a problem, try standing on someone’s shoulders.”

- Sean, when making this slide a couple hours ago

Reductions

How to show something (B) is in NP-Complete?

1. Show it is in NP.
2. Pick some known NP-Complete problem A.
3. Show that a solver for B can solve A in polynomial extra time.



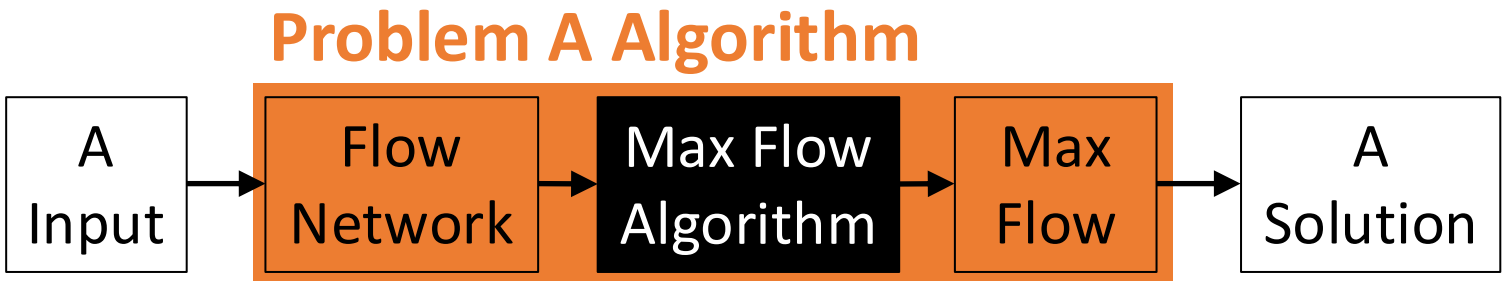
A reduces to B if A can be solved with a solver for B.

B is in NP-Complete if it satisfies two conditions:

1. $B \in NP$.
2. For some $A \in NP$-C, $A \leq_P B$.

Reductions

Reductions are a technique to solve a problem by turning into a problem we already have an algorithm for.



"If you can't solve a problem, try standing on someone's shoulders."

- Sean, when making this slide a couple hours ago

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.

work scheduling problem

Apollo 13 Filter Problem:

"We need to fit ~~this~~ into the ~~hole for this~~, using nothing but ~~that~~."

Max Flow algorithm

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.

work scheduling problem

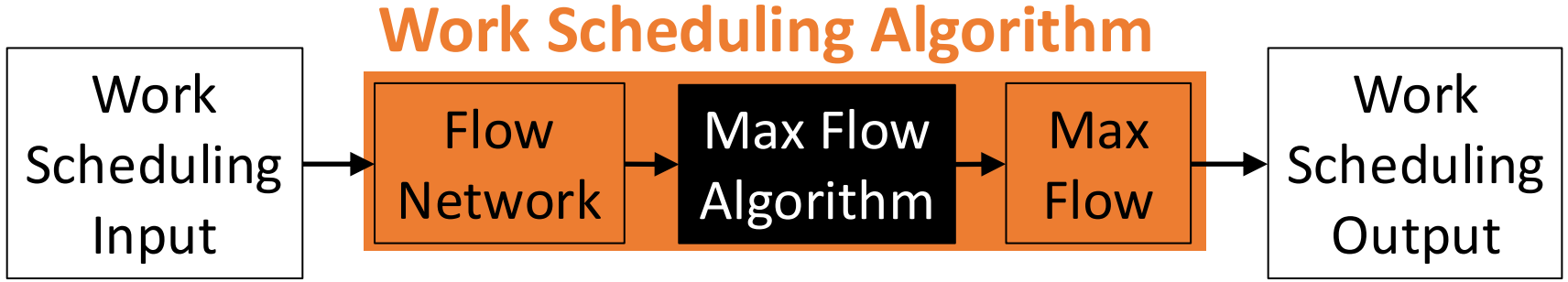
Apollo 13 Filter Problem:

"We need to fit ~~this~~ into the ~~hole for this~~, using nothing but ~~that~~."



Max Flow algorithm

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:

??

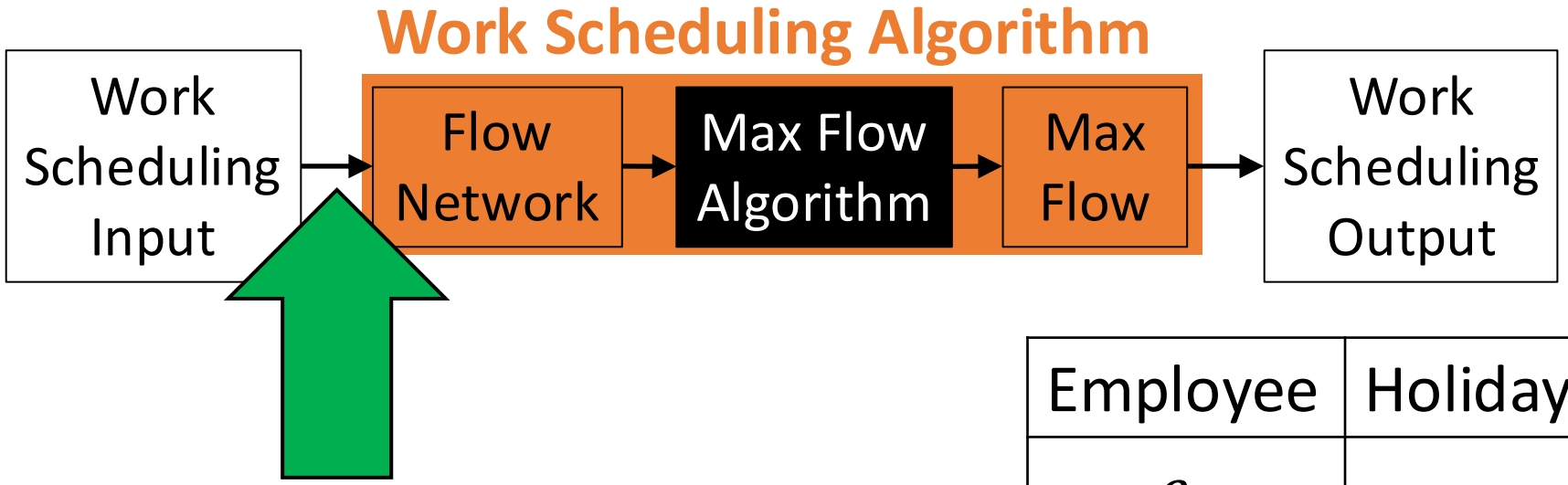
Employee	Holiday(s) they can work
e_1	h_1, h_2
e_2	h_2, h_3

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:



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

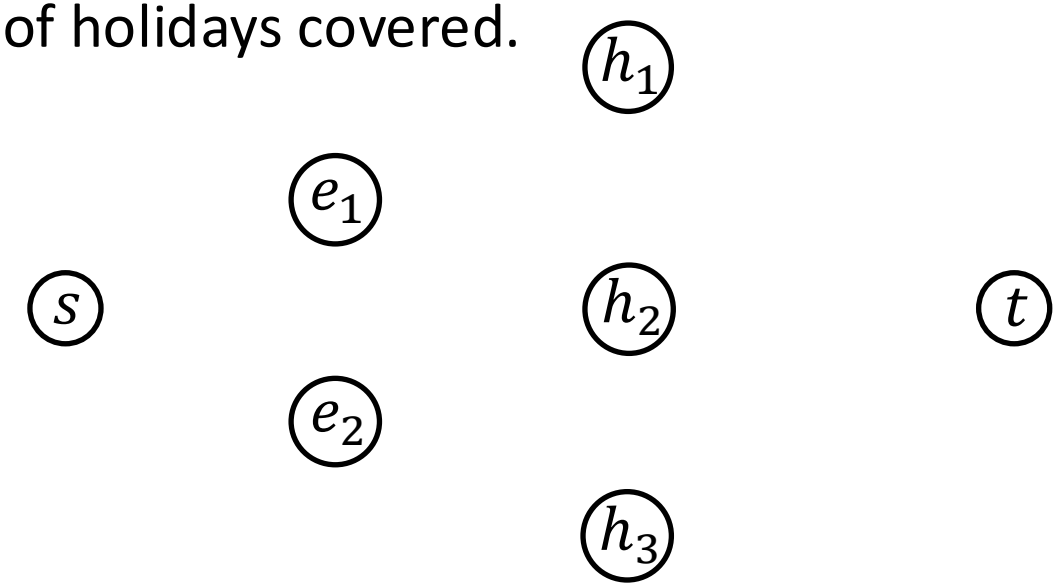
Work Scheduling

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.



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

Work Scheduling

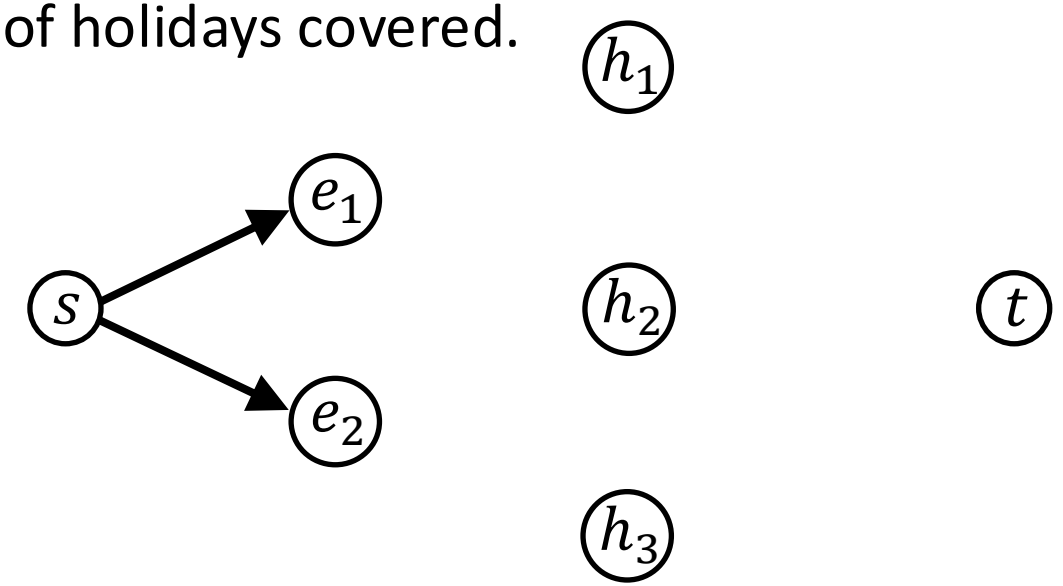
Problem: We need to make holiday schedules for our employees. Each employee has a set of holidays that they are able to work. Each employee should work at most 3 holidays. We want to maximize the number of holidays covered.

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 ???



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

Work Scheduling

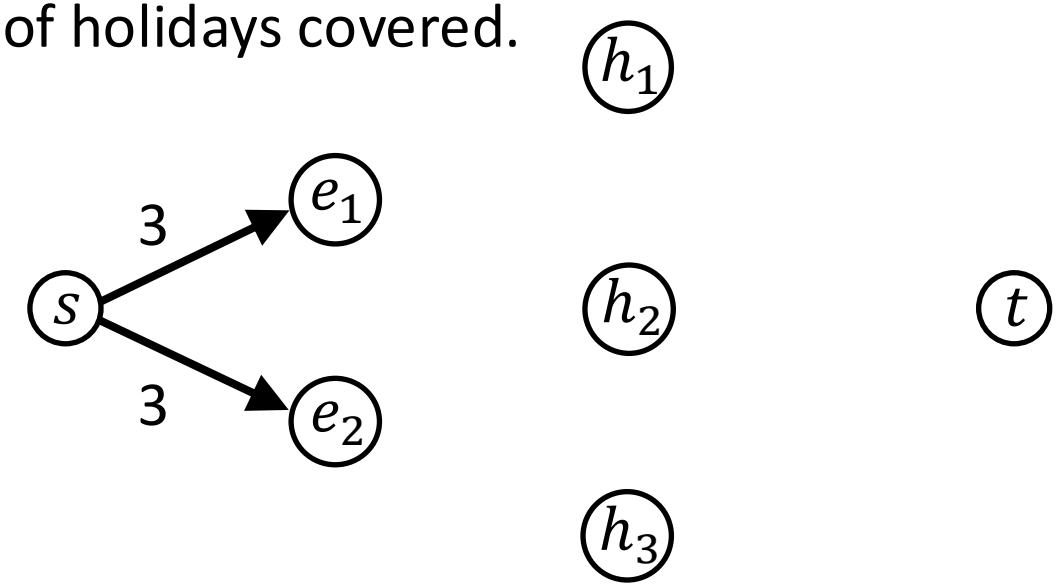
Problem: We need to make holiday schedules for our employees. Each employee has a set of holidays that they are able to work. Each employee should work at most 3 holidays. We want to maximize the number of holidays covered.

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.



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.

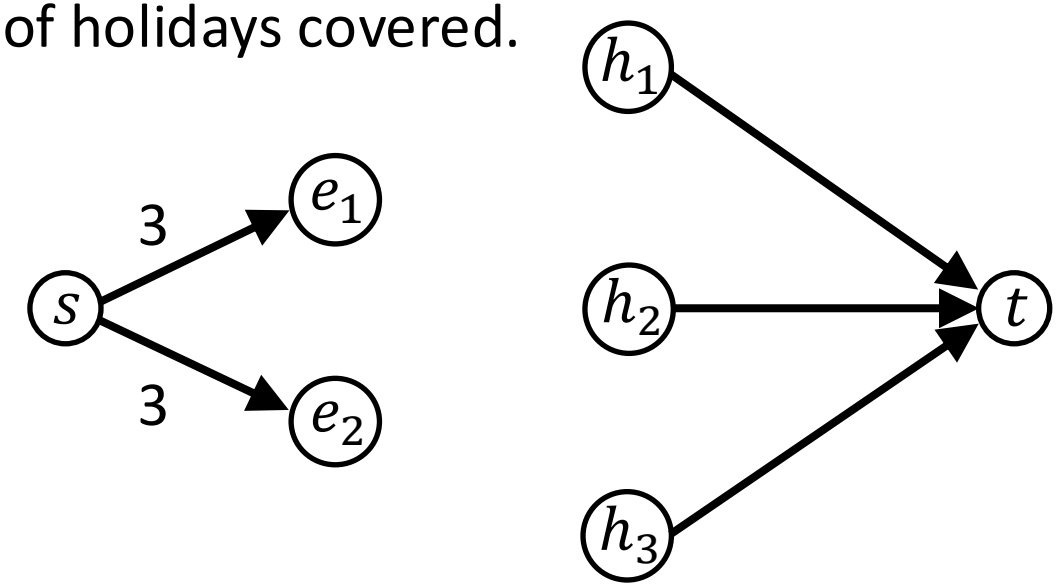
Algorithm:

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 ???



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.

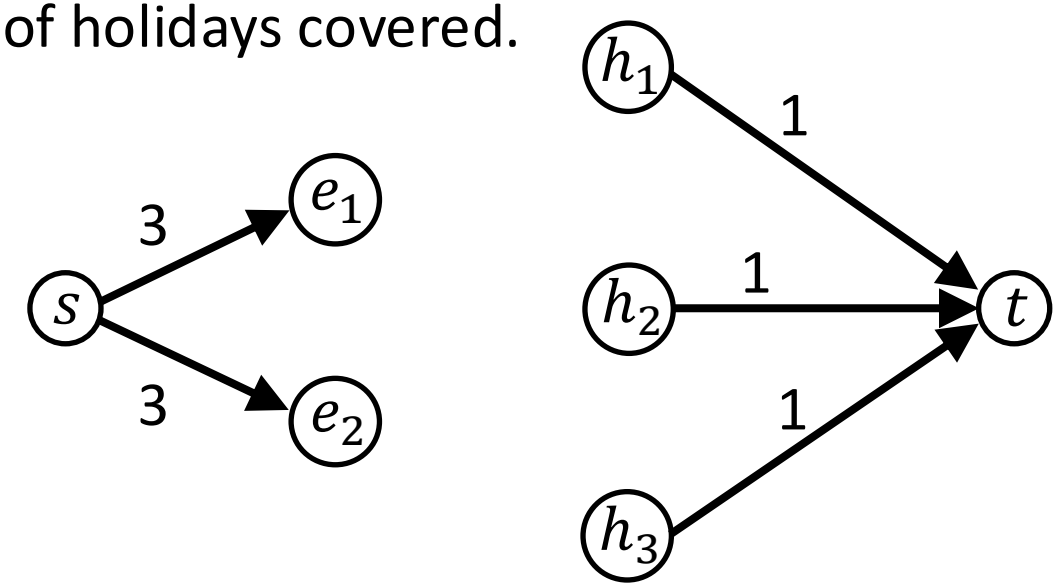
Algorithm:

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.



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.

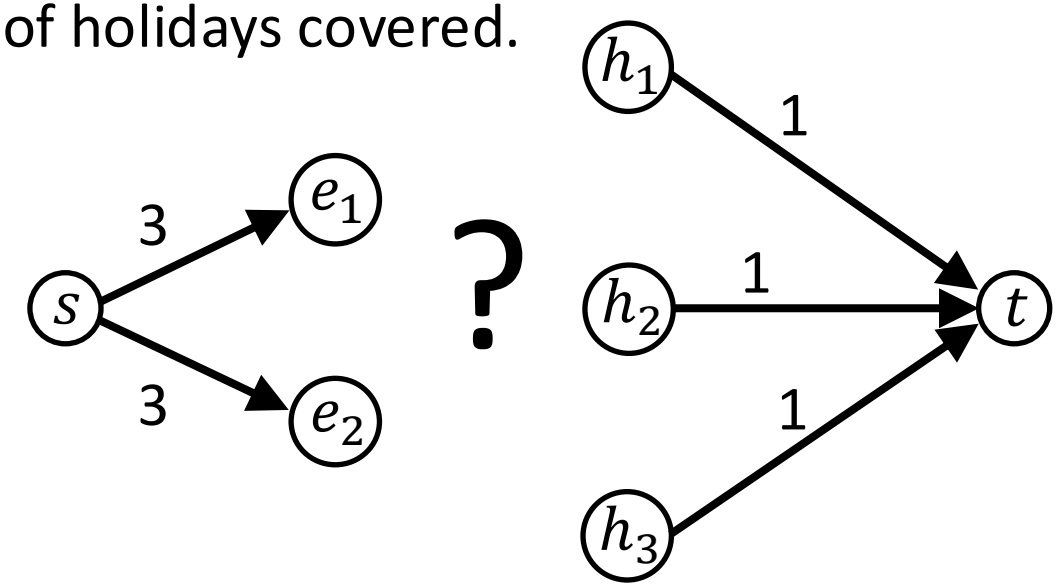
Algorithm:

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.



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:

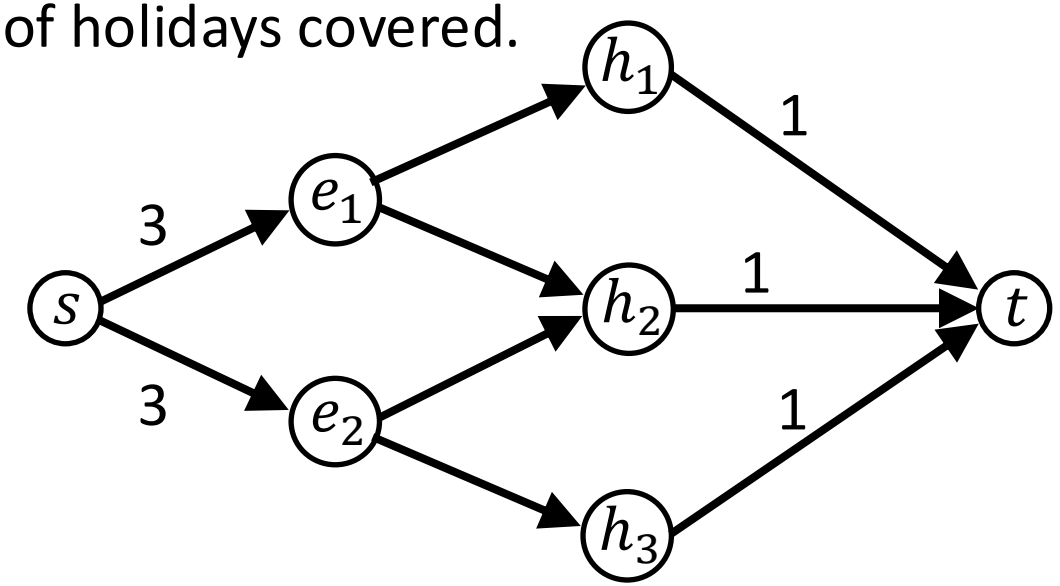
1. Build flow network:

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 ???



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:

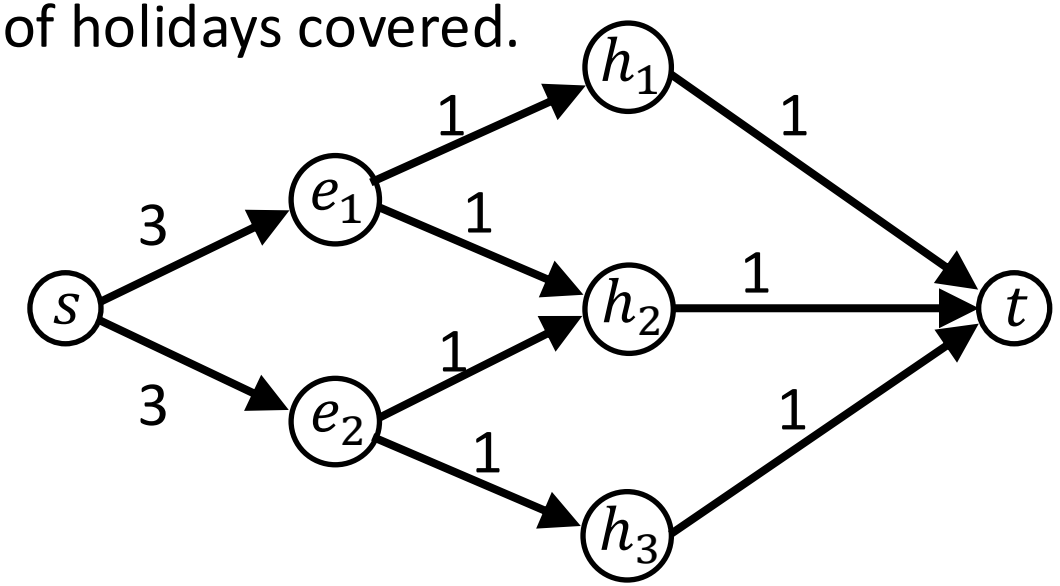
1. Build flow network:

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.

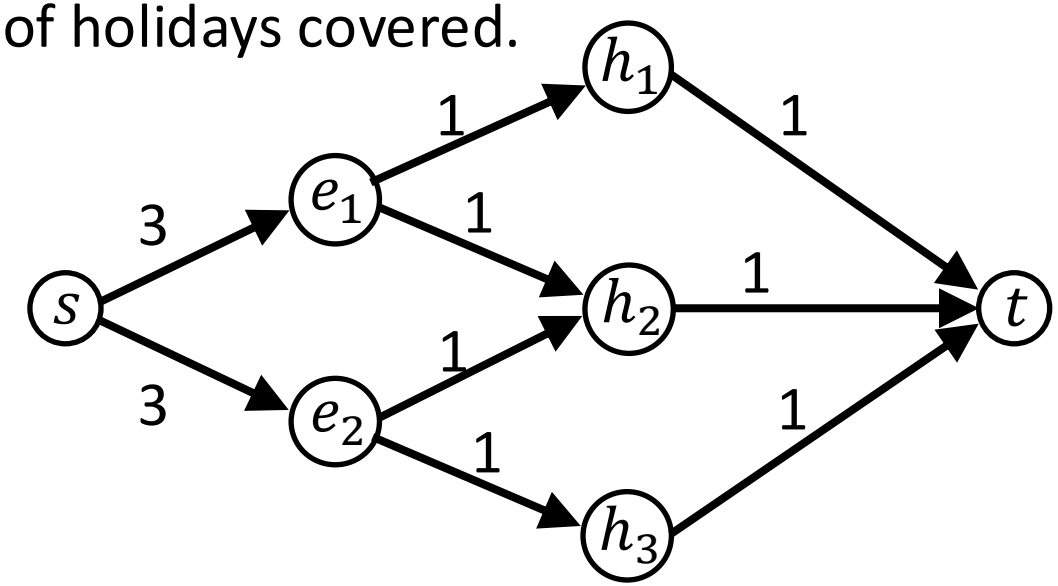
d) If an employee is able to work a holiday, connect them with a capacity of 1.



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:

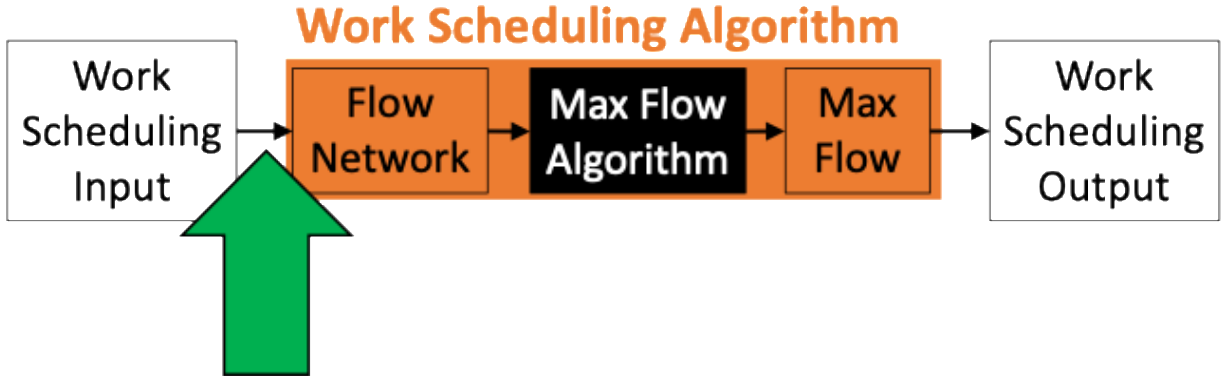
1. Build flow network:

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.

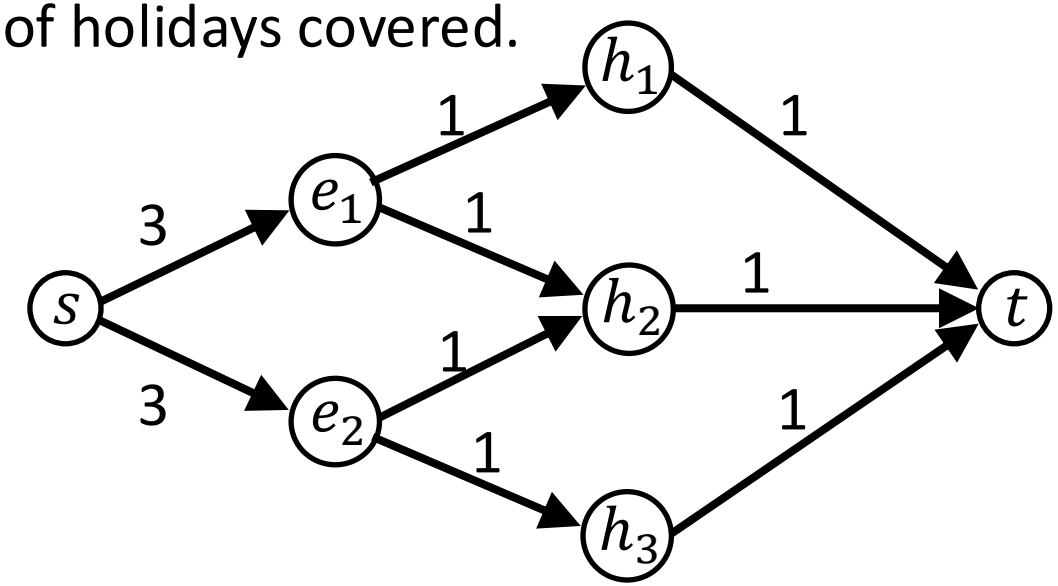


Work Scheduling

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. ??

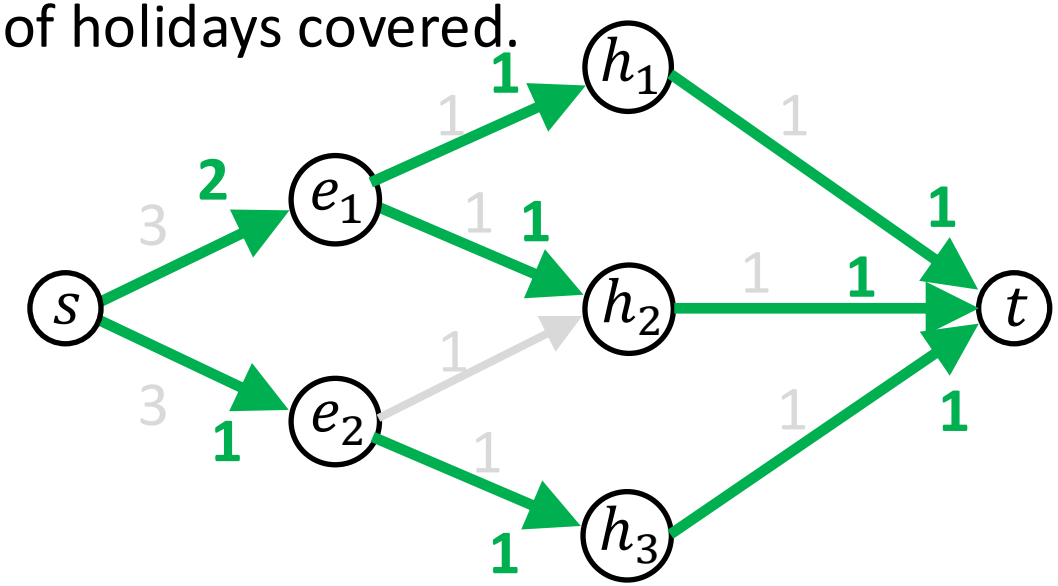


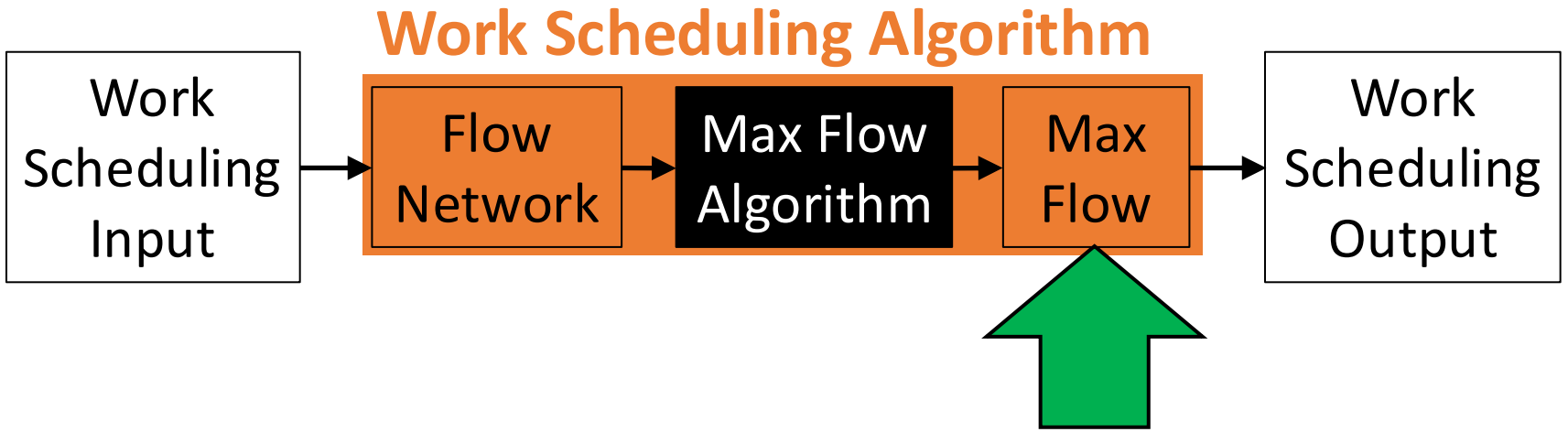
Work Scheduling

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.



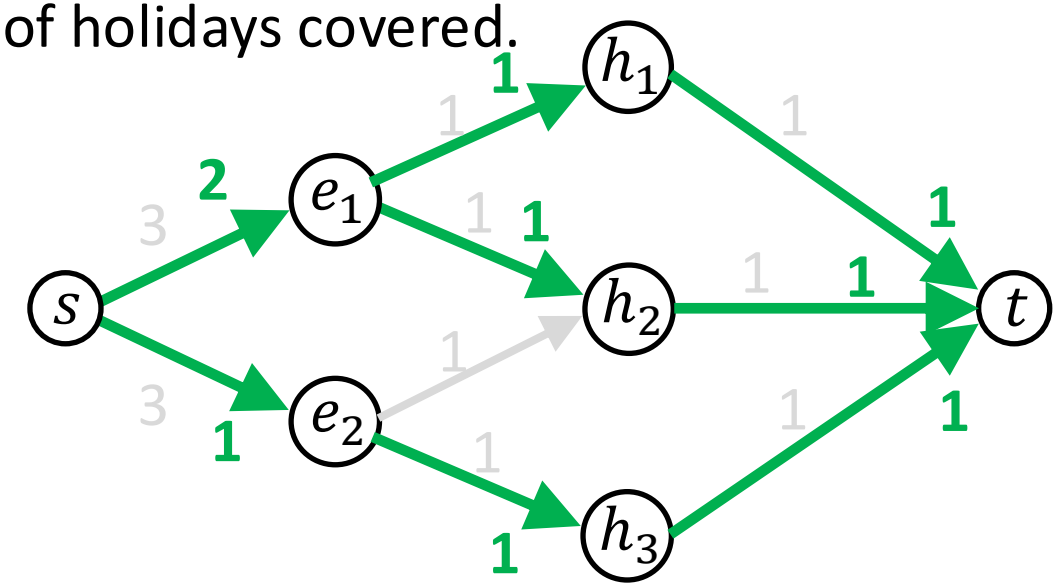


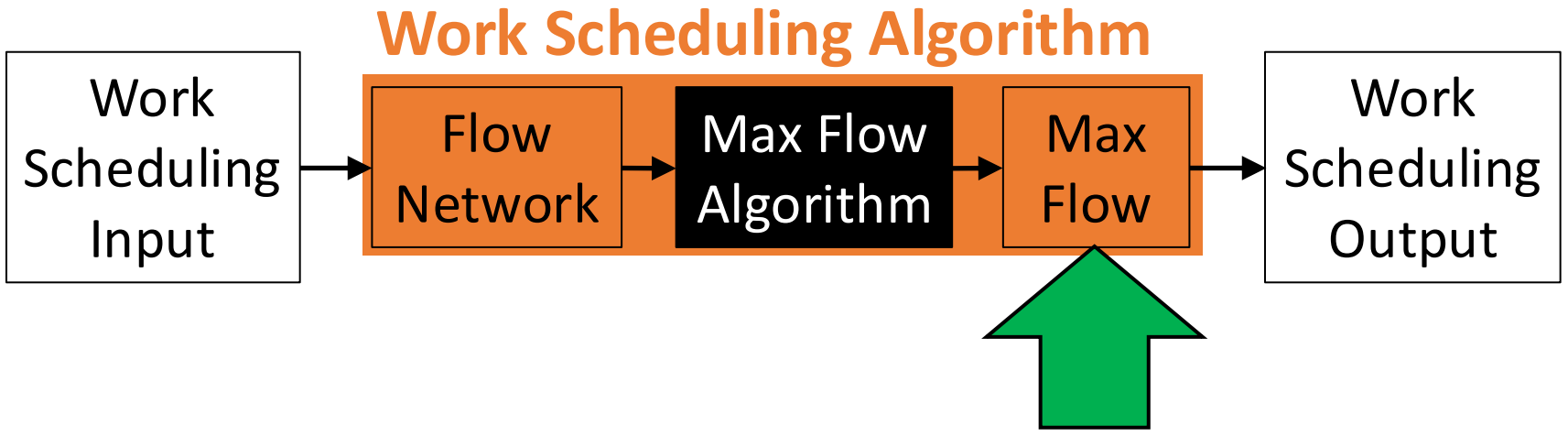
Work Scheduling

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. ??



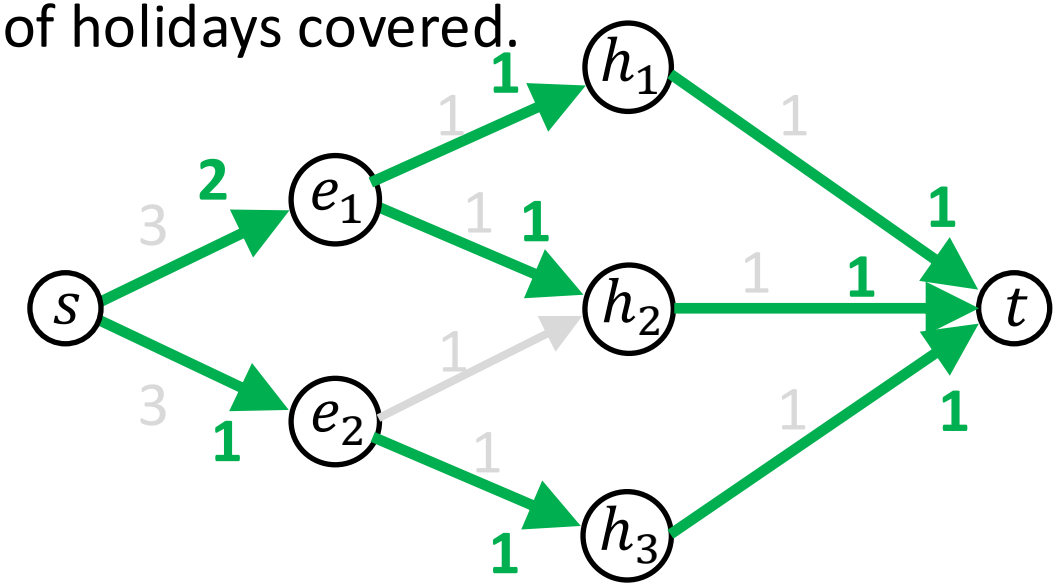


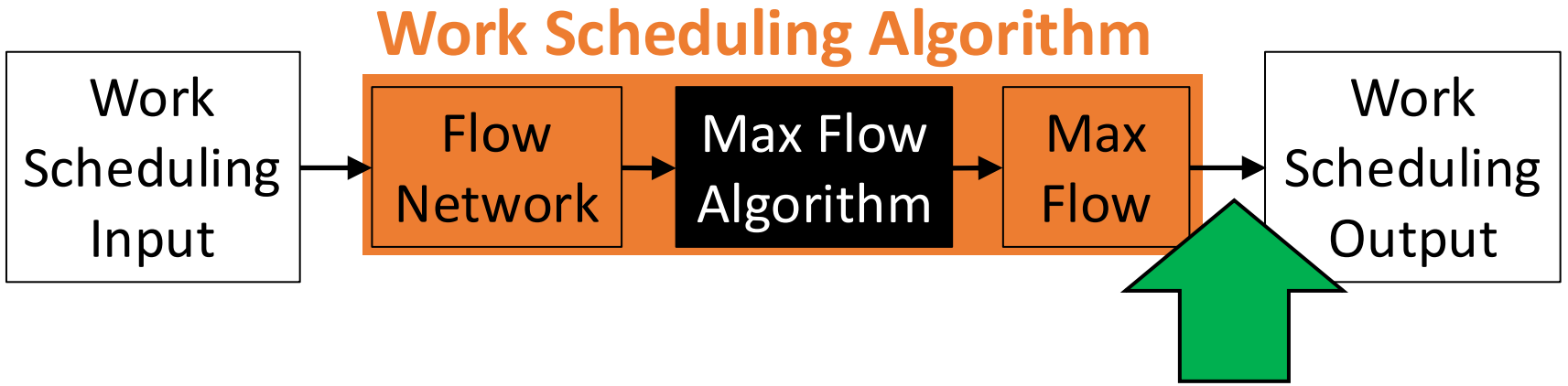
Work Scheduling

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>



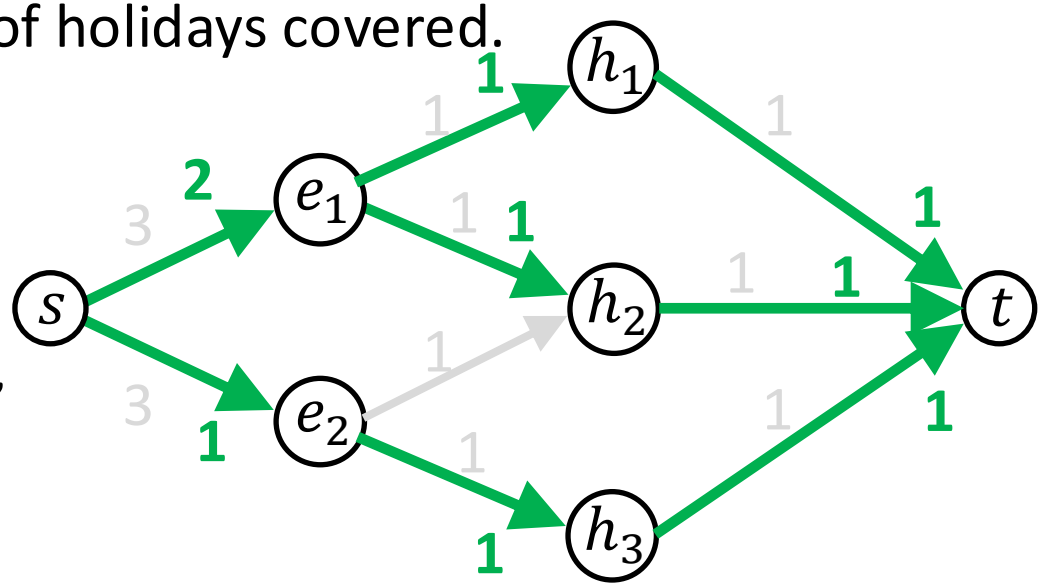


Work Scheduling

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.



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