

3 March 2024

Plan.

- * Baseball Elimination (KT § 7.12)
- * Announcements
- * Reduction to Flow.

Baseball Elimination Problem

Given: * List of teams $\langle t_0, \dots, t_k \rangle$

* Current standings $\langle w_0, \dots, w_k \rangle$

$w_i =$ current # of wins by t_i

* Remaining games $\langle g_1, \dots, g_n \rangle$

$g_j = (t_i, t_k)$

Game g_j between t_i and t_k .

Question: Can t_0 finish with the most wins?

Baseball Elimination Problem

<u>Teams</u>	<u>Wins</u>	<u>Games</u>	BOS ?
BOS	90	(BOS, NYN) (BOS, TB)	
NYN	88	(TB, BAL) (NYN, TB)	
BAL	86		
TB	91		

Baseball Elimination Problem

<u>Teams</u>	<u>Wins</u>	<u>Games</u>	BOS ?
BOS	90	(BOS, NYY) (BOS, TB)	
NYY	88	(TB, BAL)	
BAL	86	(NYY, TB)	
TB	91		

<u>Teams</u>	<u>Wins</u>	<u>Games</u>	NYY ?
BOS	90	(BOS, NYY) (BOS, TB)	
NYY	88	(TB, BAL)	
BAL	86	(NYY, TB)	
TB	90		

Baseball Elimination Problem

Given:

- * List of teams $\langle t_0, \dots, t_k \rangle$
- * Current standings $\langle w_0, \dots, w_k \rangle$
- * Remaining games $\langle g_1, \dots, g_n \rangle$

Question: Can t_0 finish with the most wins?

Observation 0. WLOG assume t_0 has no more games.

Baseball Elimination Problem

Given:

- * List of teams $\langle t_0, \dots, t_k \rangle$
- * Current standings $\langle w_0, \dots, w_k \rangle$
- * Remaining games $\langle g_1, \dots, g_n \rangle$

Question: Can t_0 finish with the most wins?

Observation 0. WLOG assume t_0 has no more games.

- * Search through all games involving t_0
- * Assign t_0 the win. (i.e. $w_0 \leftarrow w_0 + 1$)
- * Remove game [Preprocessing takes $O(n)$ time]

Baseball Elimination Problem

Given:

- * List of teams $\langle t_0, \dots, t_k \rangle$
- * Current standings $\langle w_0, \dots, w_k \rangle$
- * Remaining games $\langle g_1, \dots, g_n \rangle$

Question: Can t_0 finish with the most wins?

Observation: Every remaining game results in an additional win for some team.

Baseball Elimination Problem

Given:

- * List of teams $\langle t_0, \dots, t_k \rangle$
- * Current standings $\langle w_0, \dots, w_k \rangle$
- * Remaining games $\langle g_1, \dots, g_n \rangle$

Question: Can t_0 finish with the most wins?

Observation. Every remaining game results in an additional win for some team.

Can we allocate all of these wins (i.e. games) such that t_0 is the leader?

Announcements

* HW 2 Grades Released

* Prelim #1 Solutions Posted to Canvas

* HW3 Ongoing

Observations

- * Preprocess input so that t_0 wins every remaining game
 - * Every other game results in some team gaining a win.
-

How should we "assign" remaining games to teams?

BOS

90

(BOS, NYN)

NYN

88

(BOS, TB)

BAL

86

(TB, BAL)

TB

91

(NYN, TB)

~~BOS~~

NYN

BAL

TB

~~90~~

92

88

86

91

~~(BOS, NYN)~~

~~(BOS, TB)~~

(TB, BAL)

(NYN, TB)

~~BOS~~

NYN

BAL

TB

~~90~~

92

88

86

91

~~(BOS, NYN)~~

~~(BOS, TB)~~

(TB, BAL)

(NYN, TB)

TB, BAL

NYN, TB

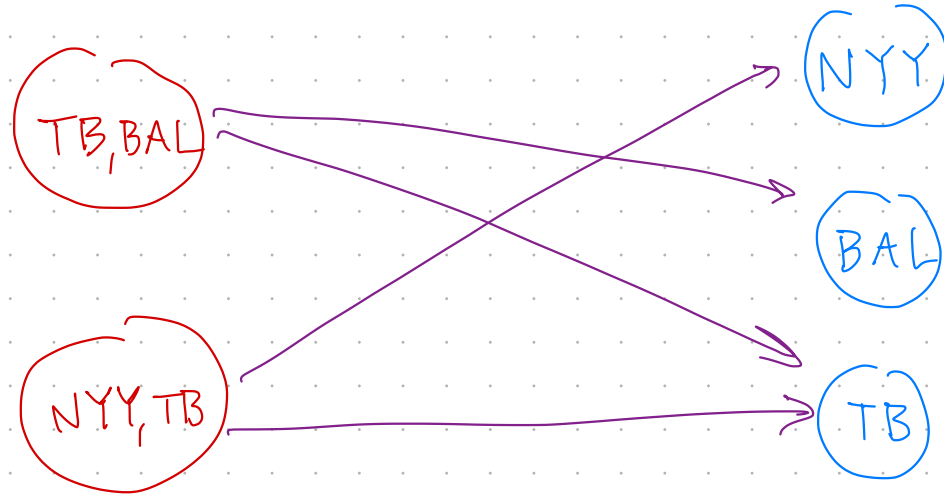
NYN

BAL

TB

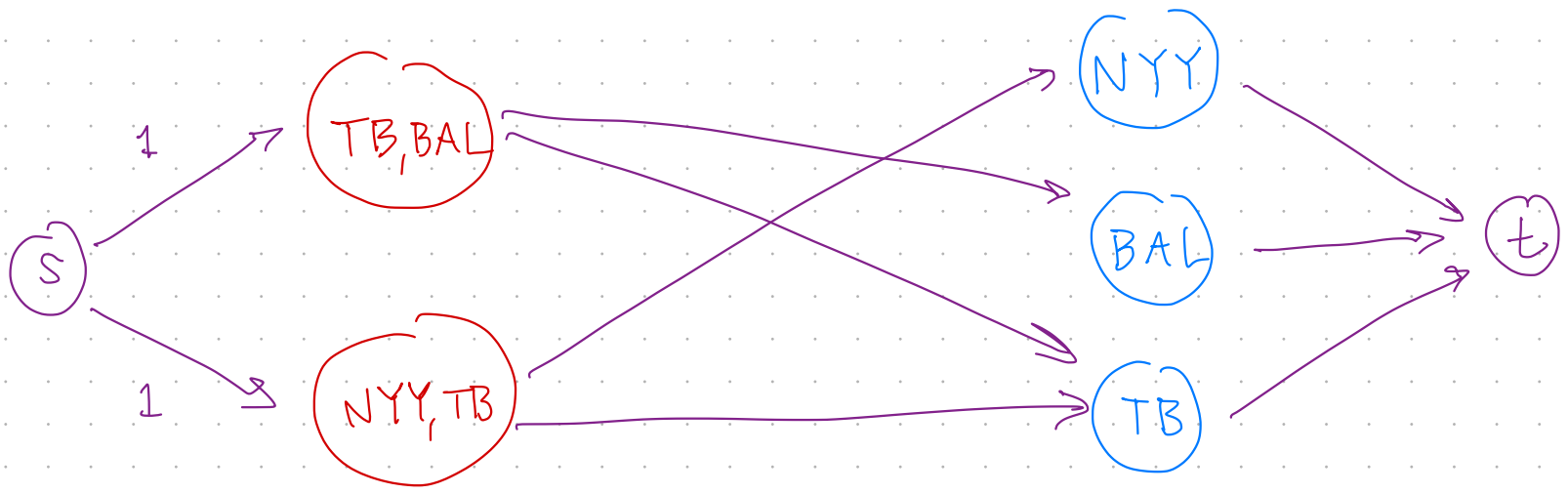
BOS	90	92
NYN	88	
BAL	86	
TB	91	

- ~~(BOS, NYN)~~
- ~~(BOS, TB)~~
- (TB, BAL)
- (NYN, TB)



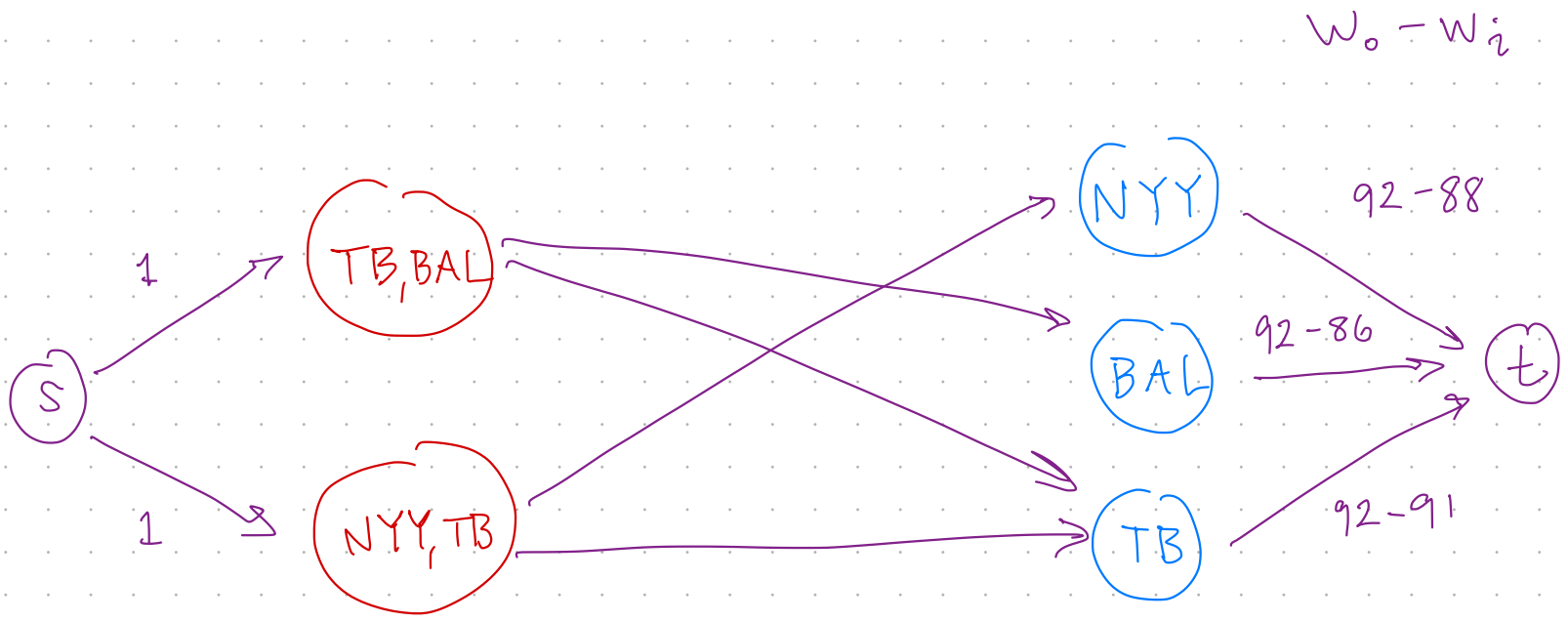
BOS	90	92
NYN	88	
BAL	86	
TB	91	

- ~~(BOS, NYN)~~
- ~~(BOS, TB)~~
- (TB, BAL)
- (NYN, TB)



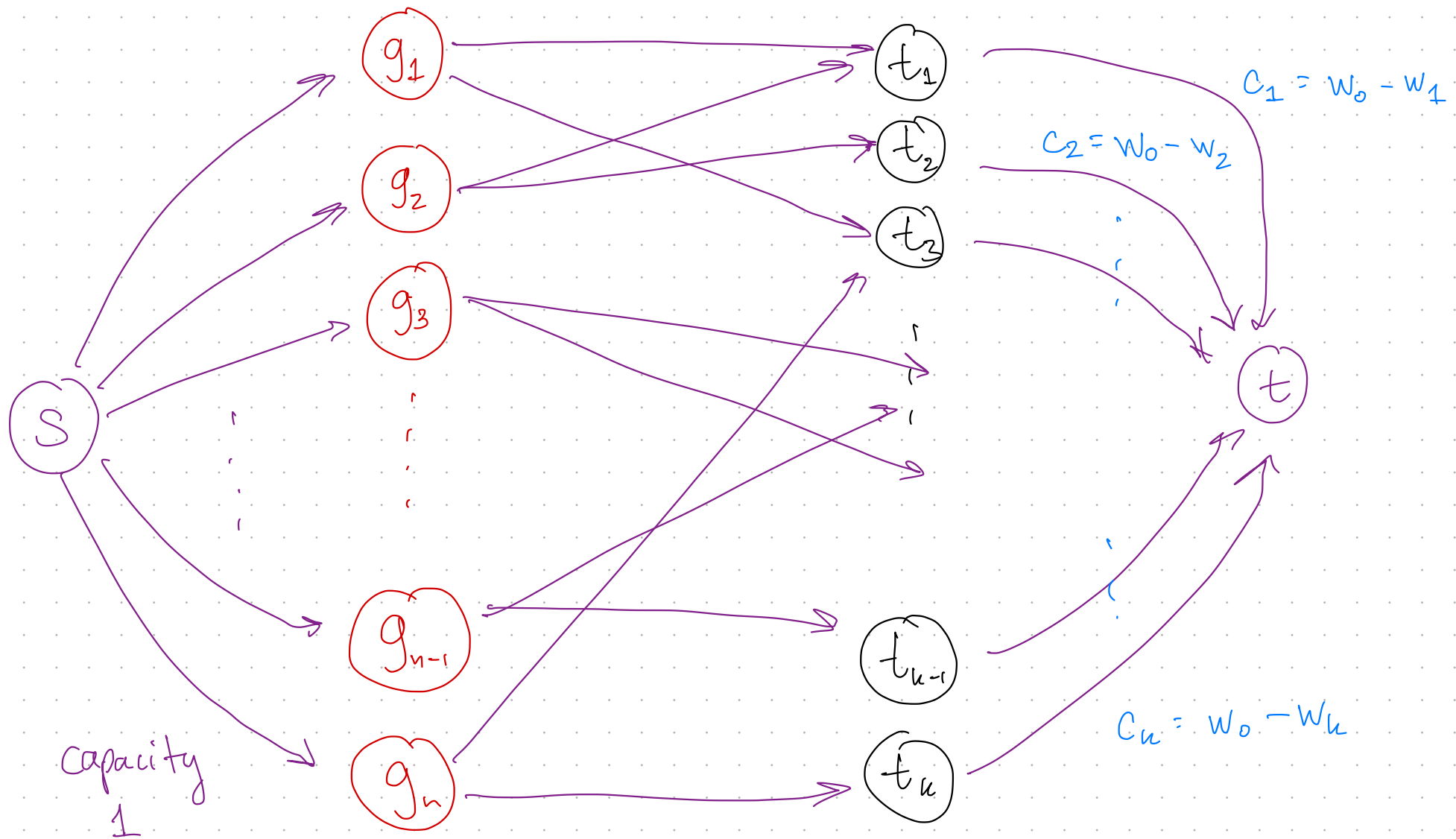
BOS	90	92
NYN	88	
BAL	86	
TB	91	

- ~~(BOS, NYN)~~
- ~~(BOS, TB)~~
- (TB, BAL)
- (NYN, TB)



$92 = w_0 = \# \text{ wins } t_0 \text{ can achieve}$

$w_i = \# \text{ wins } t_i \text{ has before remaining games assigned.}$



$$C_i = w_0 - w_i$$

← Max # of additional wins s.t. $w_i + C_i \leq w_0$.

Correctness. The max flow in G equals n
if and only if t_0 can finish with the most wins
after the remaining n games.

Correctness. The max flow in G equals n
if and only if t_0 can finish with the most wins
after the remaining n games.

(\Rightarrow) If the max flow is n ,
there is an "allocation" of wins s.t.
 t_0 finishes in 1st.

(\Leftarrow) If t_0 can finish w/ the most wins,
the max flow is n .

Assume flows are INTEGER.

(Ford-Fulkerson proves this is WLOG)

(\Rightarrow) Consider a flow f of n units

\hookrightarrow Devise an "allocation" of wins to teams as follows.

For each unit of $g_j \rightarrow t_i$ flow,

Assign 1 additional win to t_i

$$w_i \leftarrow w_i + 1$$

(\Rightarrow) Consider a flow f of n units

\hookrightarrow Devise an "allocation" of wins to teams as follows.

For each unit of $g_j \rightarrow t_i$ flow,

Assign 1 additional win to t_i

By capacity constraints. team t_i is "allocated"
at most $C_i = W_0 - W_i$ units.

\Rightarrow if team t_i wins each game allocated
then $W_i + C_i \leq W_0$.

(\Rightarrow) Consider a flow f of n units

↳ Devise an "allocation" of wins to teams as follows.

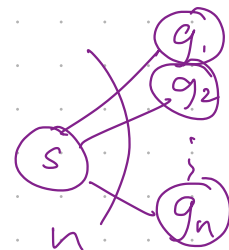
For each unit of $g_j \rightarrow t_i$ flow,

Assign 1 additional win to t_i

By capacity constraints. team t_i is "allocated"
at most $C_i = W_0 - W_i$ units.

\Rightarrow if team t_i wins each game allocated
then $W_i + C_i \leq W_0$.

By construction. n units of flow covers all
 n remaining games



(\Leftarrow) Suppose t_0 can finish w/ most wins.

\hookrightarrow Construct a flow of n units
from results.

(\Leftarrow) Suppose t_0 can finish w/ most wins.

\hookrightarrow Construct a flow of n units from results.

* Assign 1 to $s \rightarrow g$ for each game g .

* If team t_i wins game $g = (t_i, t_j)$

\hookrightarrow Assign 1 unit of flow from $g \rightarrow t_i$.

* Assign total # wins to $t_i \rightarrow t$.

Capacity & Conservation are satisfied ✓

Paradigm for solving optimization problems

* Understand problem

↳ variables to be determined

↳ constraints on variables

remaining wins



assigned to teams



s.t. to has most

Paradigm for solving optimization problems

* Understand problem

↳ variables to be determined

↳ constraints on variables

remaining wins



assigned to teams

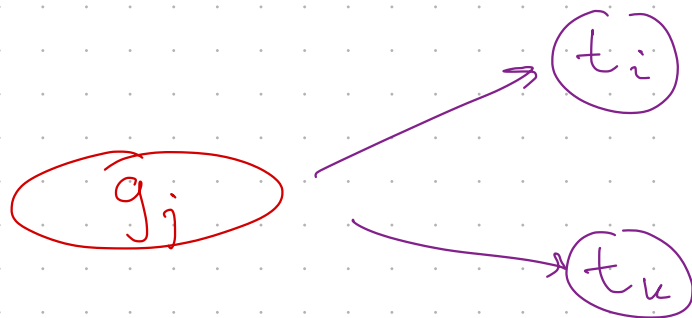


s.t. t_0 has most

* Write problem as flow instance

↳ variables = flow / cut

↳ constraints = "gadgets"



← game gadget

Paradigm for solving optimization problems

* Understand problem

↳ variables to be determined

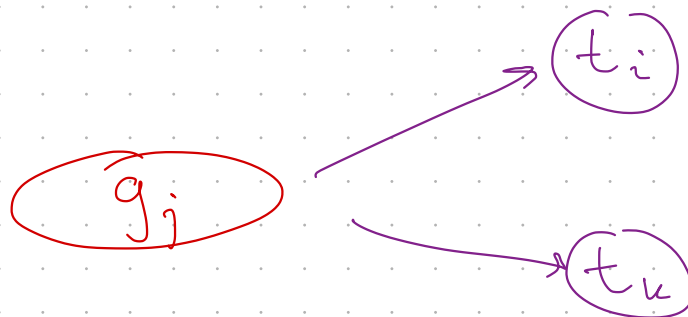
↳ constraints on variables

remaining wins
↓
assigned to teams
↓
s.t. t_i has most

* Write problem as flow instance

↳ variables = flow / cut

↳ constraints = "gadgets"



← game gadget

* Argue why solution to flow gives solution to your problem

↳ Max Flow / Min Cut

Baseball Elimination subject to constraints?

Can to win most games s.t. ?

Example Constraints

* All teams win at most k games

* Every team wins at most $\frac{2}{3}$ of their games against any other team.

Baseball Elimination subject to constraints?

Can to win most games s.t. _____ ?

Example Constraints

* All teams win at most k games

$$C(t_i, t) = \min \{k, w_0 - w_i\}$$

* Every team wins at most $2/3$ of their games against any other team.

Baseball Elimination subject to constraints?

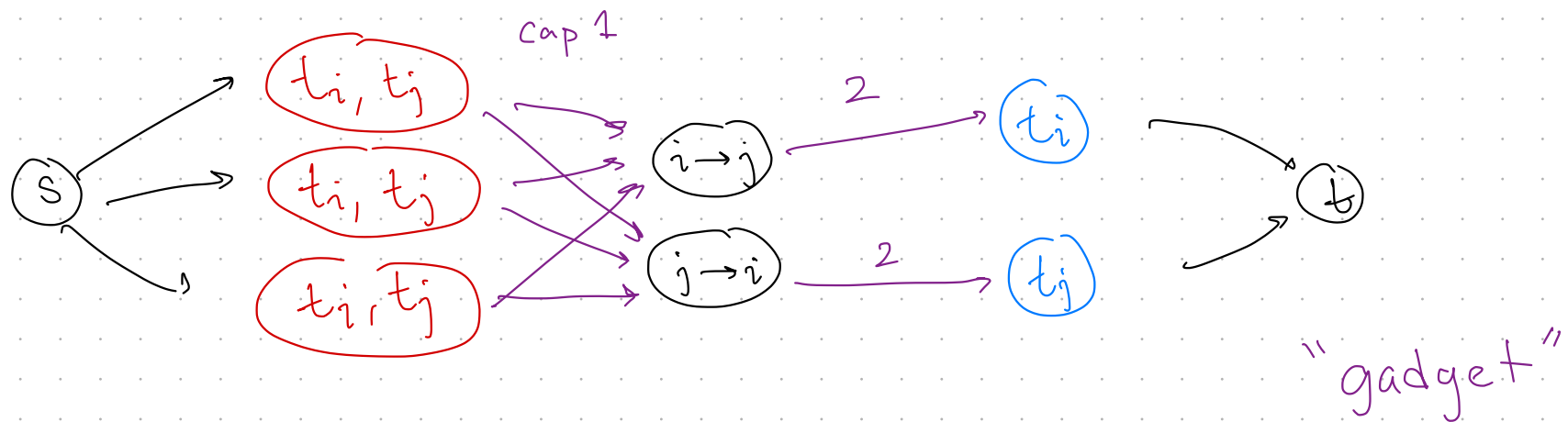
Can to win most games s.t. _____ ?

Example Constraints

* All teams win at most k games

$$C(t_i, t) = \min \{k, w_0 - w_i\}$$

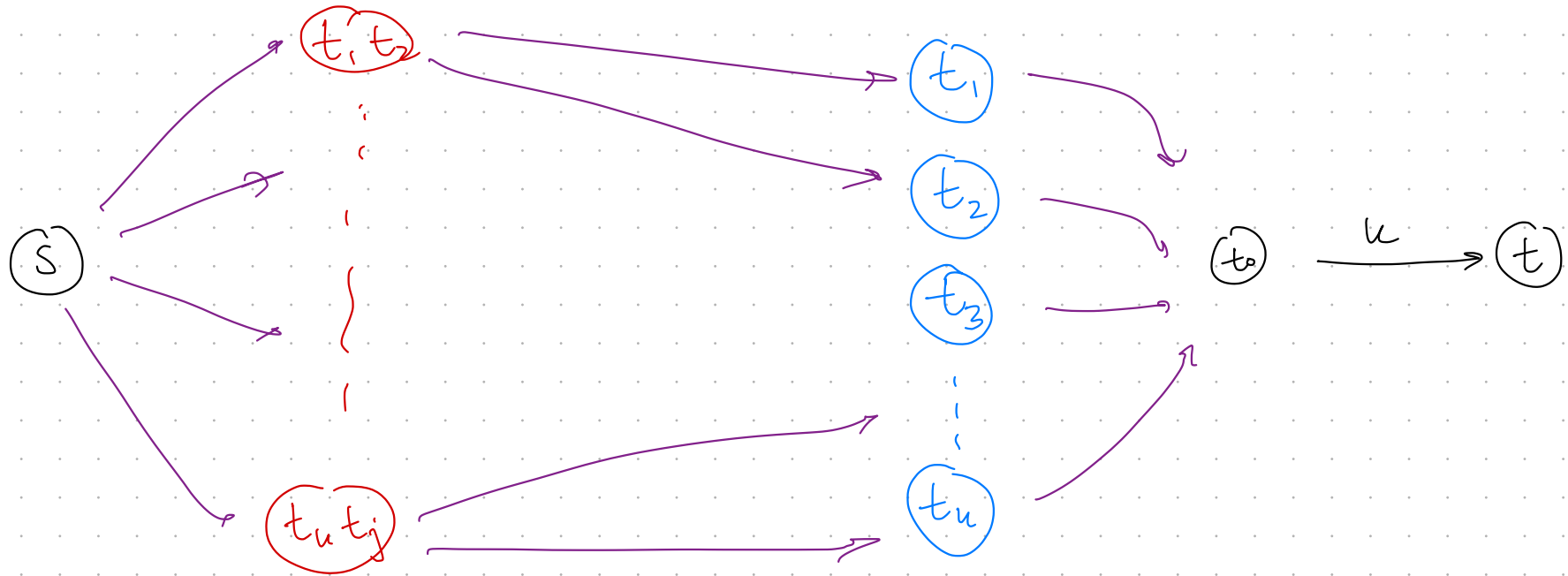
* Every team wins at most $2/3$ of their games against any other team.



Baseball Elimination subject to constraints?

Can to win most games s.t. _____?

* At most c teams win a remaining game.

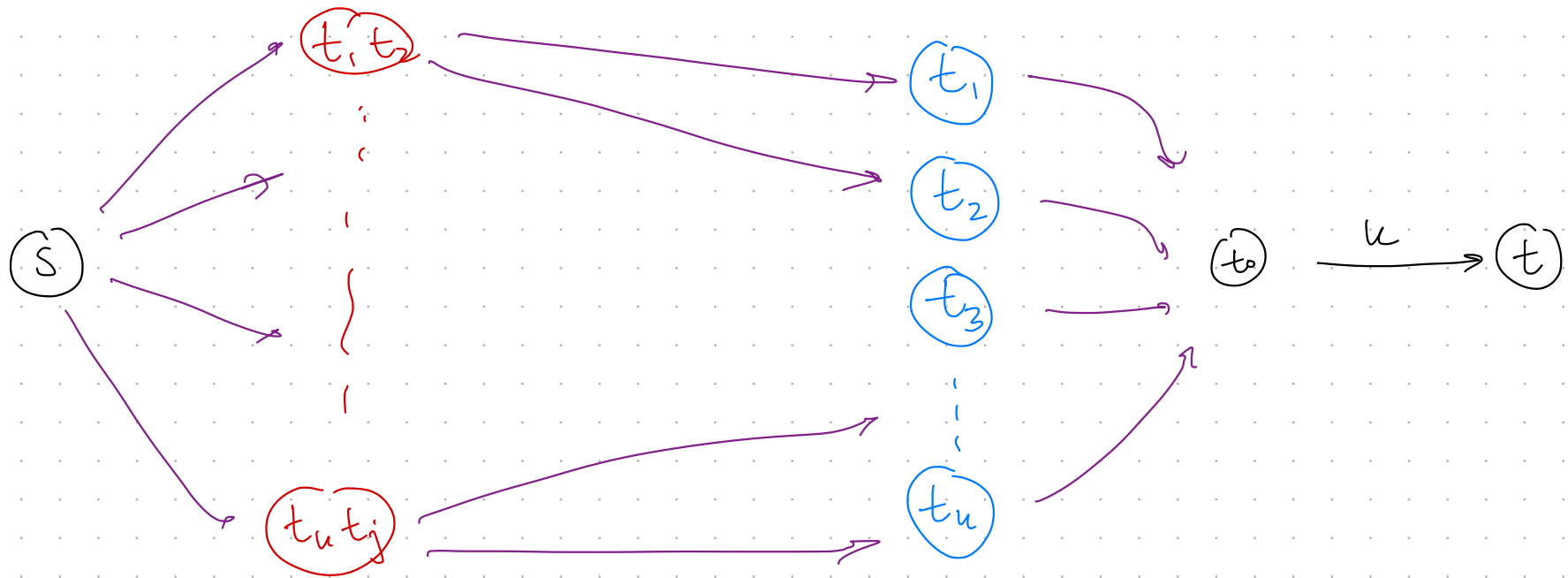


Baseball Elimination subject to constraints?

Can to win most games s.t. _____?

* At most c teams win a remaining game.

↖ constraint on teams, not on games



BE CAREFUL! Some constraints cannot be encoded as flow!