

# デモンズプラン

原案: public\_sate

テスター: haji / uku / gacho

# 問題概要

- N体の使い魔とM体の悪魔がいます。
- 各使い魔と悪魔の距離が与えられます。
- かならず全ての悪魔にちょうど1体の使い魔が訪問しなければならない。
- 使い魔は複数の悪魔を訪問できる。
- 以下の条件を満たす使い魔と悪魔の割り振り方をしてください。
  - a. 使い魔1体あたりの訪問する悪魔の数の差の最大を最小化する。
  - b. aを満たした上で訪問する悪魔と担当する使い魔の距離の最大を最小化する。

# 解法

- a. 使い魔1体あたりの訪問する悪魔の数の差の最大を最小化する。

について

悪魔の数 $M$ 、使い魔の数 $N$ としたとき  
 $M/N$ が割り切れれば

使い魔全員が $M/N$ の悪魔を担当することで差は0

割り切れなければ

$\text{floor}(M/N) + 1$ と $\text{floor}(M/N)$ を担当する悪魔が存在するので差が1

# 解法

b.  $a$ を満たした上で訪問する悪魔と担当する使い魔の距離の最大を最小化する。

について、

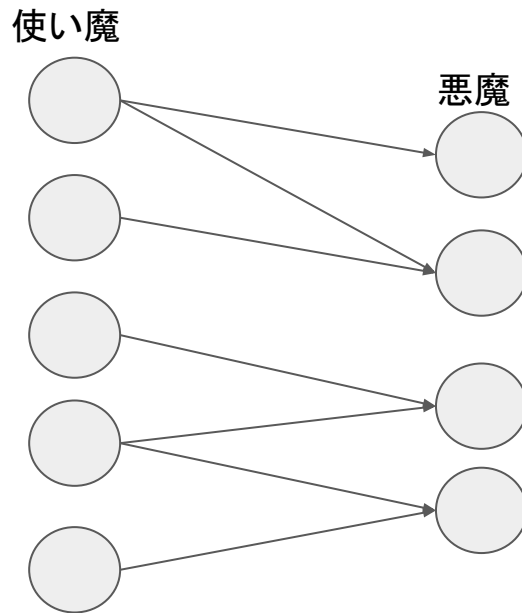
距離の最大値として許容する距離は二分探索によって求めることができる。

そのために

距離の最大値が $K$ のとき割り振ることができるかは次のような問題になる。

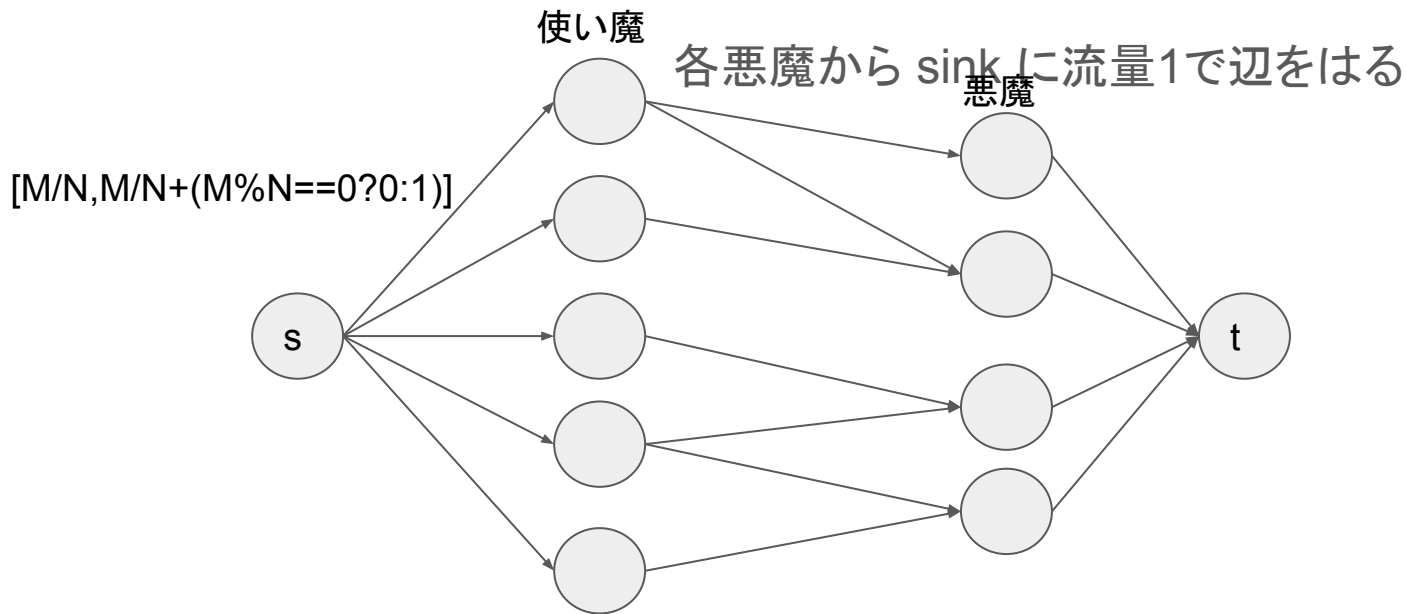
# 解法

距離K以内の使い魔→悪魔に流量1で辺を結ぶ



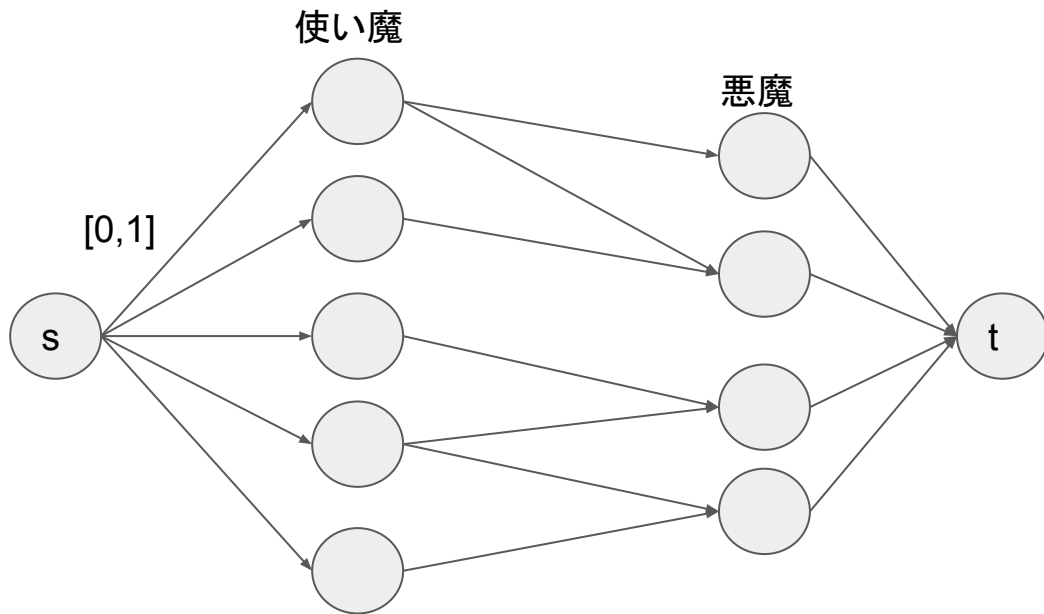
# 解法

source から 各使い魔に  $[M/N, M/N + (M \% N == 0 ? 0 : 1)]$  の流量の辺をはる



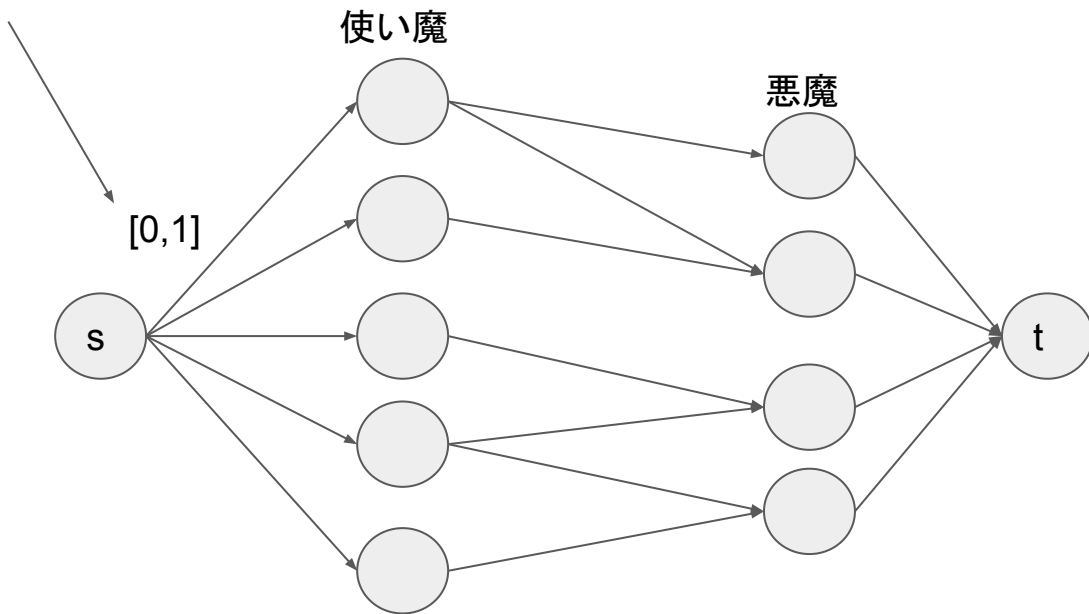
# 解法

フローを流して M だけ流れるかチェックする。



# 解法

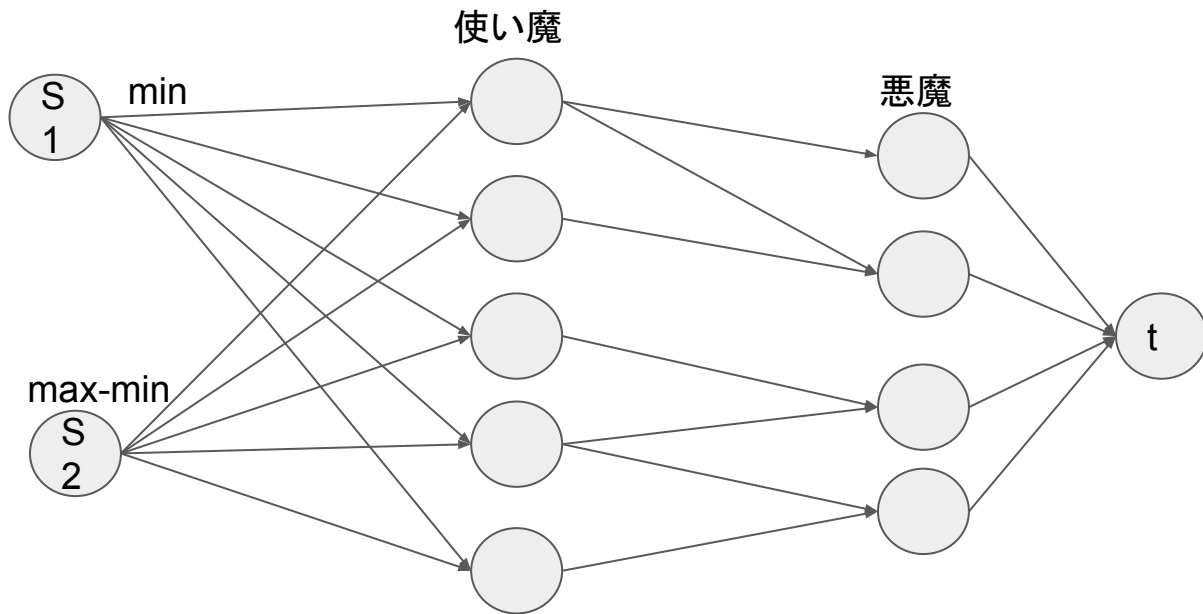
ここをなんとかする必要がある。(これはやり方がいくつかある)





# 解法(やりかた1)

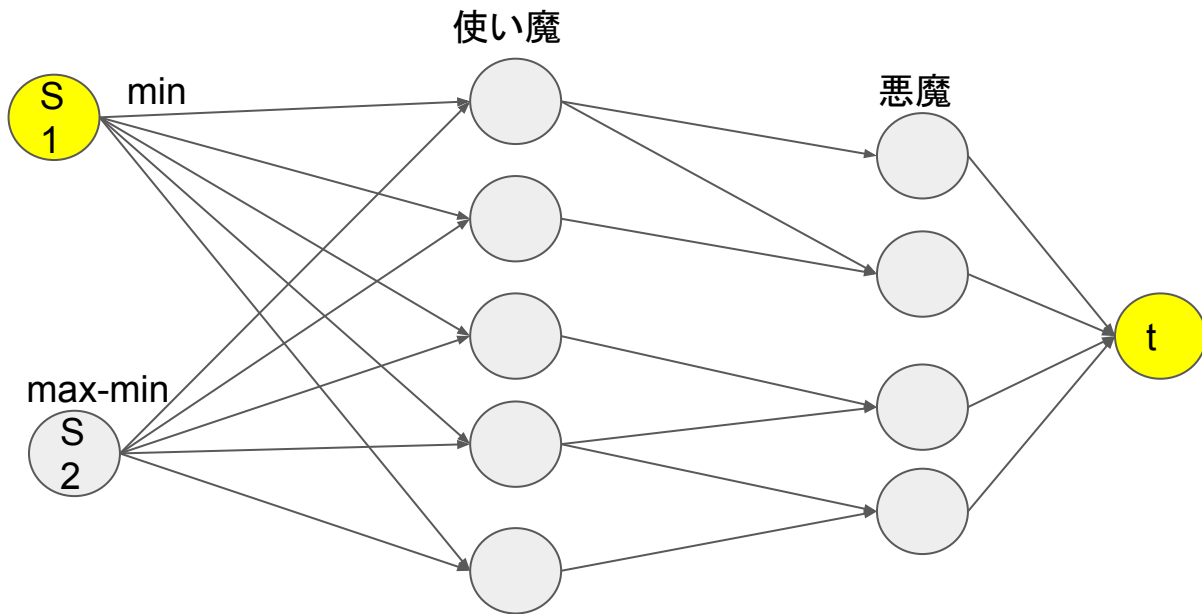
最低限流したい量(下限)をmin, 流して良い最大の流量(上限)をmaxとすると



S1から各使い魔に流量min, S2から各使い魔に流量maxを貼る。

# 解法(やりかた1)

S1 から t にめいいっぱい流す。これが  $\min * N$  以下であれば false。

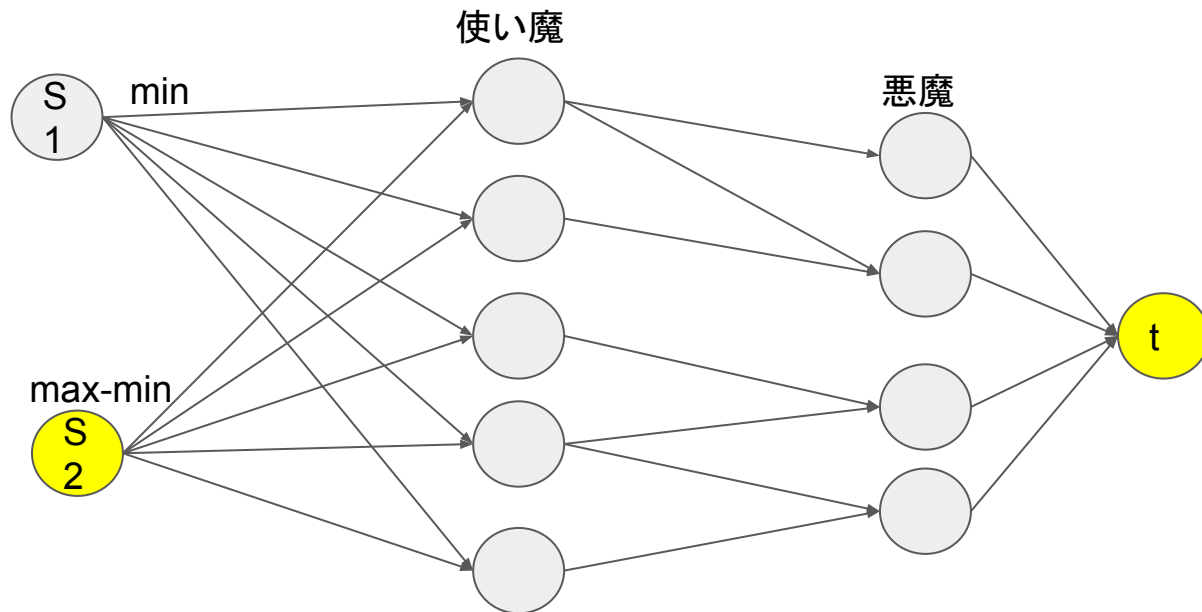


falseなのは全ての使い魔が最低限の仕事をしていないため。

# 解法(やりかた1)

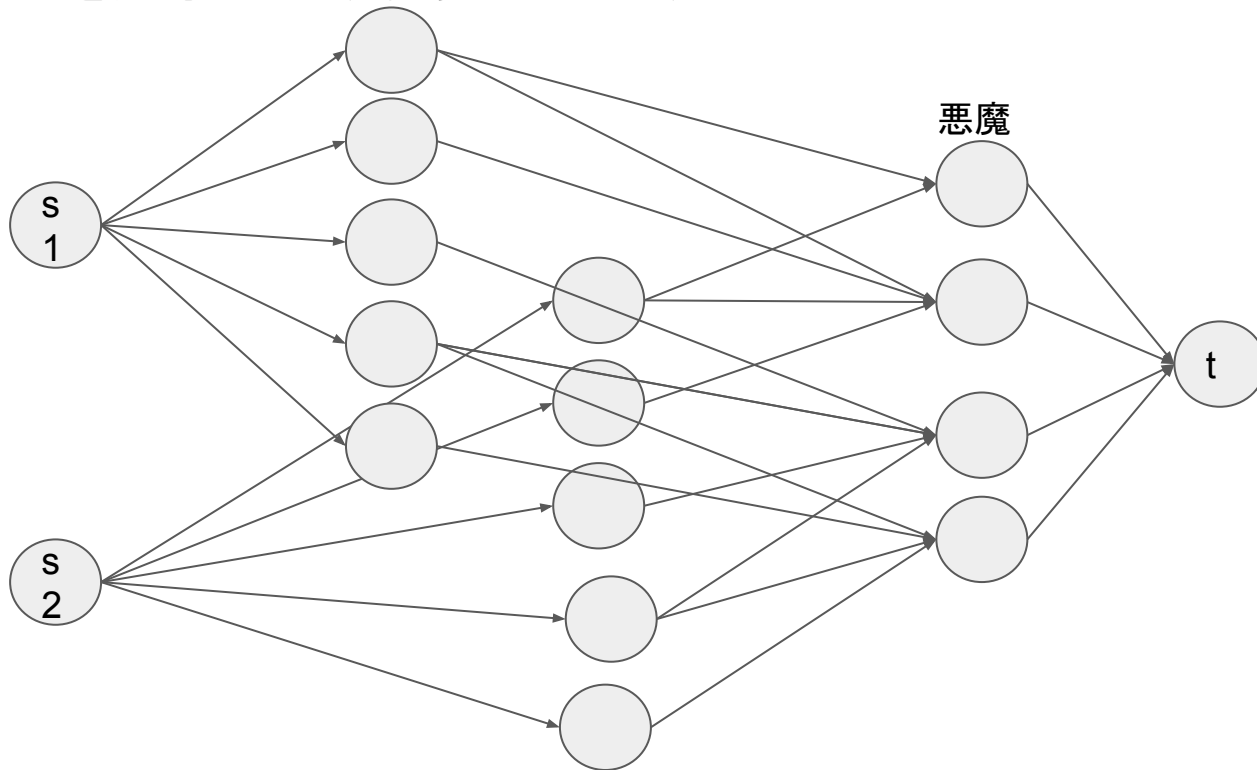
先程最低限を流しきった状態から

S2 から t にめいっぱい流す。これが  $M - \min * N$  であれば true。



# 解法(やり方2)

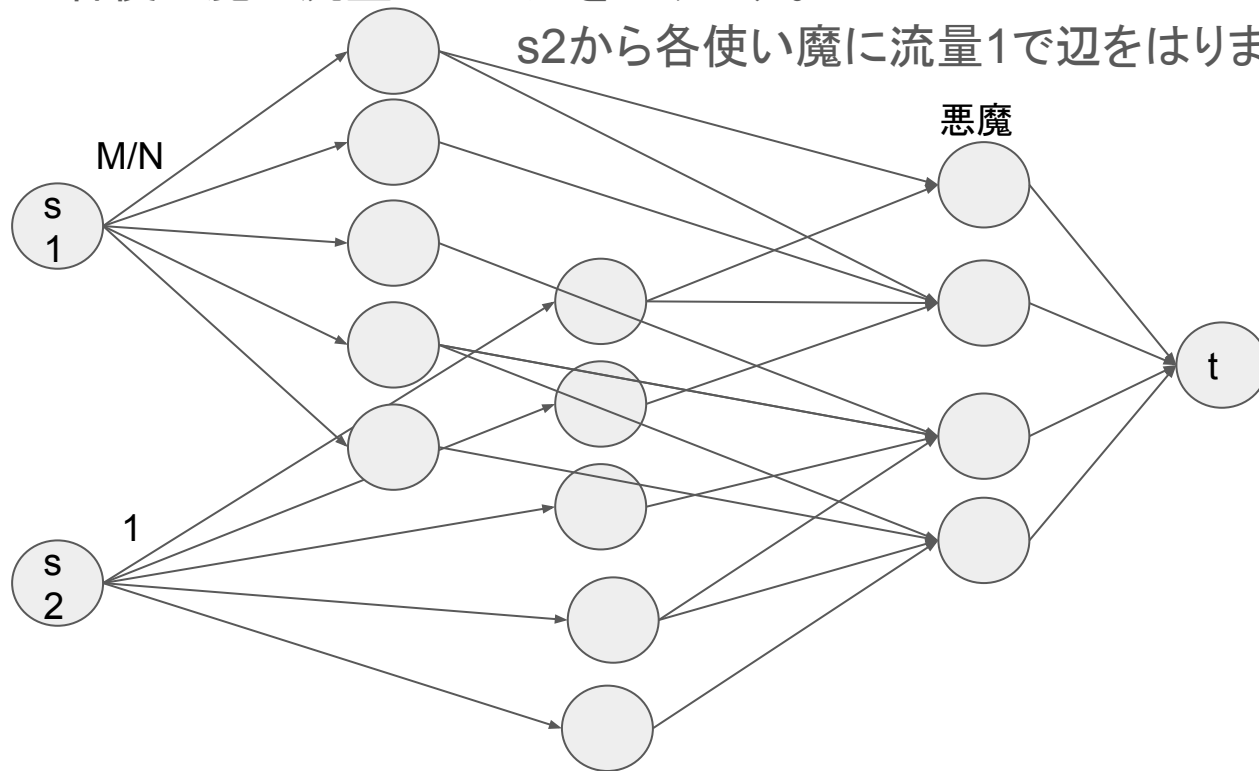
使い魔  
使い魔とsourceを分身させます。(見づらいですが許して)



# 解法(やり方2)

使い魔  
s1から分身した各使い魔に流量 $M/N$ で辺をはります。

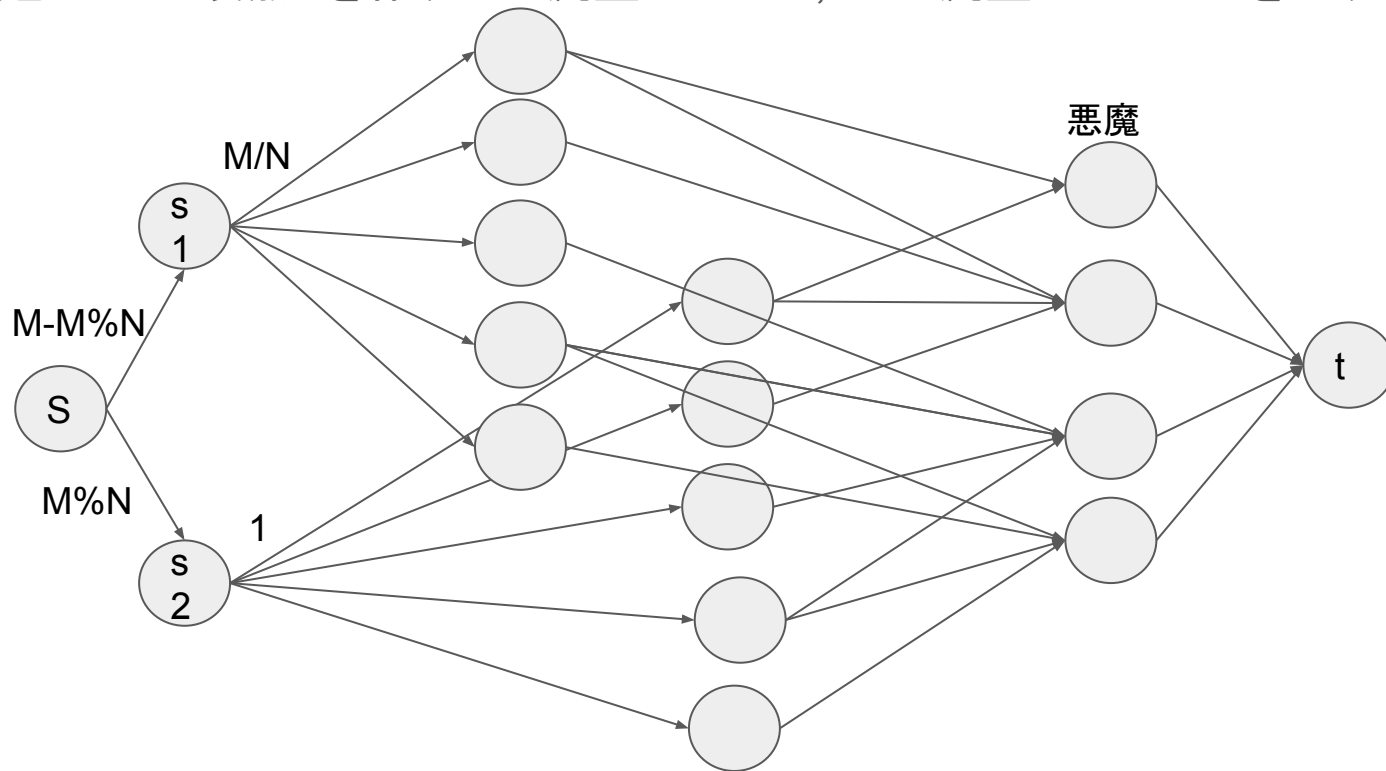
s2から各使い魔に流量1で辺をはります。



# 解法(やり方2)

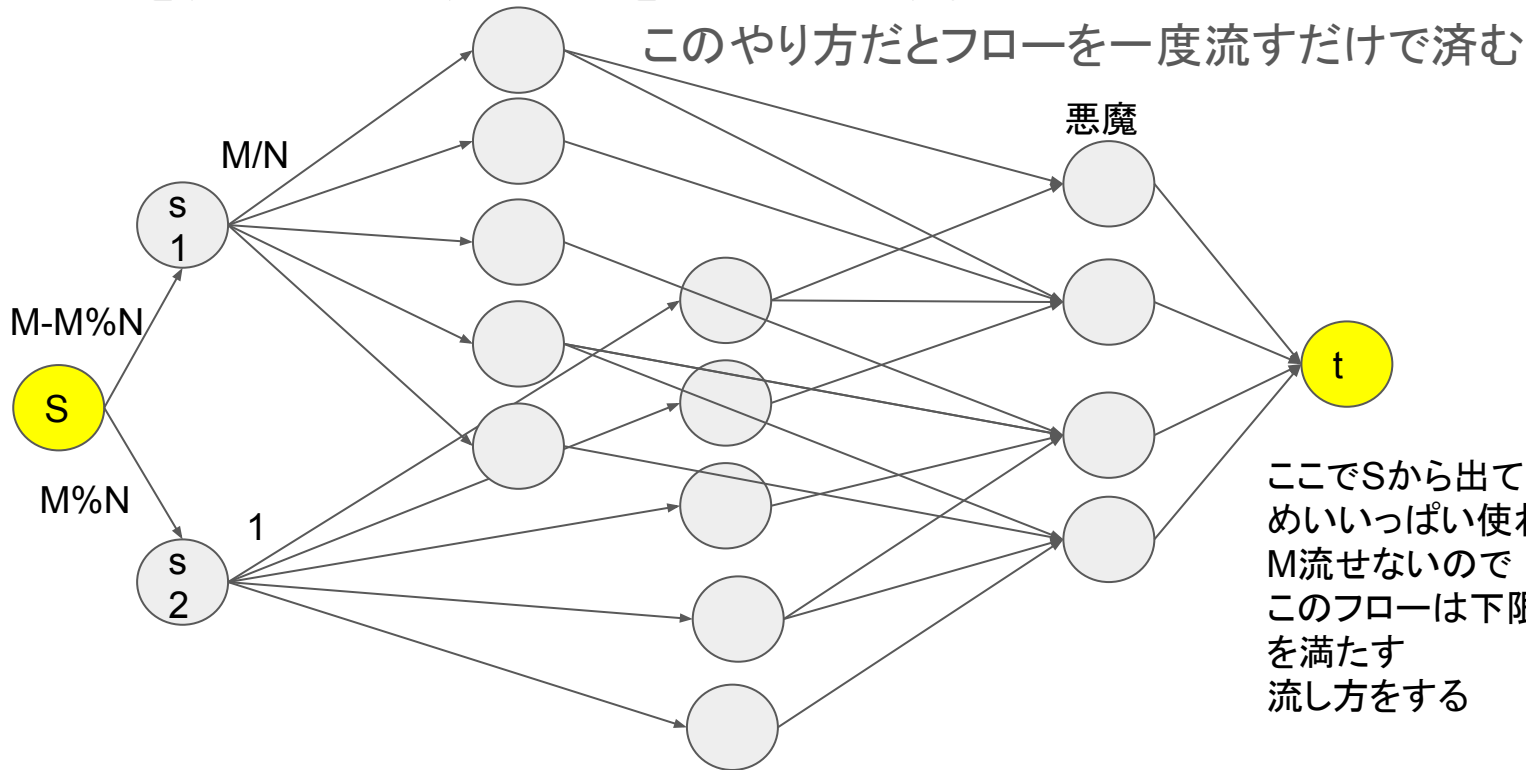
使い魔

さらに超Source頂点Sを作りs1に流量 $M - M \% N$ , s2に流量 $M \% N$ の辺をはります。



# 解法(やり方2)

使い魔  
 $S \rightarrow t$ にフローを流してMだけ流れるかをチェックします。



# 結果

- Onsite and Online
  - First Submission : biwako\_sen (2h16min)
  - First Accept : caffe (3h18min)
- SuccessRate (Accept/Submission)
  - 21.43 %



# ジャッジ解

sate	C++	112行
haji	C++	87行
uku	C++	119行
gacho	C++	76行