研究紹介

ネットワーク符号化

安永憲司

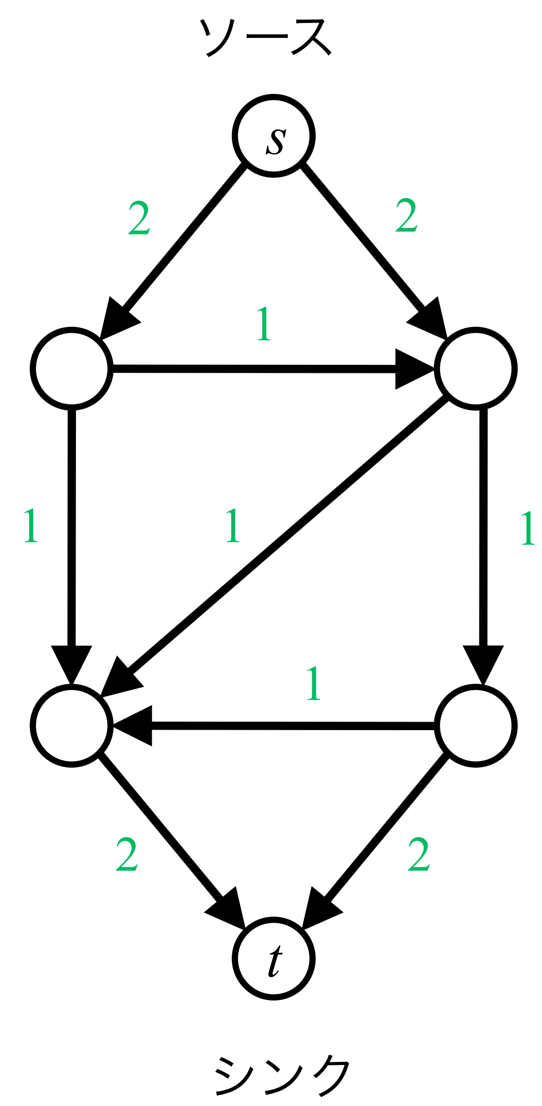
2008年5月某日

目次

- ネットワーク上の通信
- ネットワーク符号化
  - 線形ネットワーク符号化
  - ネットワーク符号化の利点・欠点
  - ランダム線形ネットワーク符号化
- まとめ
- 参考文献

ネットワーク上の通信

- ネットワーク上の各リンクは通信容量が決まっている
- ノードを経由して、ソースからシンクへ通信を行う
- ソースからシンクへの最大通信量は？

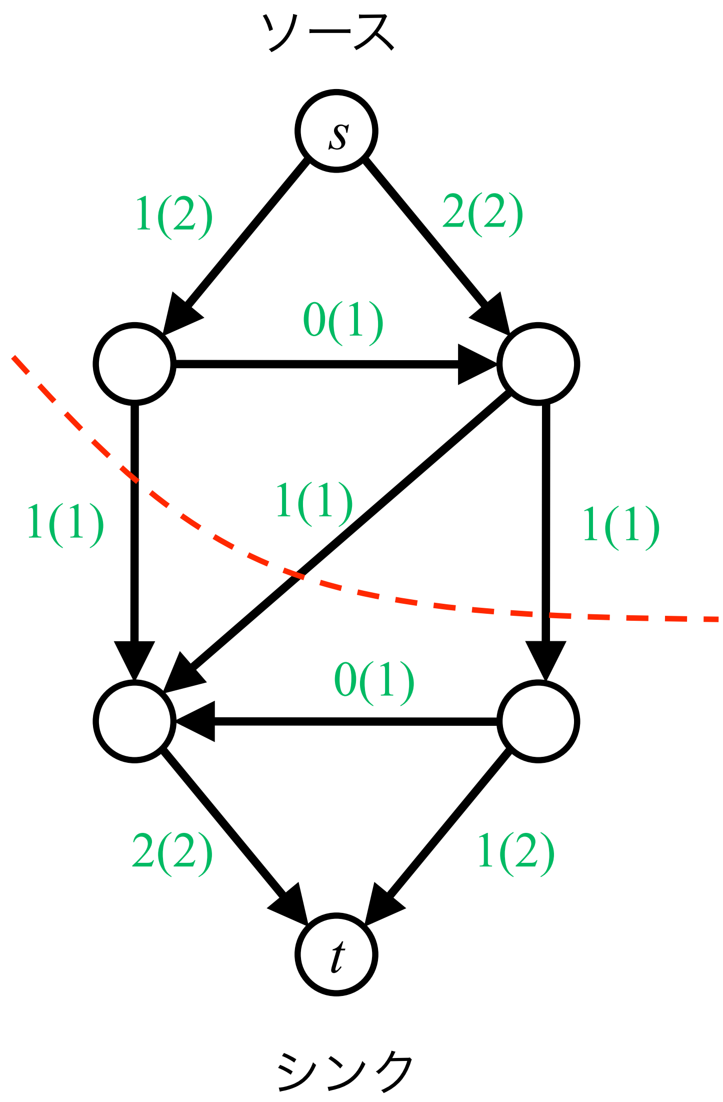


ネットワーク上の通信

- 各ノードにおいて、
  入る通信量 ＝ 出る通信量
  とする

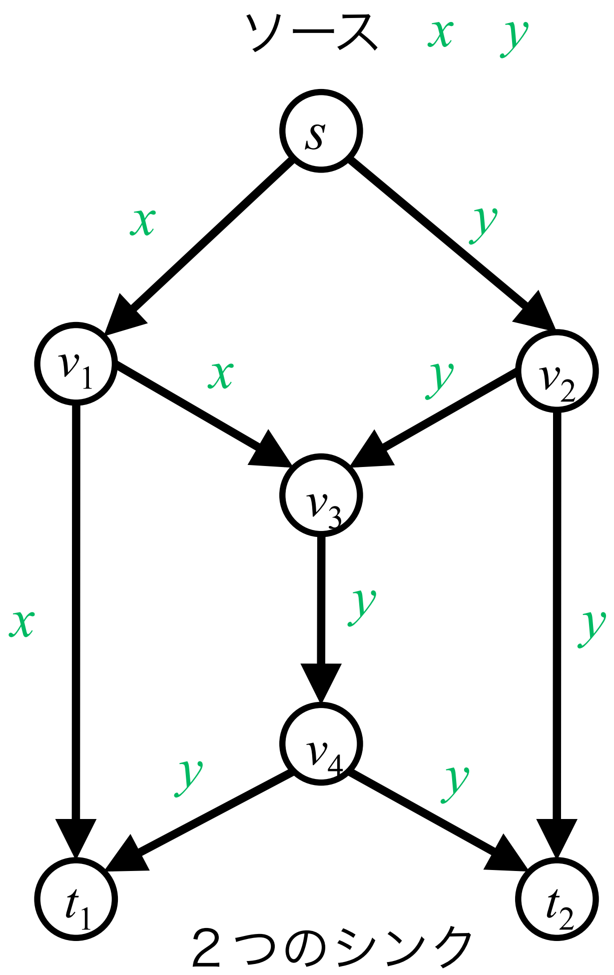
- 最大フロー最小カット定理：
  最大通信量は最小カットに等しい

- Ford-Fulkerson アルゴリズムで
  最大フローは求まる
  - 計算量：O($E \cdot mincut(s,t)$)
    E : リンク数
    $mincut(s,t)$: s - t 間の最小カット



１対２通信

- ソースが１つシンクが２つ
- マルチキャスト通信：ソースから複数のシンクへ同じ情報を伝える
- 各ソース・シンク間での最大フローはわかるが、すべてのシンクへ同時に最大フロー通信は可能か？
- 問題：v_3 から v_4 へは x, y どちらかしか送れない



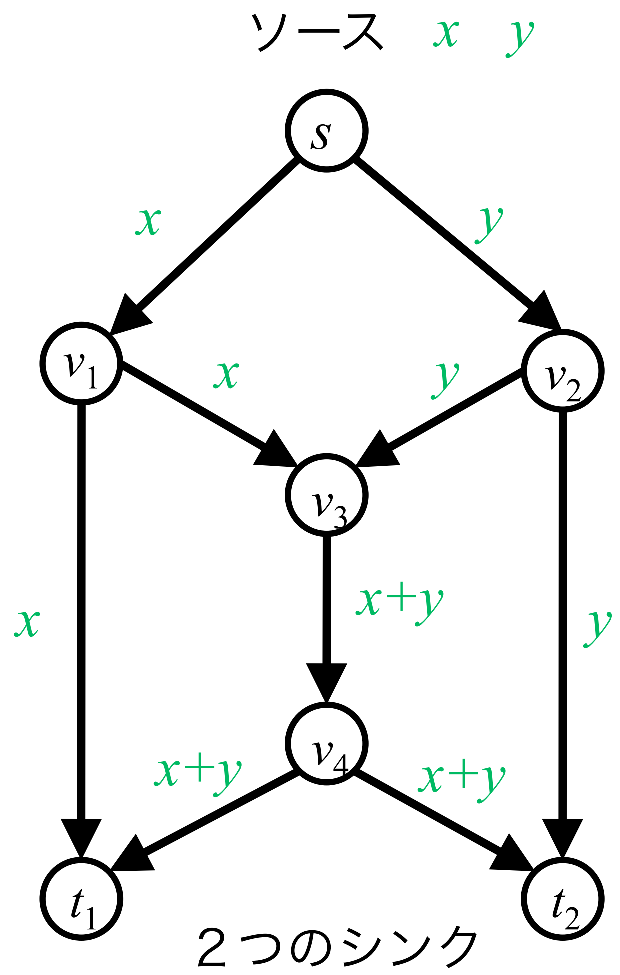
各リンクの容量は１とする

１対２通信

- 解決方法：v_3 から v_4 へ $x+y$ を送る

- ネットワーク符号化：ノードにおいて演算を許す

- ネットワーク符号化を利用したときの最大通信量は？



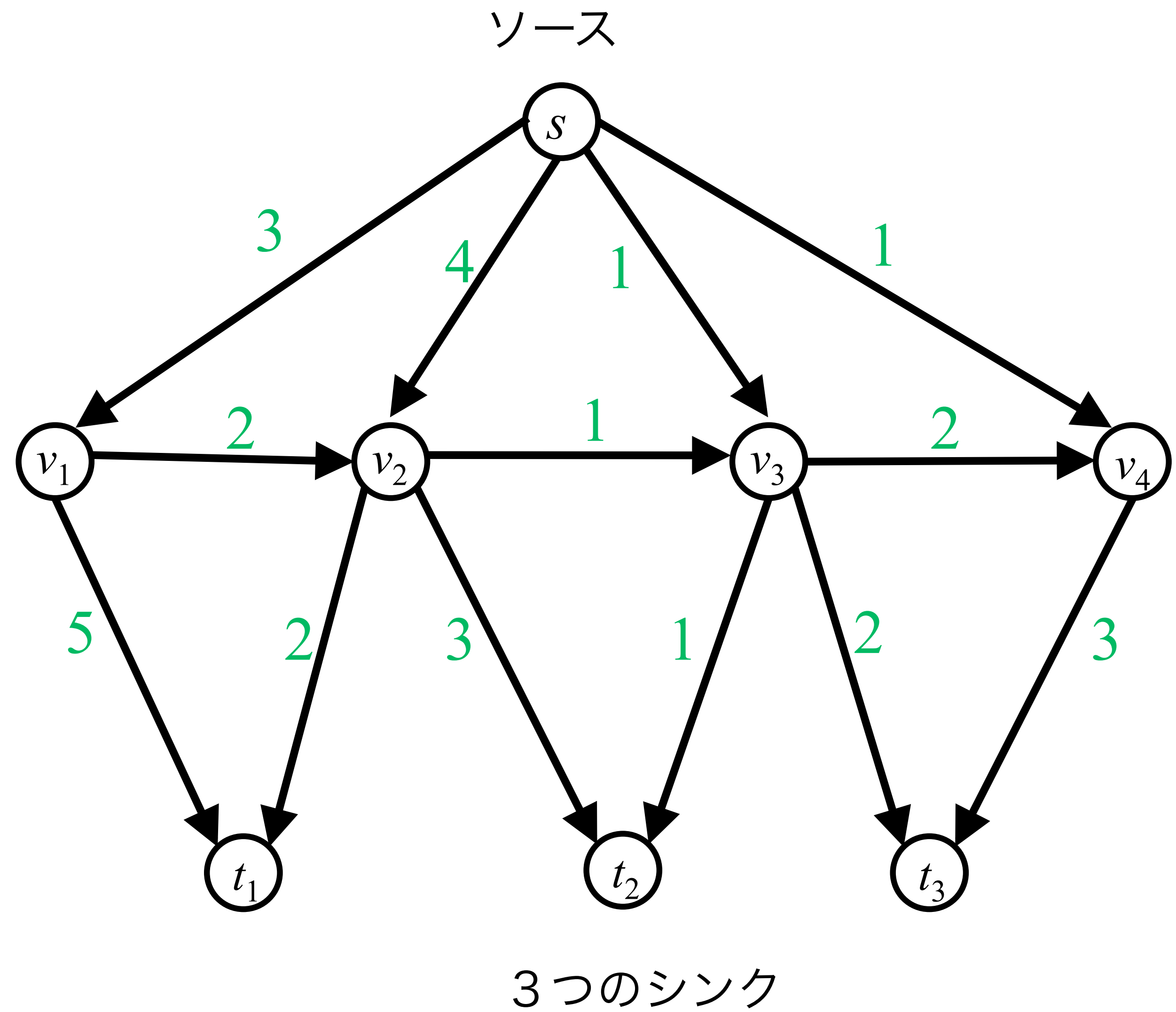
各リンクの容量は１とする

マルチキャスト通信における
最大フロー最小カット定理

- 定理 [ACLY00] ：１つのソースから複数のシンクへ同時に伝達可能な最大通信量は、各ソース・ノード間の最小カットの最小値

$$C = \min_{t \in T} mincut(s,t)$$

- 以下のネットワークにおけるマルチキャスト通信の最大通信量は？



ネットワーク符号化

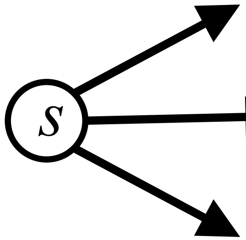
設定

- ネットワークはリンクの重複を許す有向グラフ $G = (V, E)$で表す
- 各リンクの通信容量は１（単位時間で $\mathbf{F}_q$ 上のシンボル１つを伝達可能）
- ソースノード $s \in V$, シンクノード集合 $T \subseteq V$
- １つのメッセージは l 個のシンボル集合 $X_1, X_2, \ldots, X_l$ であり s からすべての $t \in T$ へ送信
- 各ノードにおける出力は、入力シンボルの関数である

ネットワーク符号化

- ネットワーク符号は各ノードにおける関数で決まる

ソースノード

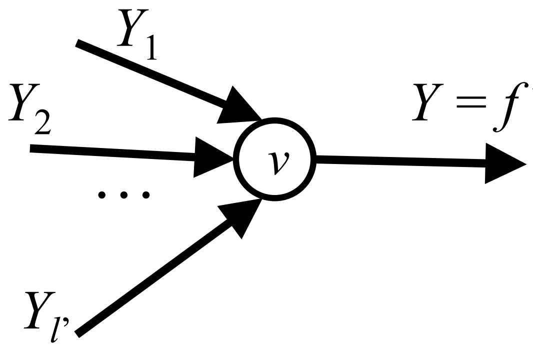


$$Y_1 = f_1^s(X_1, X_2, \ldots, X_l)$$

$$Y_2 = f_2^s(X_1, X_2, \ldots, X_l)$$

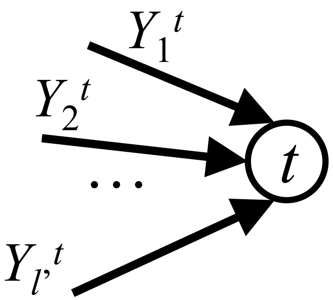
$$Y_3 = f_3^s(X_1, X_2, \ldots, X_l)$$

中間ノード



$$Y = f^v(Y_1, Y_2, \ldots, Y_{l'})$$

シンクノード



$$X_1 = f_1^t(Y_1^t, Y_2^t, \ldots, Y_{l'}^t)$$

$$\vdots$$

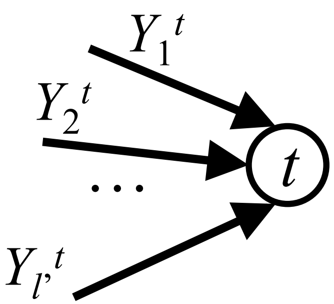
$$X_n = f_n^t(Y_1^t, Y_2^t, \ldots, Y_{l'}^t)$$

線形ネットワーク符号化

- 各ノードの符号化関数が $\mathbf{F}_q$ 上の線形関数であるもの

$$f^v(Y_1, Y_2, \ldots, Y_{l'}) = a_1 Y_1 + a_2 Y_2 + \cdots + a_{l'} Y_{l'}$$

- このとき、ソースからシンク t への通信は



$$\underbrace{\begin{bmatrix} Y_1^t \\ \vdots \\ Y_{l'}^t \end{bmatrix}}_{Y^t} = \underbrace{\begin{bmatrix} g_{1,1}^t & \cdots & g_{1,l}^t \\ \vdots & & \vdots \\ g_{l',1}^t & \cdots & g_{l',l}^t \end{bmatrix}}_{G^t} \underbrace{\begin{bmatrix} X_1 \\ \vdots \\ X_l \end{bmatrix}}_{X}$$

線形ネットワーク符号化

- G_t のランクが l ならば X を求めることができる

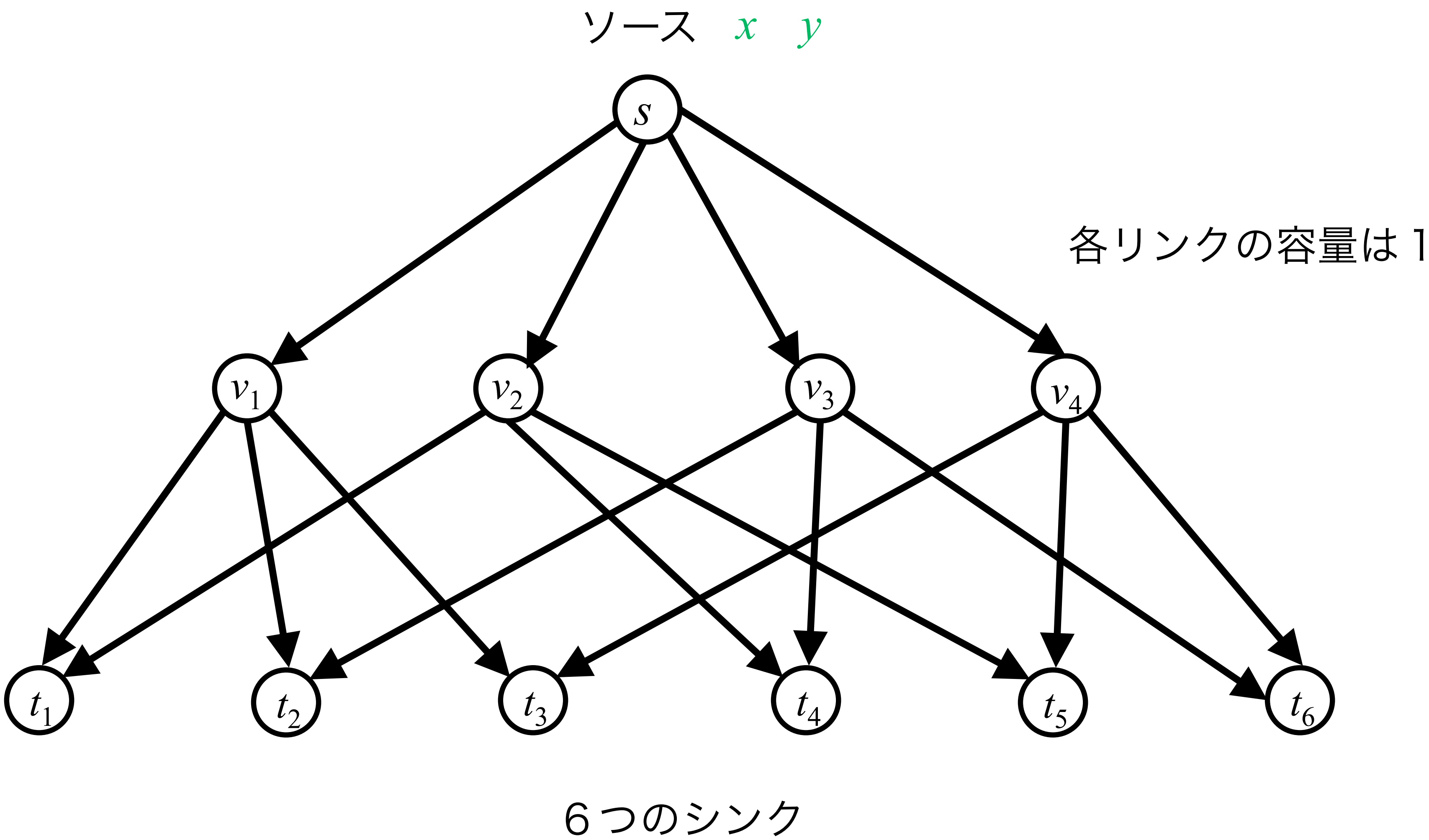
$$Y^t = G^t X$$

$$X = {G^t}^{-1} Y^t$$

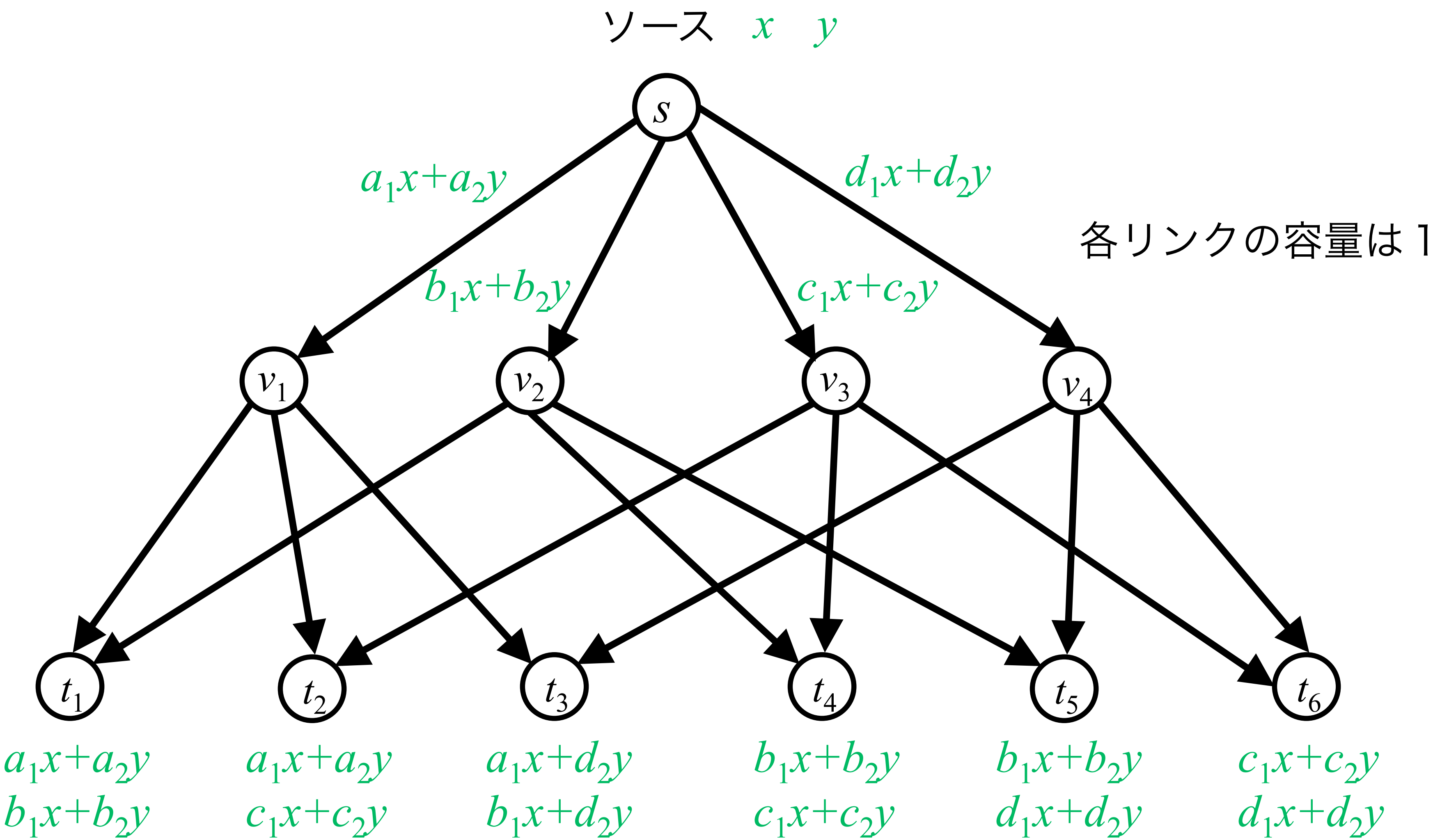
- 上記を満たすには、最大通信量が l 以上であり q が十分に大きい必要がある

- 定理 [LYC03][KM03]：マルチキャスト通信での最大通信量は、アルファベットサイズを十分に大きくした線形ネットワーク符号で達成可能
  - 決定的アルゴリズム [JSC+05]

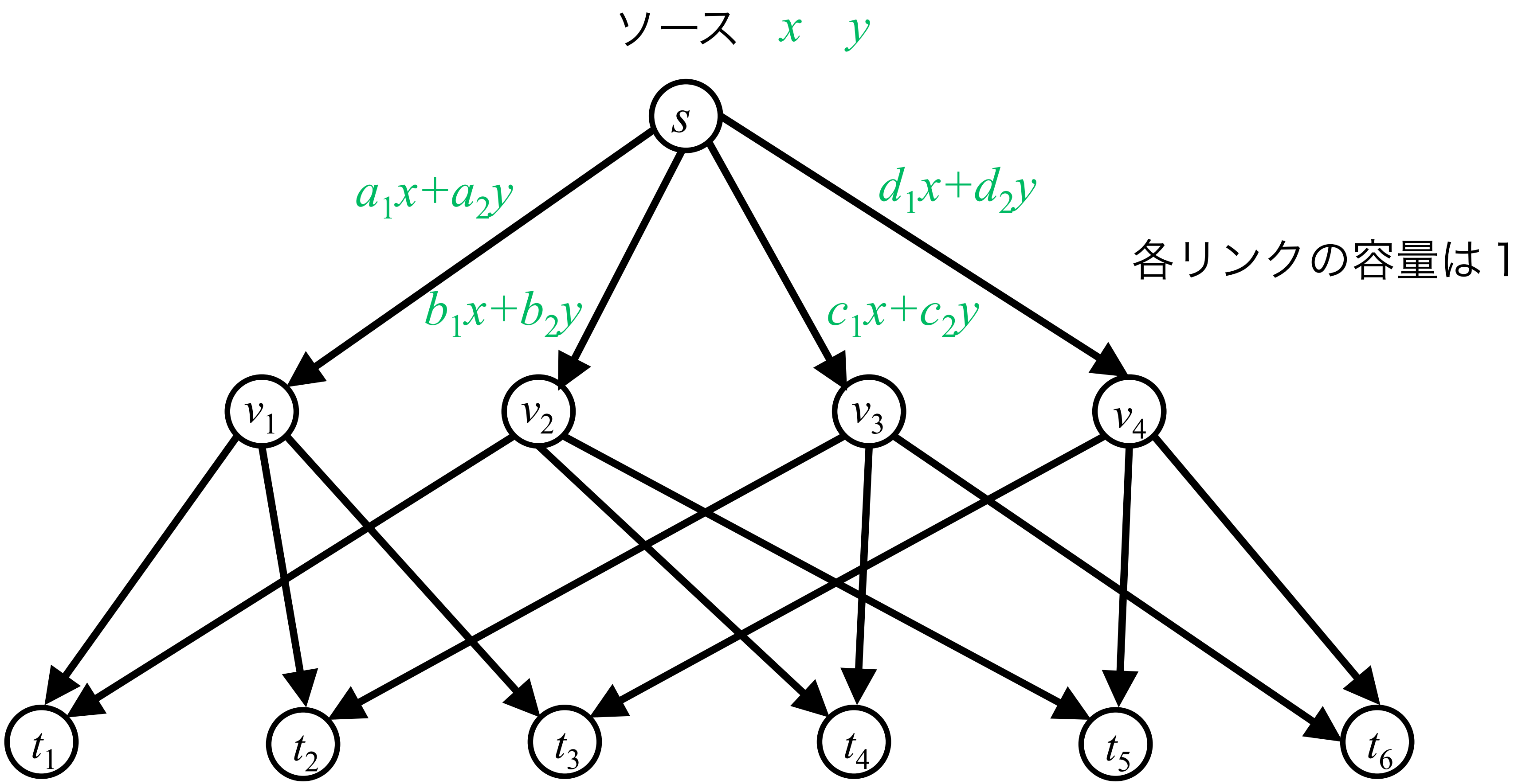
$\mathbf{F}_2$ 上では解がないマルチキャスト通信 [RL05]



$\mathbf{F}_2$ 上では解がないマルチキャスト通信 [RL05]



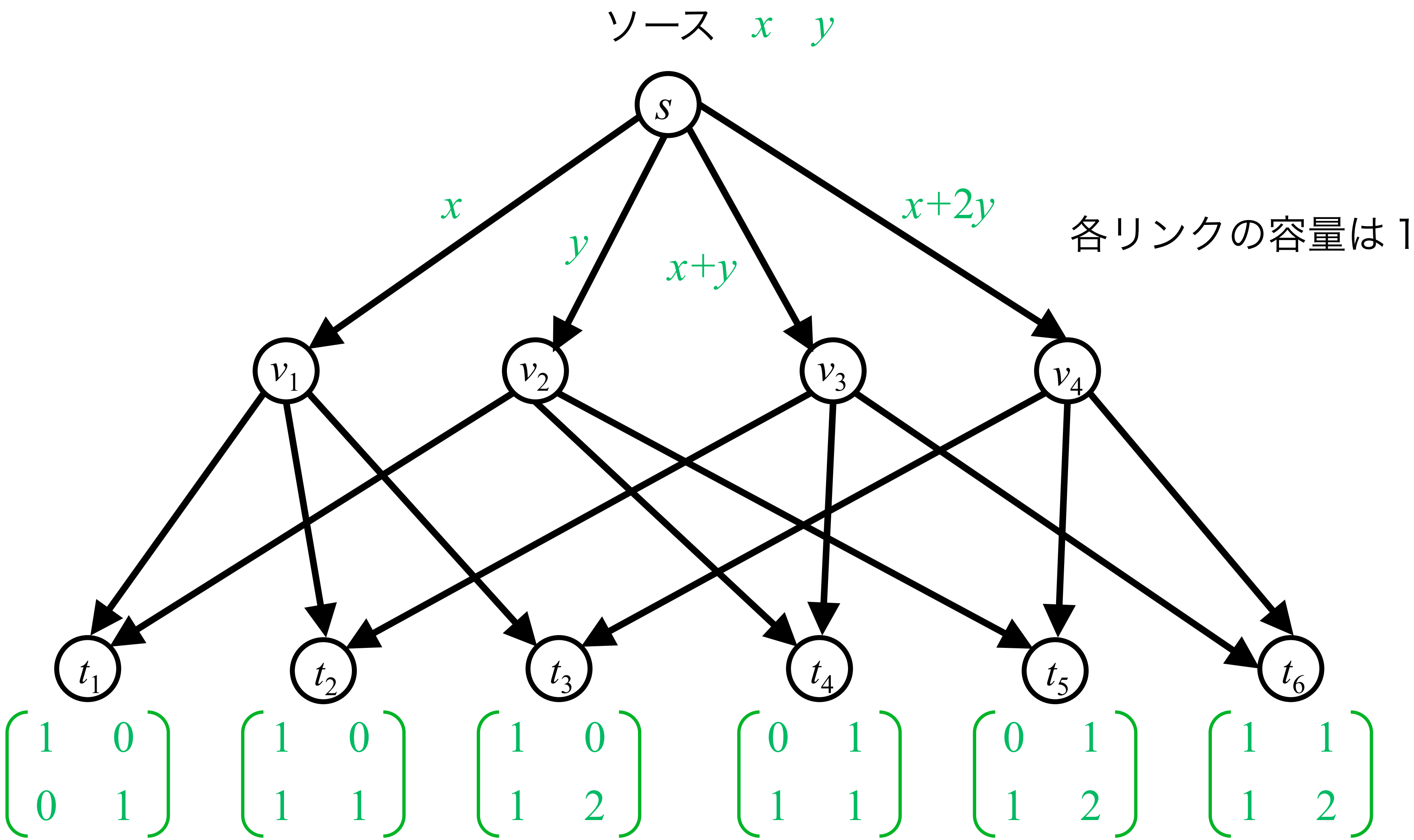
$\mathbf{F}_2$ 上では解がないマルチキャスト通信 [RL05]



$$\begin{pmatrix} a_1 & a_2 \\ b_1 & b_2 \end{pmatrix} \quad \begin{pmatrix} a_1 & a_2 \\ c_1 & c_2 \end{pmatrix} \quad \begin{pmatrix} a_1 & a_2 \\ d_1 & d_2 \end{pmatrix} \quad \begin{pmatrix} b_1 & b_2 \\ c_1 & c_2 \end{pmatrix} \quad \begin{pmatrix} b_1 & b_2 \\ d_1 & d_2 \end{pmatrix} \quad \begin{pmatrix} c_1 & c_2 \\ d_1 & d_2 \end{pmatrix}$$

６つ行列がすべてランク２をもつ必要がある

$\mathbf{F}_2$ 上では解がないマルチキャスト通信 [RL05]



$\mathbf{F}_3$ 上なら解が存在

ベクトル（パケット）による通信

- アルファベットが $\mathbf{F}_{q^n}$ のとき、$\mathbf{F}_q$ 上の n 次元ベクトルを送信していると考えることができる

- すると、１つのメッセージは長さ n のベクトル l 個

$$X_i = [X_{i,1}, \ldots, X_{i,n}]$$

- 各シンクは、独立した l 個のベクトルが受信できればよい

$$\underbrace{\begin{bmatrix} Y_{1,1} & \cdots & Y_{1,n} \\ \vdots & & \vdots \\ Y_{l',1} & \cdots & Y_{l',n} \end{bmatrix}}_{l' \times n \text{ 行列}} = \underbrace{\begin{bmatrix} g_{1,1} & \cdots & g_{1,l} \\ \vdots & & \vdots \\ g_{l',1} & \cdots & g_{l',l} \end{bmatrix}}_{l' \times l \text{ 行列}} \underbrace{\begin{bmatrix} X_{1,1} & \cdots & X_{1,n} \\ \vdots & & \vdots \\ X_{l,1} & \cdots & X_{l,n} \end{bmatrix}}_{l \times n \text{ 行列}}$$

ネットワーク符号化の利点・欠点

利点

- 通信速度向上
  - マルチキャスト通信は線形ネットワーク符号で最大通信量を達成
- 耐性が高い
  - 通信中にあるパケットが消失しても致命的にならない

欠点

- 必要なパケットが集まるまで復号できない（遅延が生じる）
- ネットワークトポロジを知る必要・符号化関数を各ノードへ教える必要がある
  - ネットワークの構成が変わってしまうと符号化関数の再構成が必要

ネットワーク符号化の利点・欠点

利点

- 通信速度向上
  - マルチキャスト通信は線形ネットワーク符号で最大通信量を達成
- 耐性が高い
  - 通信中にあるパケットが消失しても致命的にならない

欠点

- 必要なパケットが集まるまで復号できない（遅延が生じる）
- ネットワークトポロジを知る必要・符号化関数を各ノードへ教える必要がある
  - ネットワークの構成が変わってしまうと符号化関数の再構成が必要

⇒ ランダム線形ネットワーク符号化による解決

ランダム線形ネットワーク符号化 [HMK+06]

- 各ノードにおける符号化関数の係数をランダムに選ぶ
- 各パケットの先頭に、各ノードでの線形結合を記憶させるためのヘッダを用意

$$X = \underbrace{\begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix}}_{l\ \text{シンボルのヘッダ}} \underbrace{\begin{matrix} X_{1,l+1} & X_{1,l+2} & \cdots & X_{1,n} \\ X_{2,l+1} & X_{2,l+2} & \cdots & X_{2,n} \\ \vdots & \vdots & & \vdots \\ X_{l,l+1} & X_{l,l+2} & \cdots & X_{l,n} \end{matrix}}_{n-l\ \text{シンボルのデータ}}$$

ランダム線形ネットワーク符号化 [HMK+06]

- Y のランクが n になるまでパケットを集め、ガウスの消去法により X を復元
  - $X = [\,I \;\; X'\,],\ Y = G\,X = G\,[\,I \;\; X'\,] = [G \;\; GX']$
  - G の係数がランダムの場合、アルファベットサイズ q^n が大きいほど Y はフルランクになりやすい

- n を大きくとればヘッダは無視できるサイズとなる

- ネットワークトポロジを知る必要がない
  - ネットワークが変化しても方法を変えなくてよい

まとめ

- ネットワーク符号化
  - ネットワークの各ノードで演算を許したもの（通常のルーティングによる通信の一般化）
  - 通信速度向上・耐性が高い
    - マルチキャスト通信の最大通信量は線形ネットワーク符号化で達成
- 線形ランダムネットワーク符号化
  - ネットワークトポロジに依存しない通信方法
- さまざまな研究の方向が存在
  - 無向グラフ、多対多通信、一対一通信での優位性
  - セキュリティ
    - 盗聴や改ざんに耐性のあるネットワーク符号化
  - 誤り訂正
    - 誤りや消失に耐性のあるネットワーク符号化

参考文献

[ACLY00] R. Ahlswede, N. Cai, S.-Y. R. Li, R. W. Yeung, “Network information flow,” *IEEE Trans. Inform. Theory*, vol. 46, no. 4, pp. 1204-1216, July 2000.

[HMK+06] T. Ho, M. Medard, R. Koetter, D.R. Karger, M. Effros, J. Shi, B. Leong, “A random linear network coding approach to multicast,” *IEEE Trans. Inform. Theory*, vol. 52, no. 10, Oct. 2006.

[JSC+05] S. Jaggi, P. Sanders, P.A. Chou, M. Effros, S. Egner, K. Jain, L.M.G.M. Tolhuizen, “Polynomial time algorithms for multicast network code construction,” *IEEE Tran. Inform. Theory*, vol. 51, no. 6, June 2005.

[KM03] R. Koetter, M. Medard, “An algebraic approach to network coding,” *IEEE/ACM Tran. Netw.*, vol. 11, no. 5. pp. 782-795, Oct. 2003.

[LYC03] S.-Y. R. Li, R.W. Yeung, N. Cai, “Linear network coding,” *IEEE Trans. Inform. Theory*, vol. 49, no. 2, Feb. 2003.

[RL05] A. Rasala-Lehman, “Network Coding”, Ph. D. thesis, MIT, Jan. 2005.

今回の資料は、ECE1528: Multiuser Information Theory, 2007 の中の Danilo Silva, “Information-Theoretic Aspects of Network Coding” をもとに作成 (www.comm.utoronto.ca/~weiyu/ece1528_2007/slides/Danilo.ppt)