# 分散ストレージにおける再生成符号と秘密分散について

On regenerating codes and secret sharing for distributed storage

**栗原 正純**
**（電気通信大学）**

**桑門 秀典**
**（神戸大学）**

**電子情報通信学会** (IEICE) **情報理論** (IT) **研究会** 2011 **年** 1 **月奈良**

Masazumi KURIHARA (kuri@ice.uec.ac.jp) (2011/1/11/15:08)

# 分散ストレージの修復問題と再生成符号 I



1. **分散ストレージシステムの修復問題**
   [1] A.G.Dimakis, P.B.Godfrey, Y.Wu, M.J.Wainwright and K.Ramchandran,
   ”Network Coding for Distributed Storage Systems,”2010.
2. **再生成符号** (Regenerating codes)
   1. **オリジナルデータの復元** (Reconstruction)
   2. **故障ノードの修復（システムの信頼性の維持）**
      **故障ノードに保存されていたデータの複製を再生成する。**
3. **ストレージと修復バンドワイドのトレードオフ関係**
   1. **修復バンドワイドを最小にする**
      **最小バンドワイド再生成符号**
      (Minimum Bandwidth Regenerating(MBR) codes)
   2. **ストレージを最小にする最小ストレージ再生成符号**
      (Minimum Storage Regenerating(MSR) codes)

# 分散ストレージの修復問題と再生成符号 II



1. **一般的な再生成符号の構成**

   [3] K.V.Rashmi, N.B.Shah, and P.V.Kumar,
   "Optimal Exact-Regenerating Codes for Distributed Storage at the MSR and MBR Points via a Product-Matrix Construction," 2010.

   1. Product matrix **& 行列の対称性**

   [4] C.Suh and K.Ramchandran,
   "Exact Regeneration Codes for Distributed Storage Repair Using Interference Alignment," 2010.

   1. Interference alignment **& 符号内部の双対性**

# 本稿の提案：再生成符号と秘密分散 I



K.V.Rashmi, N.B.Shah, and P.V.Kumar **が提案した**
**最小ストレージ再生成** (MSR) **符号を用いて**

1. **秘密分散構造の構築方法**

**を提案し、以下の安全性を示す。**

1. **分散データの安全性**
   **（ストレージノードが保存するデータ）**
2. **再生成用データの安全性**
   **（故障ノードを修復する際に利用するデータ）**

# 復元 (Reconstruction)

$(k=3)$



$$H(u_1u_2u_3u_4u_5u_6 \mid c_{1,1}c_{1,2}c_{2,1}c_{2,2}c_{4,1}c_{4,2}) = 0$$

分散データ

Storage $\alpha = 2$

メッセージ

$u_1, u_2,$
$u_3, u_4,$
$u_5, u_6$

$B = 6$

1 $c_{1,1}, c_{1,2}$

2 $c_{2,1}, c_{2,2}$

3 $c_{3,1}, c_{3,2}$

4 $c_{4,1}, c_{4,2}$

5 $c_{5,1}, c_{5,2}$

DC

復元

$c_{1,1}, c_{1,2}$
$c_{2,1}, c_{2,2}$
$c_{4,1}, c_{4,2}$

$\Rightarrow$ $u_1, u_2,$ $u_3, u_4,$ $u_5, u_6$

ノード $i$

$u_1, u_2,$
$u_3, u_4,$
$u_5, u_6$

分散符号化

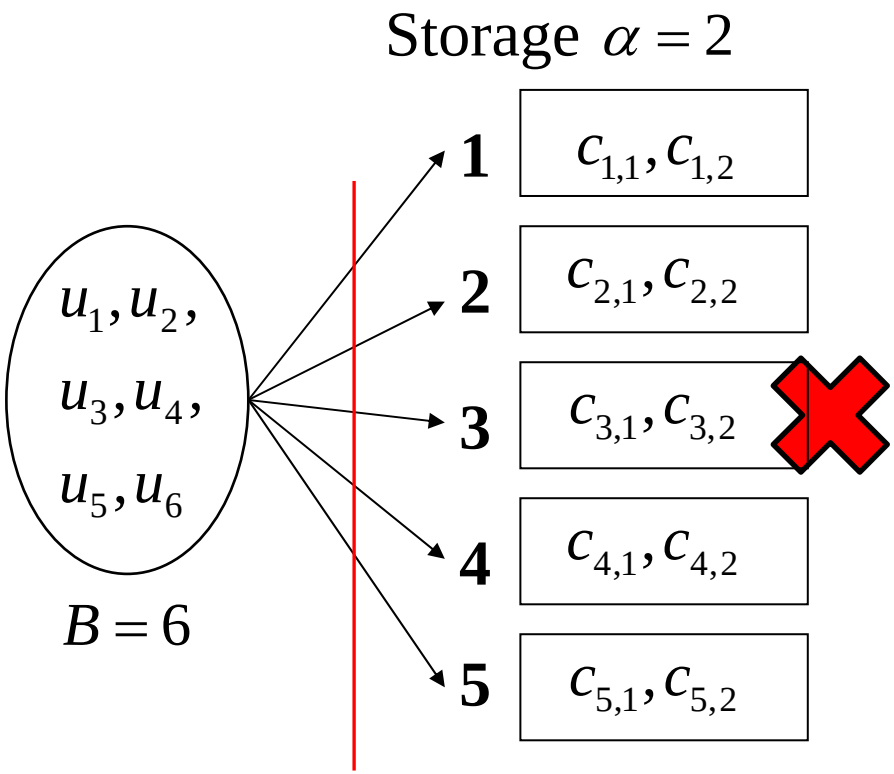
$\Rightarrow$ $c_{i,1}, c_{i,2}$

$$u_k, c_{j,k}, d_l \in GF(q)$$



Masazumi Kurihara (Univ. of Electro Communications, Tokyo)

# 故障ノードの修復問題 (Repair Problem) $(k=3)$




アクセスできない

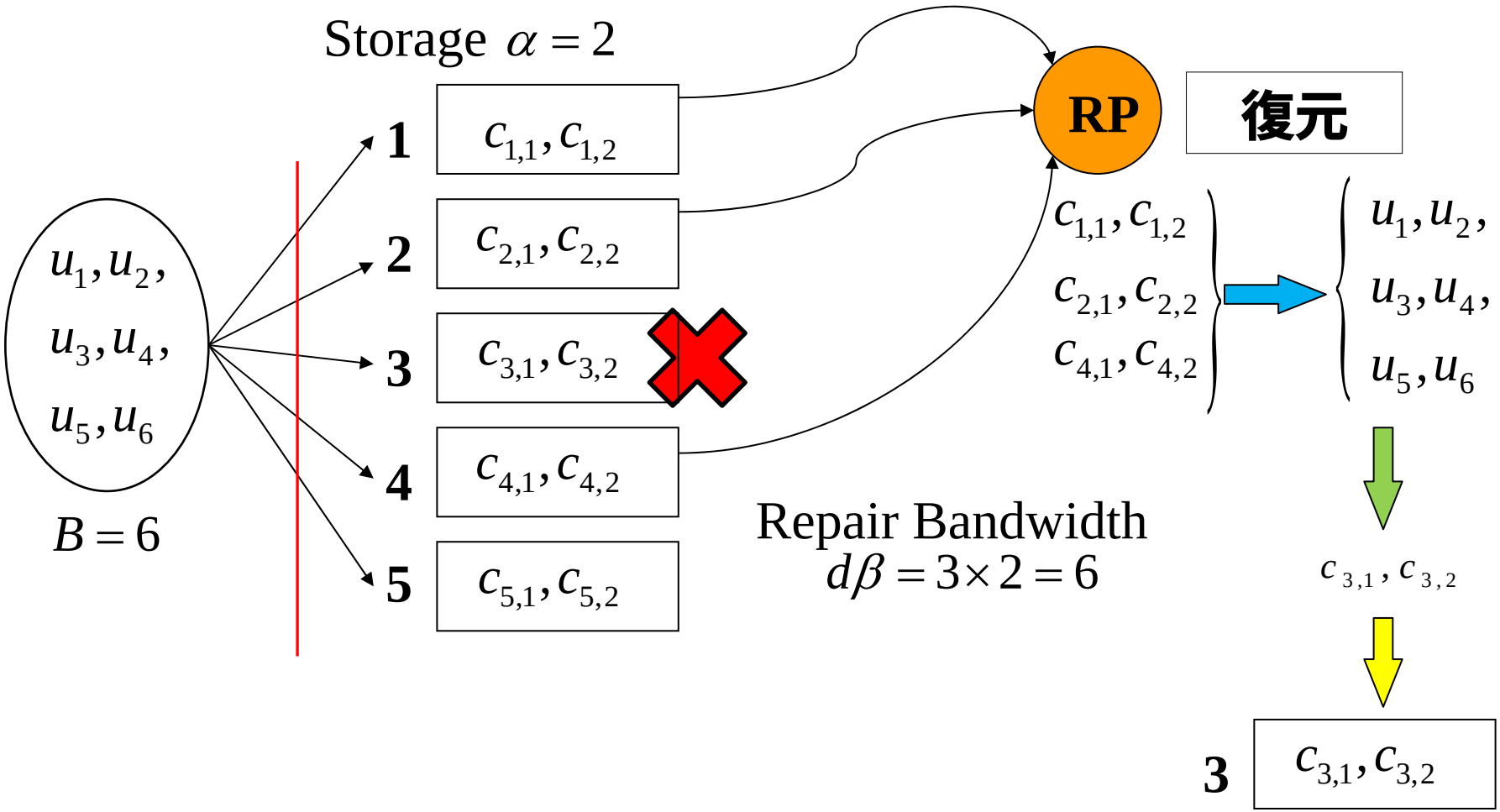
**故障ノードの修復**

故障したノードを新しいノードに置き換えて、

故障ノードが保存していた分散データと同じデータを保存したい。

ただし、再び、ソースから分散データを受信しることはできないと仮定する。

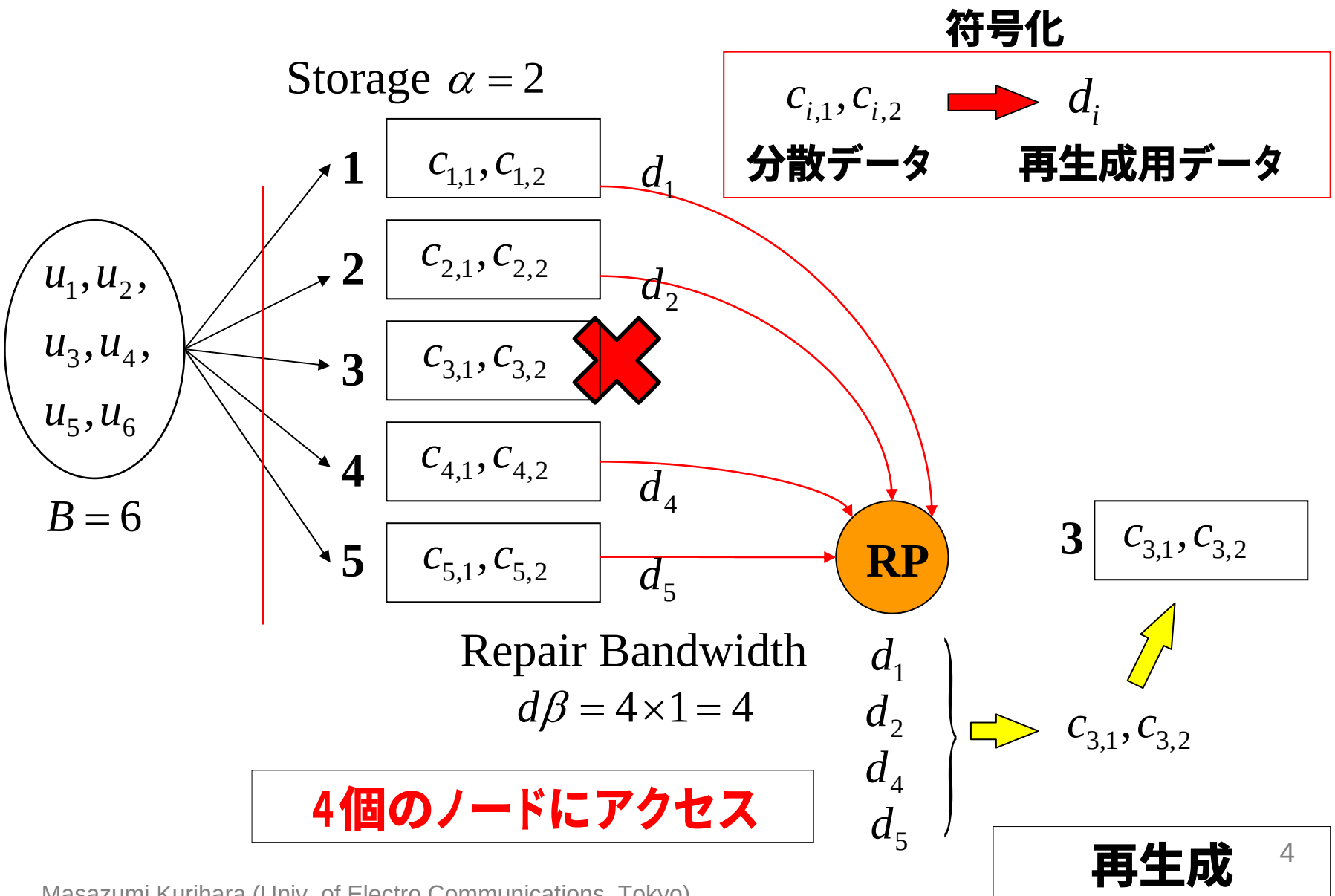
# 自明な方法：故障ノードの修復 $(k,d)=(3,3)$




**3個のノードにアクセス**

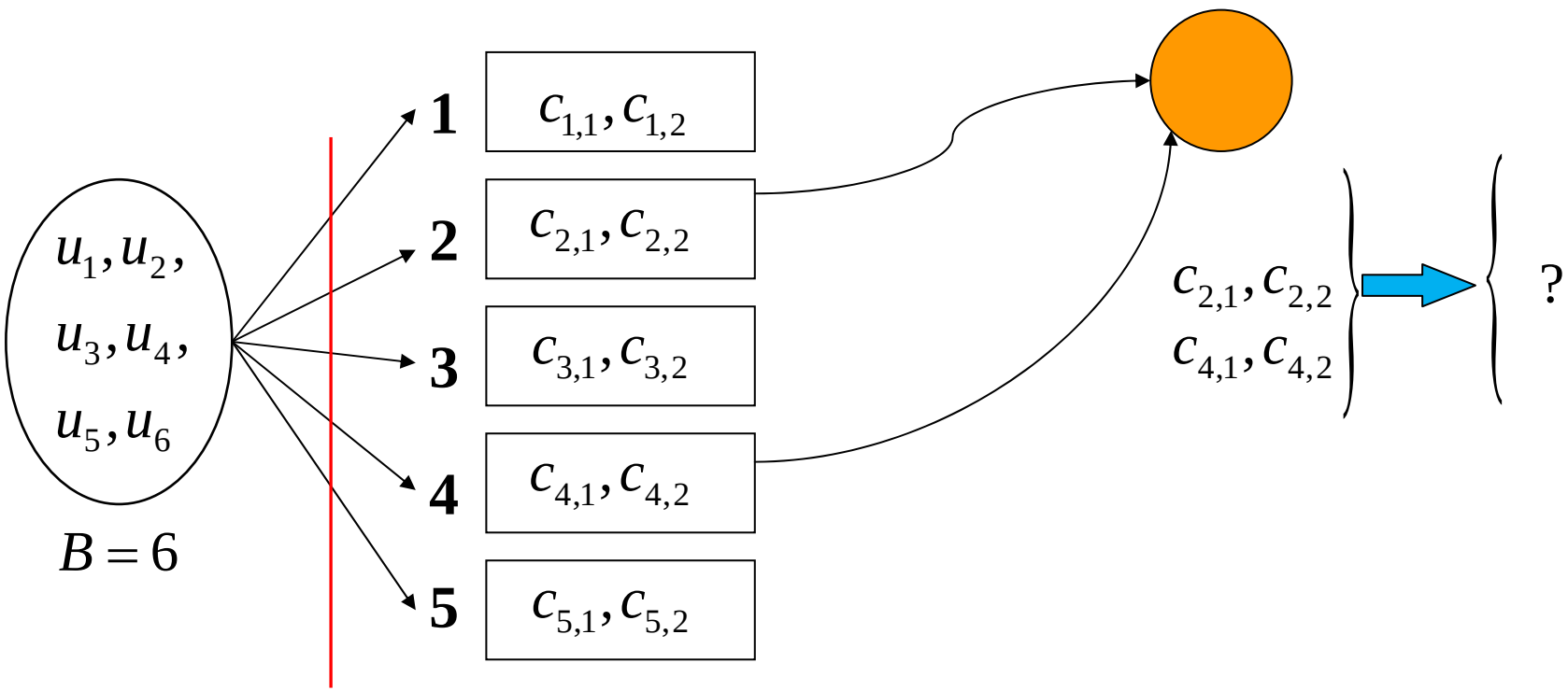
**再生成**

# 自明でない方法：故障ノードの修復 $(k,d)=(3,4)$



# 提案：分散データの安全性（秘密分散　その1）

$$H(u_2u_5)-H(u_2u_5\,|\,c_{2,1}c_{2,2}c_{4,1}c_{4,2})=0$$

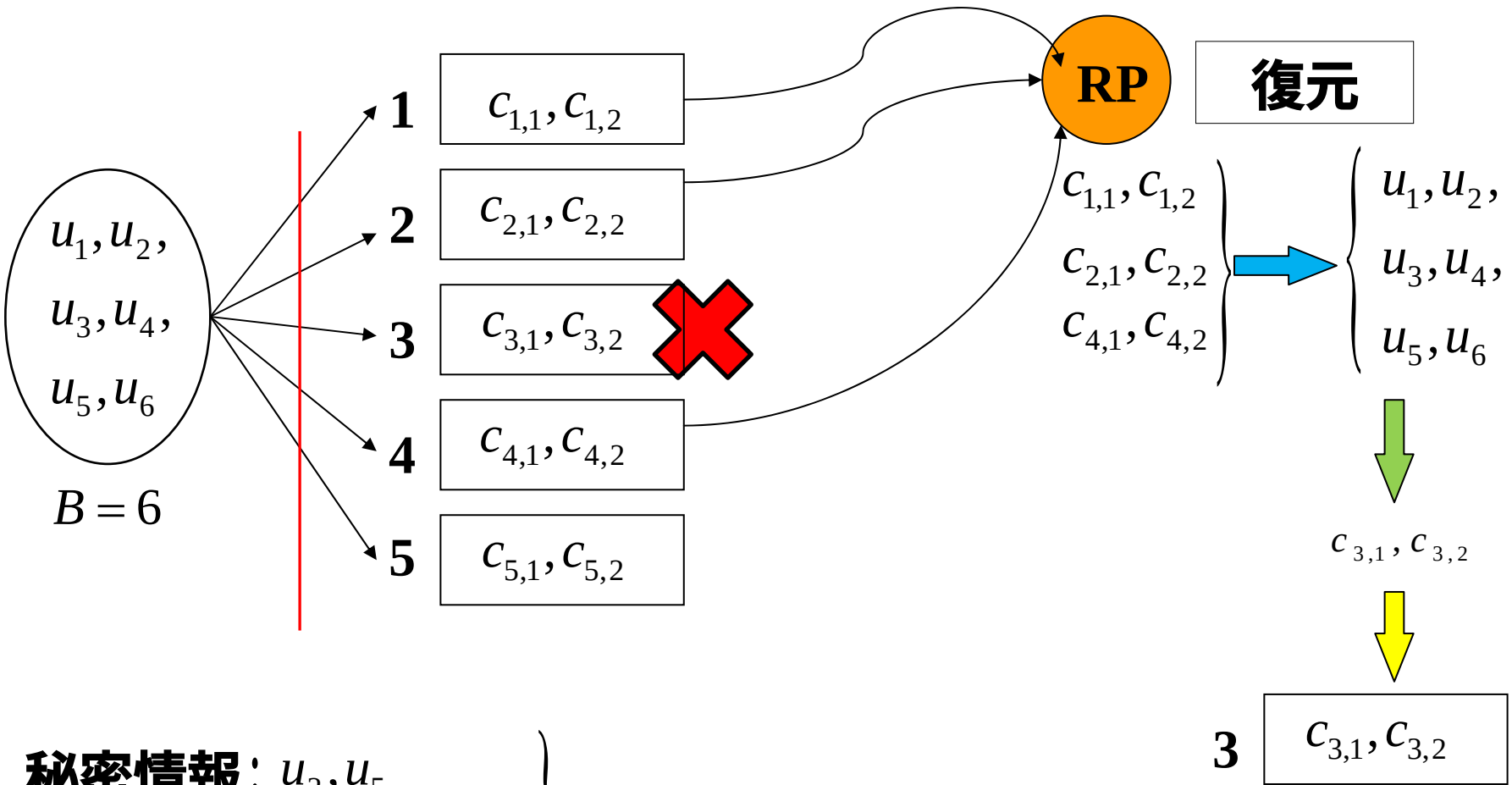



**秘密情報**：$u_2, u_5$

**乱数**：$u_1, u_3, u_4, u_6$

# 自明な方法における修復と安全性$(k,d)=(3,3)$

$$H(u_2u_5 \mid c_{1,1}c_{1,2}c_{2,1}c_{2,2}c_{4,1}c_{4,2})=0$$




**秘密情報**：$u_2,u_5$

**乱数**：$u_1,u_3,u_4,u_6$

**再生成**

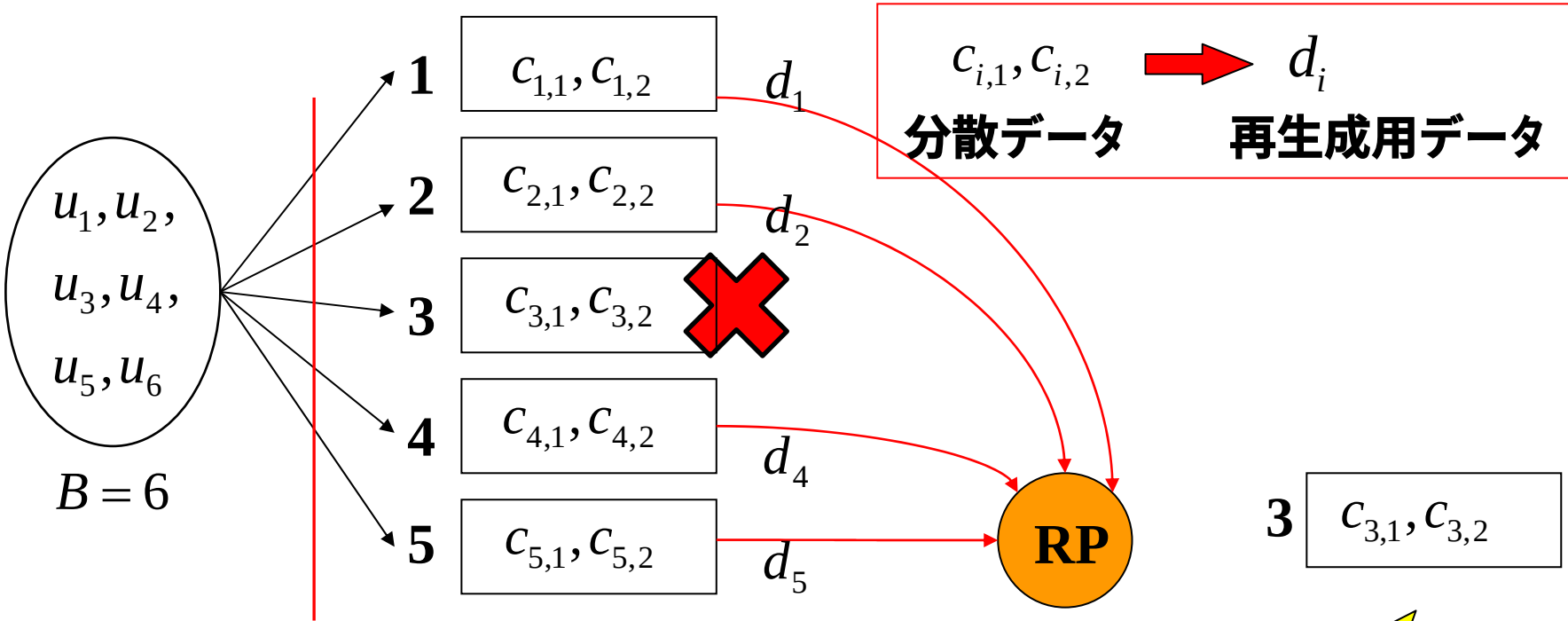
# 提案：再生成用データの安全性（秘密分散　その2）

$$H(u_2u_5)-H(u_2u_5\,|\,d_1d_2d_4d_5)=0$$

符号化

$c_{i,1},c_{i,2}$ → $d_i$

分散データ　　再生成用データ




再生成用データ　$d_1, d_2, d_4, d_5$ → $c_{3,1},c_{3,2}$

秘密情報：$u_2,u_5$

乱数：$u_1,u_3,u_4,u_6$

再生成

# Rashmi-Shah-Kumar 最小ストレージ再生成 (MSR) 符号 I



K.V.Rashmi, N.B.Shah, and P.V.Kumar が提案した
２タイプの符号, MBR 符号と MSR 符号

1. ２つのパラメータ $(d\beta, \alpha)$
   1. $d\beta$ : 修復バンドワイド
      （修復のために各ノードからダウンロードする
      データサイズ）
   2. $\alpha$ : ストレージサイズ
      （各ノードが保存できるデータサイズの上限）

1. $d\beta$ を最小にする最小バンドワイド再生成符号
   Minimum Bandwidth Regeneration(MBR) code
2. $\alpha$ を最小にする最小ストレージ再生成符号
   Minimum Storage Regeneration(MSR) code
   ⇓
   Rashmi-Shah-Kumar MSR 符号

# 符号化 (Rashmi-Shah-Kumar MSR 符号) I



1. **パラメータ** $(n, k, d, \alpha, \beta, B)$
   - $n$ : ストレージノードの個数（分散データの個数），
   - $k$ : 復元のためにアクセスするノードの個数,
   - $d$ : 修復のためにアクセスするノードの個数,
   - $\alpha$ : ノードが保存できるデータサイズの上限（分散データのサイズ），
   - $\beta$ : 修復のために各ノードからダウンロードする再生成データのサイズ,
   - $B$ : メッセージのデータサイズ.
2. **パラメータ設定**

$$
\begin{aligned}
d &= 2(k-1) = 2\alpha, \\
k &= \alpha + 1, \\
B &= \alpha(\alpha+1), \\
\beta &= 1.
\end{aligned}
$$

# 符号化 (Rashmi-Shah-Kumar MSR 符号) II



1. メッセージ $\underline{u} \in \mathbb{F}_q^B$
2. メッセージ行列 $M$ :
   $2\alpha \times \alpha$ 行列 （成分数 $= 2\alpha^2 \geq \alpha(\alpha+1) = B$）
3. ノード $i \in \{1, 2, \ldots, n\}$ と 非零な有限体の要素 $x_i \in \mathbb{F}_q$ のペア. ただし、$x_i \neq x_j$ if $i \neq j$.
4. （ノード $i$ に対する）符号化ベクトル

$$\begin{aligned} \underline{\phi}_i &= \left[1, x_i, x_i^2, \ldots, x_i^{\alpha-1}\right]^t \in \mathbb{F}_q^\alpha, \\ \underline{\psi}_i &= \left[\underline{\phi}_i^t, x_i^\alpha \underline{\phi}_i^t\right]^t \in \mathbb{F}_q^{2\alpha} \end{aligned}$$

5. ノード $i$ に保存する分散データ （分散符号化）

$$\underline{c}_i = [c_{i,1}, c_{i,2}, \ldots, c_{i,\alpha}]^t = \left[\underline{\psi}_i^t M\right]^t \in \mathbb{F}_q^\alpha$$

# 復元 (Rashmi-Shah-Kumar MSR 符号) I



1. ネットワーク上の $n$ 個のノードの中から任意の $k = \alpha + 1$ 個のノード $i_1, i_2, \ldots, i_k$ の**分散データ**

$$\underline{c}_{i_1}, \underline{c}_{i_2}, \ldots, \underline{c}_{i_k}$$

**からメッセージ $\underline{u}$ を復元できる。**

# 再生成（故障ノードの修復） (Rashmi-Shah-Kumar MSR 符号) I

1. 故障ノード $f(\in \{1,2,\ldots,n\})$ を新しいノードに置き換えて修復する。 
2. 新ノードは故障していないノードの中から任意の $d = 2\alpha$ 個のノード $h_1, h_2, \ldots, h_d$ にアクセスし、それぞれのノード $h_p$, $1 \le p \le d$, から サイズ $\beta = 1$ の**再生成用データ**

$$d_{h_p} = \underline{c}_{h_p}^t \underline{\phi}_f \in \mathbb{F}_q$$

をダウンロードする。ここで、 $\underline{\phi}_f = [1, x_f, \ldots, x_f^{\alpha-1}]^t \in \mathbb{F}_q^\alpha$ である。

3. ダウンロードした合計サイズ $d = 2\alpha$ の**再生成用データ**

$$d_{h_1}, d_{h_2}, \ldots, d_{h_d}$$

から故障ノード $f$ で保存していた分散データ $\underline{c}_f$ の複製を再生成できる。

# 秘密分散の設定 I



1. サイズ $B$ のメッセージ $\underline{u}$ の構成を**乱数**と**秘密情報**に分ける。
2. ただし、**メッセージ行列** $M$ のどの成分に乱数を対応させるかの工夫が必要である。
3. メッセージ $\underline{u}$ の内訳

$$\underline{u}_{\mathrm{r}} = ((\underline{u}_{1,1})^t, \ldots, (\underline{u}_{\alpha,\alpha})^t)^t \quad : \textbf{乱数 (サイズ } 2\alpha),$$
$$\underline{u}_{\mathrm{S}} \quad : \textbf{秘密情報 (サイズ } \alpha(\alpha-1)).$$

ここで、$B = \alpha(\alpha+1) = 2\alpha + \alpha(\alpha-1)$ である。

# 秘密分散の設定 II



4. **その工夫：メッセージ行列 $M$ での乱数 $\underline{u}_{\mathrm{r}}$ の配置**

$$M = \begin{bmatrix} u_{1,1}^{(1)} & & & \\ & u_{2,2}^{(1)} & & \\ & & \ddots & \\ & & & u_{\alpha,\alpha}^{(1)} \\ u_{1,1}^{(2)} & & & \\ & u_{2,2}^{(2)} & & \\ & & \ddots & \\ & & & u_{\alpha,\alpha}^{(2)} \end{bmatrix}$$

# 分散データの安全性（秘密分散 その１） I



1. 任意の２個のノード $i_1$ と $i_2$ が保存する合計サイズ $2\alpha$ の分散データ

$$\begin{aligned} \underline{c}_{i_1} &= [c_{i_1,1}, c_{i_1,2}, \ldots, c_{i_1,\alpha}] \\ \underline{c}_{i_2} &= [c_{i_2,1}, c_{i_2,2}, \ldots, c_{i_2,\alpha}] \end{aligned}$$

に対し、

$$\begin{bmatrix} c_{i_1,1} \\ c_{i_2,1} \\ c_{i_1,2} \\ c_{i_2,2} \\ \vdots \\ c_{i_1,\alpha} \\ c_{i_2,\alpha} \end{bmatrix} = \begin{bmatrix} C_{2\alpha} & \overline{C}_{2\alpha} \\ (2\alpha \times 2\alpha) & (2\alpha \times (\alpha(\alpha-1)) \end{bmatrix} \begin{bmatrix} \underline{u}_{\mathrm{r}} \\ \underline{u}_{\mathrm{s}} \end{bmatrix}$$

と書ける。

# 分散データの安全性（秘密分散 その１） II



2. サイズ $2\alpha$ の乱数 $\underline{u}_{\mathrm{r}}$ に対応する $2\alpha \times 2\alpha$ 行列 $C_{2\alpha}$ は，

$$C_{2\alpha} = \begin{bmatrix} 1 & x_{i_1}^{\alpha} & & & & \\ 1 & x_{i_2}^{\alpha} & & & O & \\ & & x_{i_1} & x_{i_1}^{\alpha+1} & & \\ & & x_{i_2} & x_{i_2}^{\alpha+1} & & \\ & & & & \ddots & \\ & O & & & x_{i_1}^{\alpha-1} & x_{i_1}^{2\alpha-1} \\ & & & & x_{i_2}^{\alpha-1} & x_{i_2}^{2\alpha-1} \end{bmatrix}$$

である。

# 分散データの安全性（秘密分散 その１） III



3. このとき、その行列式は

$$\begin{aligned}\det C_{2\alpha} &= (x_{i_1}x_{i_2})^{\frac{\alpha(\alpha-1)}{2}}\left(\det\begin{bmatrix}1 & x_{i_1}^{\alpha}\\ 1 & x_{i_2}^{\alpha}\end{bmatrix}\right)^{\alpha}\\ &= (x_{i_1}x_{i_2})^{\frac{\alpha(\alpha-1)}{2}}(x_{i_2}^{\alpha}-x_{i_1}^{\alpha})^{\alpha}\end{aligned}$$

となる。

4. したがって、任意の異なる２個の要素 $x_i, x_j$ に対し、$x_i^{\alpha} \neq x_j^{\alpha}$ が成り立つならば、$\det C_{2\alpha} \neq 0$ が成り立つ。
5. このとき、任意の２個のノードが保存するサイズ $2\alpha$ の分散データ $\underline{c}_{i_1}^t, \underline{c}_{i_2}^t$ から秘密情報 $\underline{u}_{\mathrm{S}}$ はまったく得られない。すなわち、

$$H(\underline{u}_{\mathrm{S}}) - H(\underline{u}_{\mathrm{S}}|\underline{c}_{i_1}^t\,\underline{c}_{i_2}^t) = 0$$

が成り立つ。

# 再生成用データの安全性（秘密分散 その２） I



1. 任意の $d = 2\alpha$ 個のノード $h_1, h_2, \ldots, h_{2\alpha}$ からダウンロードした再生成用データ $d_{h_1}, d_{h_2}, \ldots, d_{h_{2\alpha}}$ に対し、

$$\begin{bmatrix} d_{h_1} \\ d_{h_2} \\ \vdots \\ d_{h_{2\alpha}} \end{bmatrix} = \begin{bmatrix} D_{2\alpha} & \overline{D}_{2\alpha} \\ (2\alpha \times 2\alpha) & (2\alpha \times (\alpha(\alpha-1)) \end{bmatrix} \begin{bmatrix} \underline{u}_{\mathrm{r}} \\ \underline{u}_{\mathrm{S}} \end{bmatrix}$$

と書ける。

# 再生成用データの安全性（秘密分散 その２） II



2. サイズ $2\alpha$ の乱数 $\underline{u}_{\mathrm{r}}$ に対応する $2\alpha \times 2\alpha$ 行列 $D_{2\alpha}$ は，

$$D_{2\alpha} = \begin{bmatrix} \underline{x}_{h_1}^t, & (x_f x_{h_1})\underline{x}_{h_1}^t & , \ldots , & (x_f x_{h_1})^{\alpha-1}\underline{x}_{h_1}^t \\ \vdots & \vdots & \cdots & \vdots \\ \underline{x}_{h_{2\alpha}}^t, & (x_f x_{h_{2\alpha}})\underline{x}_{h_{2\alpha}}^t & , \ldots , & (x_f x_{h_{2\alpha}})^{\alpha-1}\underline{x}_{h_{2\alpha}}^t \end{bmatrix}$$

である。ここで、

$$\underline{x}_i = [1, x_i^{\alpha}]^t \in \mathbb{F}_q^2.$$

3. このとき、その行列式は

$$\det D_{2\alpha} = (-1)^{\sigma} x_f^{\alpha(\alpha-1)} \det \begin{bmatrix} 1 & x_{h_1} & x_{h_1}^2 & \cdots & x_{h_1}^{2\alpha-1} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ 1 & x_{h_{2\alpha}} & x_{h_{2\alpha}}^2 & \cdots & x_{h_{2\alpha}}^{2\alpha-1} \end{bmatrix} \neq 0$$

となる。ここで、記号 $\sigma$ は列ベクトルの基本操作による置換の回数を表す。

# 再生成用データの安全性（秘密分散 その２） III



4. ゆえに、$2\alpha$ 個の再生成用データ $d_{h_1}, \ldots, d_{h_{2\alpha}}$ から秘密情報 $\underline{u}_S$ はまったく得られない。すなわち、

$$H(\underline{u}_S^t) - H(\underline{u}_S^t | d_{h_1} \cdots d_{h_{2\alpha}}) = 0$$

が成り立つ。

# 結論



本稿では、分散ストレージシステムの修復問題において、K.V.Rashmi, N.B.Shah and P.V.Kumar が提案している Rashmi-Shah-Kumar MSR 符号に対し、

1. **秘密分散**構造の構築方法

を提案し、次の 2 点の安全性を示した。

1. 任意の 2 個のノードが保存する合計サイズ $2\alpha$ の**分散データ**に対する秘密分散の安全性、
2. 故障ノードを修復するためのサイズ $2\alpha$ の**再生成用データ**に対する秘密分散の安全性。

# 主な参考文献




1. [1] A.G.Dimakis, P.B.Godfrey, Y.Wu, M.J.Wainwright and K.Ramchandran, ”Network Coding for Distributed Storage Systems,” IEEE Trans. on Information Theory, vol.56, no.9, pp.4539–4551, Sept. 2010.
2. [2] A.G.Dimakis, K.Ramchandran, Y.Wu and C.Suh, “A Survey on Network Codes for Distributed Storage,” http://arxiv.org/abs/1004.4438
3. [3] K.V.Rashmi, N.B.Shah, and P.V.Kumar, “Optimal Exact-Regenerating Codes for Distributed Storage at the MSR and MBR Points via a Product-Matrix Construction,” http://arxiv.org/abs/1005.4178
4. [4] C.Suh and K.Ramchandran, “Exact Regeneration Codes for Distributed Storage Repair Using Interference Alignment,” http://arxiv.org/abs/1001.0107
5. [5] S.Pawar, S.E.Rouayheb, and K.Ramchandran, “On Secure Distributed Data Storage Under Repair Dynamics,” http://arxiv.org/abs/1003.0488

# 追加資料(additional slides)

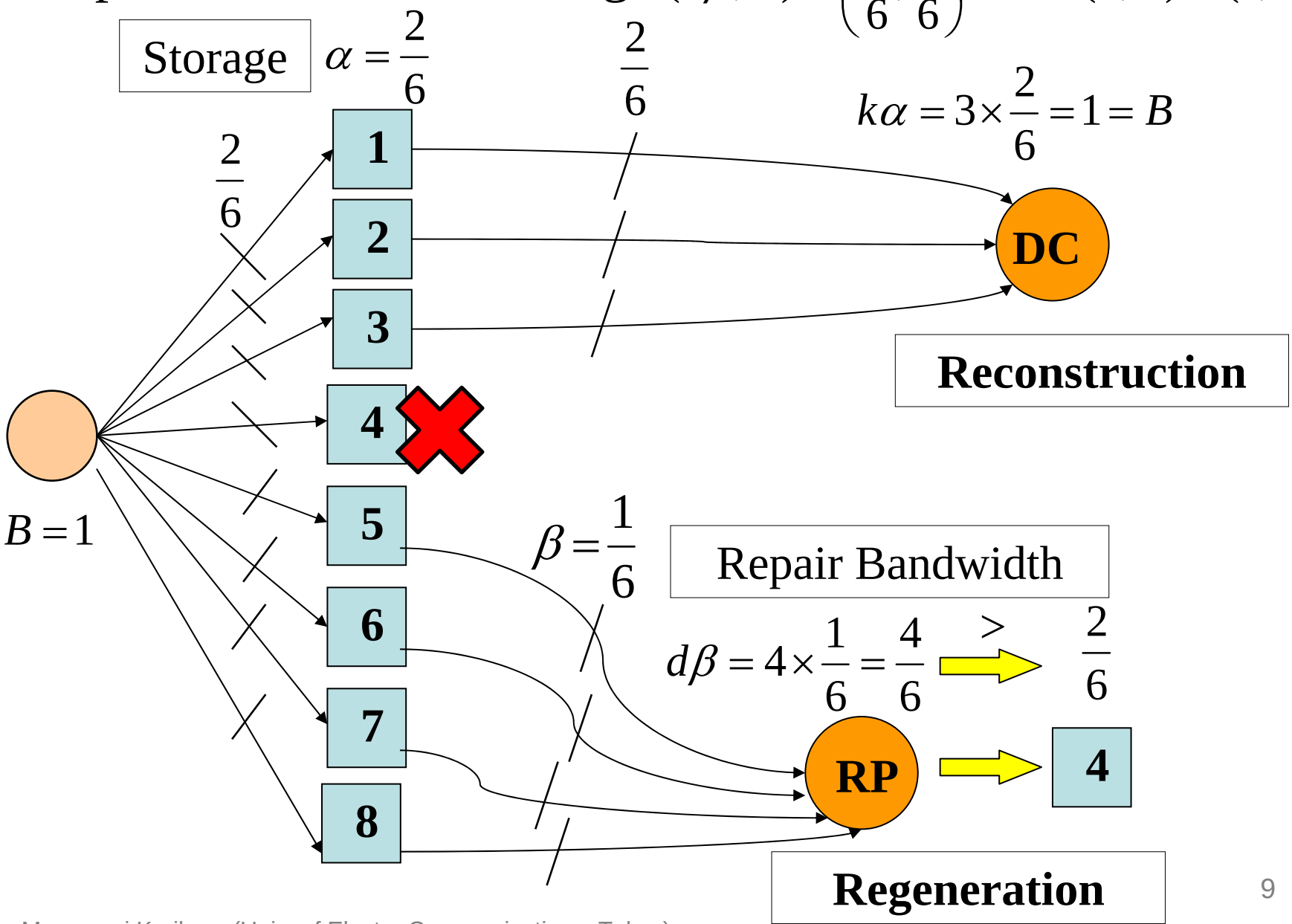
① Tradeoff between storage and repair bandwidth

② **あるMBR符号とその秘密分散の例**

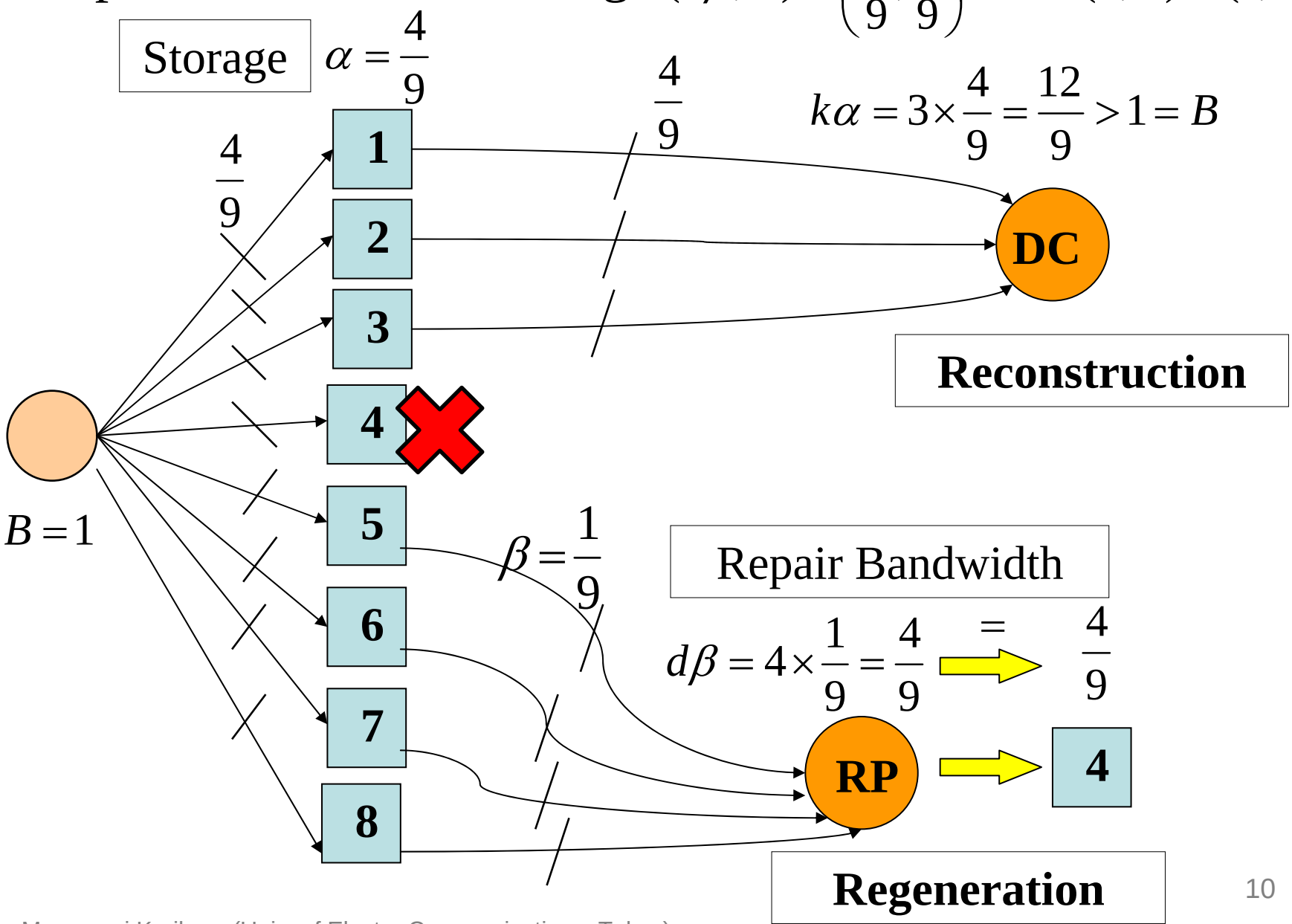
③ Rashmi-Shah-Kumar MSR**符号と秘密分散**

# Repair Bandwidth - Storage $(d\beta, \alpha) = \left(\frac{4}{6}, \frac{2}{6}\right)$ for $(k, d) = (3,4)$



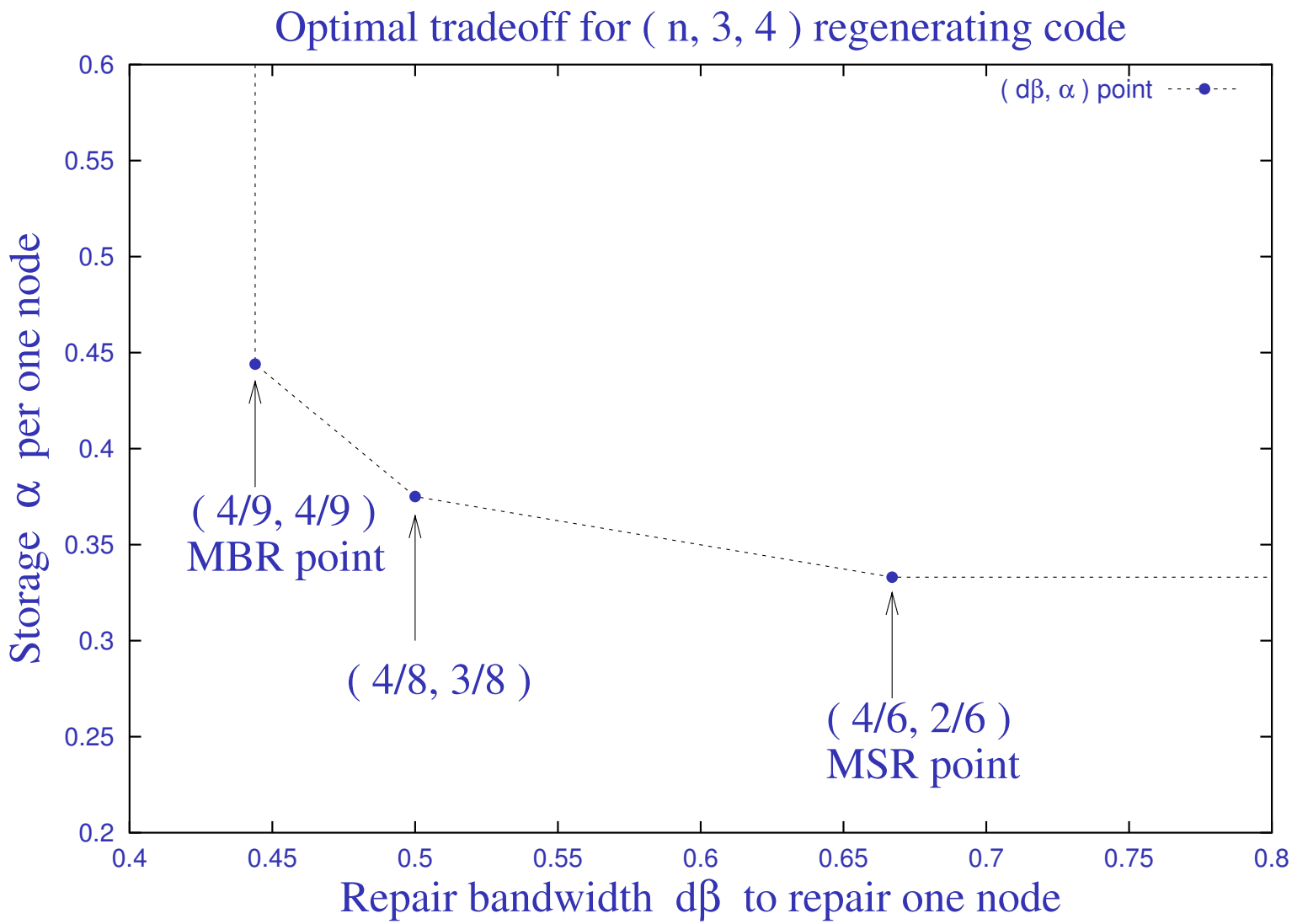


# Repair Bandwidth - Storage $(d\beta, \alpha) = \left(\frac{4}{9}, \frac{4}{9}\right)$ for $(k, d) = (3,4)$



# Tradeoff curve between strage $\alpha$ and repair bandwidth $d\beta$ I





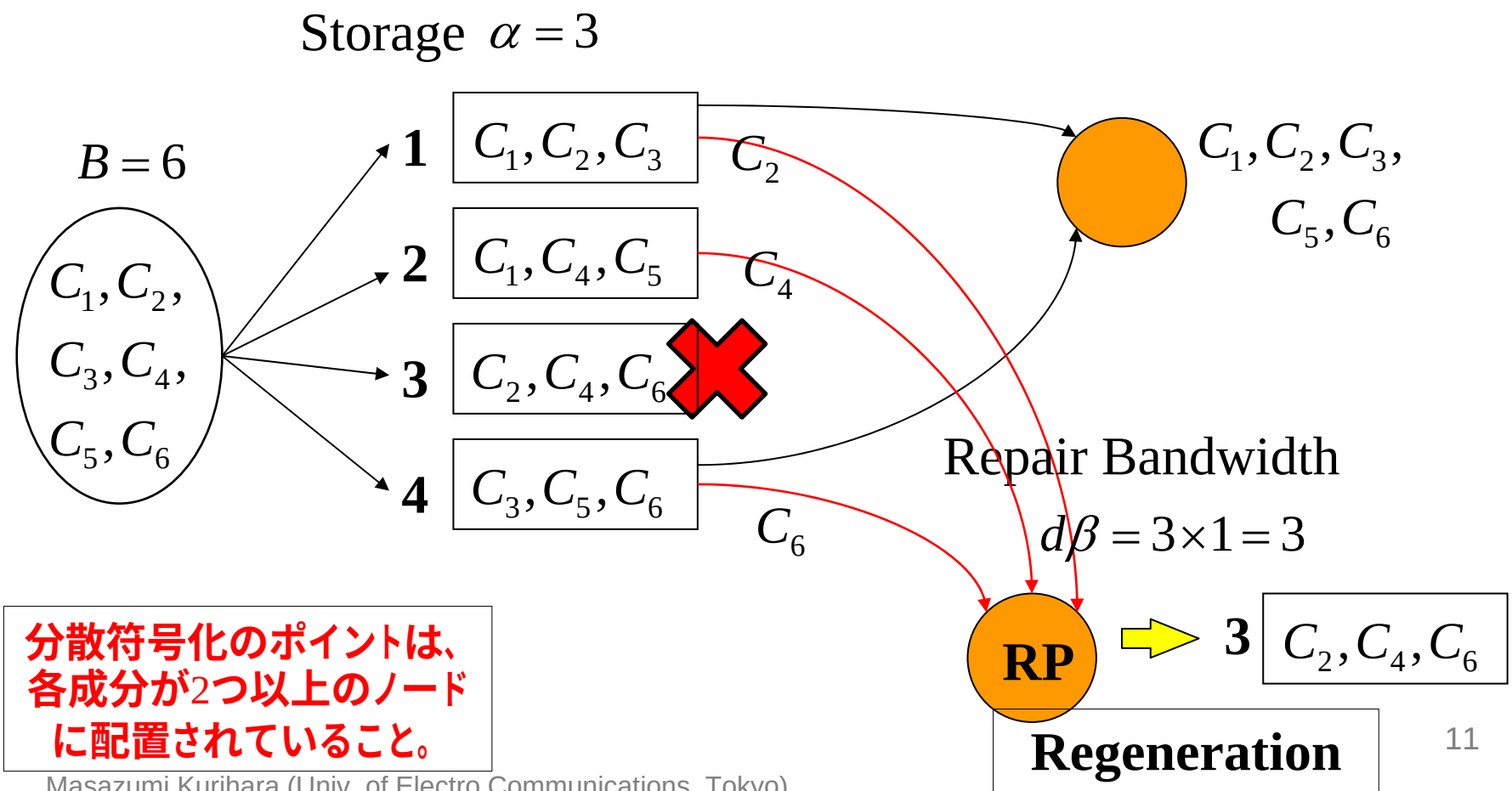

Optimal tradeoff curve between strage $\alpha$ and repair bandwidth $d\beta$ for $(k, d) = (3, 4)$ and $B = 1$.

# あるMBR符号とその秘密分散の例[5] $(k,d)=(3,3)$

**秘密情報**： $S$

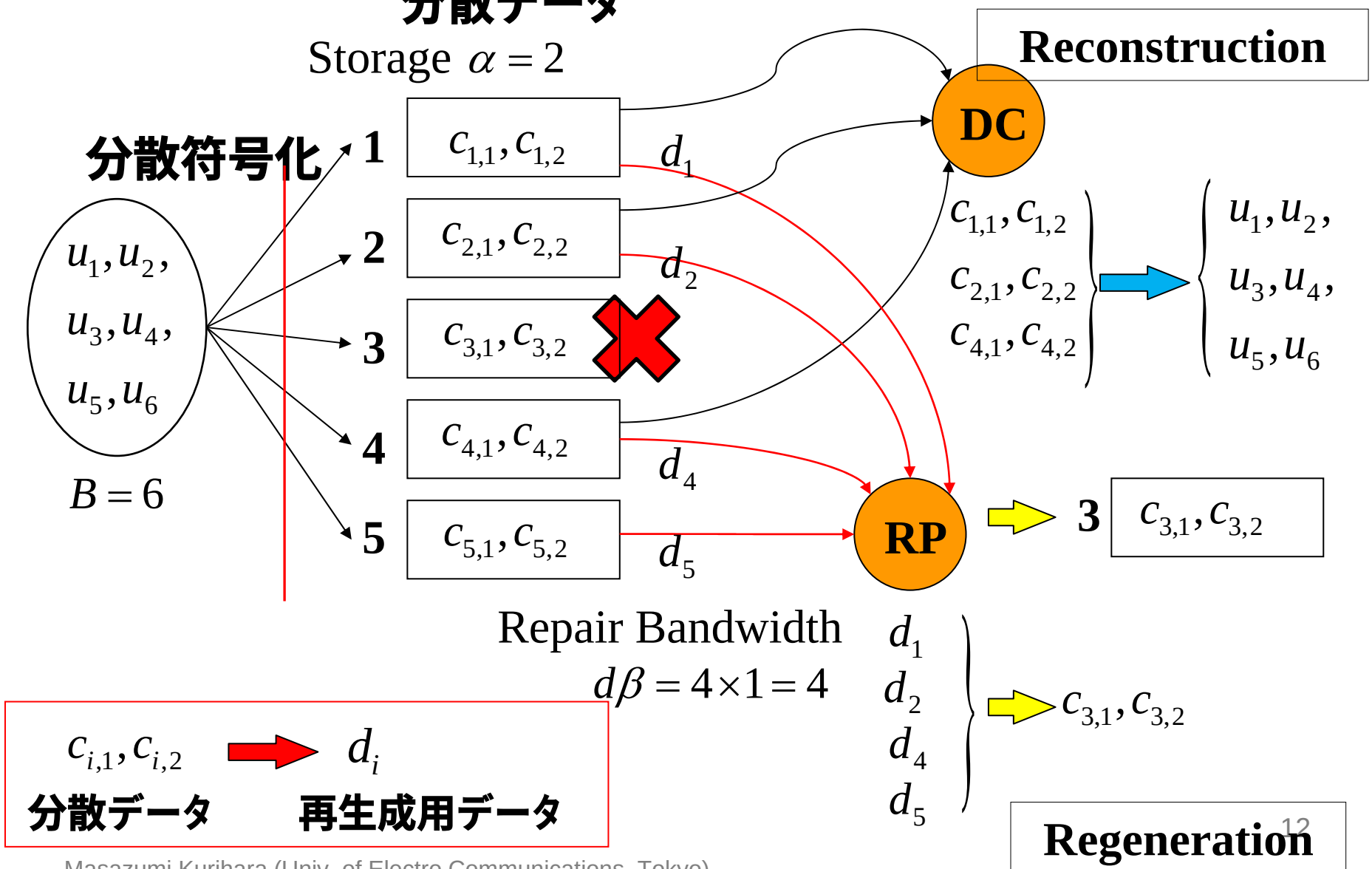
**乱数**： $R_1, R_2, R_3, R_4, R_5$

$$\Rightarrow \begin{cases} C_i = R_i, i = 1,2,3,4,5 \\ C_6 = S - (R_1 + R_2 + R_3 + R_4 + R_5) \end{cases}$$



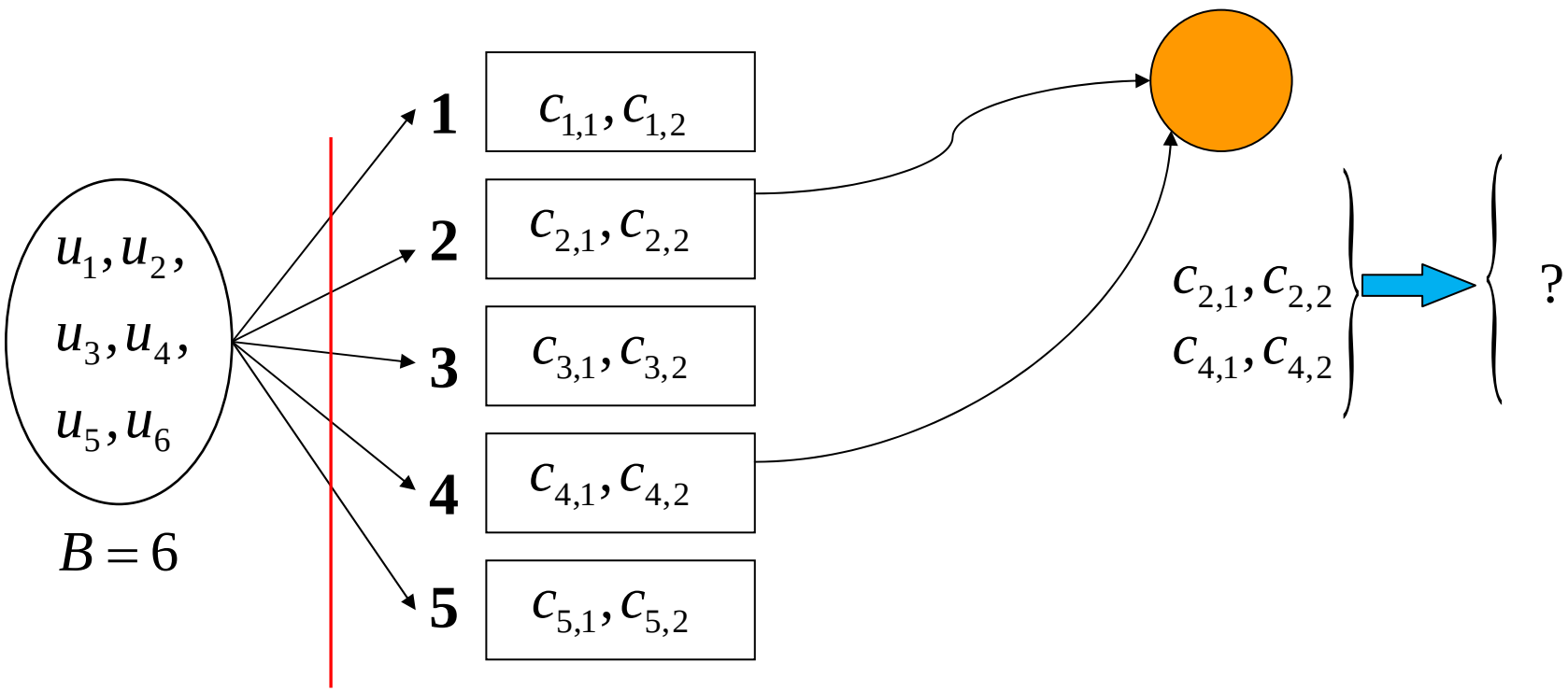

**分散符号化のポイントは、各成分が2つ以上のノードに配置されていること。**

# Rashmi-Shah-Kumar MSR符号[3] $(k,d)=(3,4)$



# 分散データ（Rashmi-Shah-Kumar MSR符号と秘密分散）

$$H(u_2u_5) - H(u_2u_5 \mid c_{2,1}c_{2,2}c_{4,1}c_{4,2}) = 0$$




**秘密情報**：$u_2, u_5$

**乱数**：$u_1, u_3, u_4, u_6$

# 再生成用データ (Rashmi-Shah-Kumar MSR符号と秘密分散)

$$H(u_2u_5) - H(u_2u_5 \mid d_1d_2d_4d_5) = 0$$