Capacity of 1-to-K Broadcast Packet Erasure Channels with Channel Output Feedback — A Packet Evolution Approach

Chih-Chun Wang, Purdue University

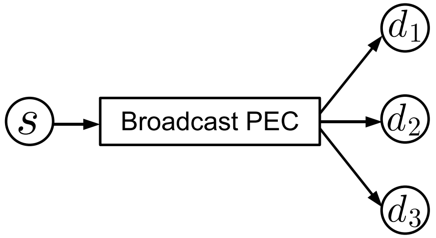
Presented in the 48-th Allerton Conference, 9/30/2010

Joint work with Y. Charlie Hu (Purdue), Ness B. Shroff (The OSU), Dimitrios Koutsonikolas, Abdallah Khreishah.

Sponsored by NSF CCF-0845968 and CNS-0905331.

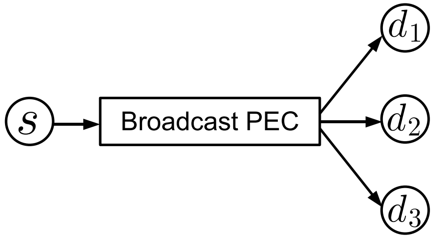
Two Ingredients

- Packet Erasure Channels (PECs):
  - Input: $X \in \mathsf{GF}(2^b)$ for large b.
  - A packet X either arrives perfectly (with the help of CRC), or is considered as erasure and discarded. (No hybrid ARQ).
  - Memoryless, time-invariant.



Two Ingredients

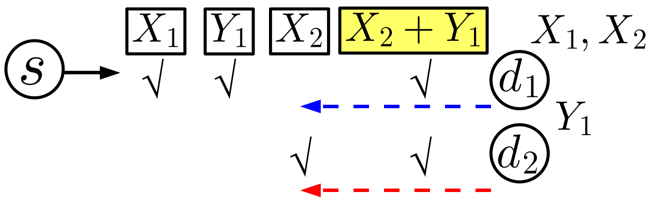
- Packet Erasure Channels (PECs):



  - Input: $X \in \mathsf{GF}(2^b)$ for large b.
  - A packet X either arrives perfectly (with the help of CRC), or is considered as erasure and discarded. (No hybrid ARQ).
  - Memoryless, time-invariant.

- The ER protocol — 1-hop cellular networks [Rozner *et al.* 07].

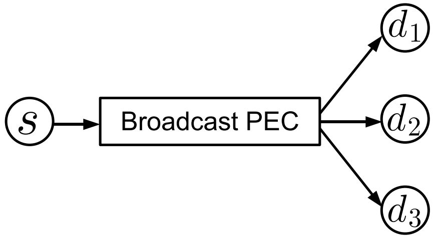
  5 transmissions w/o coding vs. 4 transmissions w. coding



  - Create its own SI through spatial diversity.
  - Empirically, 10–20% throughput improvement.

Two Ingredients

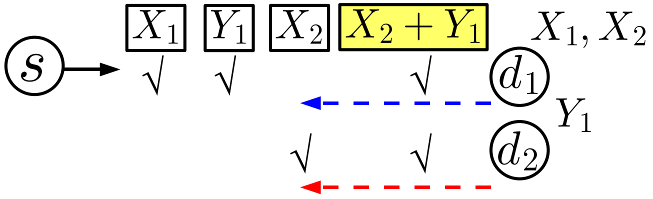
- Packet Erasure Channels (PECs):



  - Input: $X \in \mathsf{GF}(2^b)$ for large b.
  - A packet X either arrives perfectly (with the help of CRC), or is considered as erasure and discarded. (No hybrid ARQ).
  - Memoryless, time-invariant.

- The ER protocol — 1-hop cellular networks [Rozner *et al.* 07].

  5 transmissions w/o coding vs. 4 transmissions w. coding

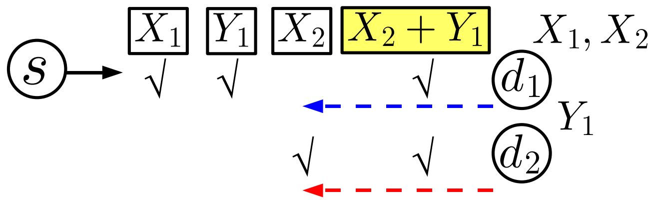


  - Create its own SI through spatial diversity.
  - Empirically, 10–20% throughput improvement.

- Our goal: Finding the Shannon capacity of PECs with channel output feedback (COF) for arbitrary number $M \geq 3$ of sessions.

Main Results & Contents

- The benefits of ER follows from the channel output feedback (COF).

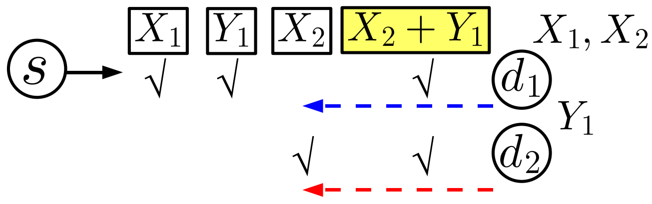


# of sessions	ER-like Protocols (Broadcast PECs w. **COF**)	Gaussian broadcast channels w. **COF**
M=2	Full capacity region [Georgiadis et al. 09]	Outer and inner bounds [Ozarow 84]
M=3	Full capacity region	?
General M	(1) Capacity for fair systems; (2) Outer and inner bounds that meet numerically.	?

Capacity-achieving schemes by code alignment.

Main Results & Contents

- The benefits of ER follows from the channel output feedback (COF).



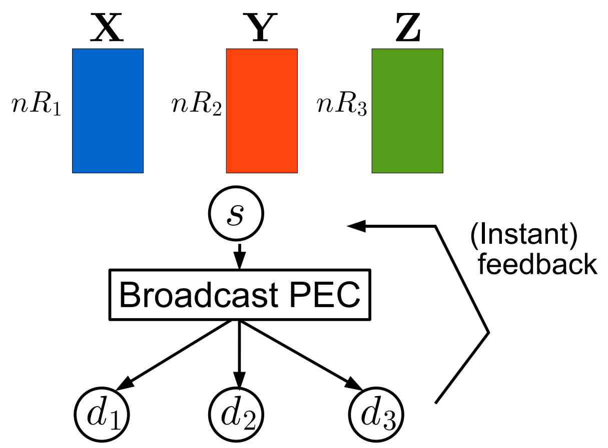
# of sessions	ER-like Protocols (Broadcast PECs w. **COF**)	Gaussian broadcast channels w. **COF**
M=2	Full capacity region [Georgiadis et al. 09]	Outer and inner bounds [Ozarow 84]
M=3	Full capacity region	?
General M	(1) Capacity for fair systems; (2) Outer and inner bounds that meet numerically.	?

Capacity-achieving schemes by code alignment.

- The problem setting.
- Existing results for $M = 2$ [Georgiadis *et al.* 09].
- New concepts of code alignment and packet evolution.
- Main theorems and numerical evaluation.

1-Hop Cellular (AP) Networks

- 1-hop access point networks. M dest.
- M can be large, say ≈ 20.
- Each session has nR_i packets.
- The source s uses the channel n times.



- Our goal is to maximize the achievable rate vector $(R_1, \cdots, R_M)$.

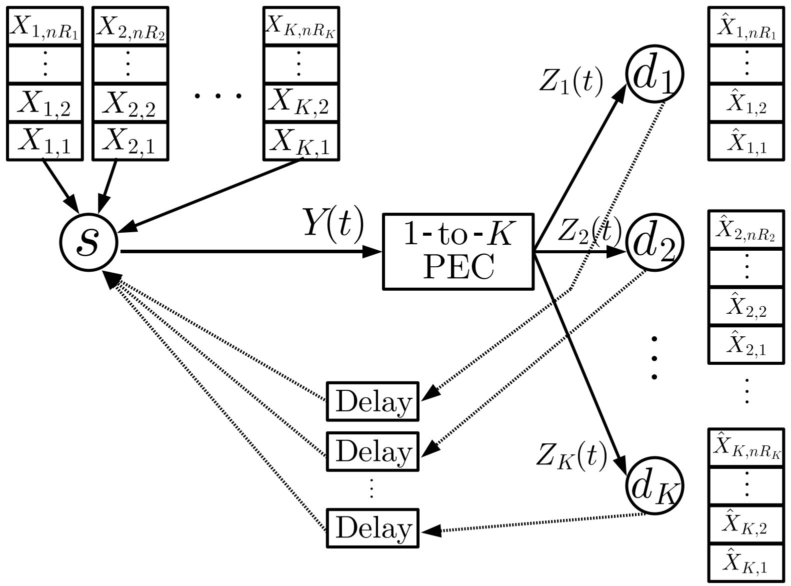
Formal Definition of Feasibility

A network code is defined by the following functions:

info. channel output feedback

$$Y(t) = f_t(\{X_{k,l} : k \in [M], l \in [nR_k]\}, \{Z_k(\tau) : k \in [M], \tau \in [t-1]\}),$$

$$\hat{\mathbf{X}}_k = g_k(\{Z_k(\tau) : \tau \in [n]\}).$$



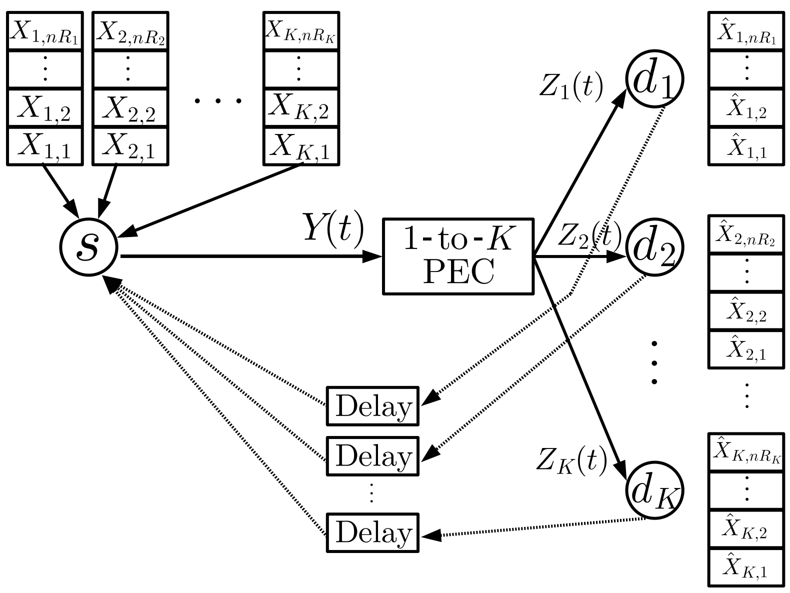
Formal Definition of Feasibility

A network code is defined by the following functions:

info. — channel output feedback

$$Y(t) = f_t(\{X_{k,l} : k \in [M], l \in [nR_k]\}, \{Z_k(\tau) : k \in [M], \tau \in [t-1]\}),$$

$$\hat{\mathbf{X}}_k = g_k(\{Z_k(\tau) : \tau \in [n]\}).$$

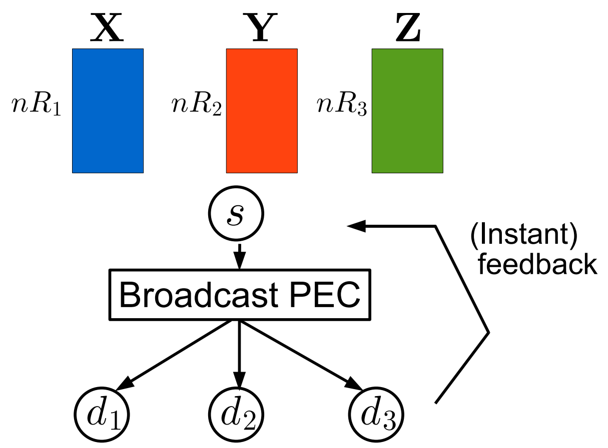


Definition 1 $(R_1, \cdots, R_M)$ *is achievable if* $\forall \epsilon > 0$, *there exist a sufficiently large* n, *a sufficiently large finite field* $\mathrm{GF}(2^b)$, *and a corresponding network code, such that for independently and uniformly distributed* $\mathbf{X}_k$, $k \in [M]$:

$$\max_{k \in [M]} \mathsf{P}\left(\hat{\mathbf{X}}_k \neq \mathbf{X}_k\right) < \epsilon.$$

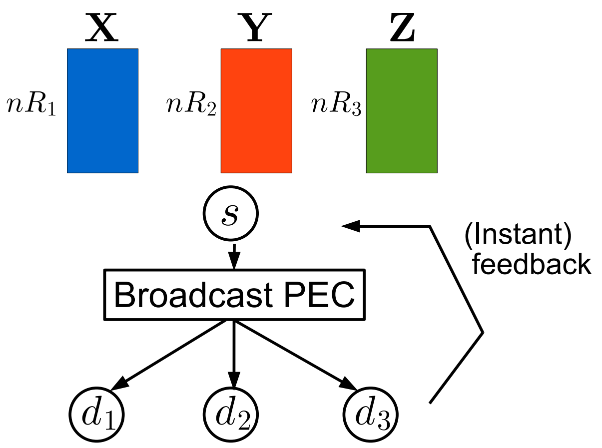
1-Hop Cellular (AP) Networks

- 1-hop access point networks. M dest.
- M can be large, say ≈ 20.
  (For 2-hop relay networks $M \leq 6$).
- Each session has nR_i packets.
- The source s uses the channel n times.



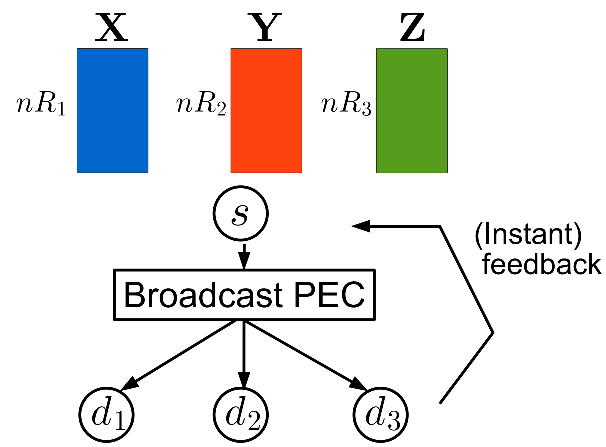
1-Hop Cellular (AP) Networks

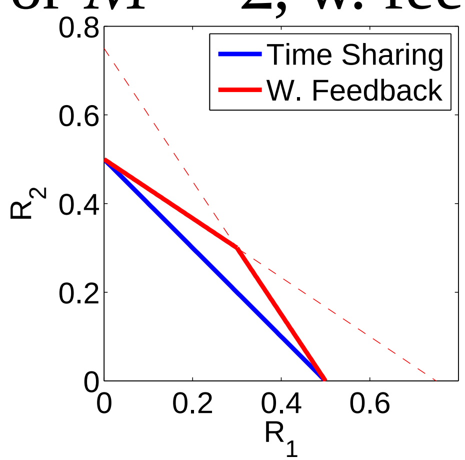
- 1-hop access point networks. M dest.
- M can be large, say ≈ 20.
  (For 2-hop relay networks $M \leq 6$).
- Each session has nR_i packets.
- The source s uses the channel n times.
- For $M = 2$, no feedback, the capacity is $\frac{R_1}{p_1} + \frac{R_2}{p_2} \leq 1$.



1-Hop Cellular (AP) Networks

- 1-hop access point networks. M dest.
- M can be large, say ≈ 20.
  (For 2-hop relay networks $M \leq 6$).
- Each session has nR_i packets.
- The source s uses the channel n times.
- For $M = 2$, no feedback, the capacity is $\frac{R_1}{p_1} + \frac{R_2}{p_2} \leq 1$.
- For $M = 2$, w. feedback, the capacity is [Georgiadis *et al.* 09].

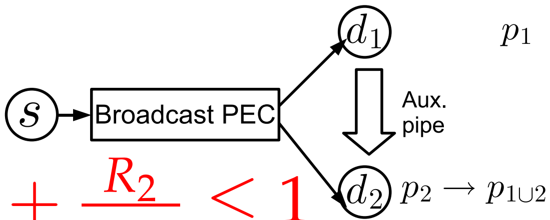




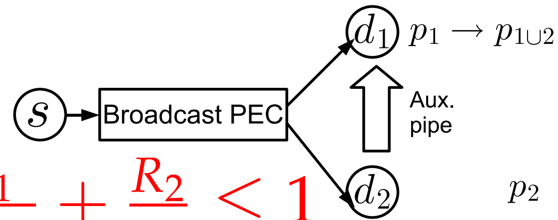
$$\begin{cases} \frac{R_1}{p_{1\cup 2}} + \frac{R_2}{p_2} \leq 1 \\ \frac{R_1}{p_1} + \frac{R_2}{p_{1\cup 2}} \leq 1 \end{cases}$$

Georgiadis' Proof

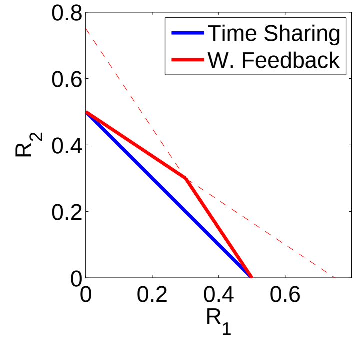
- Outer bound [Ozarow *et al.* 84]: Introduce auxiliary pipes to convert it into *physically degraded channels*, for which feedback does not increase the capacity [El Gamal 78].



$$\frac{R_1}{p_1} + \frac{R_2}{p_{1\cup 2}} \leq 1$$



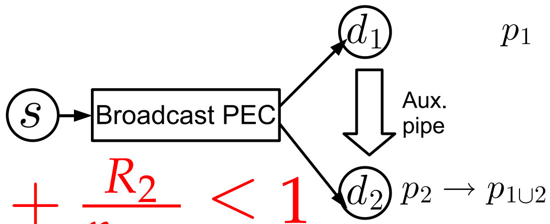
$$\frac{R_1}{p_{1\cup 2}} + \frac{R_2}{p_2} \leq 1$$



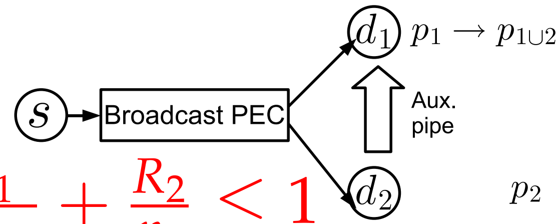
The cap. of the original CH with feedback
$\prec$ The cap. of the new physically degraded CH with feedback
$=$ The cap. of the new physically degraded CH without feedback

Georgiadis' Proof

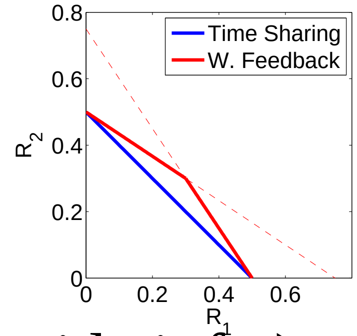
- Outer bound [Ozarow *et al.* 84]: Introduce auxiliary pipes to convert it into *physically degraded channels*, for which feedback does not increase the capacity [El Gamal 78].



$$\frac{R_1}{p_1} + \frac{R_2}{p_{1\cup 2}} \leq 1$$



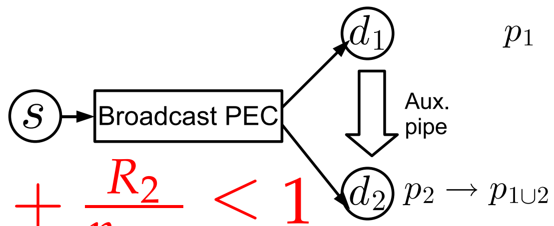
$$\frac{R_1}{p_{1\cup 2}} + \frac{R_2}{p_2} \leq 1$$



- Inner bound: A 2-phase approach. (Creating its own side info.)

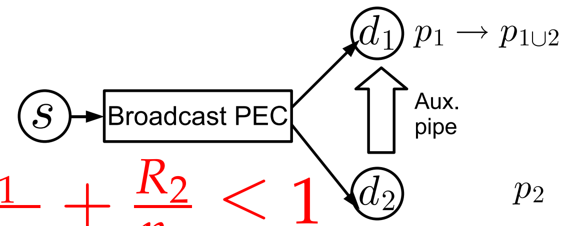
Georgiadis' Proof

- Outer bound [Ozarow *et al.* 84]: Introduce auxiliary pipes to convert it into *physically degraded channels*, for which feedback does not increase the capacity [El Gamal 78].



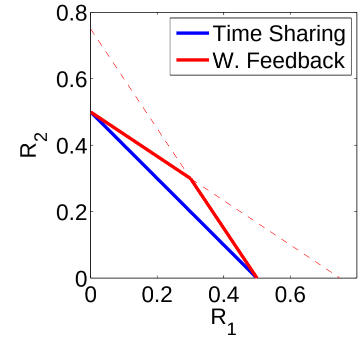
$$\frac{R_1}{p_1} + \frac{R_2}{p_{1\cup 2}} \leq 1$$

Phase 1

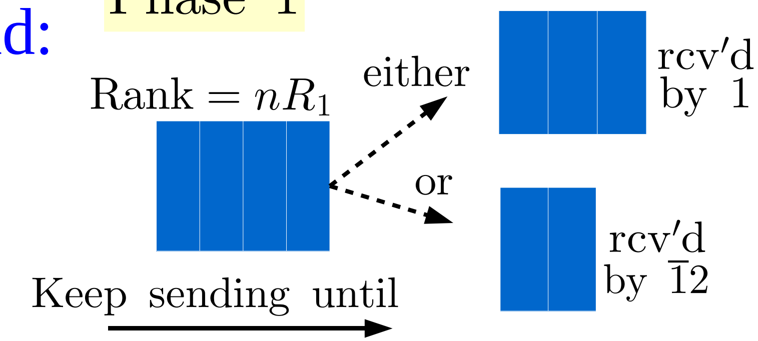


$$\frac{R_1}{p_{1\cup 2}} + \frac{R_2}{p_2} \leq 1$$

Phase 2

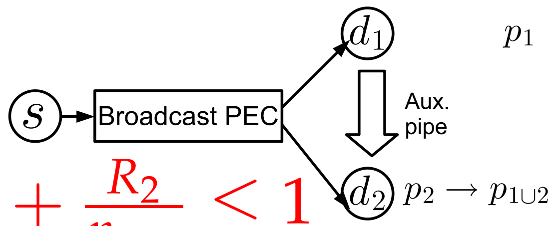


- Inner bound:



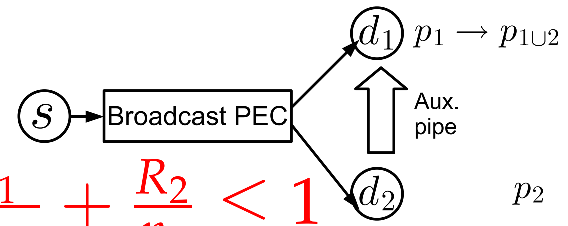
Georgiadis' Proof

- Outer bound [Ozarow *et al.* 84]: Introduce auxiliary pipes to convert it into *physically degraded channels*, for which feedback does not increase the capacity [El Gamal 78].



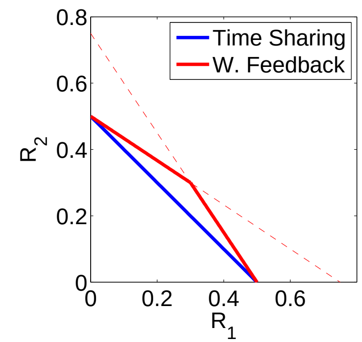
$$\frac{R_1}{p_1}+\frac{R_2}{p_{1\cup 2}}\leq 1$$

Phase 1

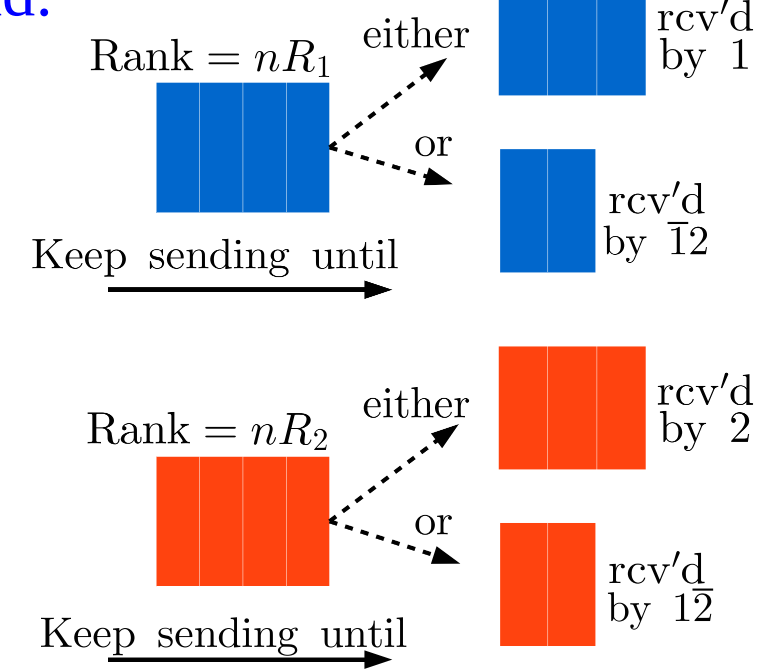


$$\frac{R_1}{p_{1\cup 2}}+\frac{R_2}{p_2}\leq 1$$

Phase 2

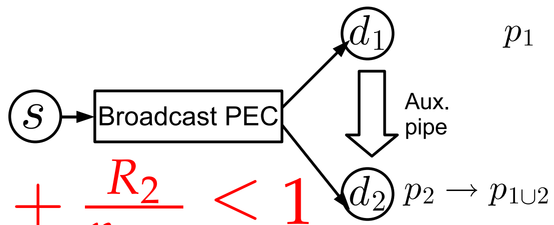


- Inner bound:

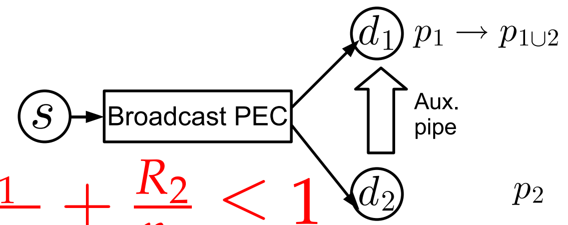


Georgiadis' Proof

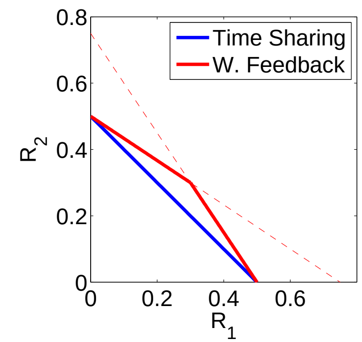
- Outer bound [Ozarow *et al.* 84]: Introduce auxiliary pipes to convert it into *physically degraded channels*, for which feedback does not increase the capacity [El Gamal 78].



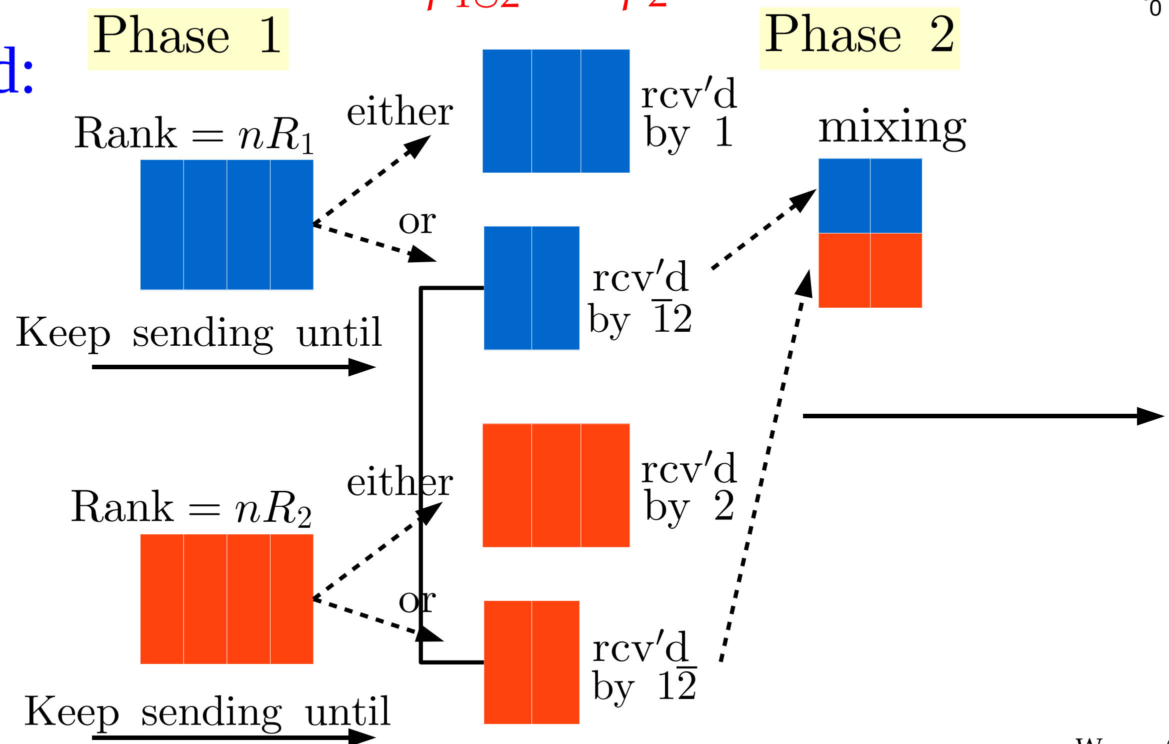
$$\frac{R_1}{p_1} + \frac{R_2}{p_{1\cup 2}} \leq 1$$



$$\frac{R_1}{p_{1\cup 2}} + \frac{R_2}{p_2} \leq 1$$

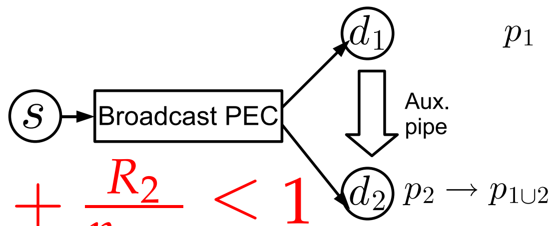


- Inner bound:

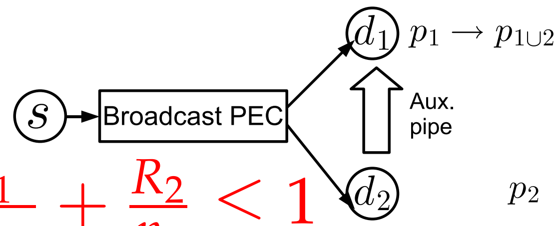


Georgiadis' Proof

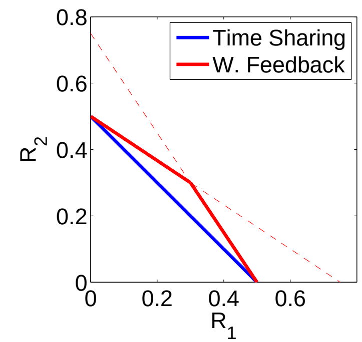
- Outer bound [Ozarow *et al.* 84]: Introduce auxiliary pipes to convert it into *physically degraded channels*, for which feedback does not increase the capacity [El Gamal 78].



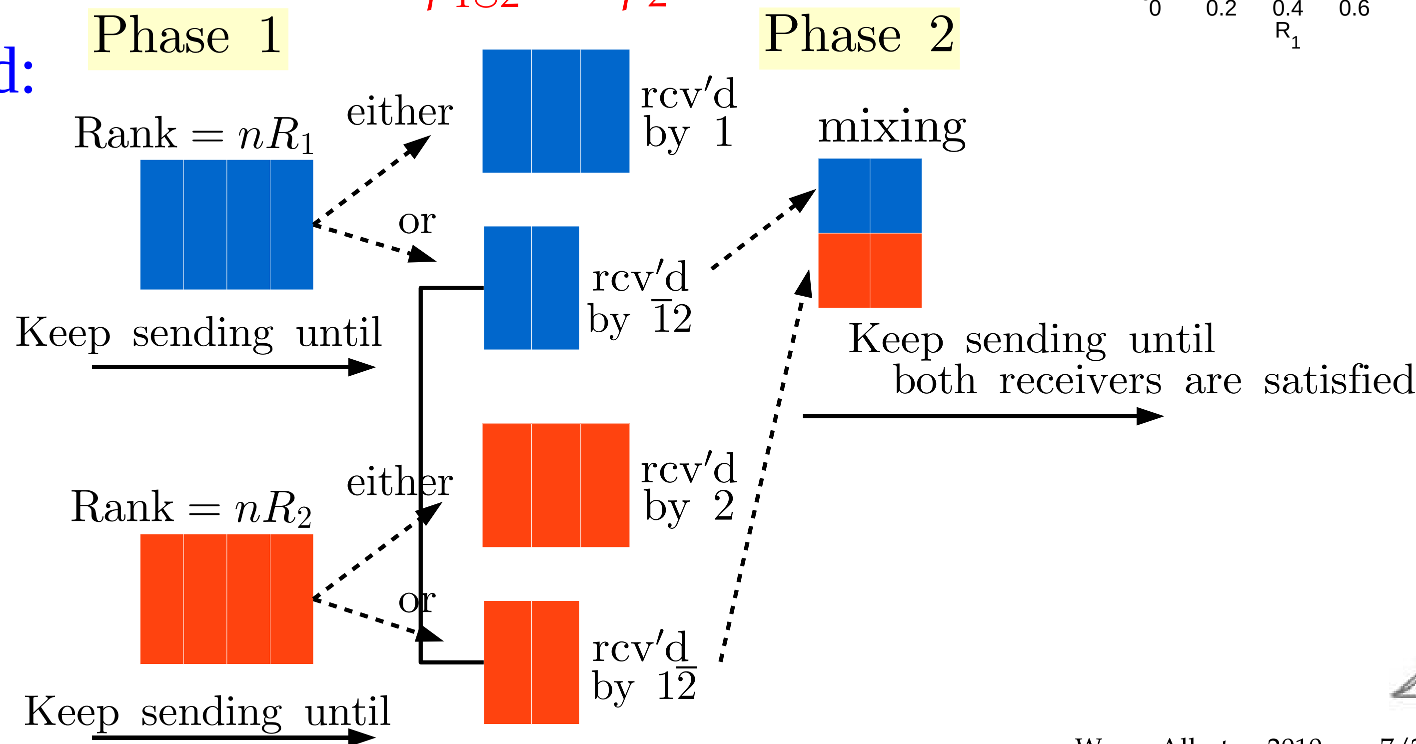
$$\frac{R_1}{p_1} + \frac{R_2}{p_{1\cup 2}} \leq 1$$



$$\frac{R_1}{p_{1\cup 2}} + \frac{R_2}{p_2} \leq 1$$

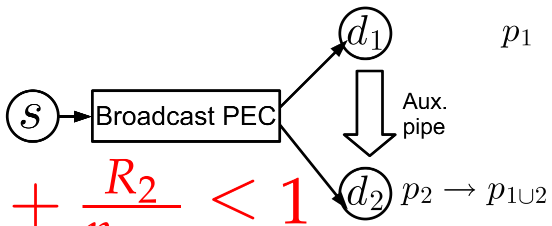


- Inner bound:

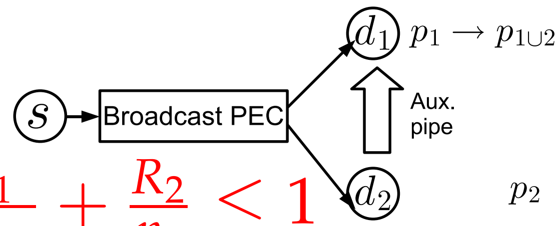


Georgiadis' Proof

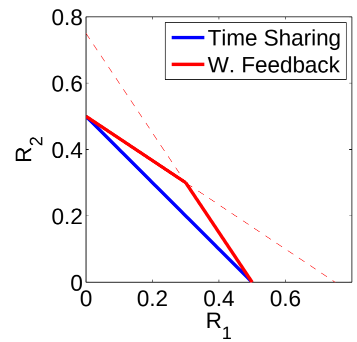
- Outer bound [Ozarow *et al.* 84]: Introduce auxiliary pipes to convert it into *physically degraded channels*, for which feedback does not increase the capacity [El Gamal 78].



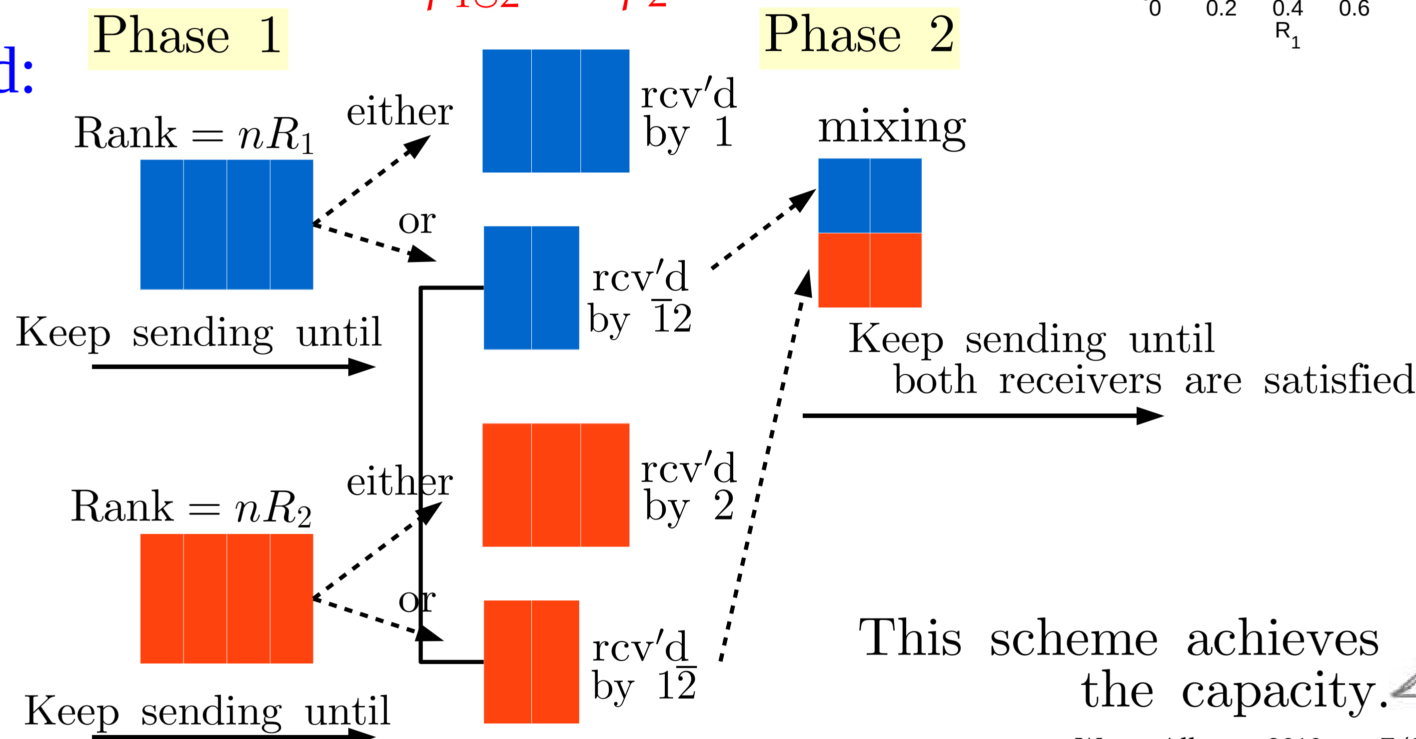
$$\frac{R_1}{p_1} + \frac{R_2}{p_{1\cup 2}} \leq 1$$



$$\frac{R_1}{p_{1\cup 2}} + \frac{R_2}{p_2} \leq 1$$

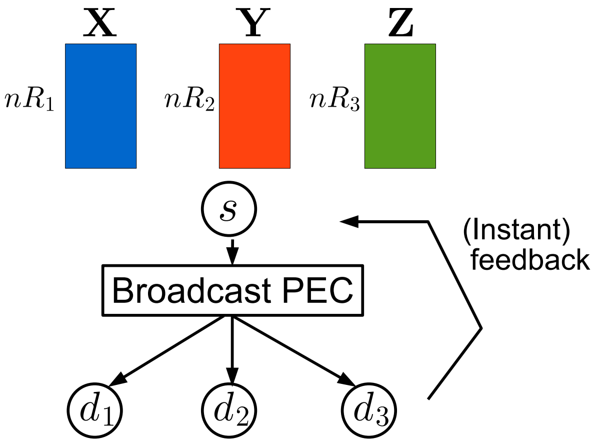


- Inner bound:



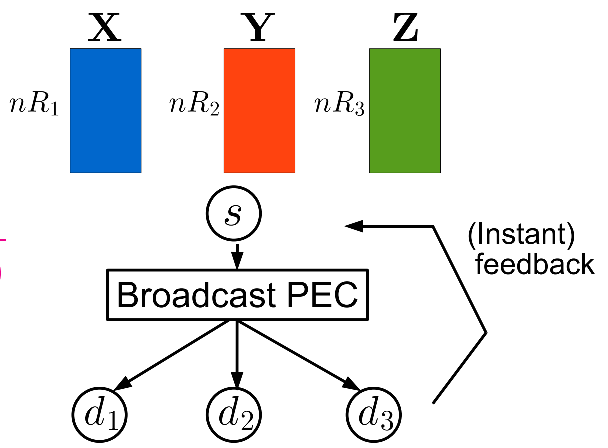
This scheme achieves the capacity.

What if $M \geq 3$?



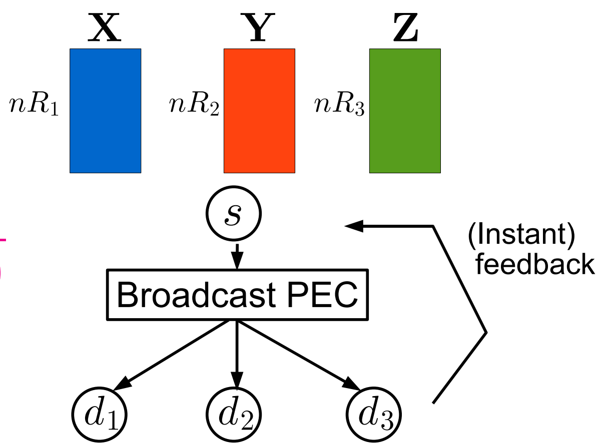
What if $M \geq 3$?

- The CH. parameters become more involved.
  - $M = 2$: p_{12}, p_{12^c}, p_{1^c2}, $p_{1^c2^c}$.
  - $M \geq 3$: the success probability $p_{S\overline{([M]\backslash S)}}$ that a packet is received *by and only by* $d_i \in S$. We have 2^M such parameters.



What if $M \geq 3$?

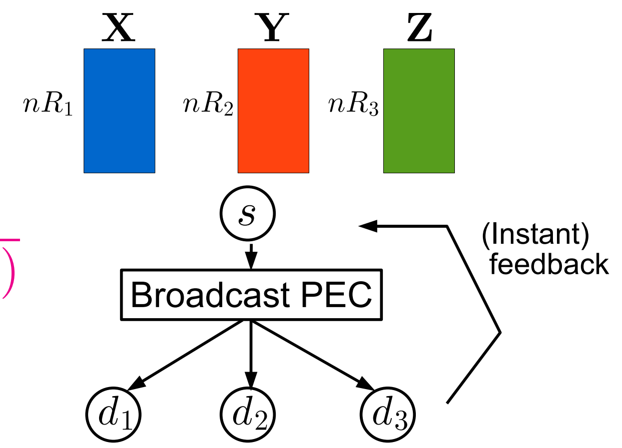
- The CH. parameters become more involved.
  - $M = 2$: p_{12}, p_{12^c}, p_{1^c2}, $p_{1^c2^c}$.
  - $M \geq 3$: the success probability $p_{S\overline{([M]\backslash S)}}$ that a packet is received *by and only by* $d_i \in S$. We have 2^M such parameters.



- Can we also quantify the Shannon capacity for $M \geq 3$?

What if $M \geq 3$?

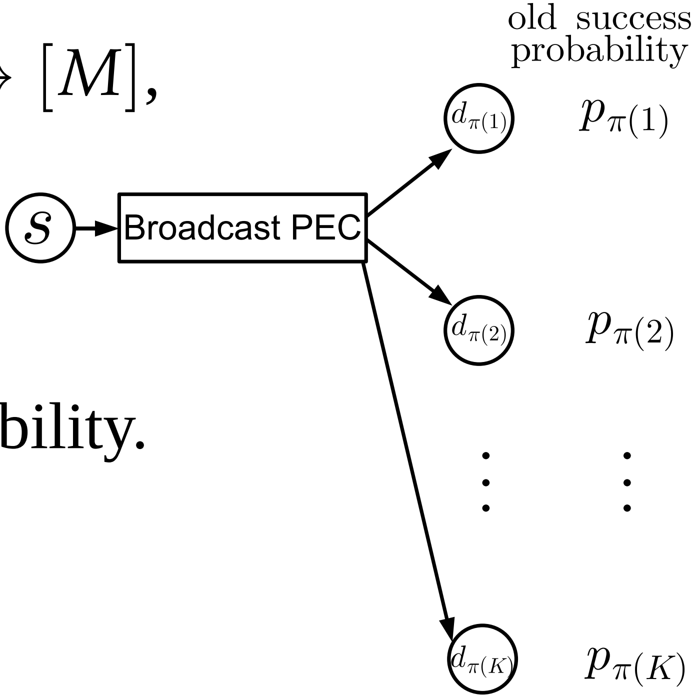
- The CH. parameters become more involved.
  - $M = 2$: $p_{12}, p_{12^c}, p_{1^c2}, p_{1^c2^c}$.
  - $M \geq 3$: the success probability $p_{S\overline{([M]\backslash S)}}$ that a packet is received *by and only by* $d_i \in S$. We have 2^M such parameters.



- Can we also quantify the Shannon capacity for $M \geq 3$?
  - Generalization of the outer bound is straightforward.
  - Generalization of the inner bound is more difficult.

Simple Cap. Outer Bound

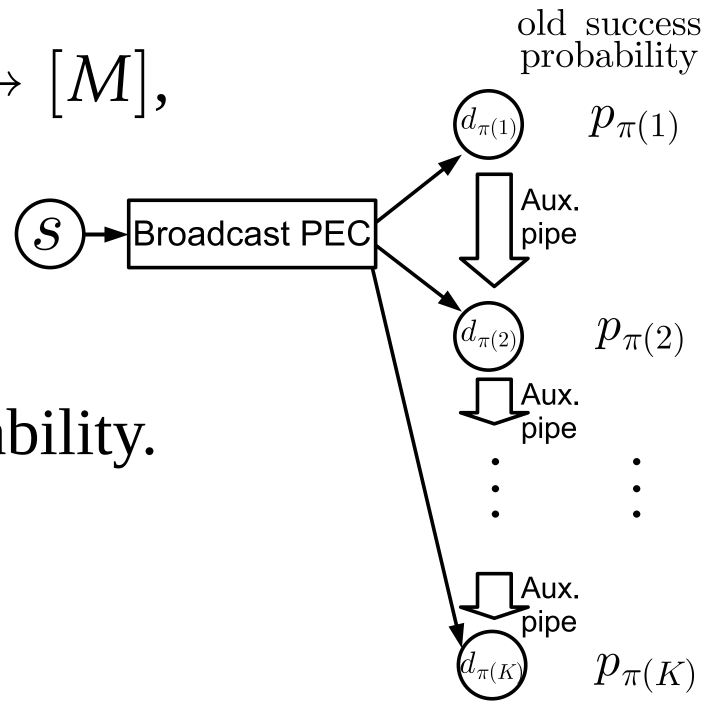
- For any permutation $\pi : [M] \mapsto [M]$,



- p_k: The marginal success probability.

Simple Cap. Outer Bound

- For any permutation $\pi : [M] \mapsto [M]$,
- p_k: The marginal success probability.

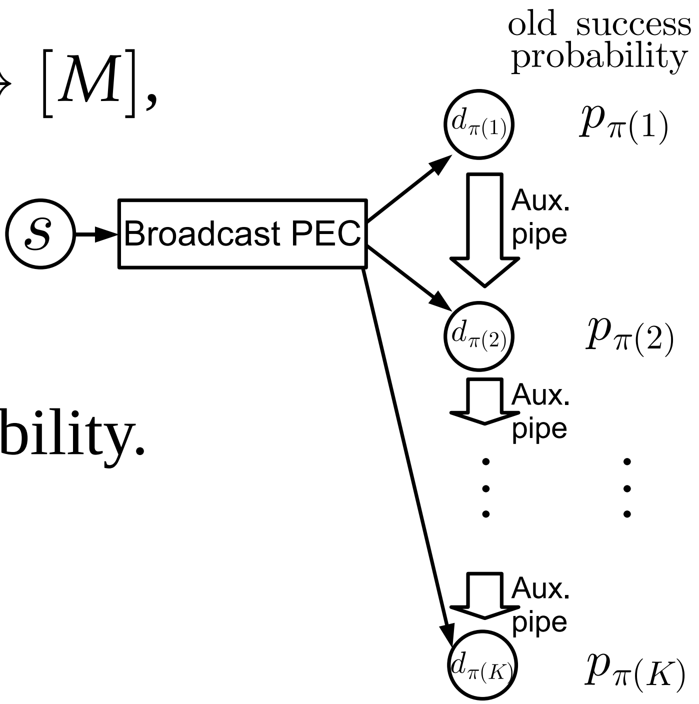


Simple Cap. Outer Bound

- For any permutation $\pi : [M] \mapsto [M]$,

$$\begin{aligned} &\text{Cap. of the original CH with feedback} \\ \prec\ &\text{Cap. of the new CH with feedback} \\ =\ &\text{Cap. of the new CH without feedback} \end{aligned}$$

- p_k: The marginal success probability.

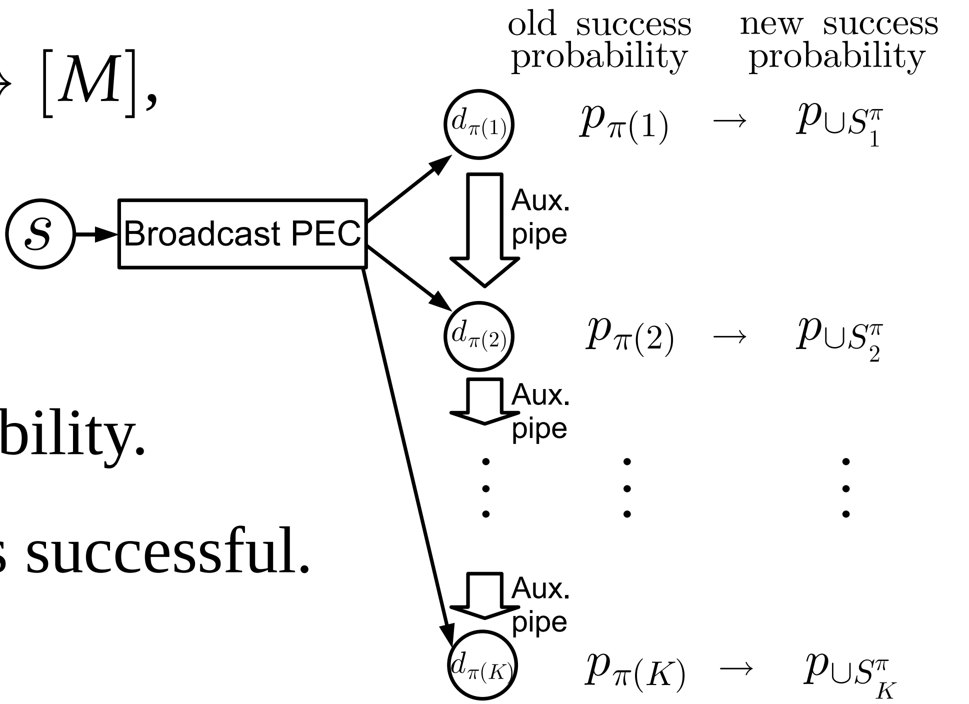


Simple Cap. Outer Bound

- For any permutation $\pi : [M] \mapsto [M]$,

Cap. of the original CH with feedback
$\prec$ Cap. of the new CH with feedback
$=$ Cap. of the new CH without feedback

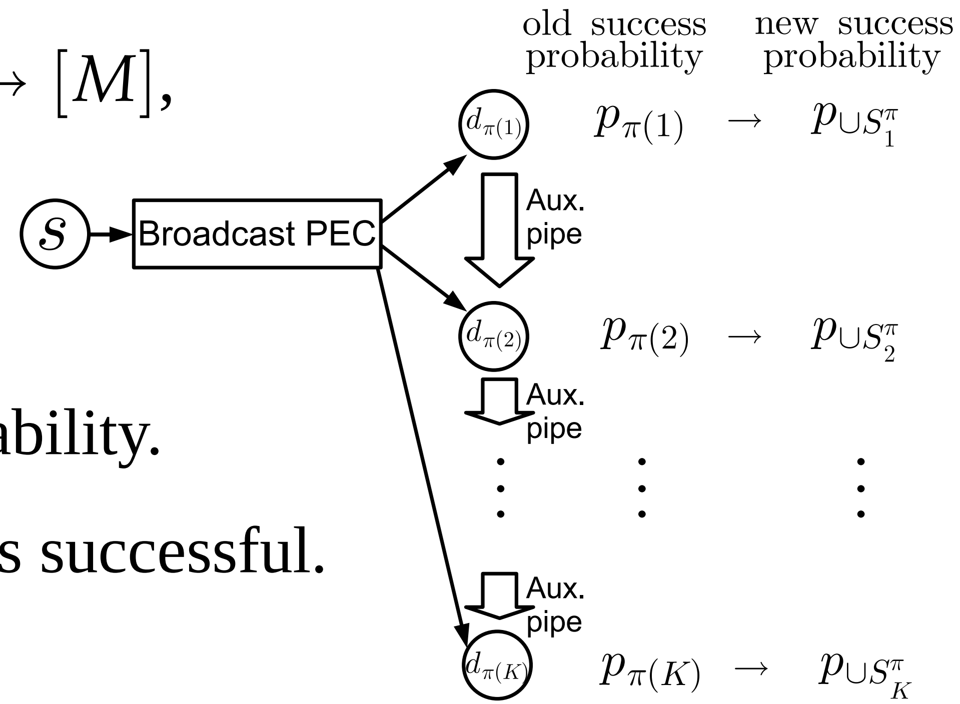
- p_k: The marginal success probability.
- $p_{\cup S}$: Prob. at least one $d_i \in S$ is successful.
  $S_k^\pi = \{\pi(j) : \forall j = 1, \cdots, k\}$.



Simple Cap. Outer Bound

- For any permutation $\pi : [M] \mapsto [M]$,

Cap. of the original CH with feedback
$\prec$ Cap. of the new CH with feedback
$=$ Cap. of the new CH without feedback



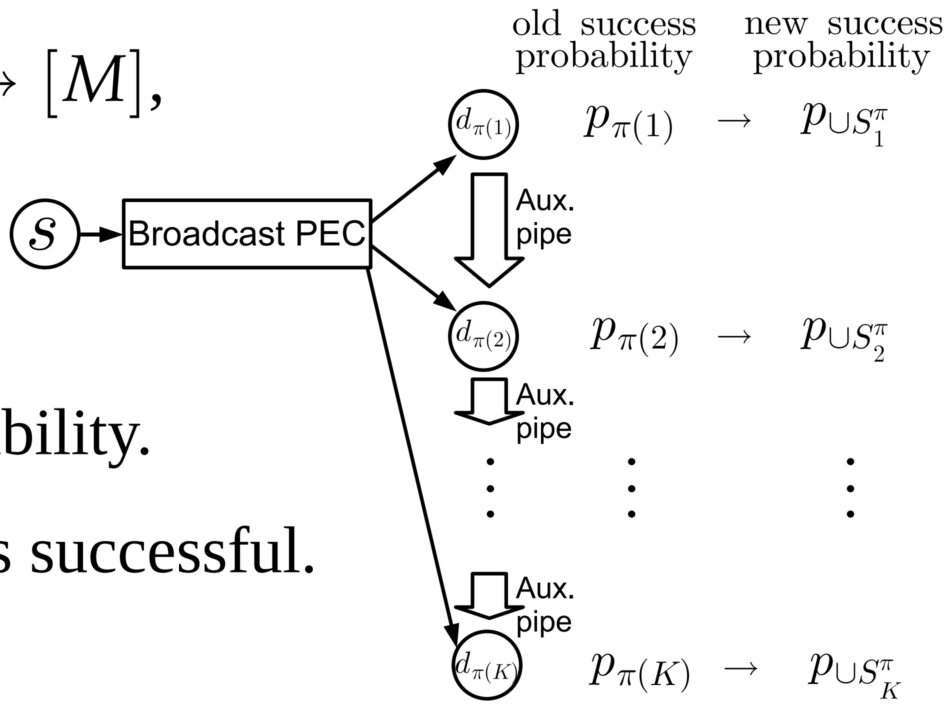
- p_k: The marginal success probability.
- $p_{\cup S}$: Prob. at least one $d_i \in S$ is successful. $S_k^\pi = \{\pi(j) : \forall j = 1, \cdots, k\}$.
- For each π, the capacity of the degraded channel is

$$\sum_{k=1}^{M} \frac{R_{\pi(k)}}{p_{\cup S_k^\pi}} \leq 1.$$

Simple Cap. Outer Bound

- For any permutation $\pi : [M] \mapsto [M]$,

$$\begin{aligned} &\text{Cap. of the original CH with feedback} \\ \prec\ &\text{Cap. of the new CH with feedback} \\ =\ &\text{Cap. of the new CH without feedback} \end{aligned}$$



- p_k: The marginal success probability.
- $p_{\cup S}$: Prob. at least one $d_i \in S$ is successful.
  $S_k^\pi = \{\pi(j) : \forall j = 1, \cdots, k\}$.
- For each π, the capacity of the degraded channel is

$$\sum_{k=1}^{M} \frac{R_{\pi(k)}}{p_{\cup S_k^\pi}} \leq 1.$$

- A capacity outer bound is thus $\forall \pi, \quad \sum_{k=1}^{M} \frac{R_{\pi(k)}}{p_{\cup S_k^\pi}} \leq 1.$

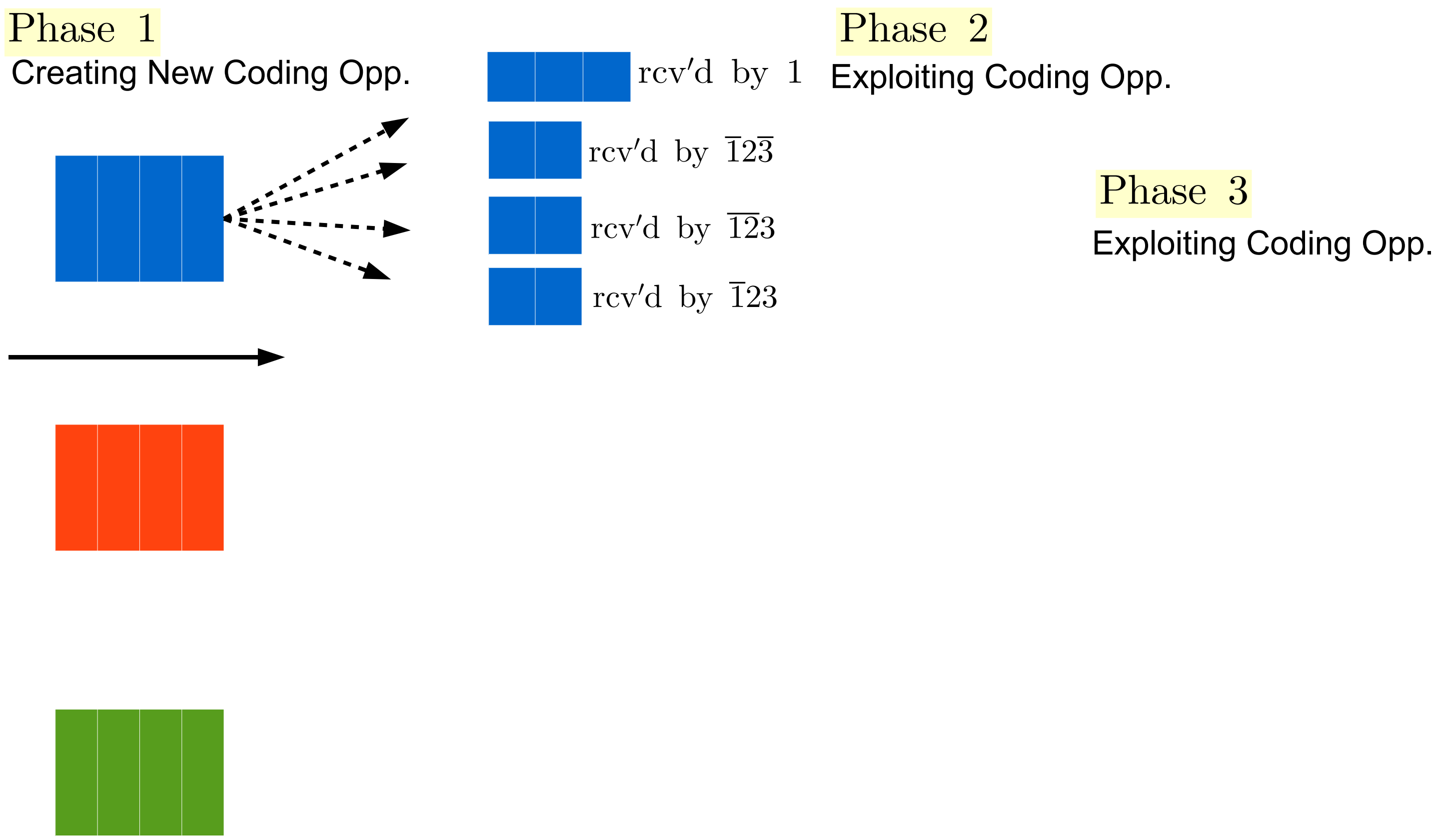
Cap. Inner Bound?

How to achieve the outer bound: $\forall \pi, \sum_{k=1}^{M} \frac{R_{\pi(k)}}{p_{\cup S_k^\pi}} \leq 1$

Cap. Inner Bound?

How to achieve the outer bound: $\forall \pi, \ \sum_{k=1}^{M} \frac{R_{\pi(k)}}{p_{\cup S_k^{\pi}}} \leq 1$

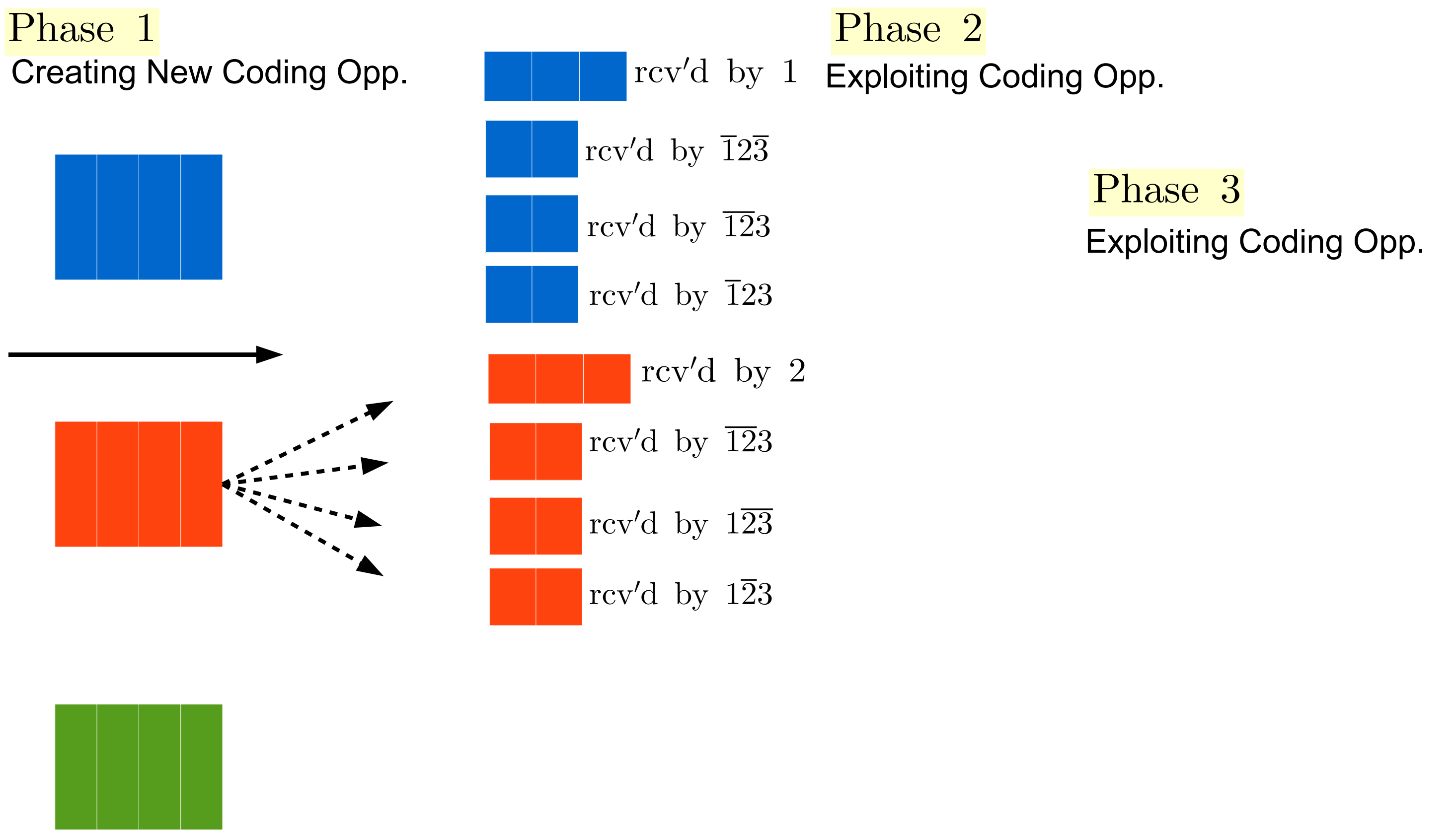
First try was by [Larsson *et al.* 06], an M-phase approach.



Cap. Inner Bound?

How to achieve the outer bound: $\forall \pi, \sum_{k=1}^{M} \frac{R_{\pi(k)}}{p_{\cup S_k^{\pi}}} \leq 1$

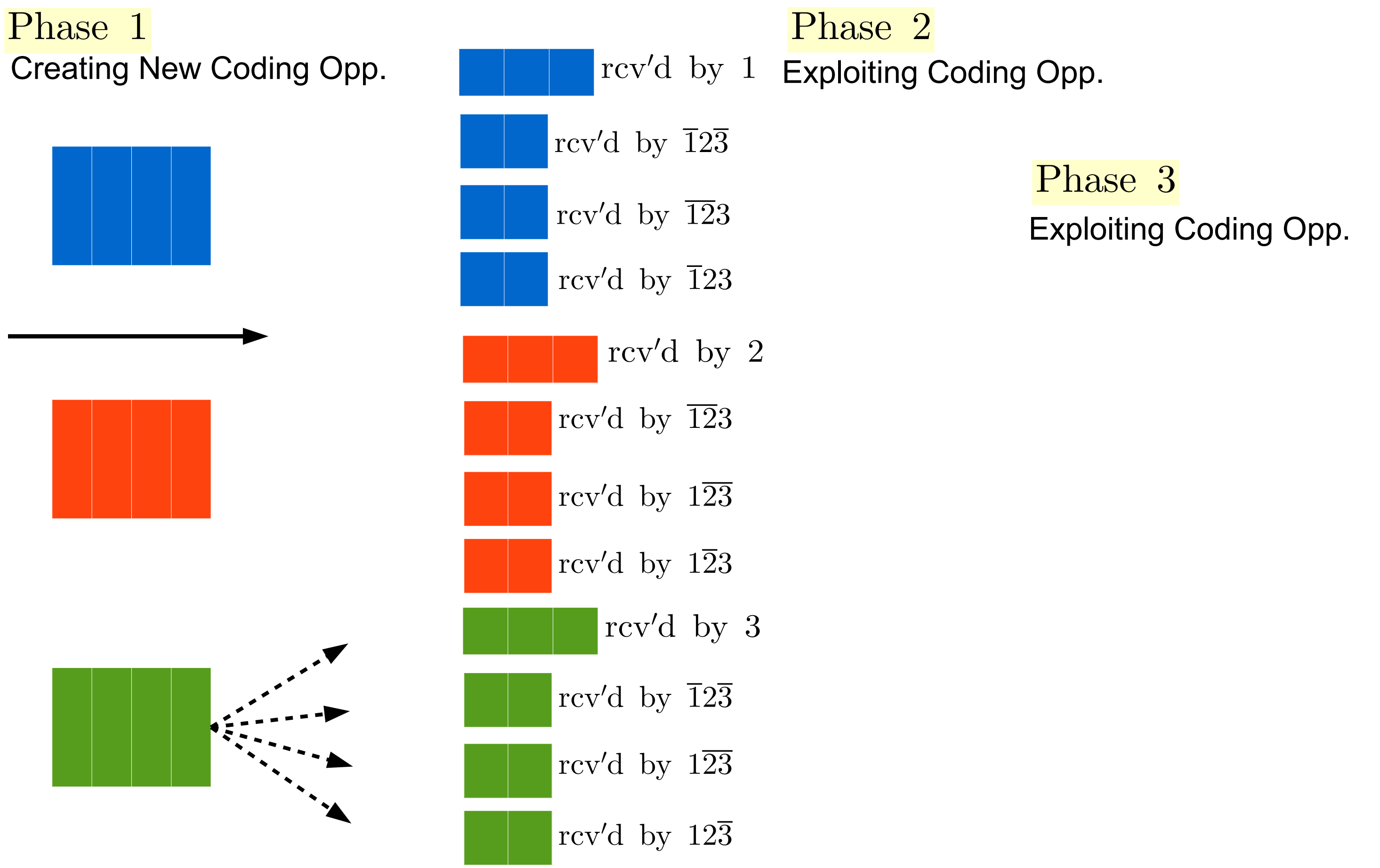
First try was by [Larsson *et al.* 06], an M-phase approach.



Cap. Inner Bound?

How to achieve the outer bound: $\forall \pi, \sum_{k=1}^{M} \frac{R_{\pi(k)}}{p_{\cup S_k^{\pi}}} \leq 1$

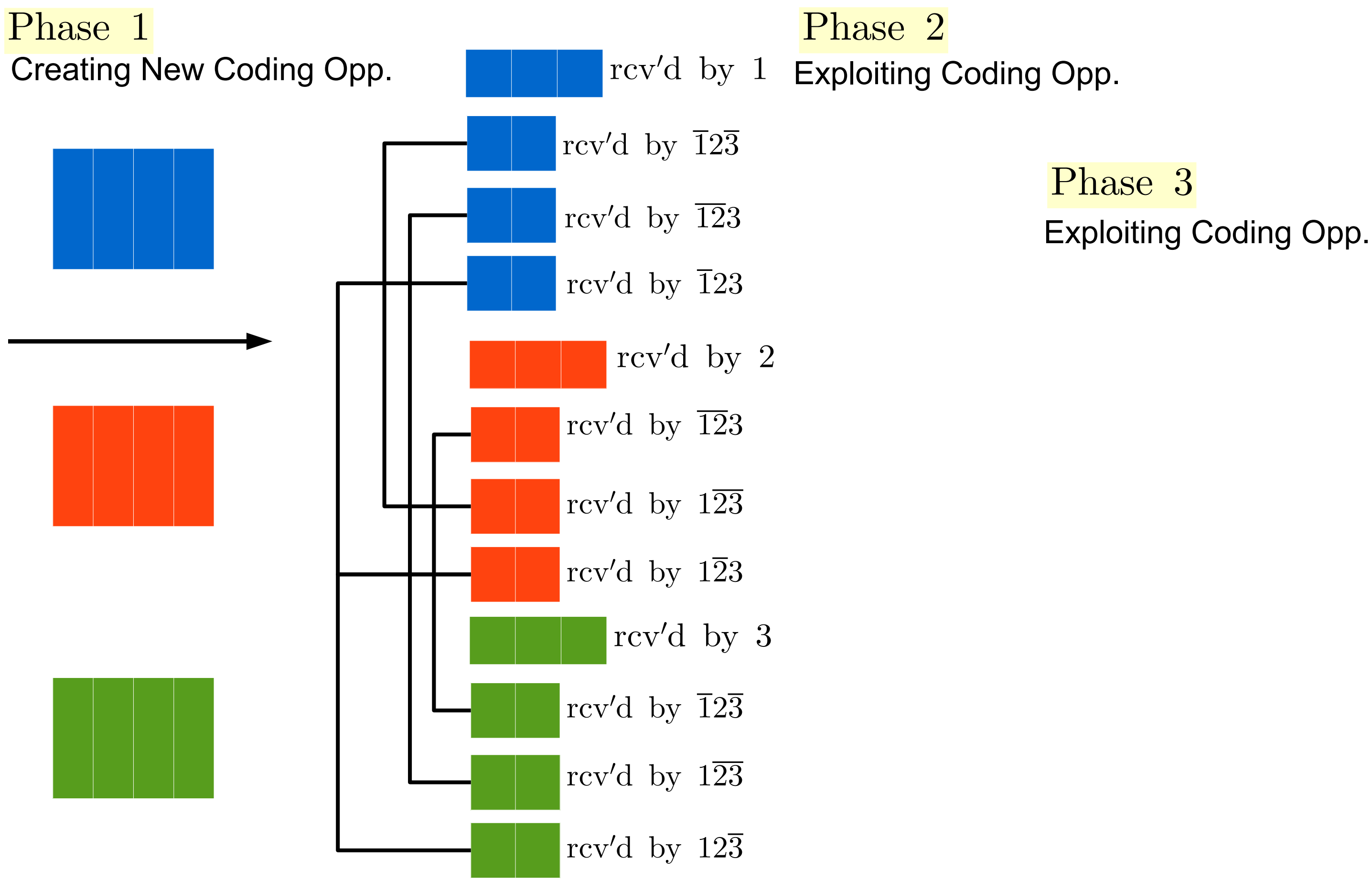
First try was by [Larsson *et al.* 06], an M-phase approach.



Cap. Inner Bound?

How to achieve the outer bound: $\forall \pi, \sum_{k=1}^{M} \frac{R_{\pi(k)}}{p_{\cup S_k^\pi}} \leq 1$

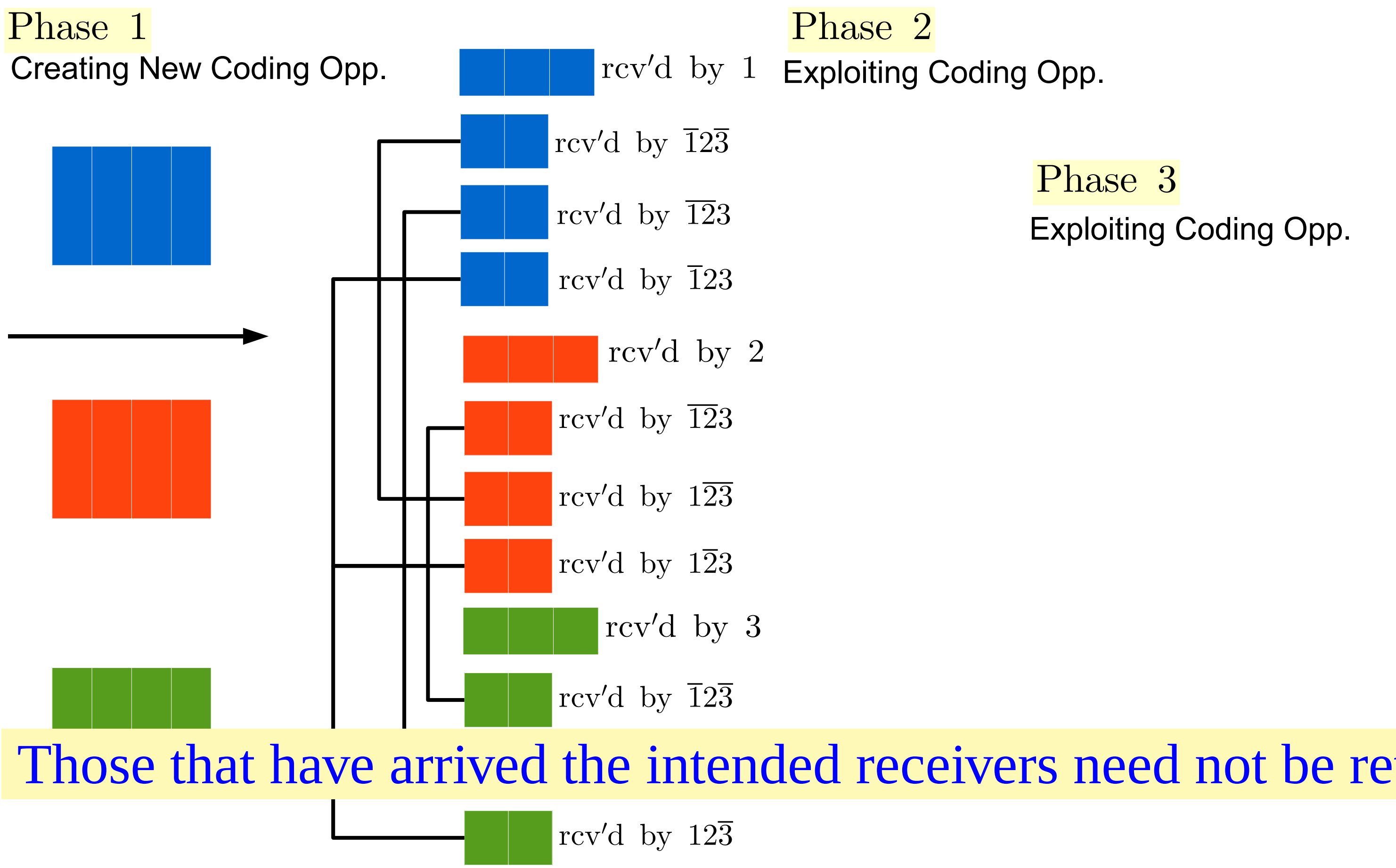
First try was by [Larsson *et al.* 06], an M-phase approach.



Cap. Inner Bound?

How to achieve the outer bound: $\forall \pi, \sum_{k=1}^{M} \frac{R_{\pi(k)}}{p_{\cup S_k^{\pi}}} \leq 1$

First try was by [Larsson *et al.* 06], an M-phase approach.

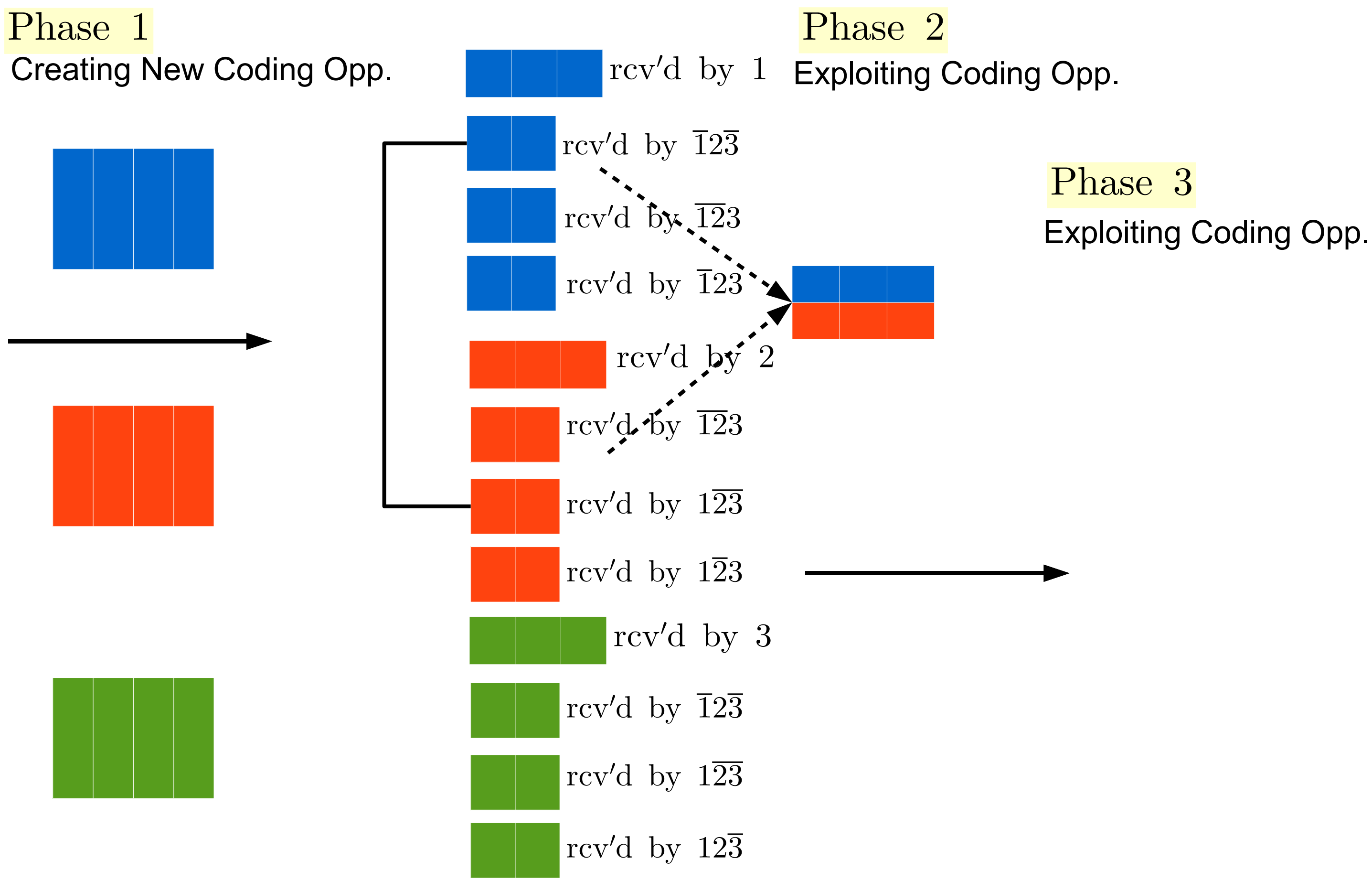


Those that have arrived the intended receivers need not be retransmitted!

Cap. Inner Bound?

How to achieve the outer bound: $\forall \pi, \; \sum_{k=1}^{M} \frac{R_{\pi(k)}}{p_{\cup S_k^{\pi}}} \leq 1$

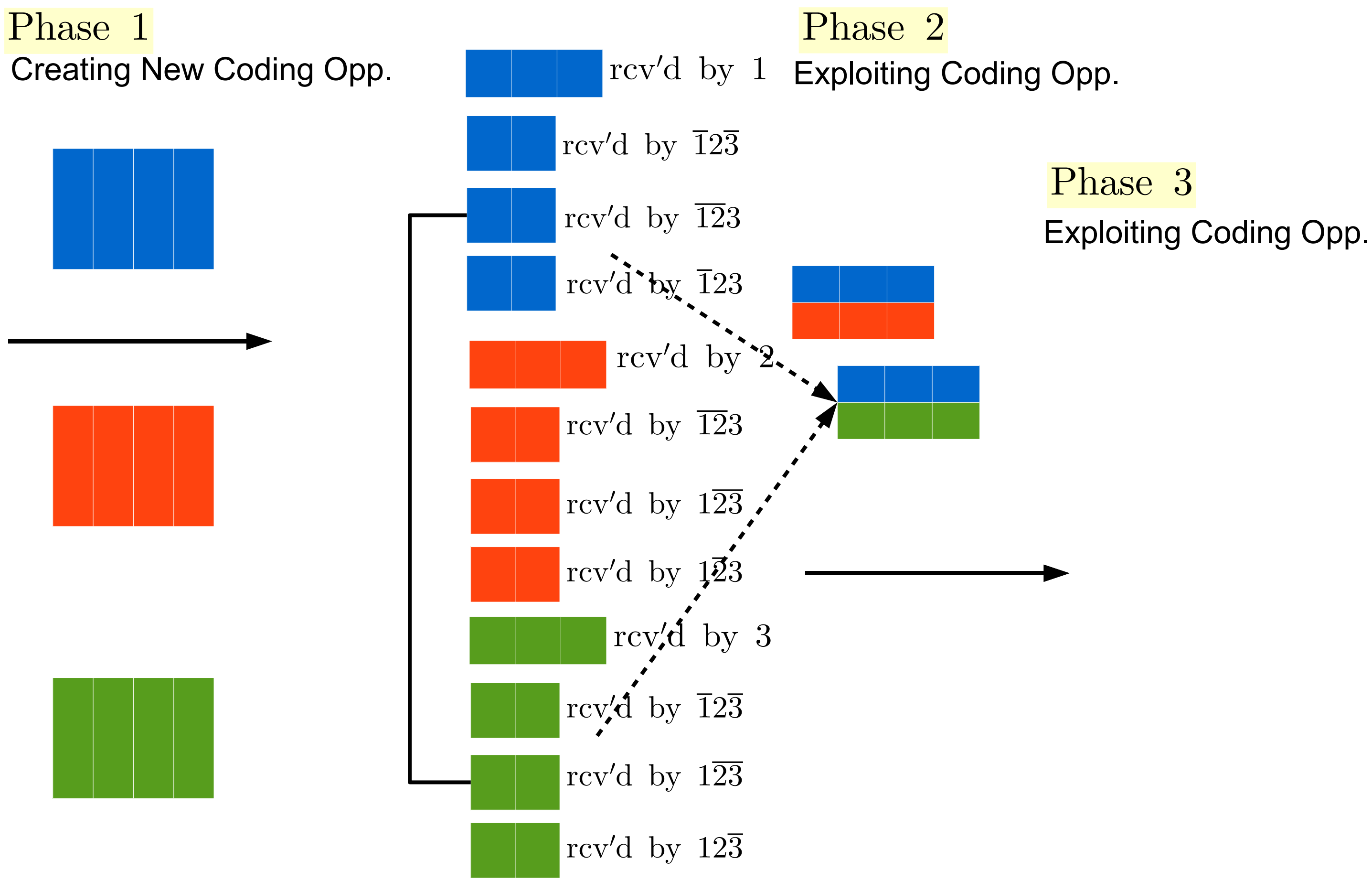
First try was by [Larsson *et al.* 06], an M-phase approach.



Cap. Inner Bound?

How to achieve the outer bound: $\forall \pi, \; \sum_{k=1}^{M} \frac{R_{\pi(k)}}{p_{\cup S_k^\pi}} \leq 1$

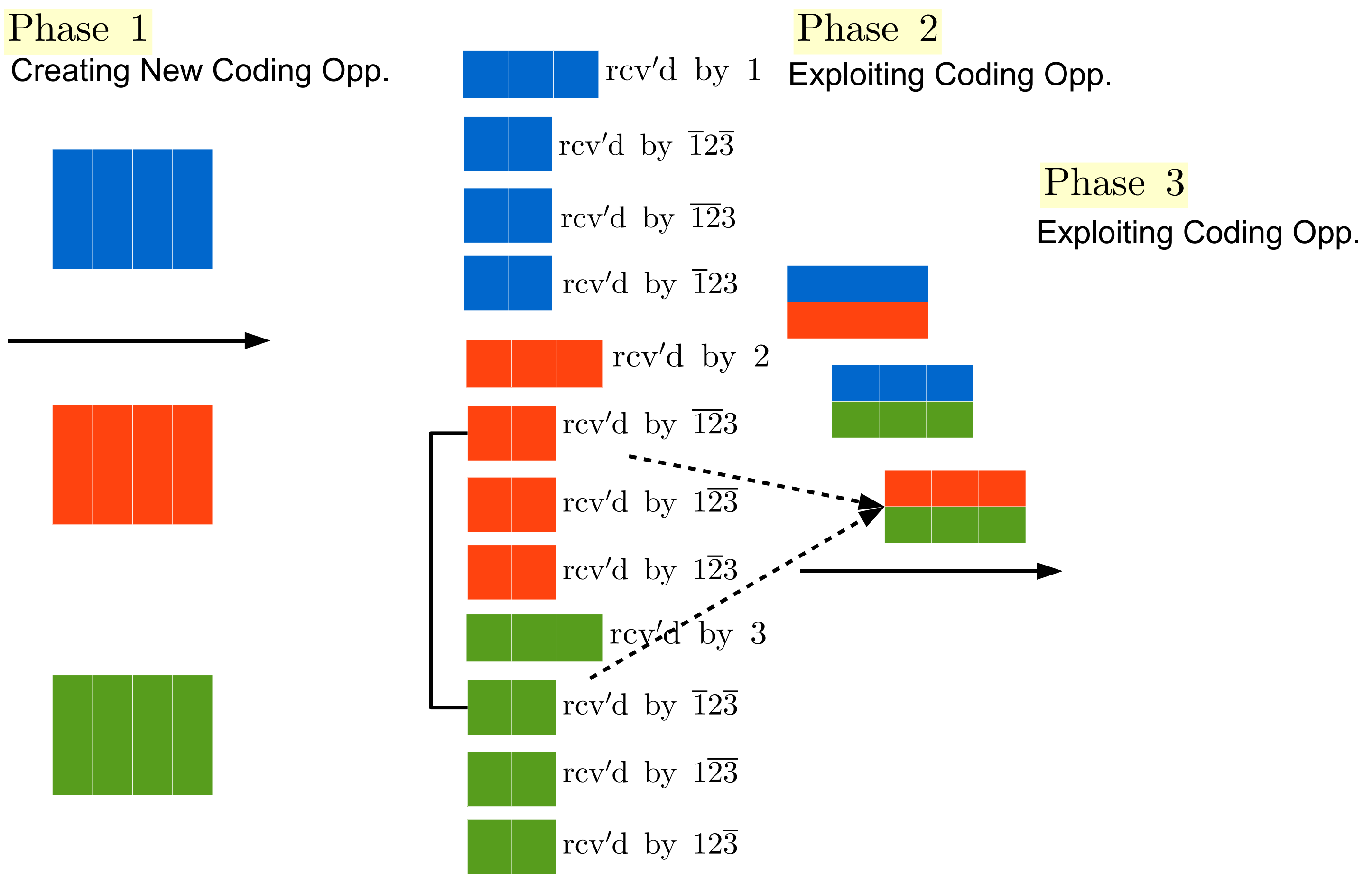
First try was by [Larsson *et al.* 06], an M-phase approach.



Cap. Inner Bound?

How to achieve the outer bound: $\forall \pi, \sum_{k=1}^{M} \frac{R_{\pi(k)}}{p_{\cup S_k^{\pi}}} \leq 1$

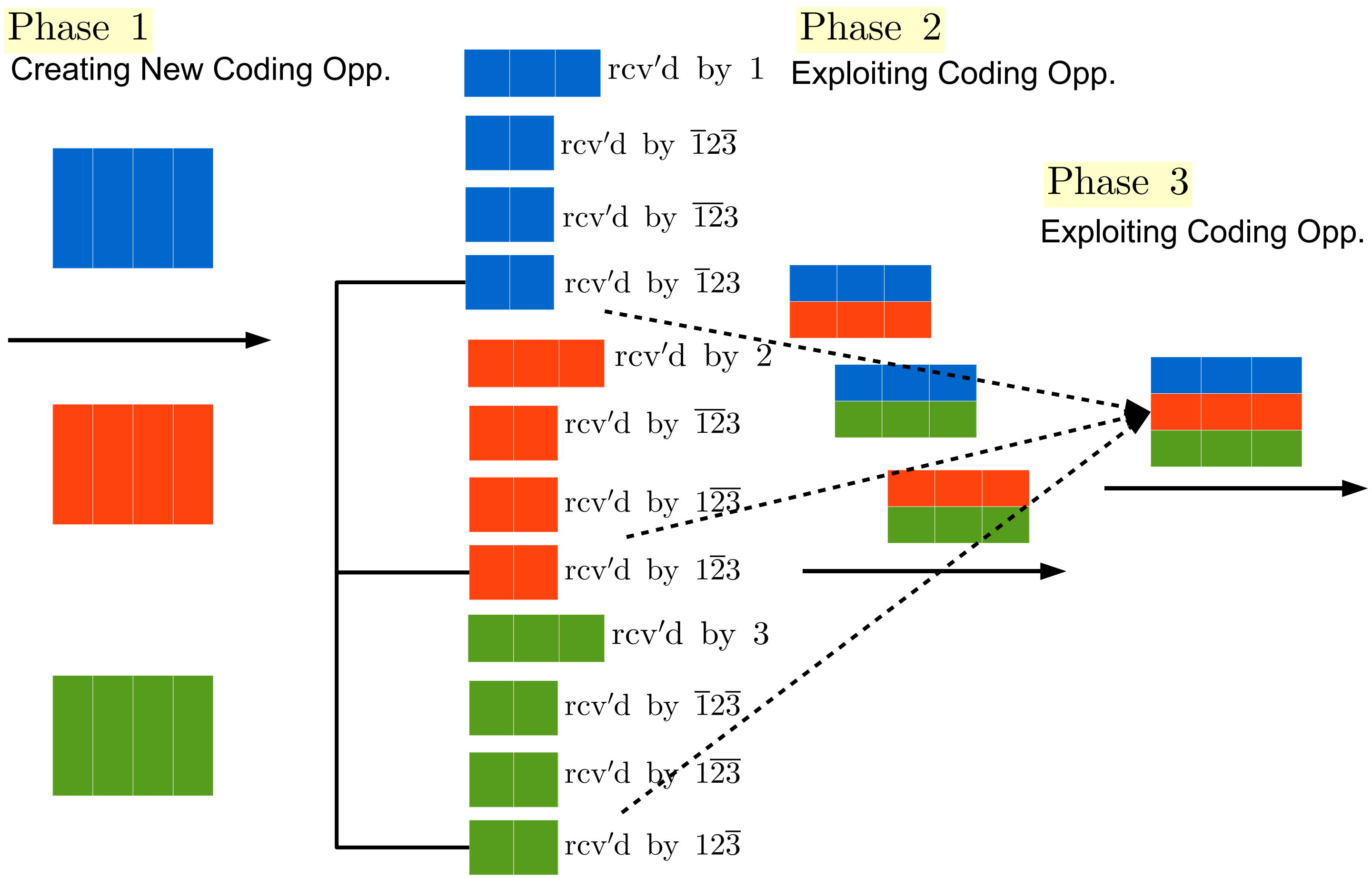
First try was by [Larsson *et al.* 06], an M-phase approach.



Cap. Inner Bound?

How to achieve the outer bound: $\forall \pi, \ \sum_{k=1}^{M} \frac{R_{\pi(k)}}{p_{\cup S_k^{\pi}}} \leq 1$

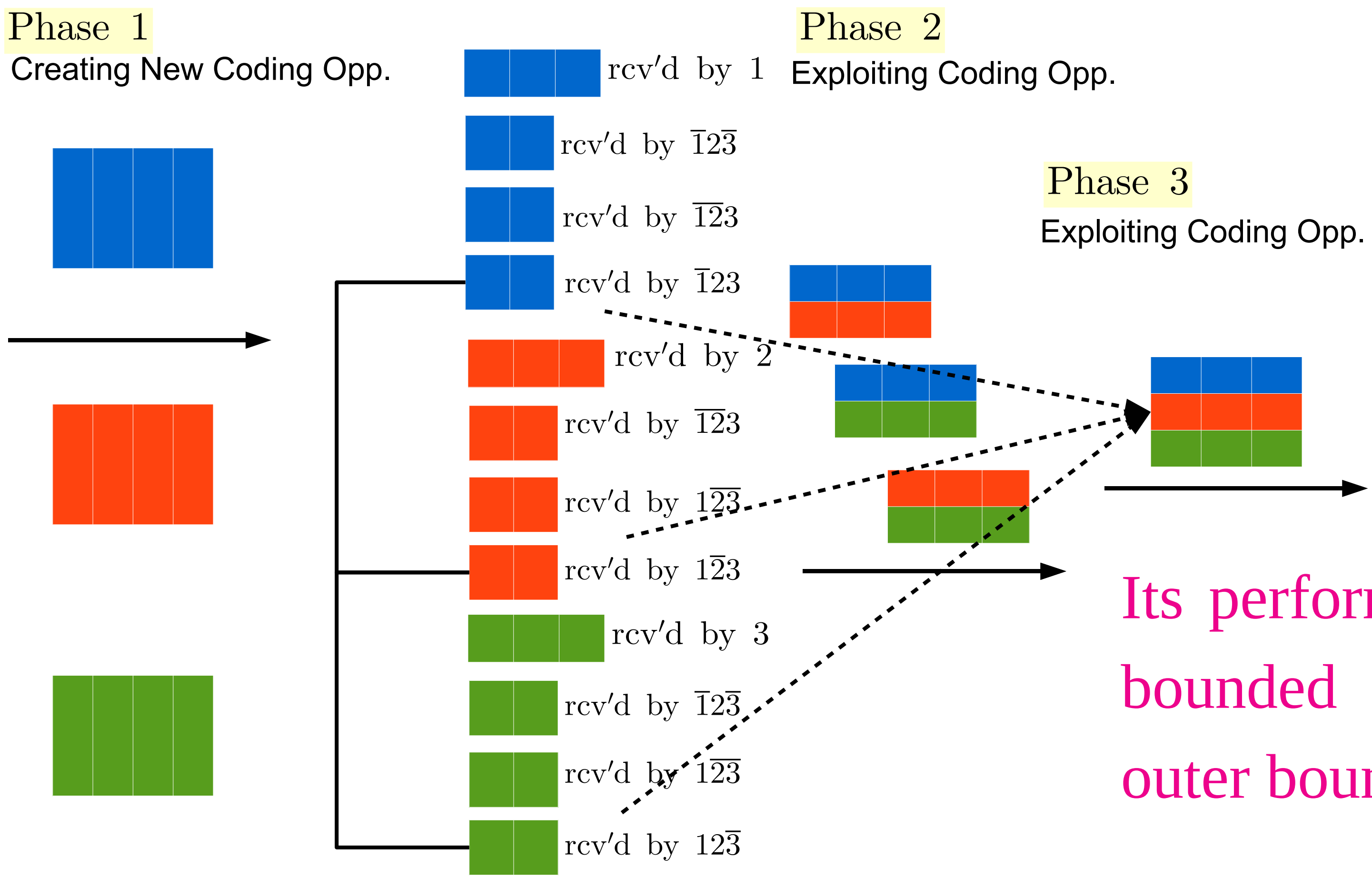
First try was by [Larsson *et al.* 06], an M-phase approach.



Cap. Inner Bound?

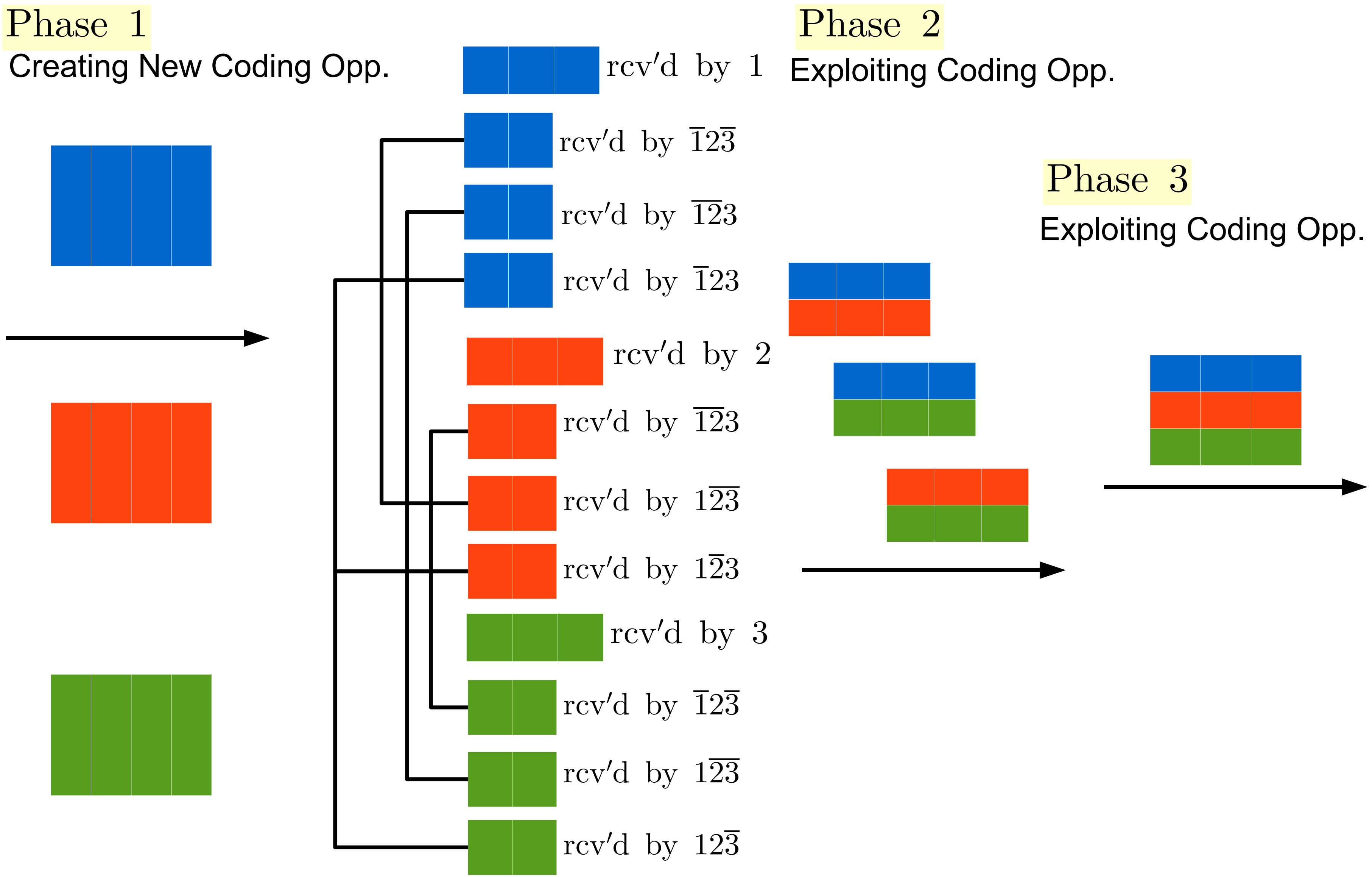
How to achieve the outer bound: $\forall \pi, \sum_{k=1}^{M} \frac{R_{\pi(k)}}{p_{\cup S_k^{\pi}}} \leq 1$

First try was by [Larsson *et al.* 06], an M-phase approach.

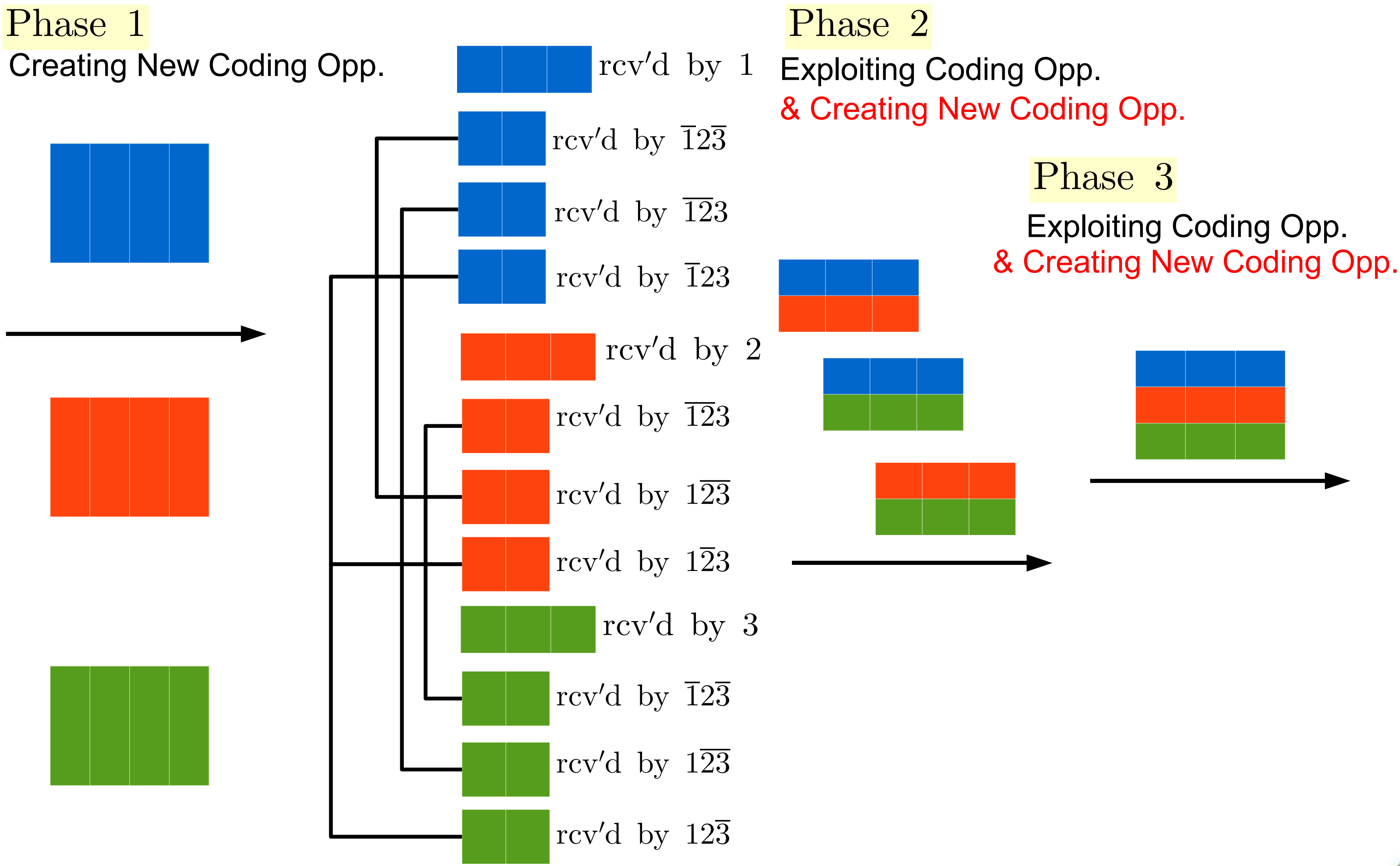


Its performance is strictly bounded away from the outer bound.

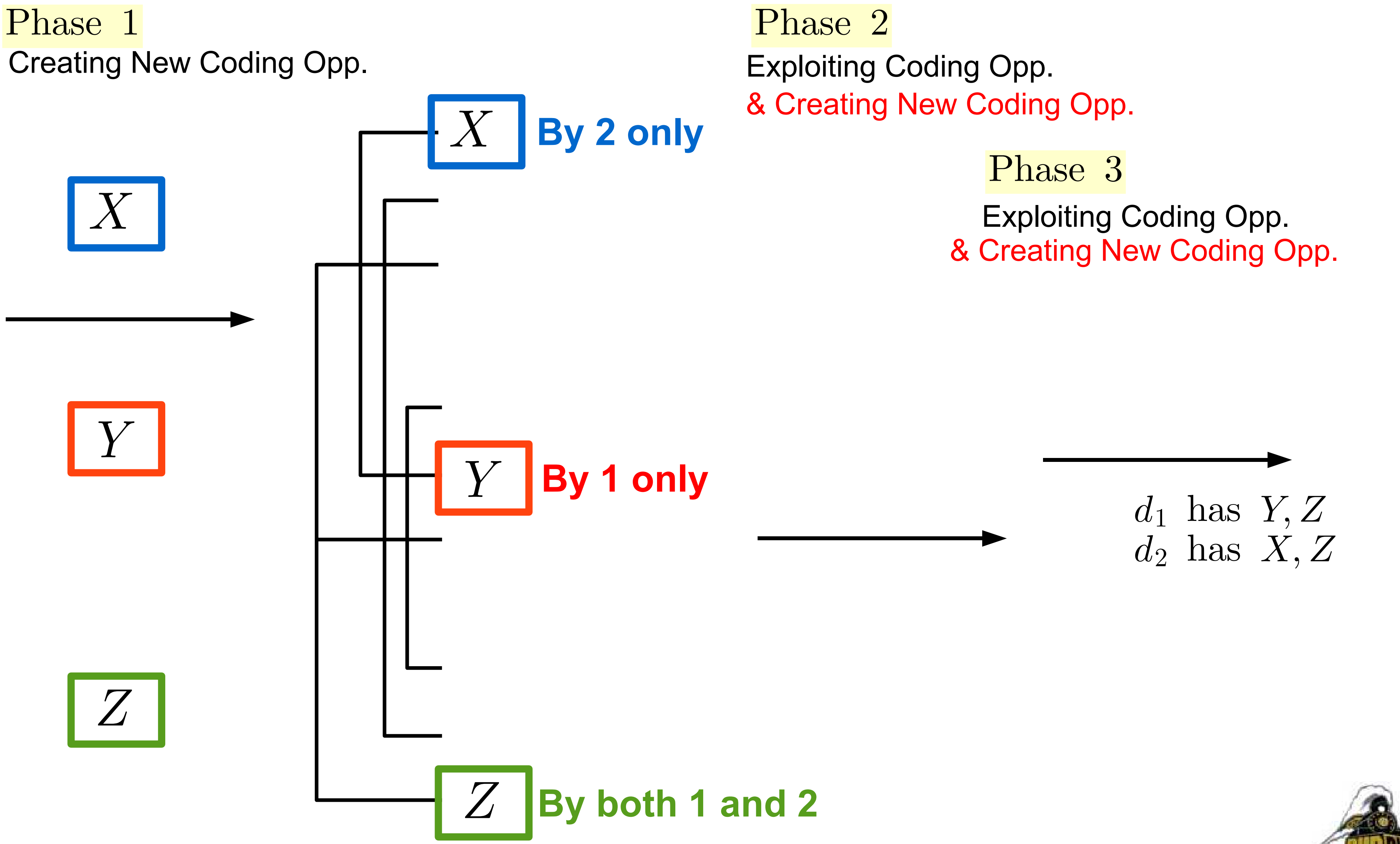
What Went Wrong?



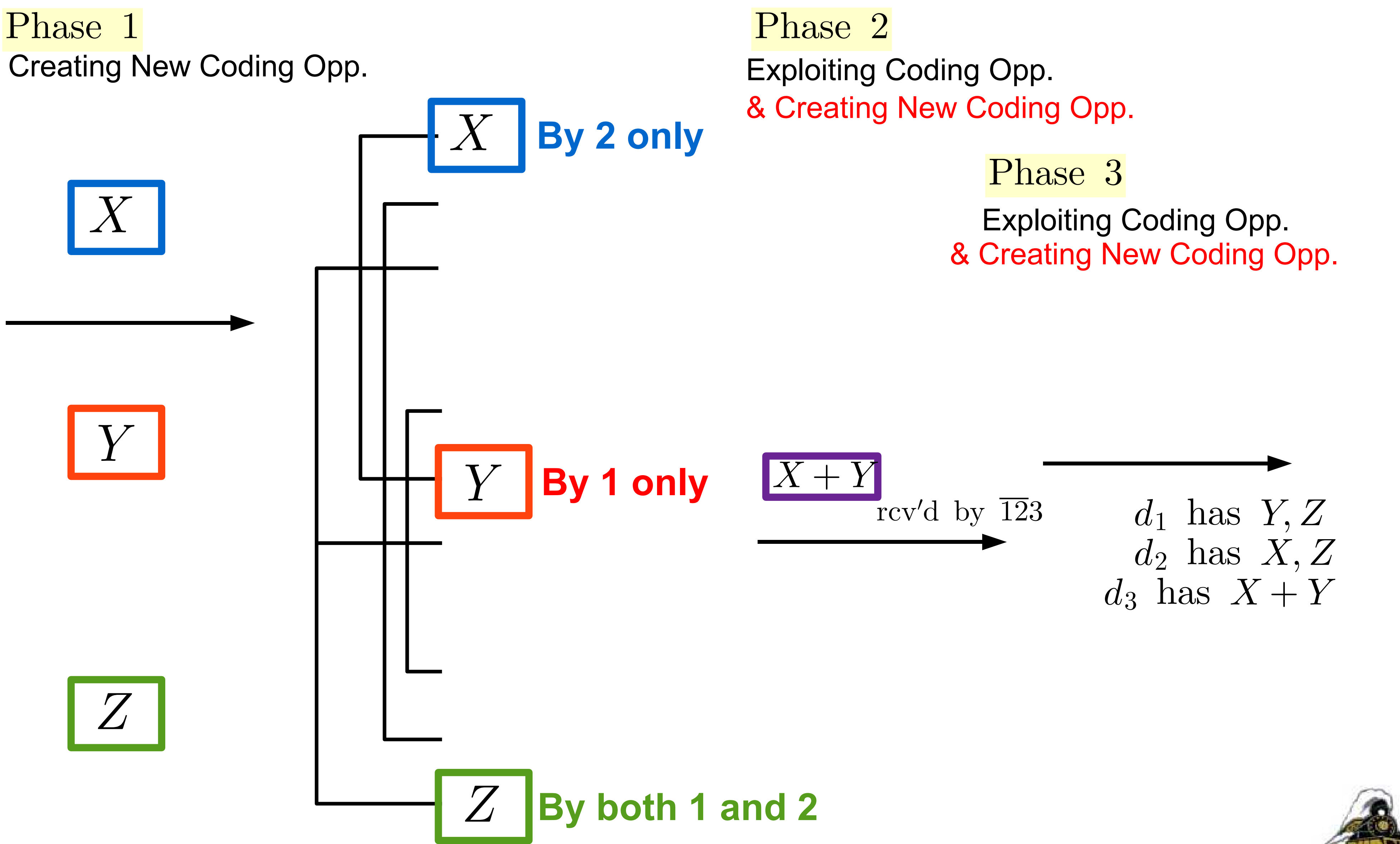
What Went Wrong?



What Went Wrong?



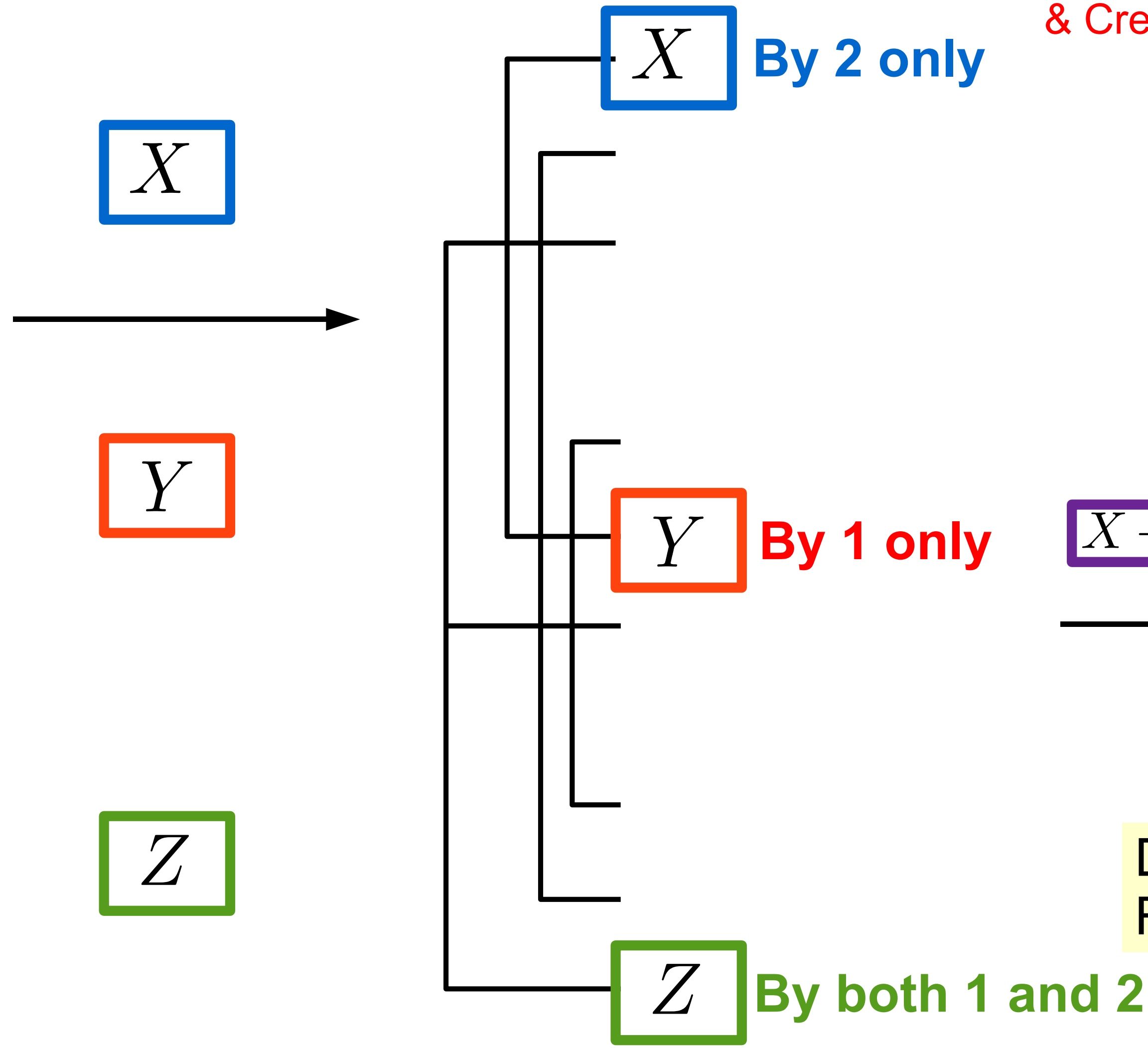
What Went Wrong?



What Went Wrong?

Phase 1

Creating New Coding Opp.



Phase 2

Exploiting Coding Opp.

& Creating New Coding Opp.

Phase 3

Exploiting Coding Opp.

& Creating New Coding Opp.

$X+Y$ rcv'd by $\overline{1}23$

d_1 has Y, Z

d_2 has X, Z

d_3 has $X+Y$

Discard it => Suboptimal

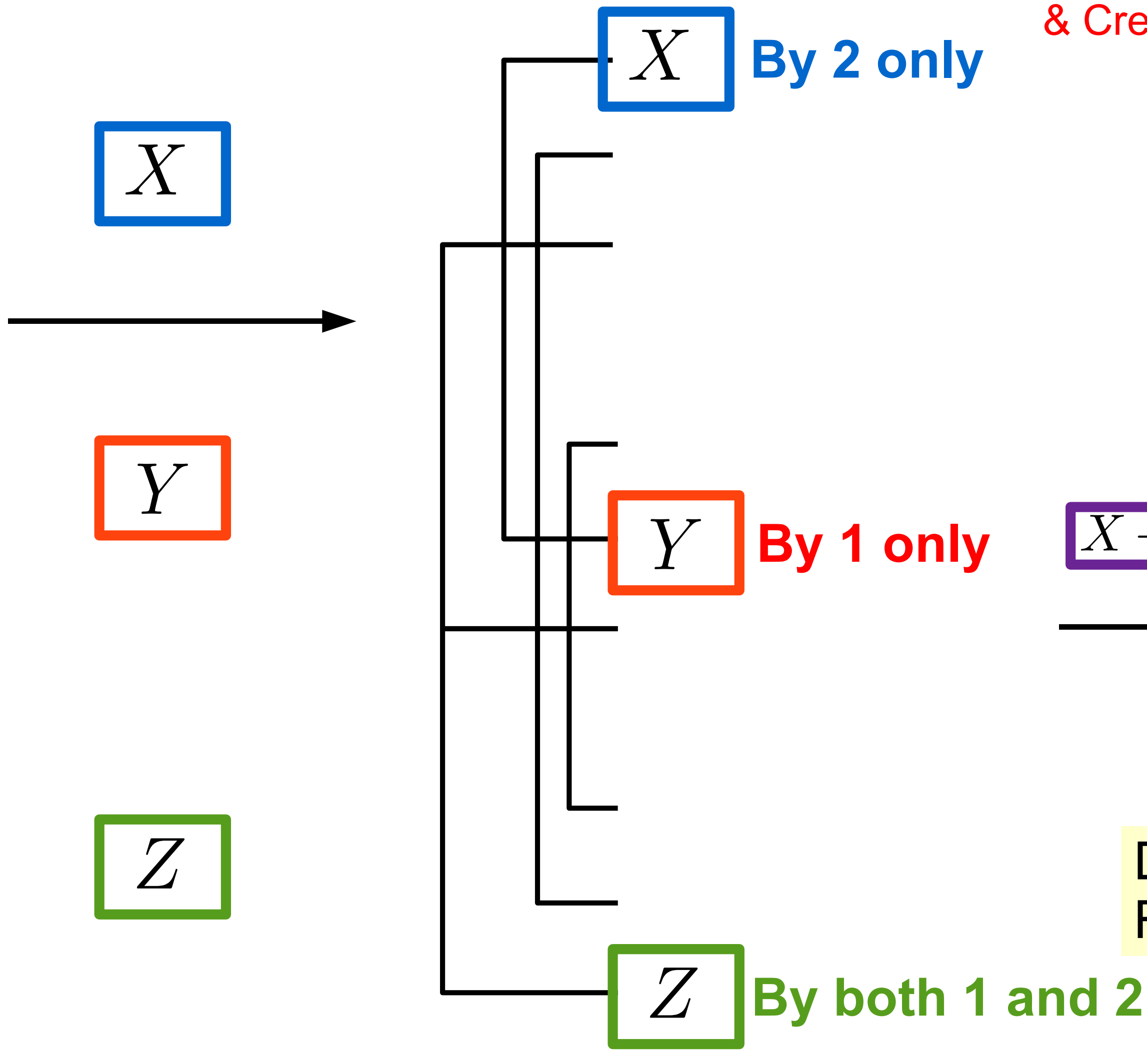
Recoup it for new coding opp.

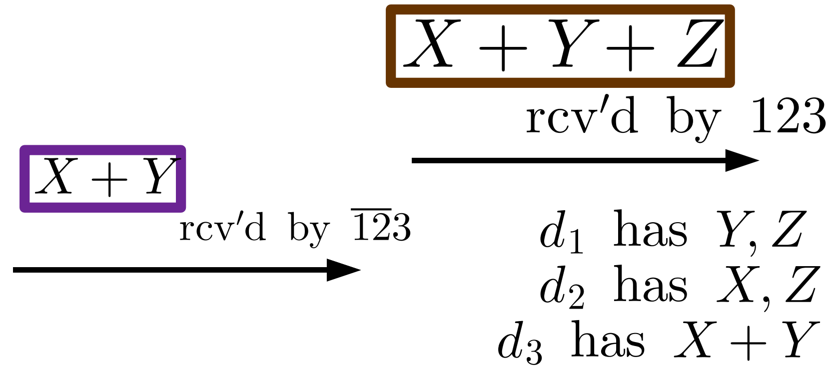
What Went Wrong?

Phase 1
Creating New Coding Opp.

Phase 2
Exploiting Coding Opp.
& Creating New Coding Opp.

Phase 3
Exploiting Coding Opp.
& Creating New Coding Opp.





Discard it => Suboptimal
Recoup it for new coding opp.

New Cap. Inner Bound

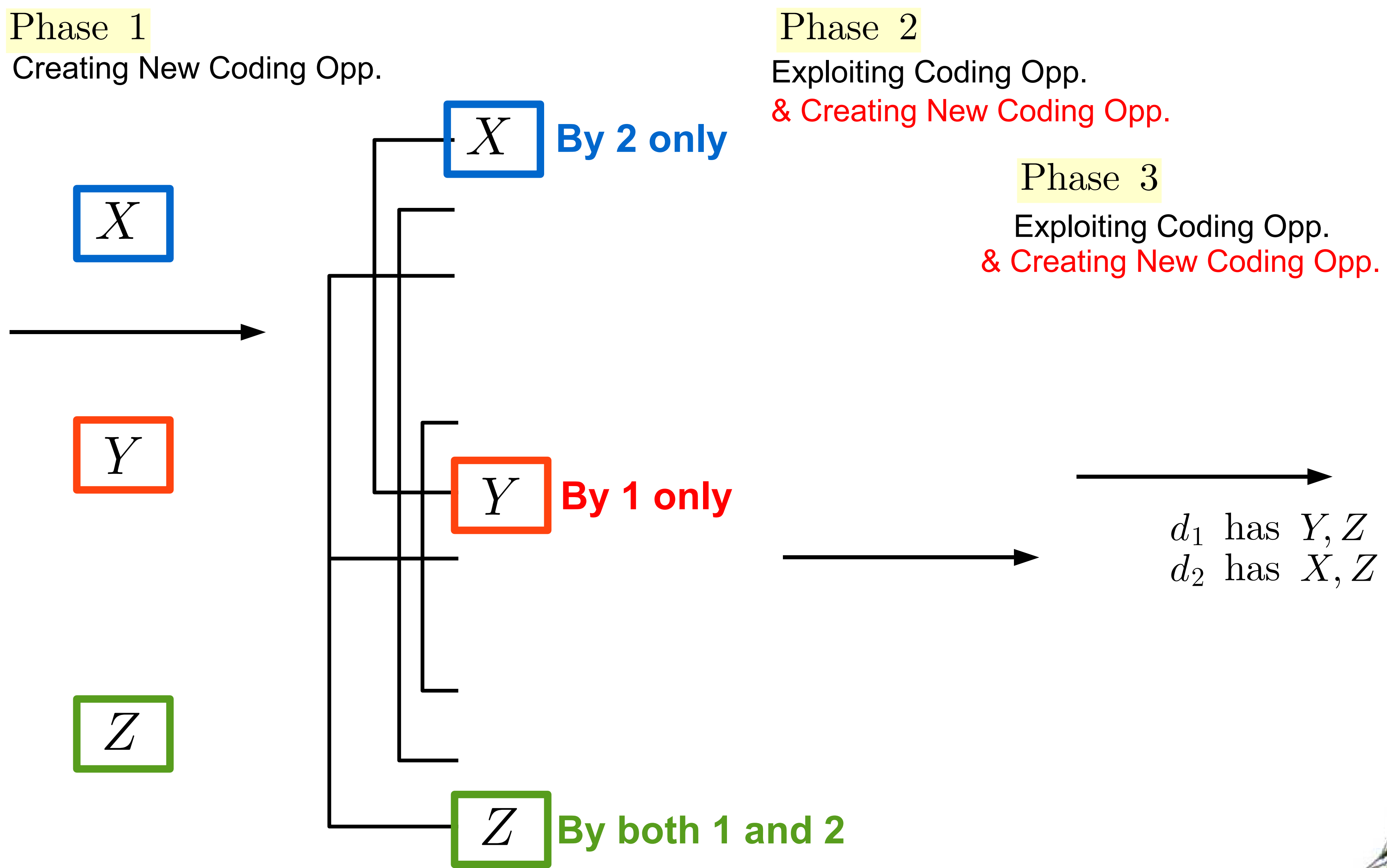
- We need code alignment [W. ISIT10] in order to recoup the overheard coding opportunities during Phases 2 to M.
- That is, the overheard coding vector $[X + Y]$ has to remain aligned in the subsequent mixing stages.
  - $[\alpha(X + Y) + \beta Z]$ serves all three destinations, but
  - $[\alpha X + \beta Y + \gamma Z]$ does not.

New Cap. Inner Bound

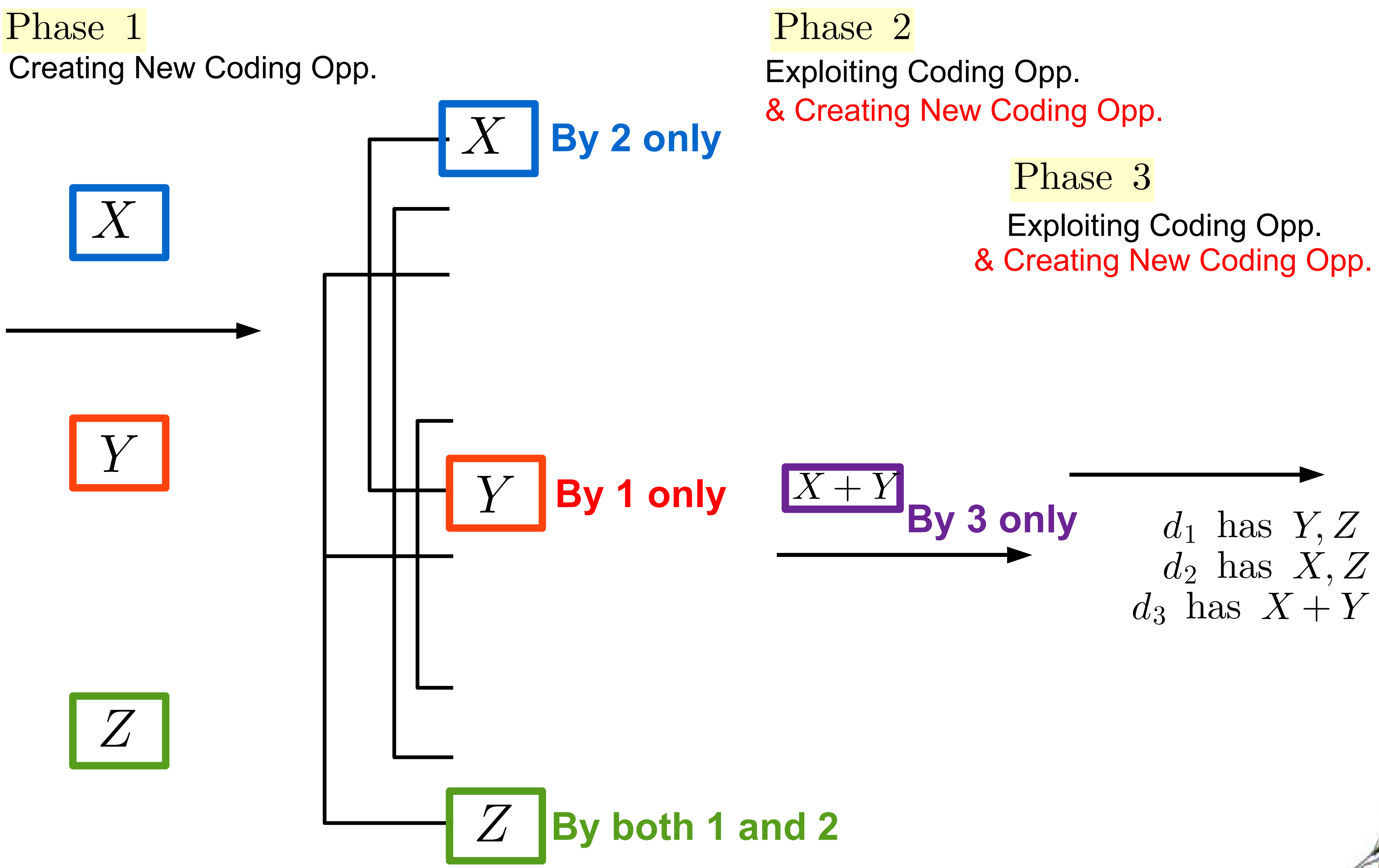
- We need code alignment [W. ISIT10] in order to recoup the overheard coding opportunities during Phases 2 to M.
- That is, the overheard coding vector $[X+Y]$ has to remain aligned in the subsequent mixing stages.
  - $[\alpha(X+Y)+\beta Z]$ serves all three destinations, but
  - $[\alpha X+\beta Y+\gamma Z]$ does not.
- We propose a new Packet Evolution scheme.
- Each information packet (payload) is expanded to

  (payload, overhearing status, representative coding vector)
  - overhearing status keeps evolving to create more coding opportunities.
  - representative coding vector keeps evolving to ensure code alignment.

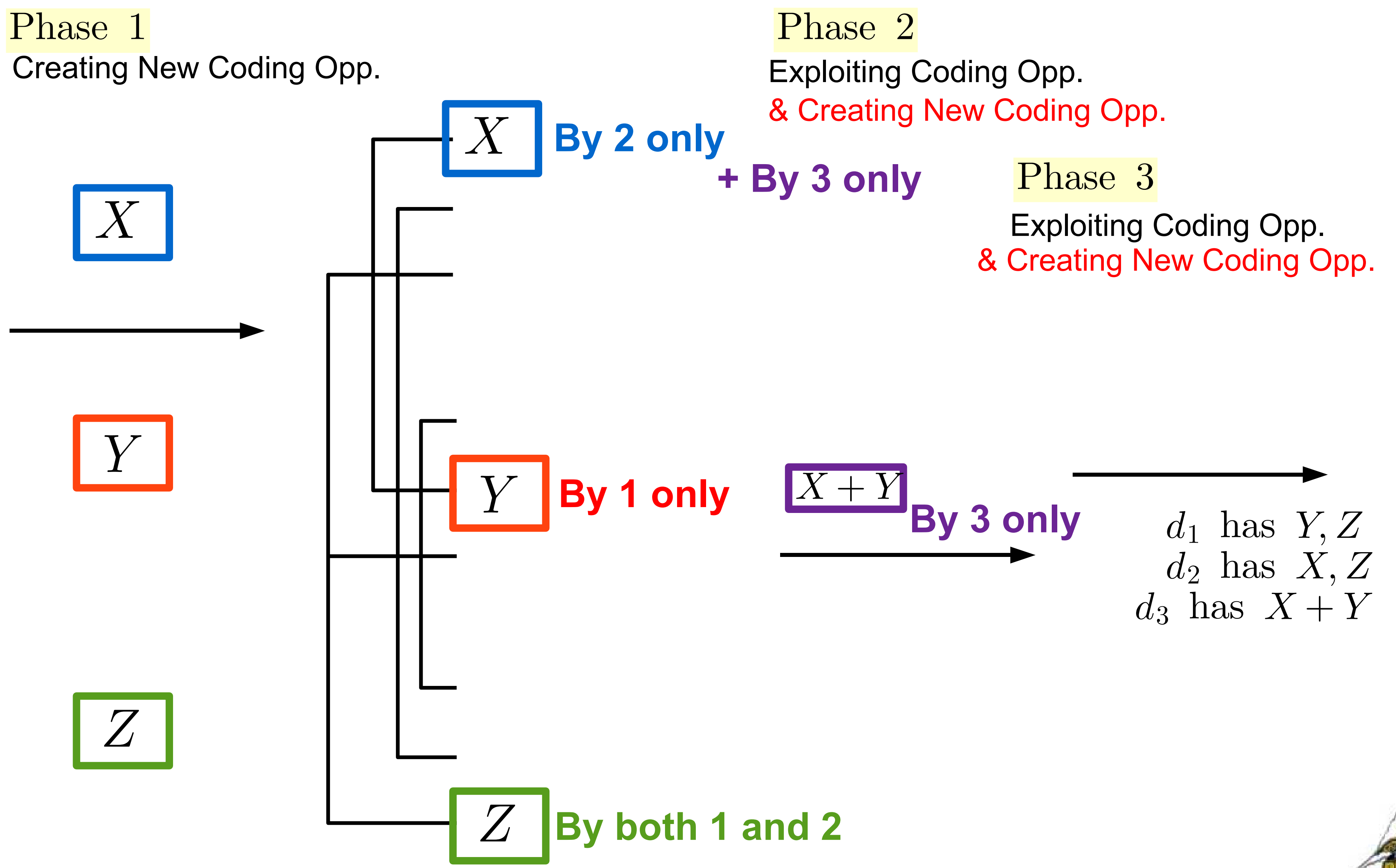
The Packet Evolution Scheme



The Packet Evolution Scheme



The Packet Evolution Scheme

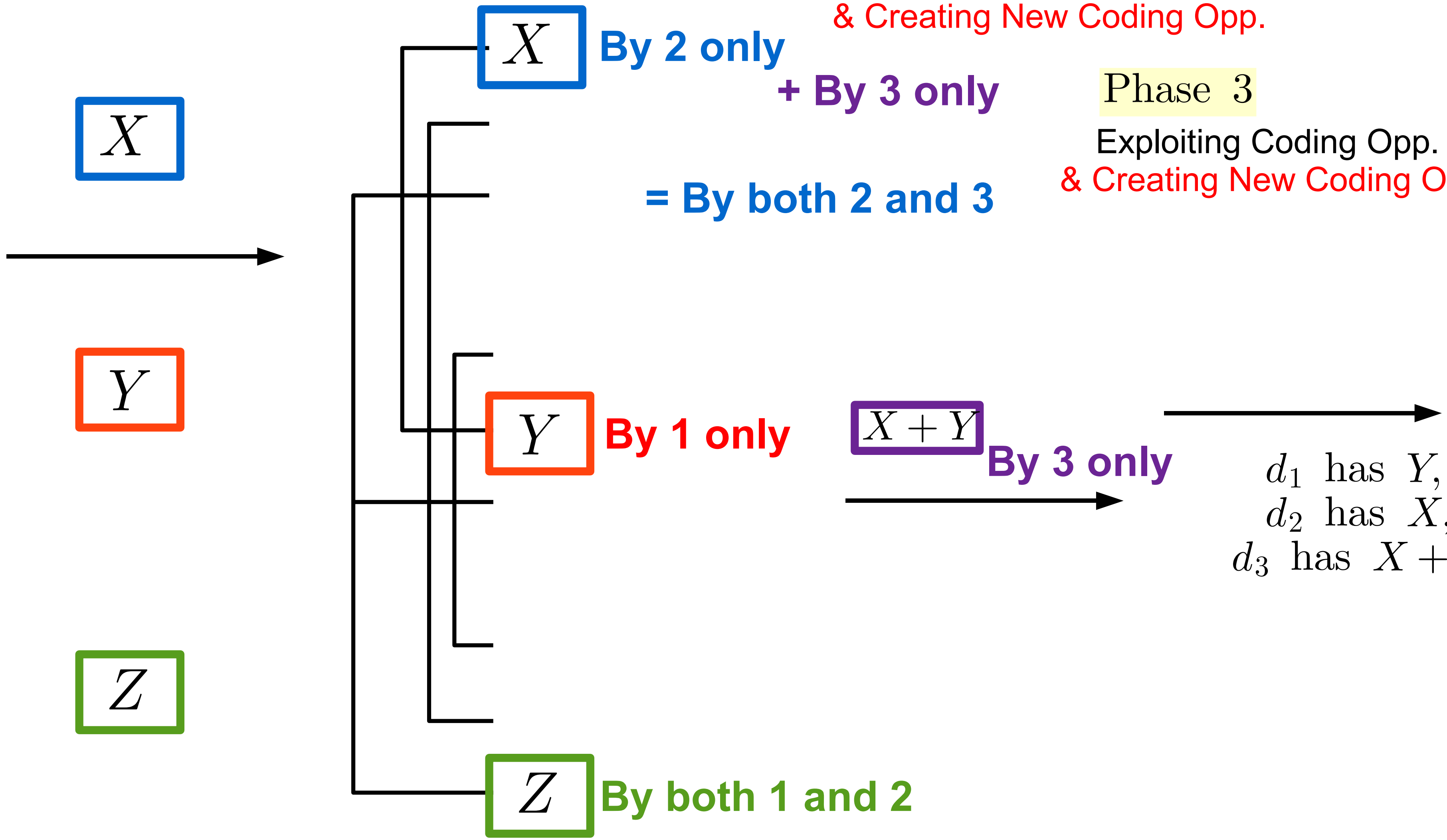


The Packet Evolution Scheme

Phase 1
Creating New Coding Opp.

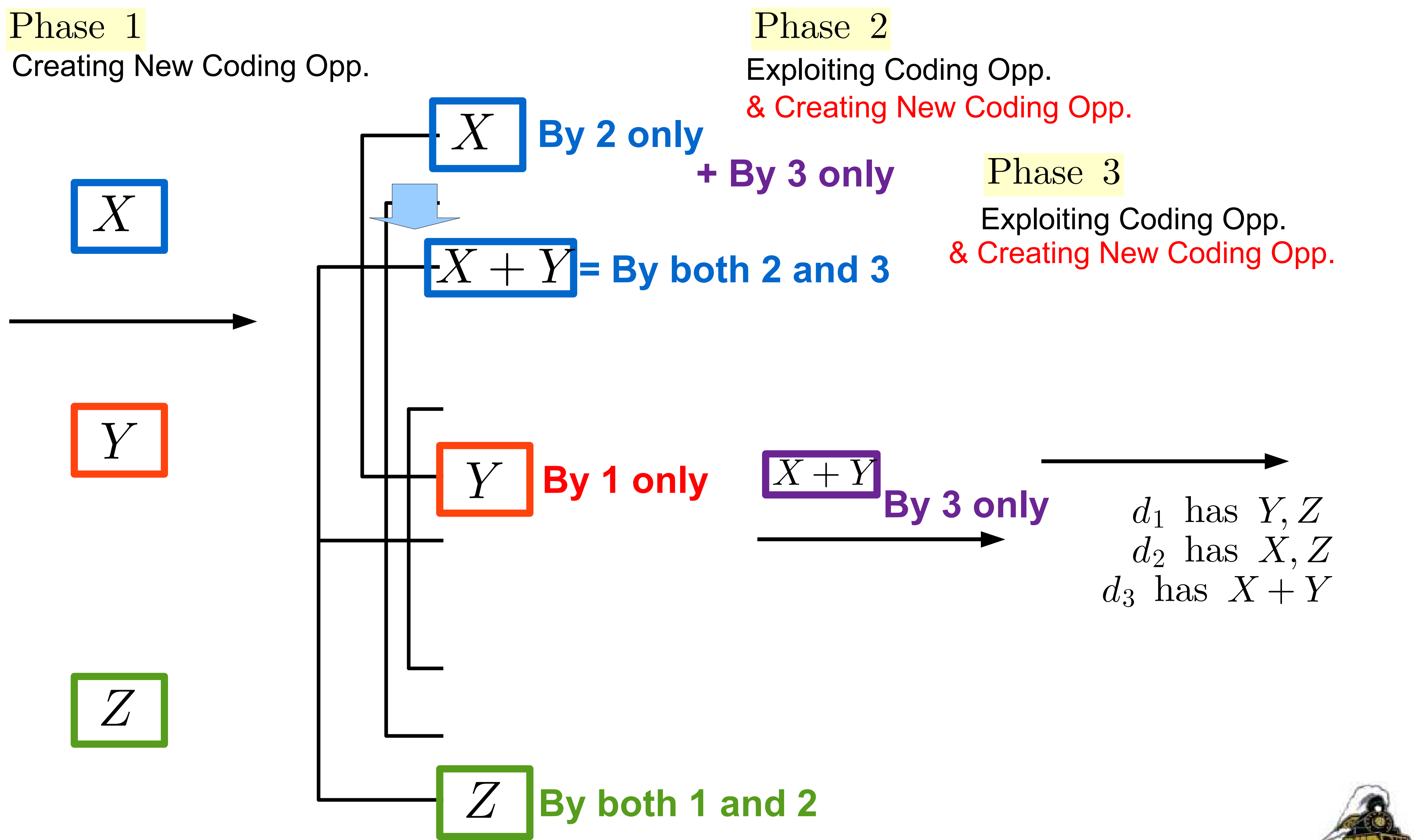
Phase 2
Exploiting Coding Opp.
& Creating New Coding Opp.

Phase 3
Exploiting Coding Opp.
& Creating New Coding Opp.

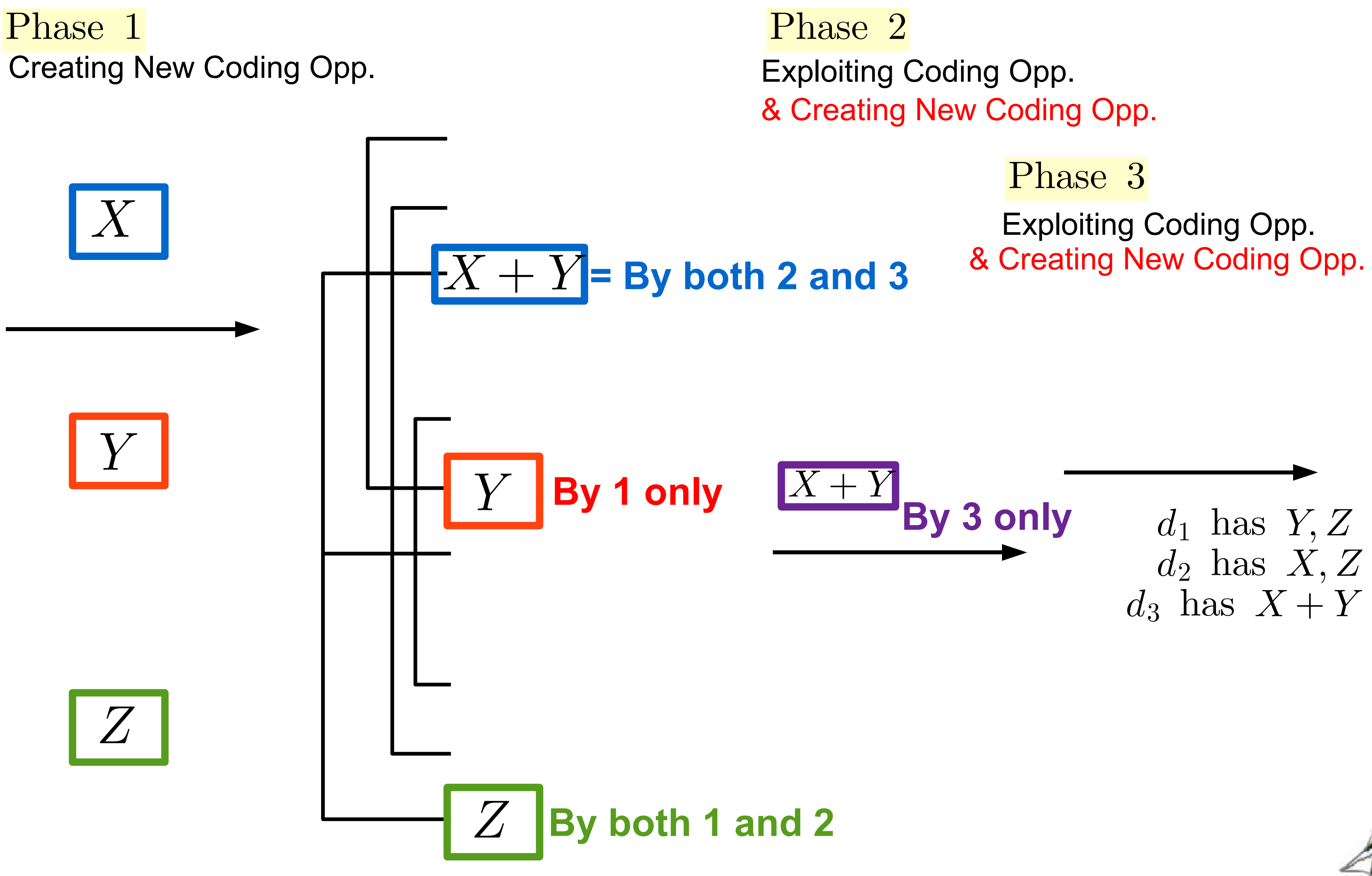


d_1 has Y, Z
d_2 has X, Z
d_3 has $X+Y$

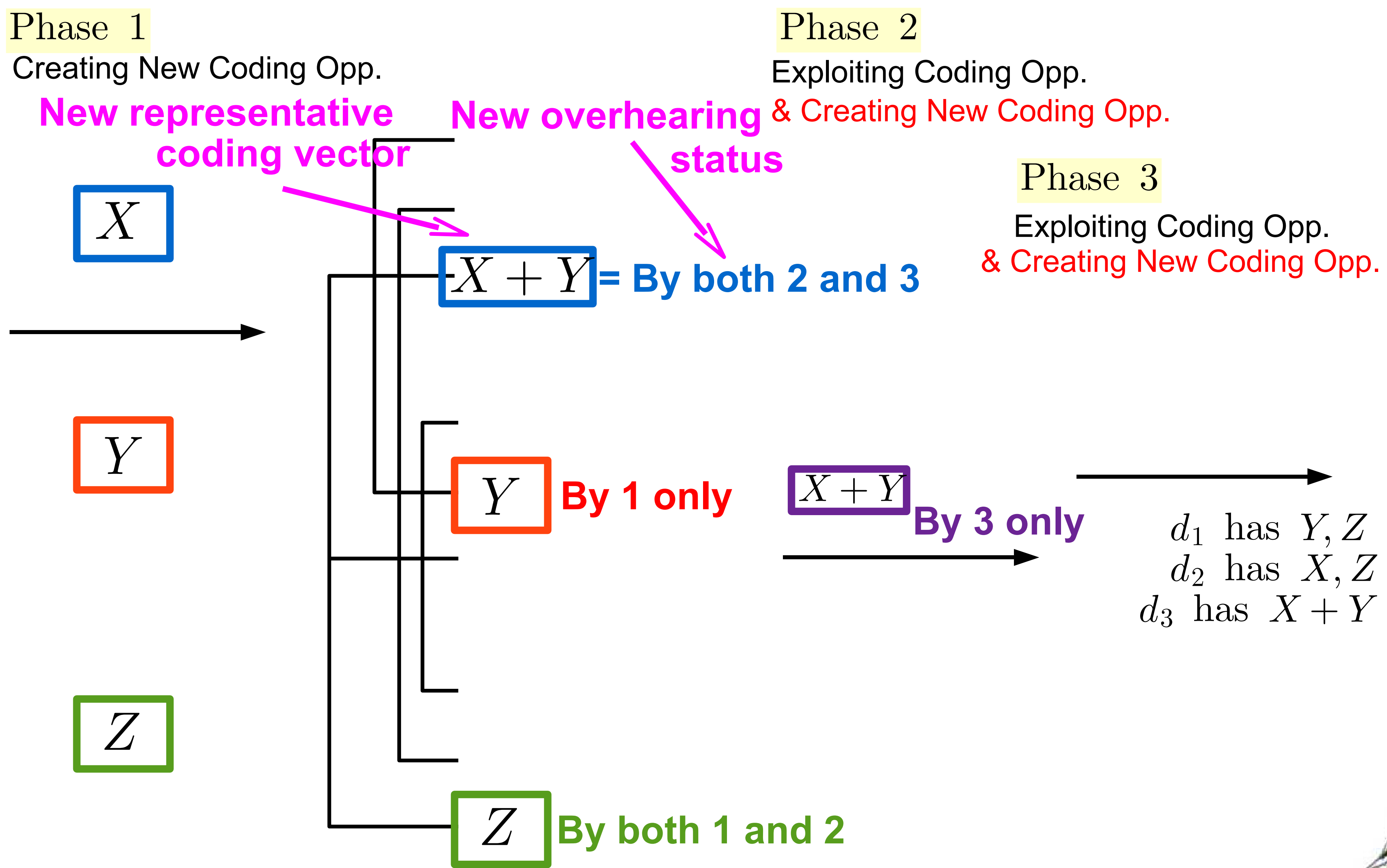
The Packet Evolution Scheme



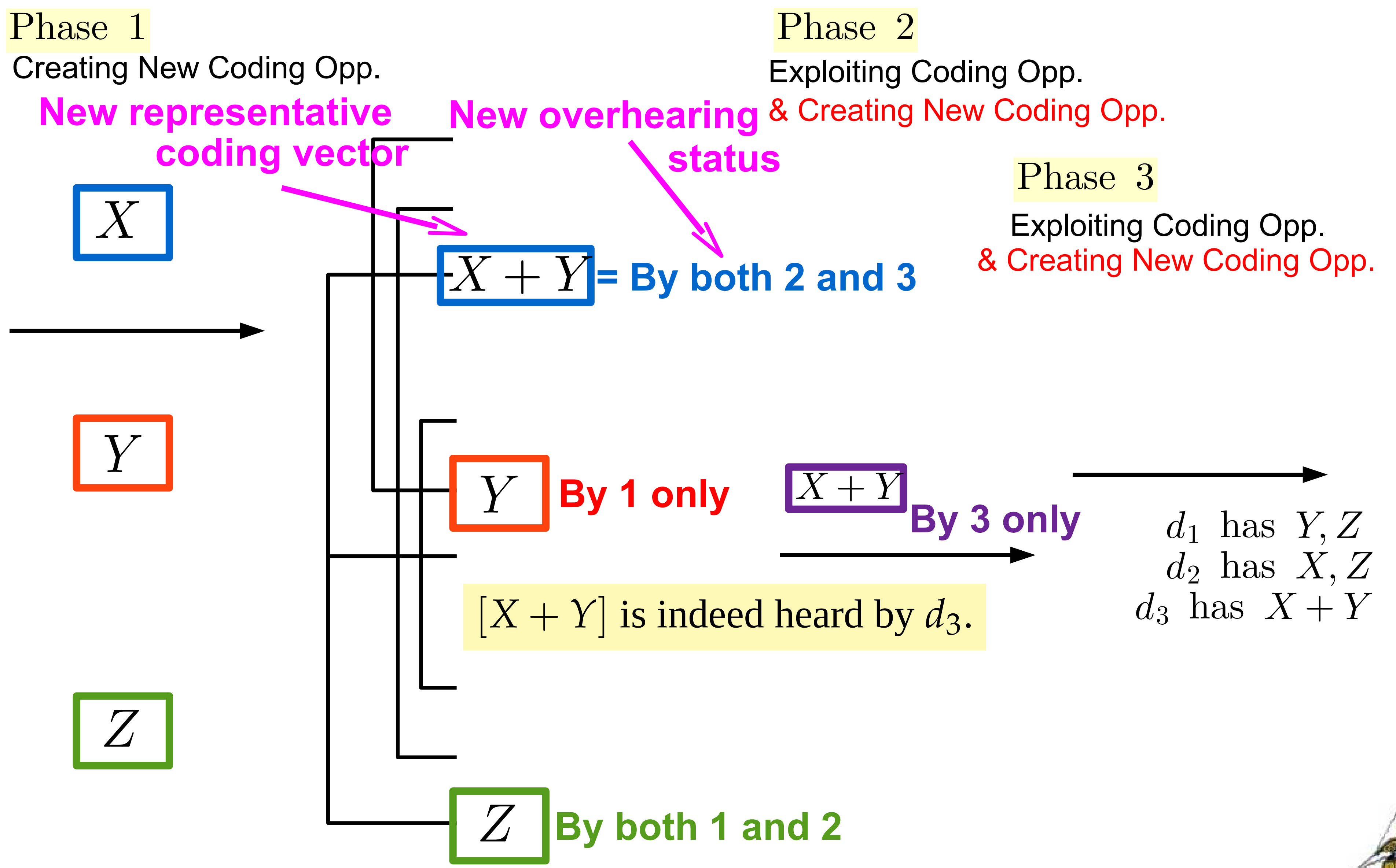
The Packet Evolution Scheme



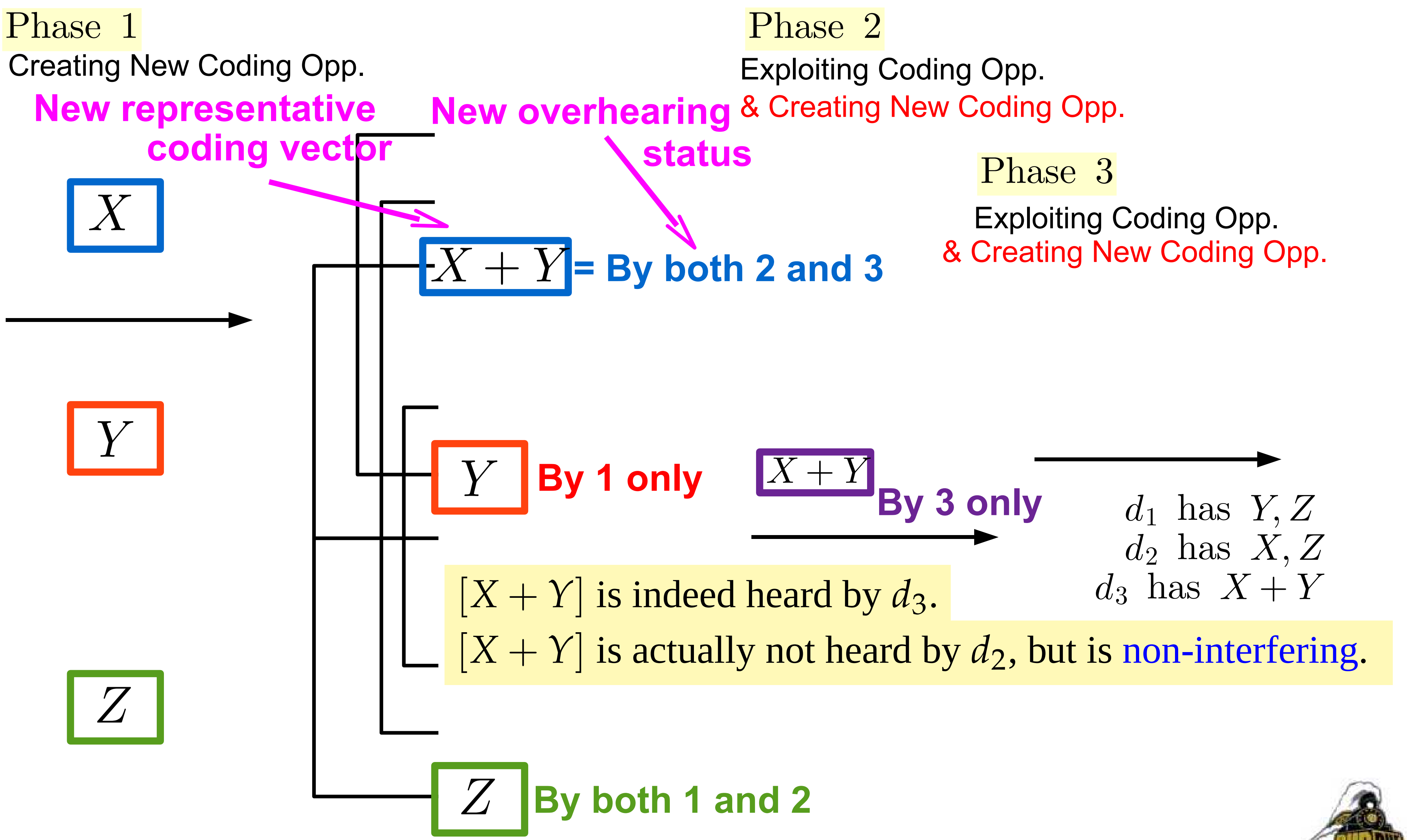
The Packet Evolution Scheme



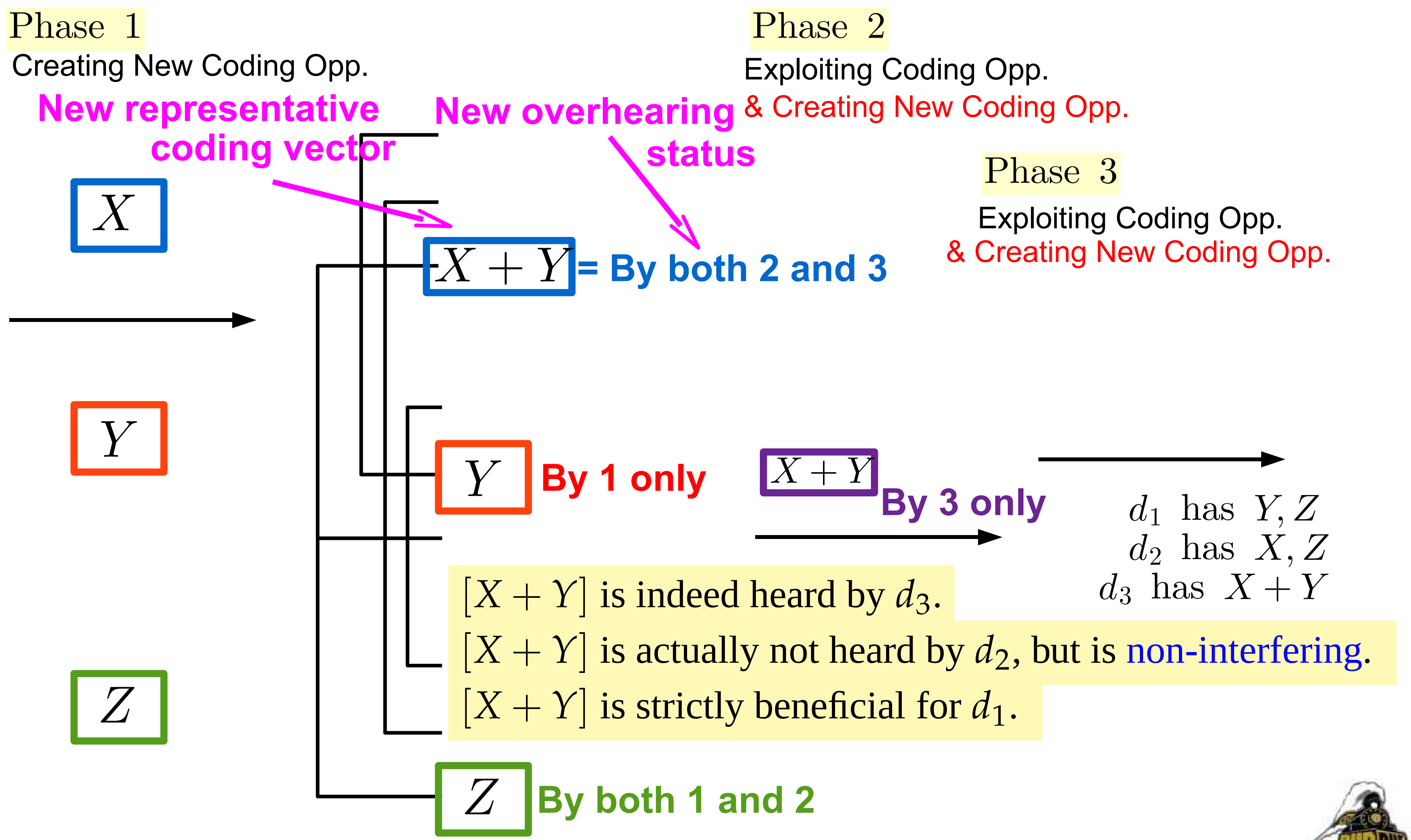
The Packet Evolution Scheme



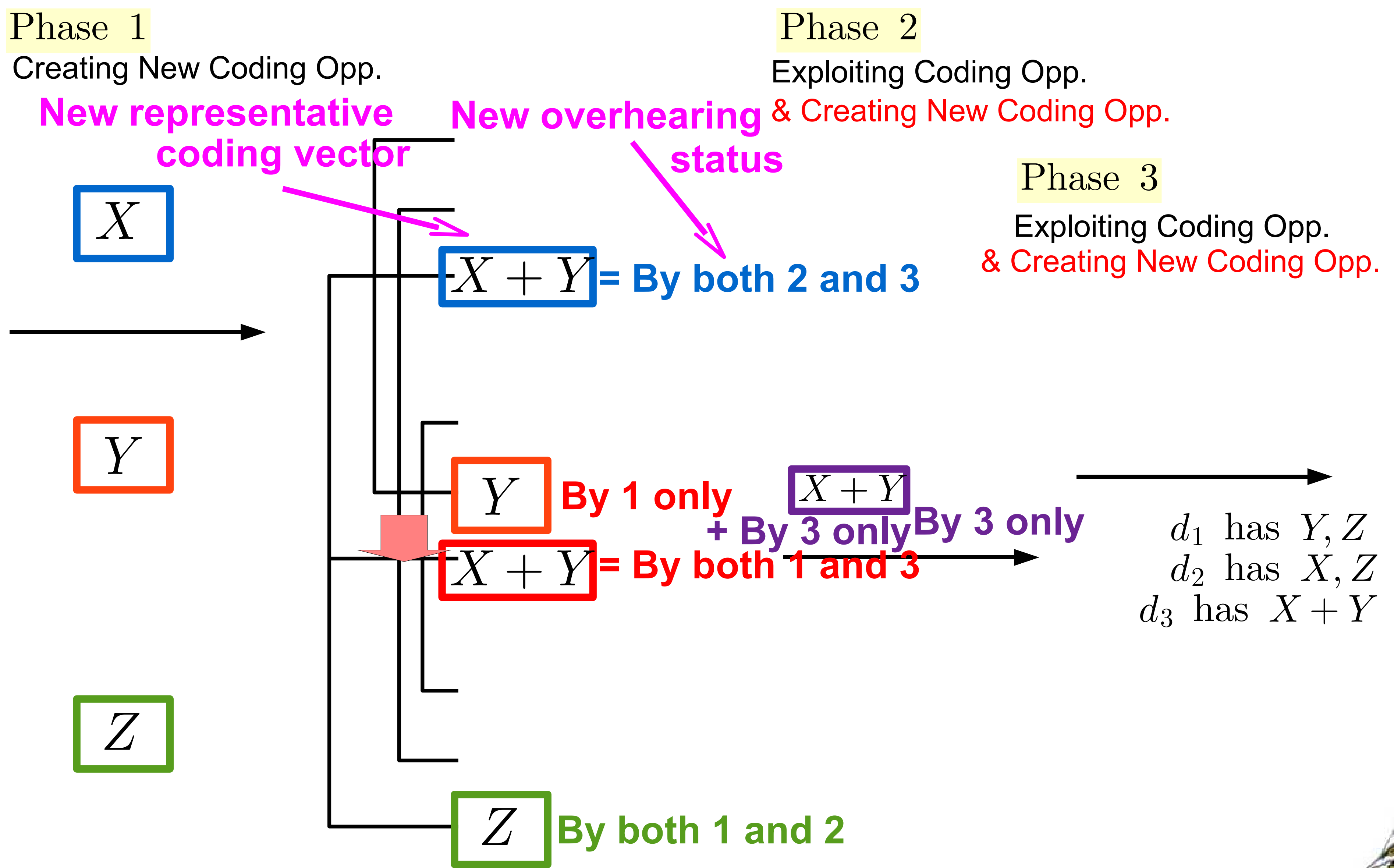
The Packet Evolution Scheme



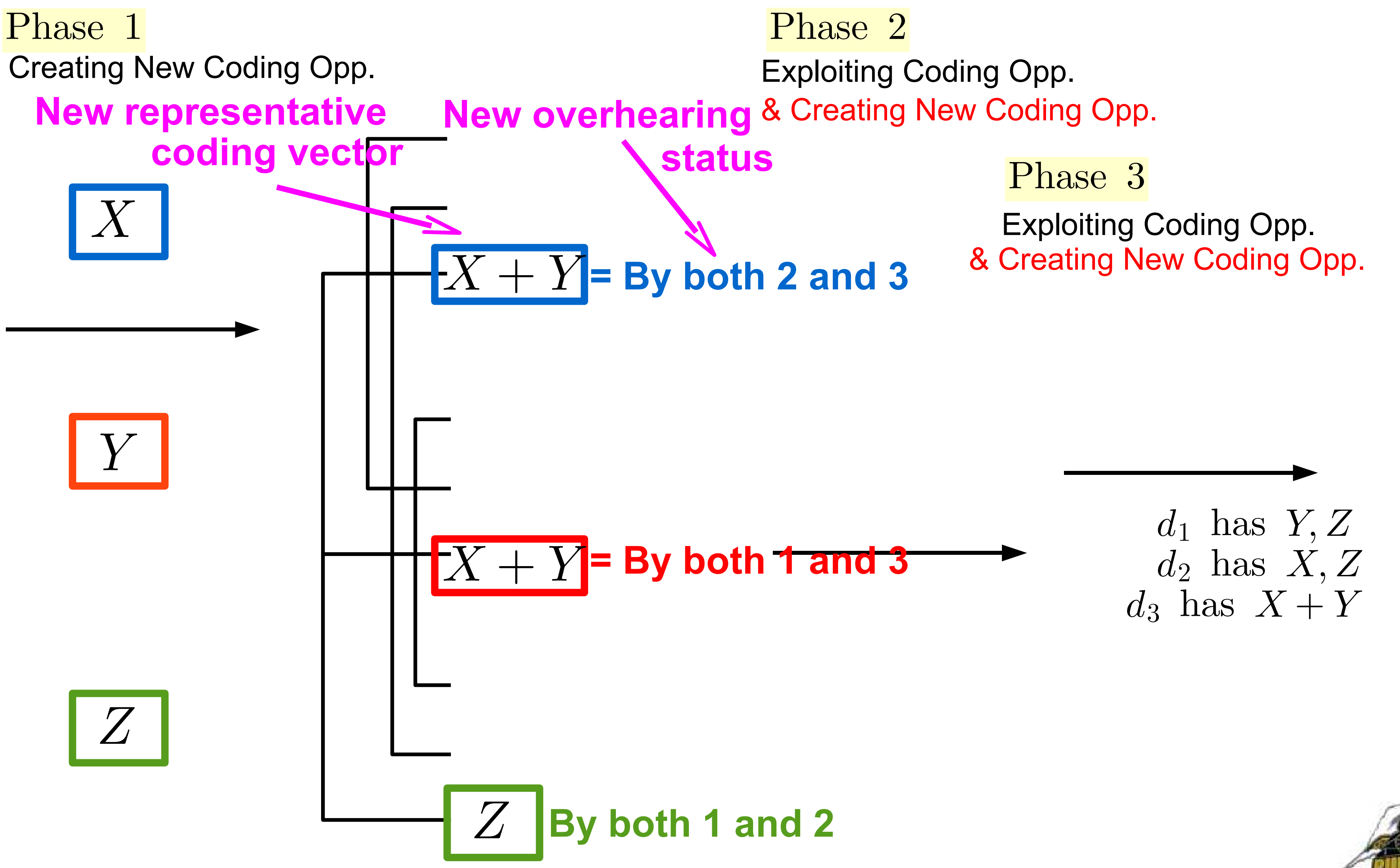
The Packet Evolution Scheme



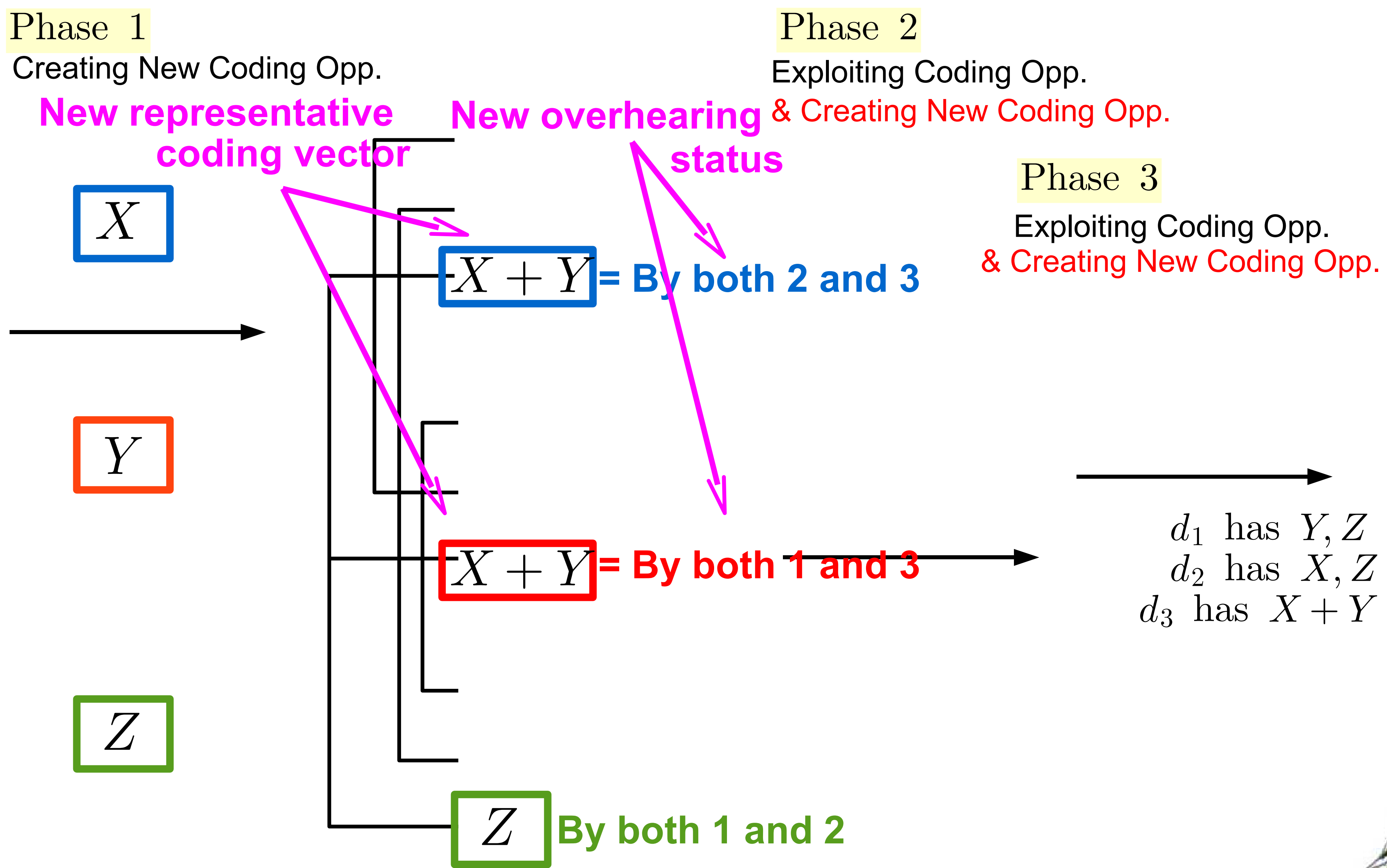
The Packet Evolution Scheme



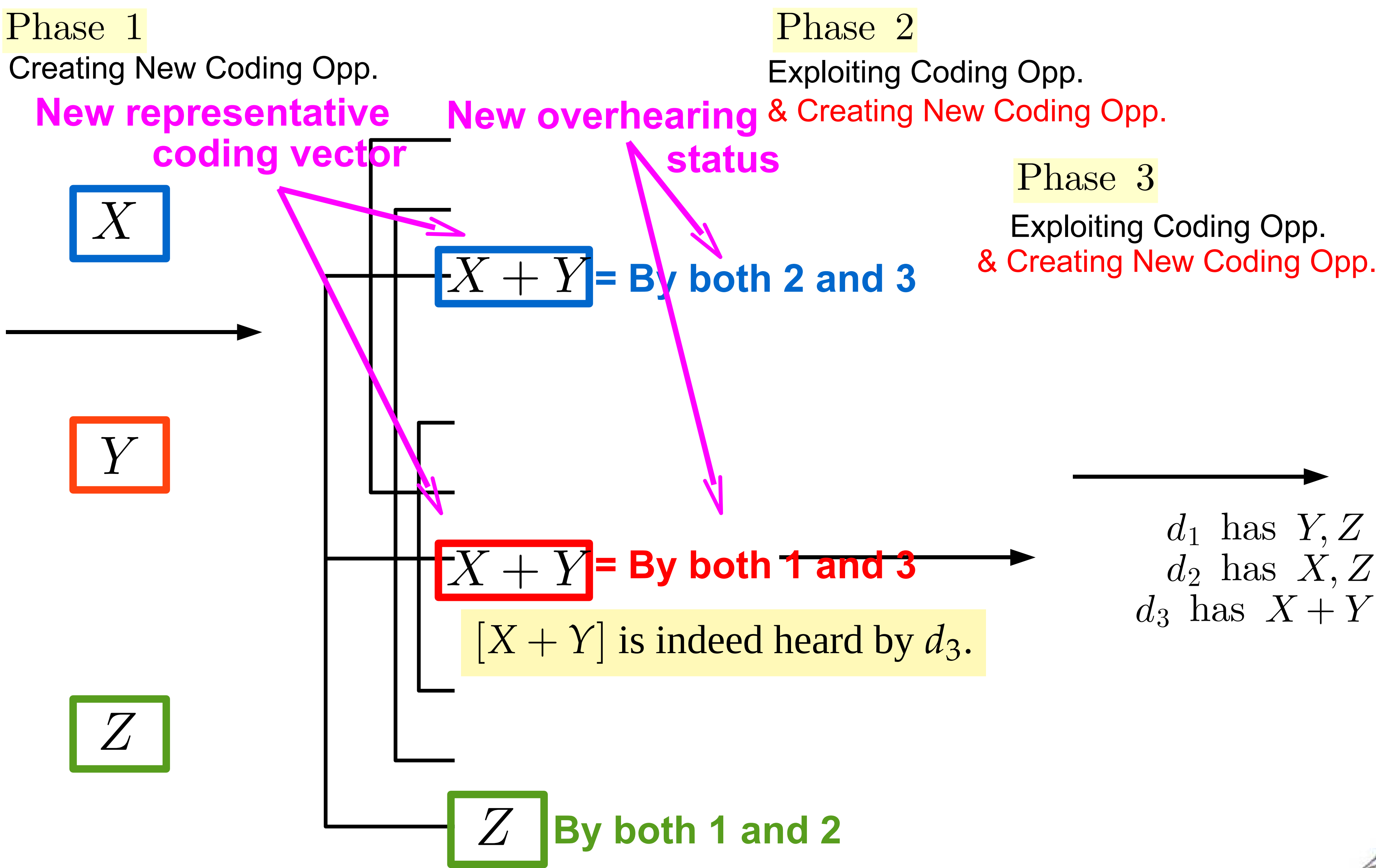
The Packet Evolution Scheme



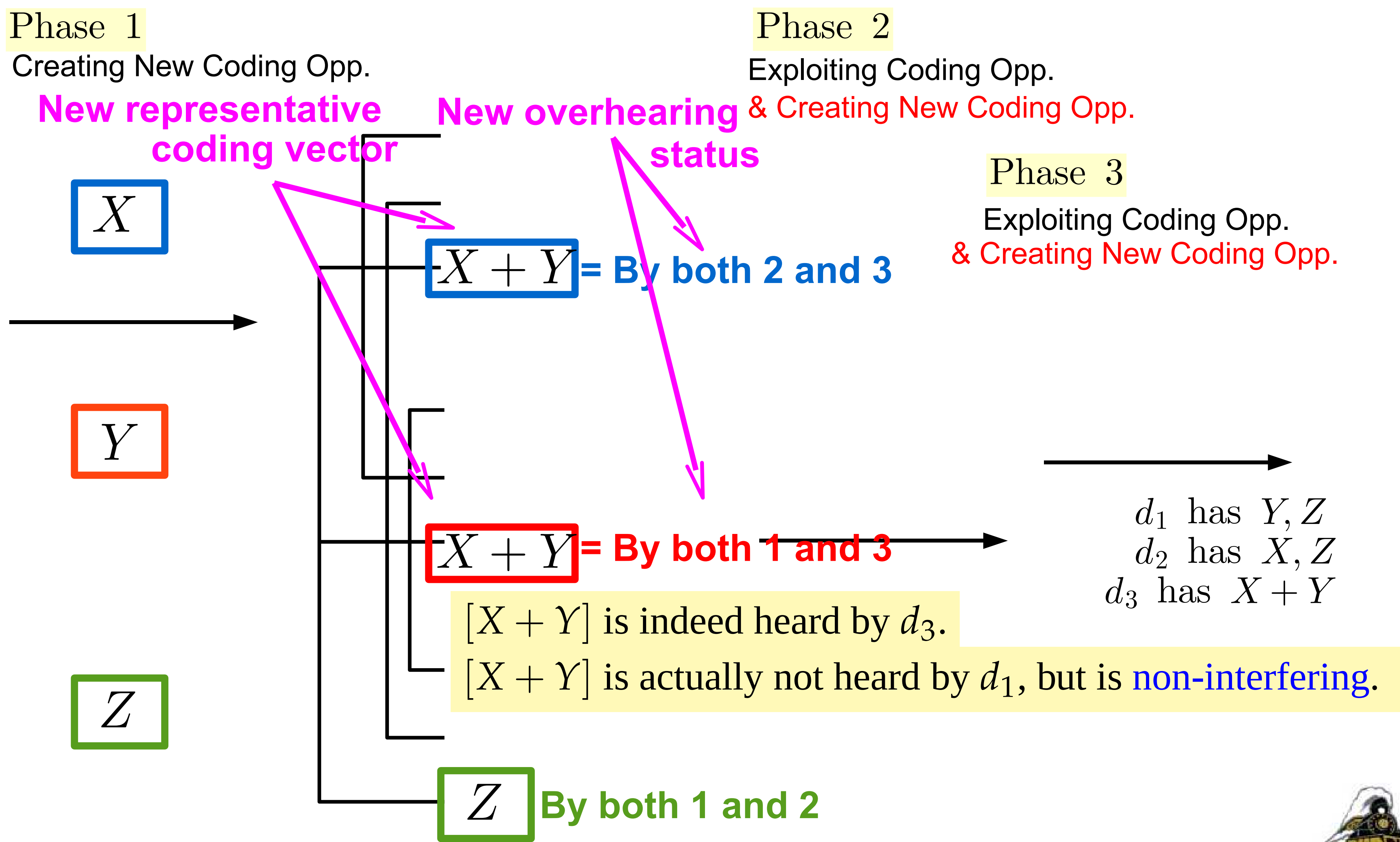
The Packet Evolution Scheme



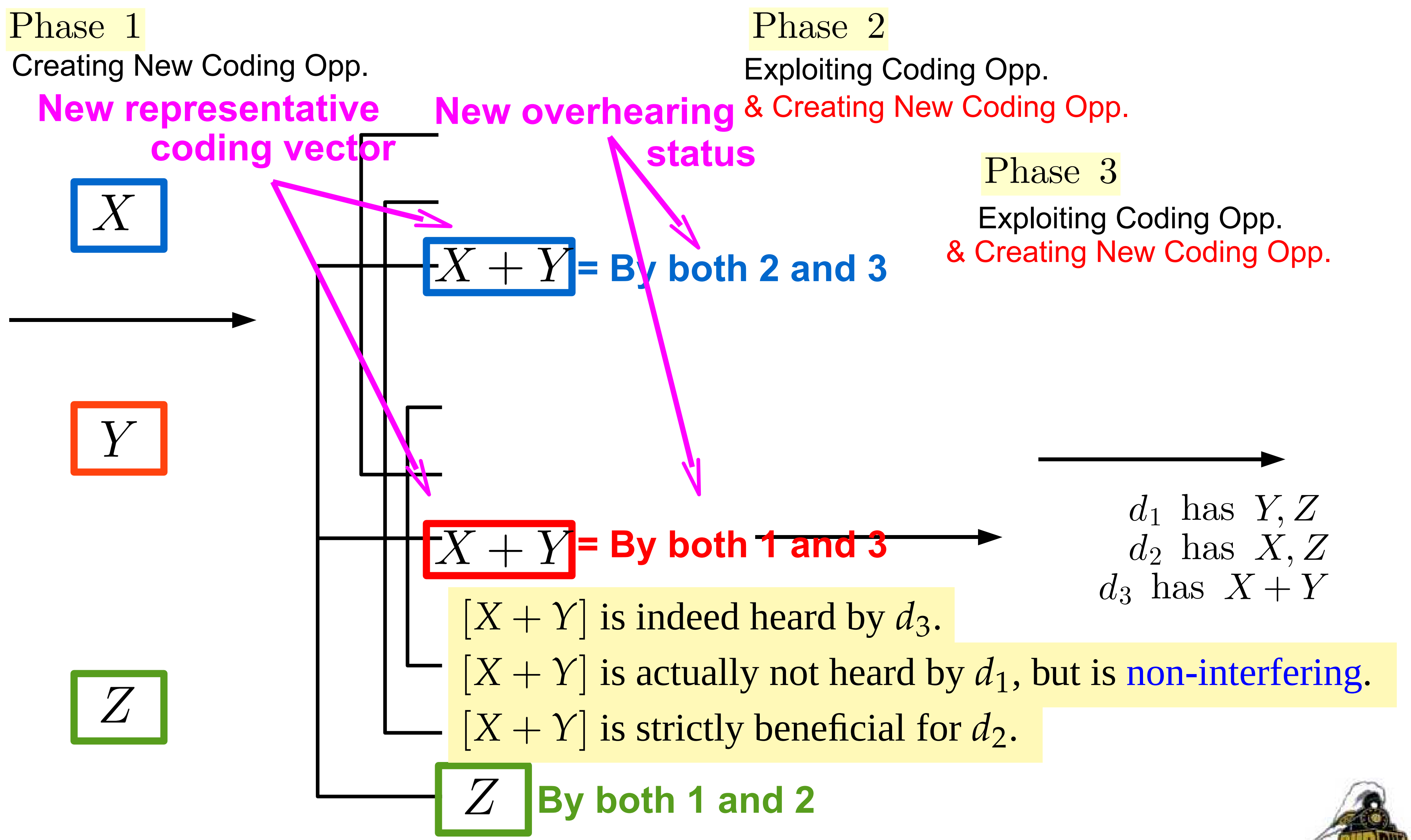
The Packet Evolution Scheme



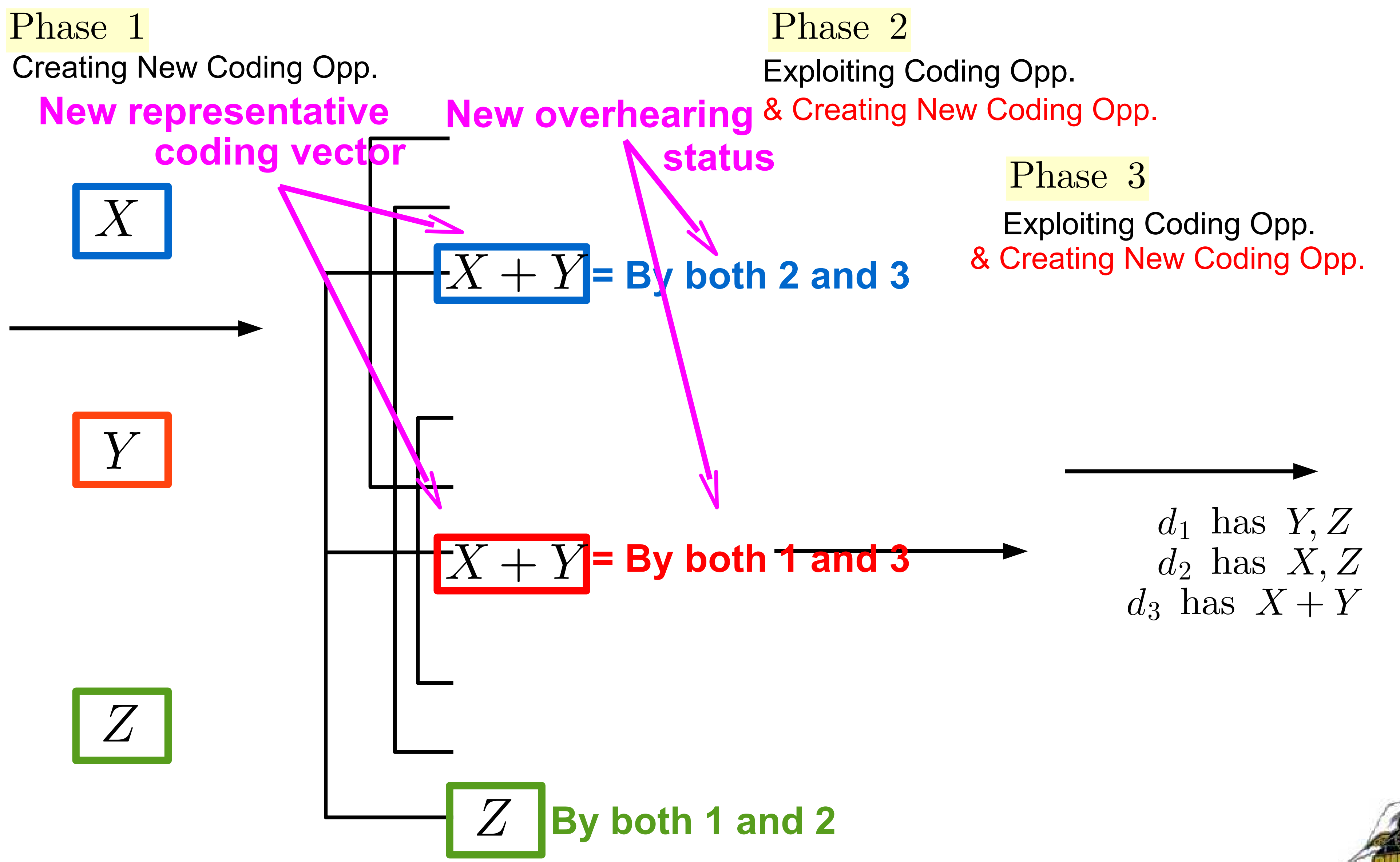
The Packet Evolution Scheme



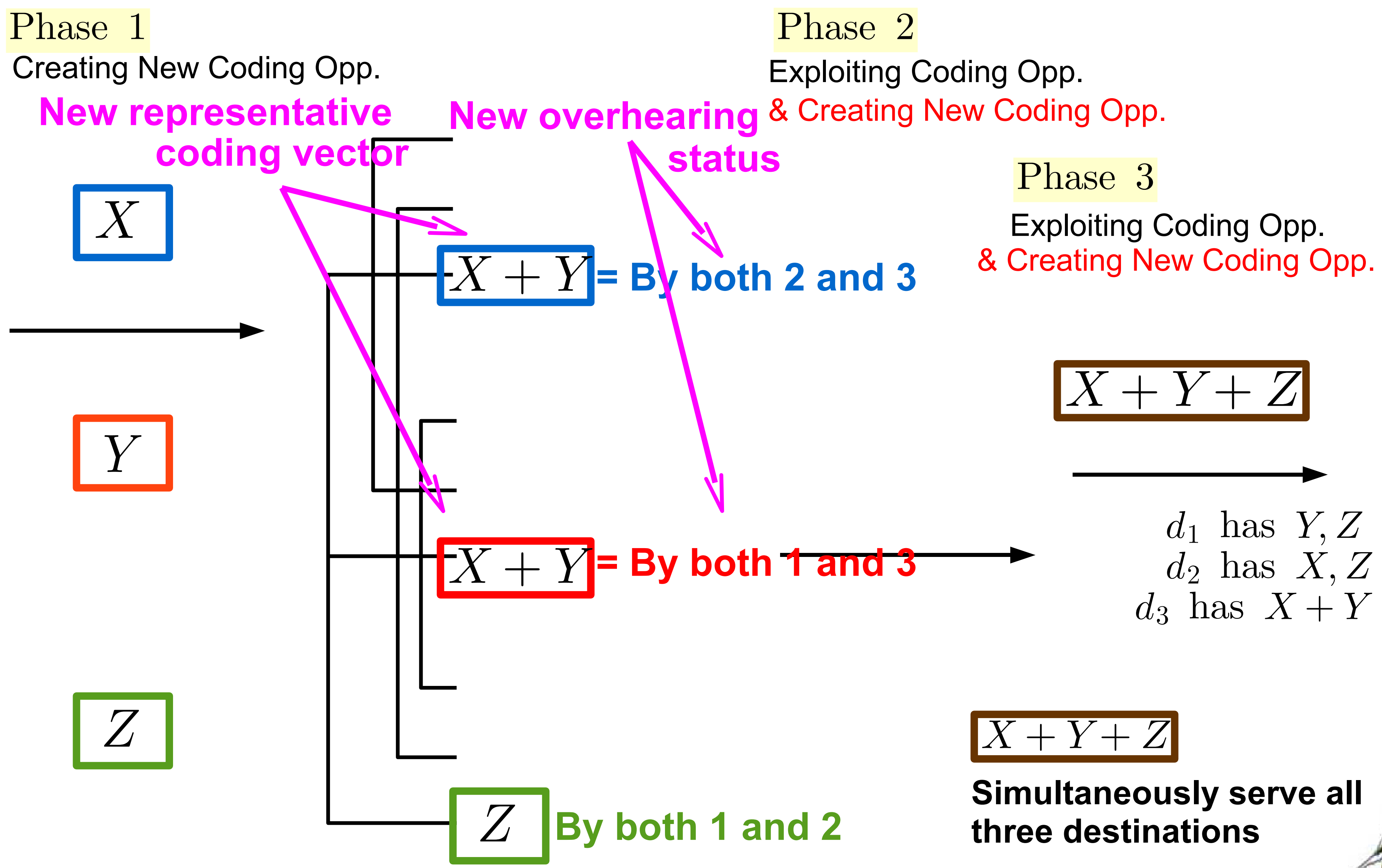
The Packet Evolution Scheme



The Packet Evolution Scheme



The Packet Evolution Scheme



An Example of $M = 4$

- $M = 4$ sessions, each d_k has $X_{k,1}$ to $X_{k,100}$ packets.
- Each $X_{k,l}, \forall k \in [4], l \in [100]$ has an overhearing status $S(X_{k,l}) \subseteq \{1,2,3,4\}$, and a representative coding vector $\mathbf{v}(X_{k,l})$ being a 400-dimensional vector.

An Example of $M = 4$

- $M = 4$ sessions, each d_k has $X_{k,1}$ to $X_{k,100}$ packets.
- Each $X_{k,l}, \forall k \in [4], l \in [100]$ has an overhearing status $S(X_{k,l}) \subseteq \{1,2,3,4\}$, and a representative coding vector $\mathbf{v}(X_{k,l})$ being a 400-dimensional vector.
- Use $S(X_{k,l})$ to decide which packets to be coded together.
  - Suppose we plan to encode sessions 1, 2, 3 together. We choose three packets X_{1,l_1}, X_{2,l_2}, and X_{3,l_3} such that $S(X_{1,l_1}) = \{2,3\}$, $S(X_{2,l_2}) = \{1,3\}$, and $S(X_{3,l_3}) = \{1,2\}$

An Example of $M = 4$

- $M = 4$ sessions, each d_k has $X_{k,1}$ to $X_{k,100}$ packets.
- Each $X_{k,l}, \forall k \in [4], l \in [100]$ has an overhearing status $S(X_{k,l}) \subseteq \{1,2,3,4\}$, and a representative coding vector $\mathbf{v}(X_{k,l})$ being a 400-dimensional vector.
- Use $S(X_{k,l})$ to decide which packets to be coded together.
  - Suppose we plan to encode sessions 1, 2, 3 together. We choose three packets X_{1,l_1}, X_{2,l_2}, and X_{3,l_3} such that $S(X_{1,l_1}) = \{2,3\}$, $S(X_{2,l_2}) = \{1,3\}$, and $S(X_{3,l_3}) = \{1,2\}$
- Instead of mixing X_{1,l_1} to X_{3,l_3}, we mix $\mathbf{v}(X_{1,l_1})$ to $\mathbf{v}(X_{3,l_3})$.
  - Generate $\mathbf{v}_{\text{tx}}$ by $\mathbf{v}_{\text{tx}} = c_1\mathbf{v}(X_{1,l_1}) + c_2\mathbf{v}(X_{2,l_2}) + c_3\mathbf{v}(X_{3,l_3})$.
  - Transmit $Y = \mathbf{v}_{\text{tx}}(X_{1,1}, \cdots, X_{4,100})^{\mathrm{T}}$.

An Example of $M = 4$

- Use $S(X_{k,l})$ to decide which packets to be coded together.
  - Suppose we plan to encode sessions 1, 2, 3 together. We choose three packets X_{1,l_1}, X_{2,l_2}, and X_{3,l_3} such that $S(X_{1,l_1}) = \{2,3\}$, $S(X_{2,l_2}) = \{1,3\}$, and $S(X_{3,l_3}) = \{1,2\}$
- Instead of mixing X_{1,l_1} to X_{3,l_3}, we mix $\mathbf{v}(X_{1,l_1})$ to $\mathbf{v}(X_{3,l_3})$.
  - Generate $\mathbf{v}_{\text{tx}}$ by $\mathbf{v}_{\text{tx}} = c_1\mathbf{v}(X_{1,l_1}) + c_2\mathbf{v}(X_{2,l_2}) + c_3\mathbf{v}(X_{3,l_3})$.
  - Transmit $Y = \mathbf{v}_{\text{tx}}(X_{1,1}, \cdots, X_{4,100})^{\mathrm{T}}$.

An Example of $M = 4$

- Use $S(X_{k,l})$ to decide which packets to be coded together.
  - Suppose we plan to encode sessions 1, 2, 3 together. We choose three packets X_{1,l_1}, X_{2,l_2}, and X_{3,l_3} such that $S(X_{1,l_1}) = \{2,3\}$, $S(X_{2,l_2}) = \{1,3\}$, and $S(X_{3,l_3}) = \{1,2\}$
- Instead of mixing X_{1,l_1} to X_{3,l_3}, we mix $\mathbf{v}(X_{1,l_1})$ to $\mathbf{v}(X_{3,l_3})$.
  - Generate $\mathbf{v}_{\text{tx}}$ by $\mathbf{v}_{\text{tx}} = c_1\mathbf{v}(X_{1,l_1}) + c_2\mathbf{v}(X_{2,l_2}) + c_3\mathbf{v}(X_{3,l_3})$.
  - Transmit $Y = \mathbf{v}_{\text{tx}}(X_{1,1}, \cdots, X_{4,100})^{\text{T}}$.
- Upon receiving a feedback, say "$\{d_3, d_4\}$ receive Y":
  - Augment overhearing status $S(x_{k,l})$ and update representative coding vector $\mathbf{v}(x_{k,l})$:

$$
\begin{aligned}
S(X_{1,l_1}) &\leftarrow S(X_{1,l_1}) \cup \{3,4\} = \{2,3,4\}, & \mathbf{v}(X_{1,l_1}) &\leftarrow \mathbf{v}_{\text{tx}} \\
S(X_{2,l_2}) &\leftarrow S(X_{2,l_2}) \cup \{3,4\} = \{1,3,4\}, & \mathbf{v}(X_{2,l_2}) &\leftarrow \mathbf{v}_{\text{tx}} \\
S(X_{3,l_3}) &\leftarrow S(X_{3,l_3}) \cup \{3,4\} = \{1,2,3,4\}, & \mathbf{v}(X_{3,l_3}) &\leftarrow \mathbf{v}_{\text{tx}}
\end{aligned}
$$

An Example of $M = 4$

- Use $S(X_{k,l})$ to decide which packets to be coded together.
  - Suppose we plan to encode sessions 1, 2, 3 together. We choose three packets X_{1,l_1}, X_{2,l_2}, and X_{3,l_3} such that $S(X_{1,l_1}) = \{2,3\}$, $S(X_{2,l_2}) = \{1,3\}$, and $S(X_{3,l_3}) = \{1,2\}$
- Instead of mixing X_{1,l_1} to X_{3,l_3}, we mix $\mathbf{v}(X_{1,l_1})$ to $\mathbf{v}(X_{3,l_3})$.
  - Generate $\mathbf{v}_{\text{tx}}$ by $\mathbf{v}_{\text{tx}} = c_1\mathbf{v}(X_{1,l_1}) + c_2\mathbf{v}(X_{2,l_2}) + c_3\mathbf{v}(X_{3,l_3})$.
  - Transmit $Y = \mathbf{v}_{\text{tx}}(X_{1,1}, \cdots, X_{4,100})^{\text{T}}$.
- Upon receiving a feedback, say "$\{d_3, d_4\}$ receive Y":
  - Augment overhearing status $S(x_{k,l})$ and update representative coding vector $\mathbf{v}(x_{k,l})$:

Create more coding Opp.

$$S(X_{1,l_1}) \leftarrow S(X_{1,l_1}) \cup \{3,4\} = \{2,3,4\}, \qquad \mathbf{v}(X_{1,l_1}) \leftarrow \mathbf{v}_{\text{tx}}$$
$$S(X_{2,l_2}) \leftarrow S(X_{2,l_2}) \cup \{3,4\} = \{1,3,4\}, \qquad \mathbf{v}(X_{2,l_2}) \leftarrow \mathbf{v}_{\text{tx}}$$
$$S(X_{3,l_3}) \leftarrow S(X_{3,l_3}) \cup \{3,4\} = \{1,2,3,4\}, \qquad \mathbf{v}(X_{3,l_3}) \leftarrow \mathbf{v}_{\text{tx}}$$

An Example of $M = 4$

- Use $S(X_{k,l})$ to decide which packets to be coded together.
  - Suppose we plan to encode sessions 1, 2, 3 together. We choose three packets X_{1,l_1}, X_{2,l_2}, and X_{3,l_3} such that $S(X_{1,l_1}) = \{2,3\}$, $S(X_{2,l_2}) = \{1,3\}$, and $S(X_{3,l_3}) = \{1,2\}$
- Instead of mixing X_{1,l_1} to X_{3,l_3}, we mix $\mathbf{v}(X_{1,l_1})$ to $\mathbf{v}(X_{3,l_3})$.
  - Generate $\mathbf{v}_{\text{tx}}$ by $\mathbf{v}_{\text{tx}} = c_1\mathbf{v}(X_{1,l_1}) + c_2\mathbf{v}(X_{2,l_2}) + c_3\mathbf{v}(X_{3,l_3})$.
  - Transmit $Y = \mathbf{v}_{\text{tx}}(X_{1,1}, \cdots, X_{4,100})^{\mathrm{T}}$.
- Upon receiving a feedback, say "$\{d_3, d_4\}$ receive Y":
  - Augment overhearing status $S(x_{k,l})$ and update representative coding vector $\mathbf{v}(x_{k,l})$:

Create more coding Opp. $S(X_{1,l_1}) \leftarrow S(X_{1,l_1}) \cup \{3,4\} = \{2,3,4\}, \quad \mathbf{v}(X_{1,l_1}) \leftarrow \mathbf{v}_{\text{tx}}$

Create more coding Opp. $S(X_{2,l_2}) \leftarrow S(X_{2,l_2}) \cup \{3,4\} = \{1,3,4\}, \quad \mathbf{v}(X_{2,l_2}) \leftarrow \mathbf{v}_{\text{tx}}$

$S(X_{3,l_3}) \leftarrow S(X_{3,l_3}) \cup \{3,4\} = \{1,2,3,4\}, \quad \mathbf{v}(X_{3,l_3}) \leftarrow \mathbf{v}_{\text{tx}}$

An Example of $M = 4$

- Use $S(X_{k,l})$ to decide which packets to be coded together.
  - Suppose we plan to encode sessions 1, 2, 3 together. We choose three packets X_{1,l_1}, X_{2,l_2}, and X_{3,l_3} such that $S(X_{1,l_1}) = \{2,3\}$, $S(X_{2,l_2}) = \{1,3\}$, and $S(X_{3,l_3}) = \{1,2\}$
- Instead of mixing X_{1,l_1} to X_{3,l_3}, we mix $\mathbf{v}(X_{1,l_1})$ to $\mathbf{v}(X_{3,l_3})$.
  - Generate $\mathbf{v}_{\text{tx}}$ by $\mathbf{v}_{\text{tx}} = c_1\mathbf{v}(X_{1,l_1}) + c_2\mathbf{v}(X_{2,l_2}) + c_3\mathbf{v}(X_{3,l_3})$.
  - Transmit $Y = \mathbf{v}_{\text{tx}}(X_{1,1}, \cdots, X_{4,100})^{\mathrm{T}}$.
- Upon receiving a feedback, say "$\{d_3, d_4\}$ receive Y":
  - Augment overhearing status $S(x_{k,l})$ and update representative coding vector $\mathbf{v}(x_{k,l})$:

Create more coding Opp.	$S(X_{1,l_1}) \leftarrow S(X_{1,l_1}) \cup \{3,4\} = \{2,3,4\},$	$\mathbf{v}(X_{1,l_1}) \leftarrow \mathbf{v}_{\text{tx}}$
Create more coding Opp.	$S(X_{2,l_2}) \leftarrow S(X_{2,l_2}) \cup \{3,4\} = \{1,3,4\},$	$\mathbf{v}(X_{2,l_2}) \leftarrow \mathbf{v}_{\text{tx}}$
X_{3,l_3} has arrived d_3	$S(X_{3,l_3}) \leftarrow S(X_{3,l_3}) \cup \{3,4\} = \{1,2,3,4\},$	$\mathbf{v}(X_{3,l_3}) \leftarrow \mathbf{v}_{\text{tx}}$

An Example of $M = 4$

- Use $S(X_{k,l})$ to decide which packets to be coded together.
  - Suppose we plan to encode sessions 1, 2, 3 together. We choose three packets X_{1,l_1}, X_{2,l_2}, and X_{3,l_3} such that $S(X_{1,l_1}) = \{2,3\}$, $S(X_{2,l_2}) = \{1,3\}$, and $S(X_{3,l_3}) = \{1,2\}$
- Instead of mixing X_{1,l_1} to X_{3,l_3}, we mix $\mathbf{v}(X_{1,l_1})$ to $\mathbf{v}(X_{3,l_3})$.
  - Generate $\mathbf{v}_{\text{tx}}$ by $\mathbf{v}_{\text{tx}} = c_1\mathbf{v}(X_{1,l_1}) + c_2\mathbf{v}(X_{2,l_2}) + c_3\mathbf{v}(X_{3,l_3})$.
  - Transmit $Y = \mathbf{v}_{\text{tx}}(X_{1,1}, \cdots, X_{4,100})^{\text{T}}$. Achieve Code Alignment
- Upon receiving a feedback, say "$\{d_3, d_4\}$ receive Y":
  - Augment overhearing status $S(x_{k,l})$ and update representative coding vector $\mathbf{v}(x_{k,l})$:

Achieve Code Alignment

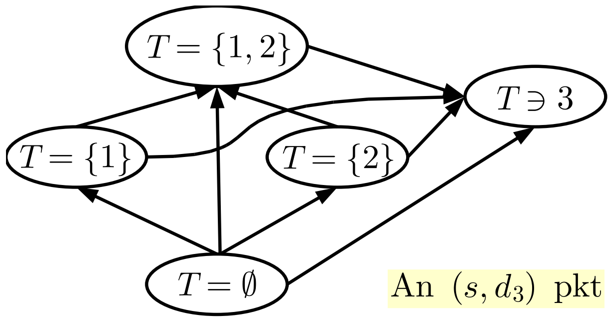
Create more coding Opp. $S(X_{1,l_1}) \leftarrow S(X_{1,l_1}) \cup \{3,4\} = \{2,3,4\}$, $\mathbf{v}(X_{1,l_1}) \leftarrow \mathbf{v}_{\text{tx}}$

Create more coding Opp. $S(X_{2,l_2}) \leftarrow S(X_{2,l_2}) \cup \{3,4\} = \{1,3,4\}$, $\mathbf{v}(X_{2,l_2}) \leftarrow \mathbf{v}_{\text{tx}}$

X_{3,l_3} has arrived d_3 $S(X_{3,l_3}) \leftarrow S(X_{3,l_3}) \cup \{3,4\} = \{1,2,3,4\}$, $\mathbf{v}(X_{3,l_3}) \leftarrow \mathbf{v}_{\text{tx}}$

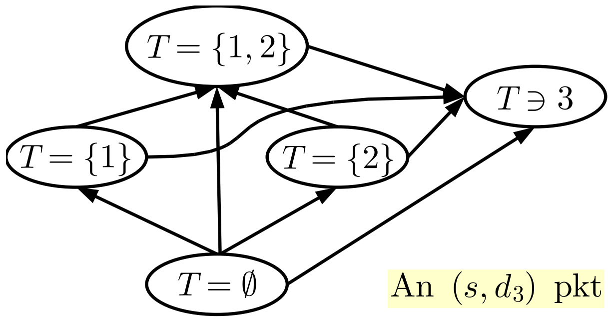
Analysis of The PE Schemes

- In the packet evolution scheme, each packet evolves independently.

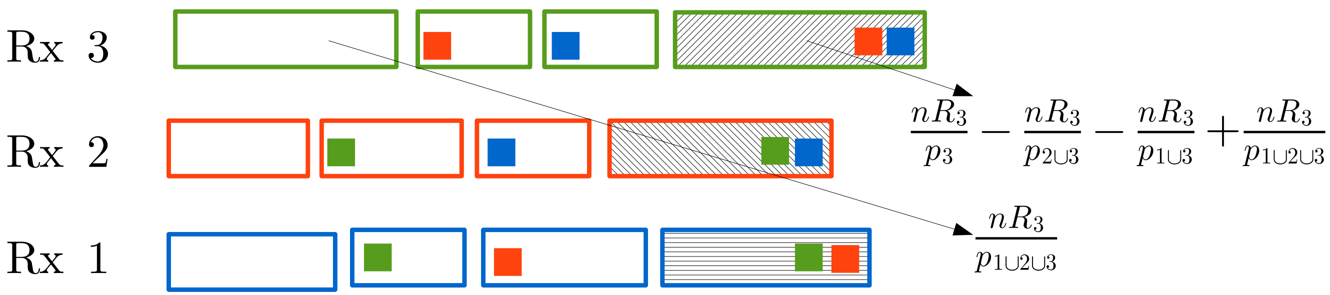


Analysis of The PE Schemes

- In the packet evolution scheme, each packet evolves independently.

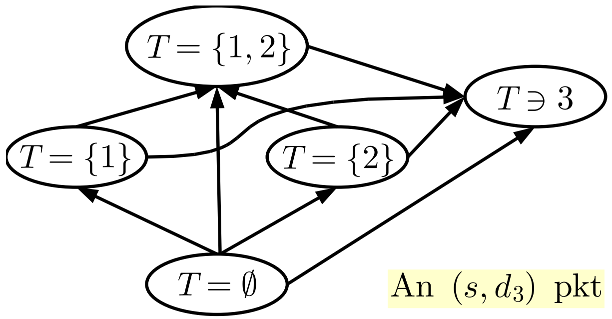


- We can quantify the number of slots that a packet has overhearing status T.

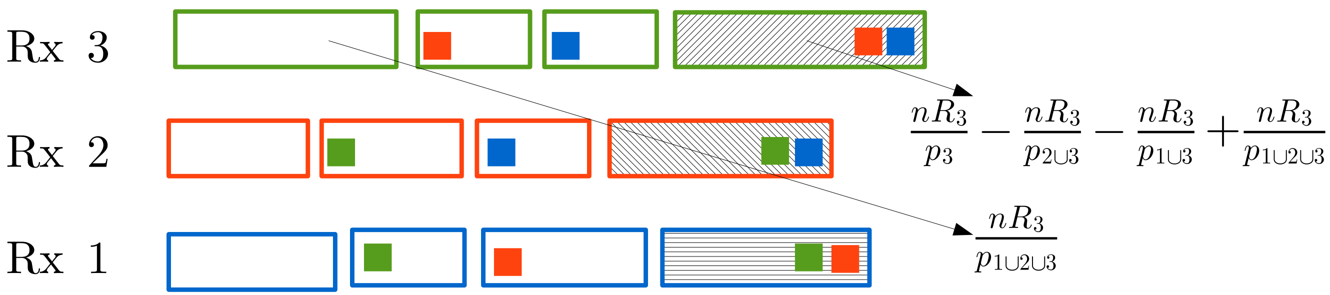


Analysis of The PE Schemes

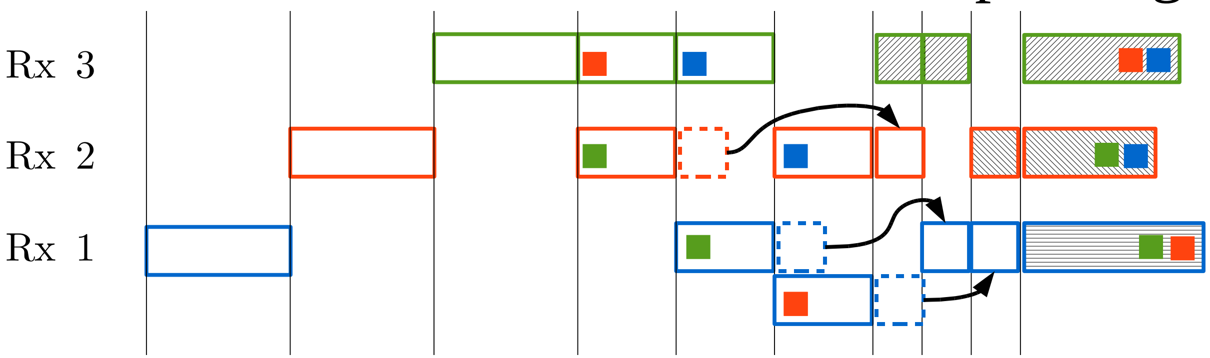
- In the packet evolution scheme, each packet evolves independently.



- We can quantify the number of slots that a packet has overhearing status T.

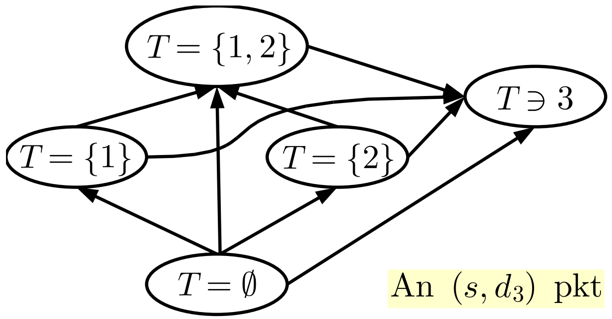


- The analysis of PE schemes becomes a time-slot packing problem:



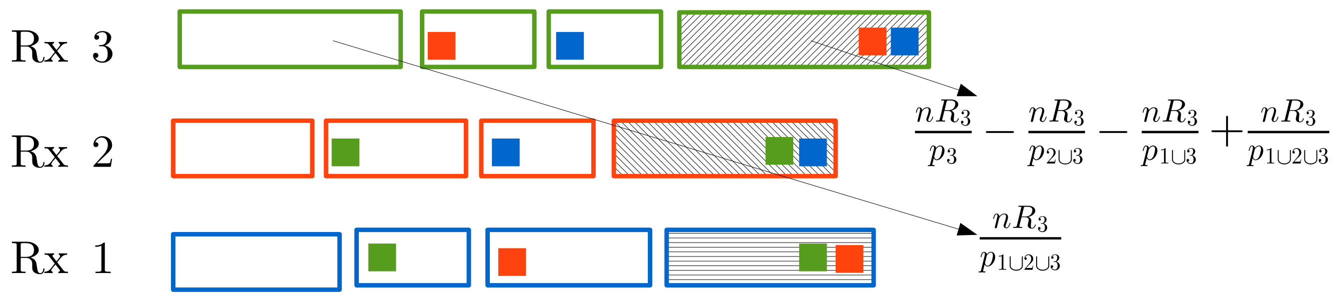
Analysis of The PE Schemes

- In the packet evolution scheme, each packet evolves independently.

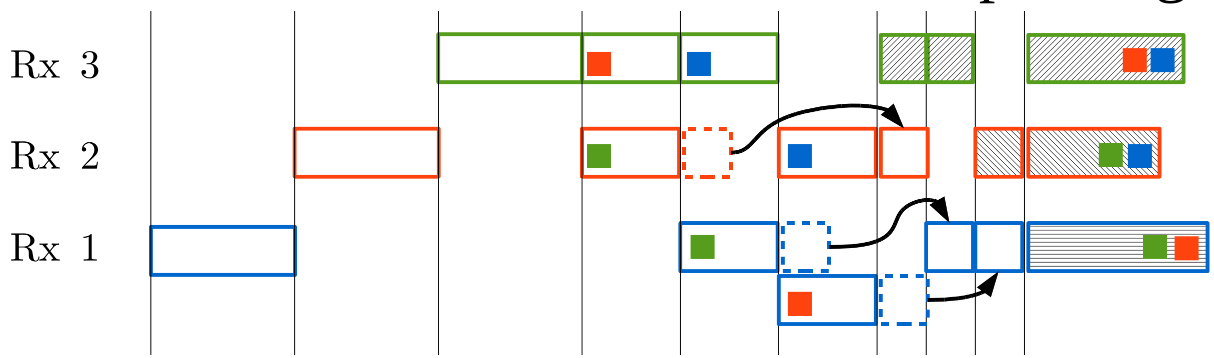


- We can quantify the number of slots that a packet has overhearing status T.

The joint success prob. $p_{S\overline{\{1,2,3\}\backslash S}}$ affects the duration of each status, and thus how to pack them.



- The analysis of PE schemes becomes a time-slot packing problem:

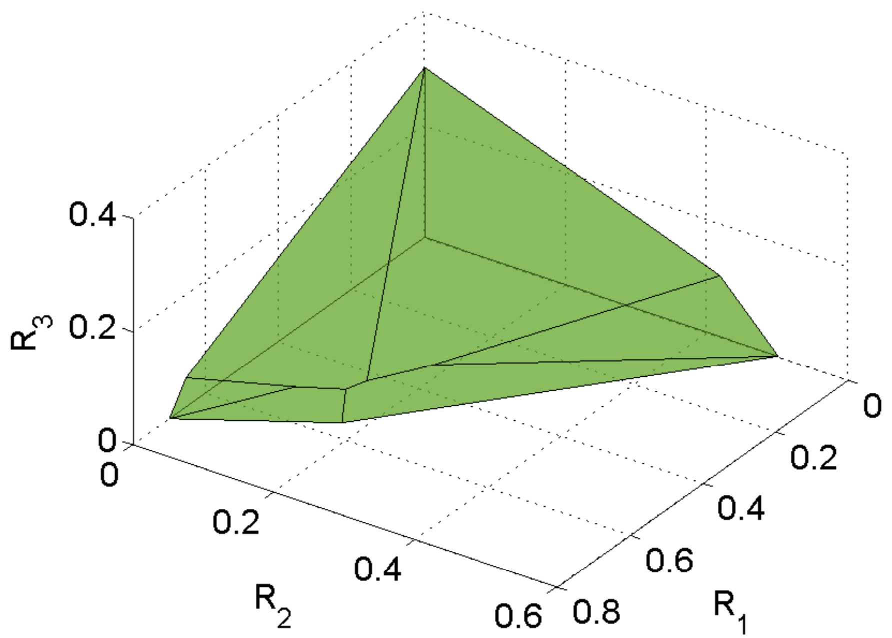


Capacity Results $M = 3$

Based on the Packet Evolution method, we have:

Proposition 1 *Consider any 1-to-3 broadcast PEC with channel output feedback with arbitrary parameters* $p_{S\overline{(\{1,2,3\}\backslash S)}}$ *for all* $S \subseteq \{1,2,3\}$.

The capacity region is indeed $\forall \pi, \sum_{k=1}^{M} \frac{R_{\pi(k)}}{p_{\cup S_k^\pi}} \leq 1$.



6 facets $\Leftrightarrow$ 6 different permutations π

Capacity Results $M \geq 4$

Outer bound: $\forall \pi, \quad \sum_{k=1}^{M} \frac{R_{\pi(k)}}{p_{\cup S_k^{\pi}}} \leq 1.$

Settings with general M>3 values	Capacity inner bound results
General $p_{S\overline{[M]\backslash S}}$	
Spatially symmetric broadcast PECs	
Spatially independent broadcast PECs	

Capacity Results $M \geq 4$

Outer bound: $\forall \pi, \quad \sum_{k=1}^{M} \frac{R_{\pi(k)}}{p_{\cup S_k^\pi}} \leq 1.$

Settings with general M>3 values	Capacity inner bound results
General $p_{S\overline{[M]\backslash S}}$	* A cap. Inner bound by using LP solvers to find the tightest time-slot packing * Numerically meets the outer bound for all our experiments
Spatially symmetric broadcast PECs	
Spatially independent broadcast PECs	

Capacity Results $M \geq 4$

Outer bound: $\forall \pi, \quad \sum_{k=1}^{M} \frac{R_{\pi(k)}}{p_{\cup S_k^\pi}} \leq 1.$

Settings with general M>3 values	Capacity inner bound results
General $p_{S\overline{[M]\backslash S}}$	* A cap. Inner bound by using LP solvers to find the tightest time-slot packing * Numerically meets the outer bound for all our experiments
Spatially symmetric broadcast PECs $p_{S_1\overline{[M]\backslash S_1}} = p_{S_2\overline{[M]\backslash S_2}}$ if $\lvert S_1\rvert = \lvert S_2\rvert$	
Spatially independent broadcast PECs	

Capacity Results $M \geq 4$

Outer bound: $\forall \pi, \quad \sum_{k=1}^{M} \frac{R_{\pi(k)}}{p_{\cup S_k^\pi}} \leq 1.$

Settings with general M>3 values	Capacity inner bound results
General $p_{S\overline{[M]\backslash S}}$	* A cap. Inner bound by using LP solvers to find the tightest time-slot packing * Numerically meets the outer bound for all our experiments
Spatially symmetric broadcast PECs $p_{S_1\overline{[M]\backslash S_1}} = p_{S_2\overline{[M]\backslash S_2}}$ if $\lvert S_1\rvert = \lvert S_2\rvert$	The inner and outer bounds always meet. → Full capacity region.
Spatially independent broadcast PECs	

Capacity Results $M \geq 4$

Outer bound: $\forall \pi, \quad \sum_{k=1}^{M} \frac{R_{\pi(k)}}{p_{\cup S_k^{\pi}}} \leq 1.$

Settings with general M>3 values	Capacity inner bound results
General $p_{S\overline{[M]\backslash S}}$	* A cap. Inner bound by using LP solvers to find the tightest time-slot packing * Numerically meets the outer bound for all our experiments
Spatially symmetric broadcast PECs $p_{S_1\overline{[M]\backslash S_1}} = p_{S_2\overline{[M]\backslash S_2}}$ if $\lvert S_1\rvert = \lvert S_2\rvert$	The inner and outer bounds always meet. → Full capacity region.
Spatially independent broadcast PECs $p_{S\overline{[M]\backslash S}} = \prod_{k\in S} p_k \prod_{j\in[M]\backslash S}(1-p_j)$	

Capacity Results $M \geq 4$

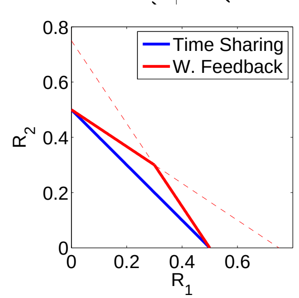
Outer bound: $\forall \pi, \quad \sum_{k=1}^{M} \frac{R_{\pi(k)}}{p_{\cup S_k^\pi}} \leq 1.$

Settings with general M>3 values	Capacity inner bound results
General $p_{S\overline{[M]\backslash S}}$	* A cap. Inner bound by using LP solvers to find the tightest time-slot packing * Numerically meets the outer bound for all our experiments
Spatially symmetric broadcast PECs $p_{S_1\overline{[M]\backslash S_1}} = p_{S_2\overline{[M]\backslash S_2}}$ if $\|S_1\| = \|S_2\|$	The inner and outer bounds always meet. → Full capacity region.
Spatially independent broadcast PECs $p_{S\overline{[M]\backslash S}} = \prod_{k\in S} p_k \prod_{j\in[M]\backslash S}(1-p_j)$	The inner and outer bounds meet when $(R_1, \cdots, R_M)$ are one-sided fair (when $R_1 \approx R_2 \approx \cdots \approx R_M$)

Capacity Results $M \geq 4$

Outer bound: $\forall \pi, \quad \sum_{k=1}^{M} \frac{R_{\pi(k)}}{p_{\cup S_k^{\pi}}} \leq 1.$

Settings with general M>3 values	Capacity inner bound results
General $p_{S\overline{[M]\backslash S}}$	* A cap. Inner bound by **using LP solvers** to find the **tightest time-slot packing** * **Numerically meets** the outer bound for all our experiments
Spatially symmetric broadcast PECs $p_{S_1\overline{[M]\backslash S_1}} = p_{S_2\overline{[M]\backslash S_2}}$ if $\lvert S_1 \rvert = \lvert S_2 \rvert$	The inner and outer bounds always meet. → **Full capacity region**.
Spatially independent broadcast PECs $p_{S\overline{[M]\backslash S}} = \prod_{k\in S} p_k \prod_{j\in[M]\backslash S}($	The inner and outer bounds meet when $(R_1, \cdots, R_M)$ are **one-sided fair** $R_1 \approx R_2 \approx \cdots \approx R_M)$



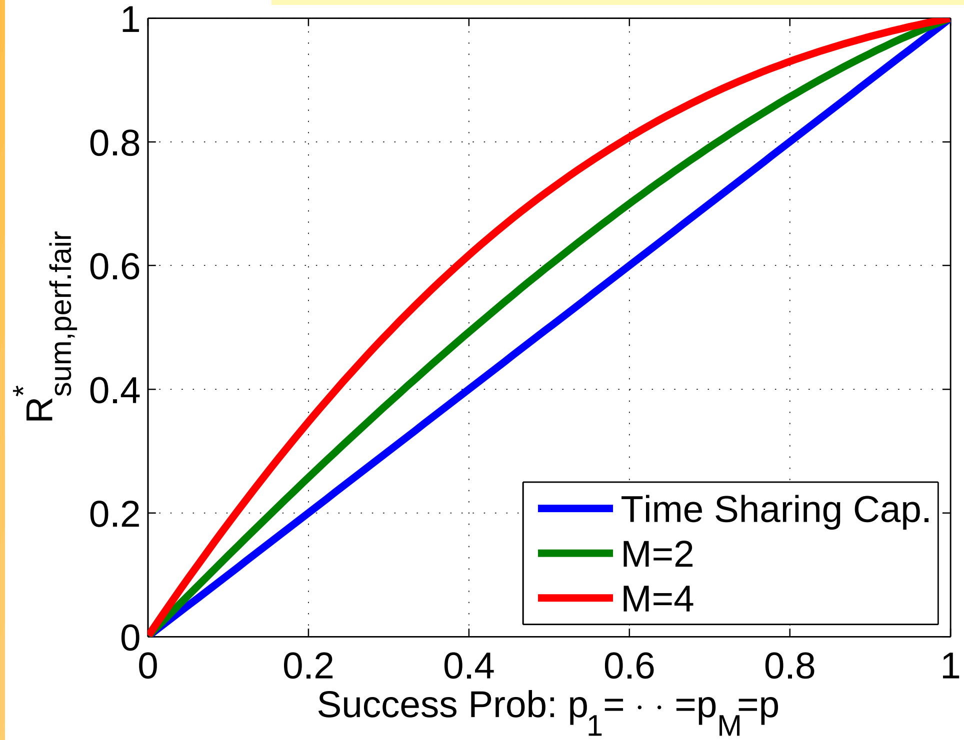
Numerical Evaluation

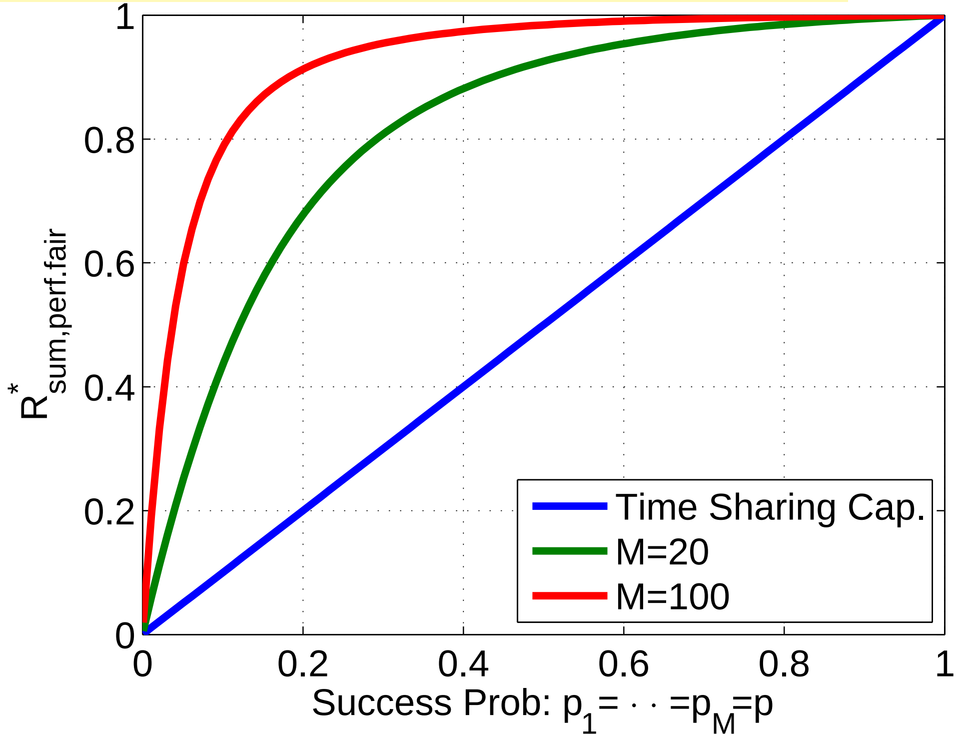
$$\forall \pi, \sum_{k=1}^{M} \frac{R_{\pi(k)}}{p_{\cup S_k^{\pi}}} \leq 1.$$

Symmetric spatially independent PECs: $p_1 = p_2 = \cdots = p_M = p$

Perfectly fair systems: $R_1 = R_2 = \cdots = R_M$

Sum rate capacity $\sum_{k=1}^{M} R_k$ vs. marginal success prob. p.





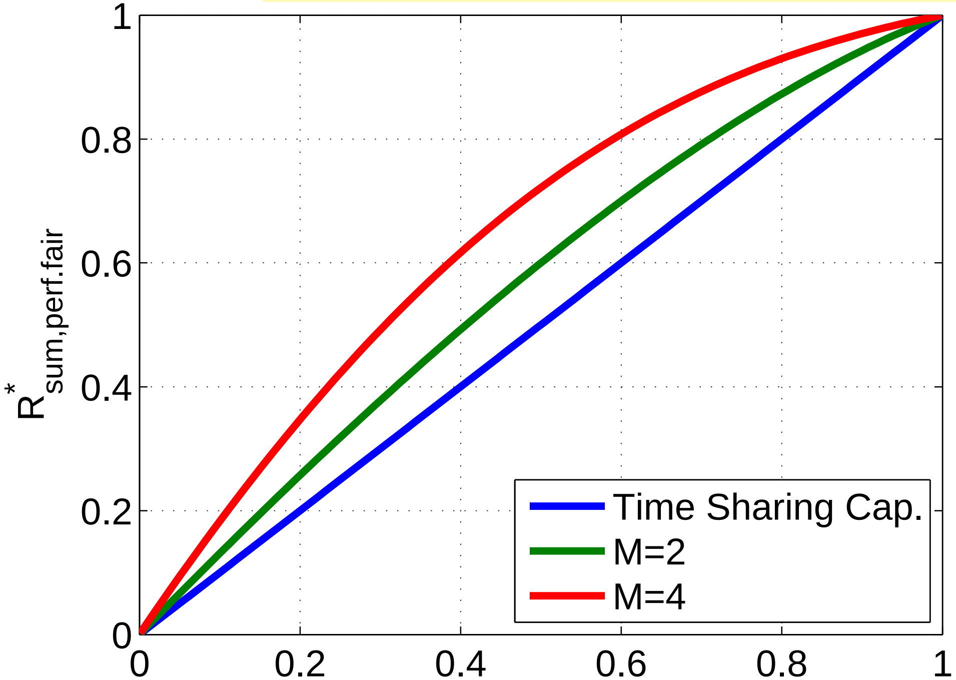
Numerical Evaluation

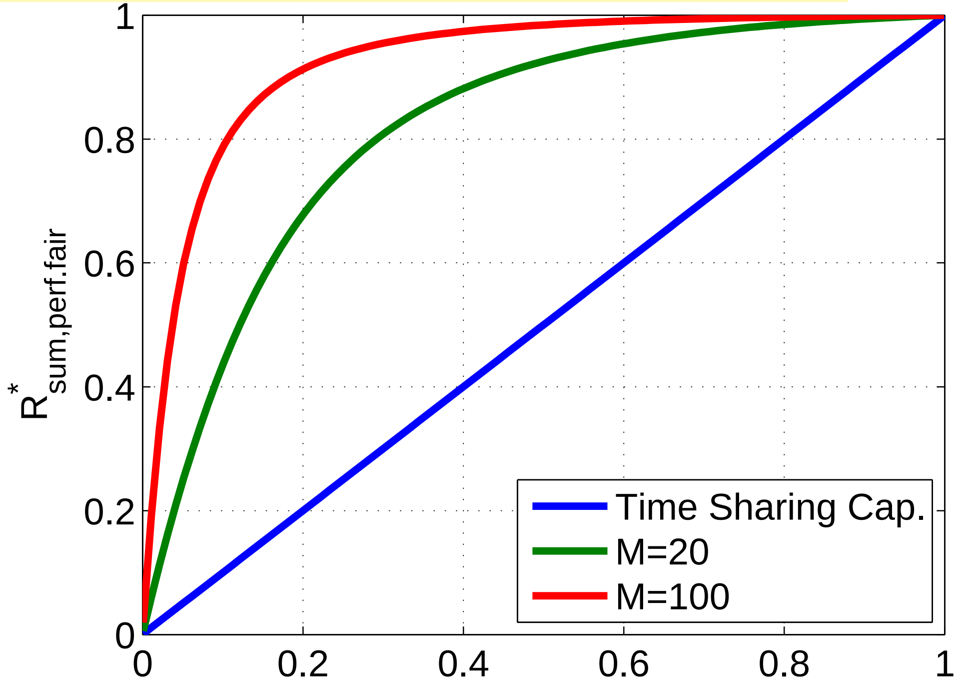
$$\forall \pi, \sum_{k=1}^{M} \frac{R_{\pi(k)}}{p_{\cup S_k^{\pi}}} \leq 1.$$

Symmetric spatially independent PECs: $p_1 = p_2 = \cdots = p_M = p$

Perfectly fair systems: $R_1 = R_2 = \cdots = R_M$

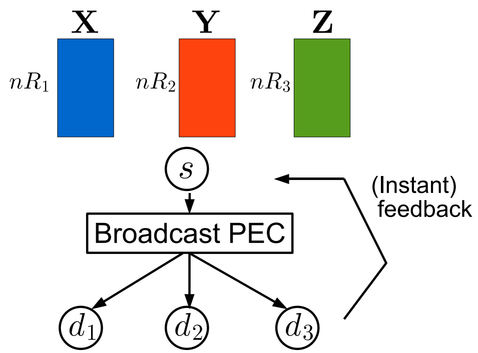
Sum rate capacity $\sum_{k=1}^{M} R_k$ vs. marginal success prob. p.





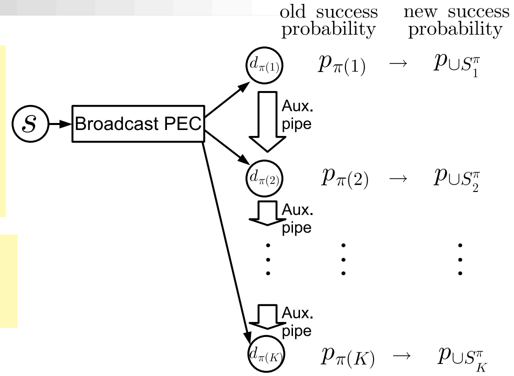
Corollary : When $M \to \infty$, the channel becomes effectively noiseless. [Larsson *et al.* 06]

Summary



Outer bound: $\forall \pi, \quad \sum_{k=1}^{M} \frac{R_{\pi(k)}}{p_{\cup S_k^\pi}} \leq 1.$

Tight for $M = 3$



Settings with general M>3 values	Capacity inner bound results
General $p_{S\overline{[M]\backslash S}}$	* A cap. Inner bound by using LP solvers to find the tightest time-slot packing * Numerically meets the outer bound for all our experiments
Spatially symmetric broadcast PECs $p_{S_1\overline{[M]\backslash S_1}} = p_{S_2\overline{[M]\backslash S_2}}$ if $\|S_1\| = \|S_2\|$	The inner and outer bounds always meet. → Full capacity region.
Spatially independent broadcast PECs $p_{S\overline{[M]\backslash S}} = \prod_{k\in S} p_k \prod_{j\in[M]\backslash S}(1-p_j)$	The inner and outer bounds meet when $(R_1, \cdots, R_M)$ are one-sided fair (when $R_1 \approx R_2 \approx \cdots \approx R_M$)

One-Sided Fairness

Definition 2 *The one-sidedly fair region Λ_{osf} contains all rate vectors $(R_1, \cdots, R_M)$ satisfying*

$$\forall i, j \text{ satisfying } p_i < p_j, \text{ we have } R_i(1 - p_i) \geq R_j(1 - p_j).$$

Remark 1: A perfectly fair vector $(R, \cdots, R)$ belongs to Λ_{osf}.

Remark 2: A proportionally fair vector $(p_1 R, \cdots, p_M R)$ belongs to Λ_{osf} if $\min\{p_k : \forall k \in [M]\} \geq 0.5$.