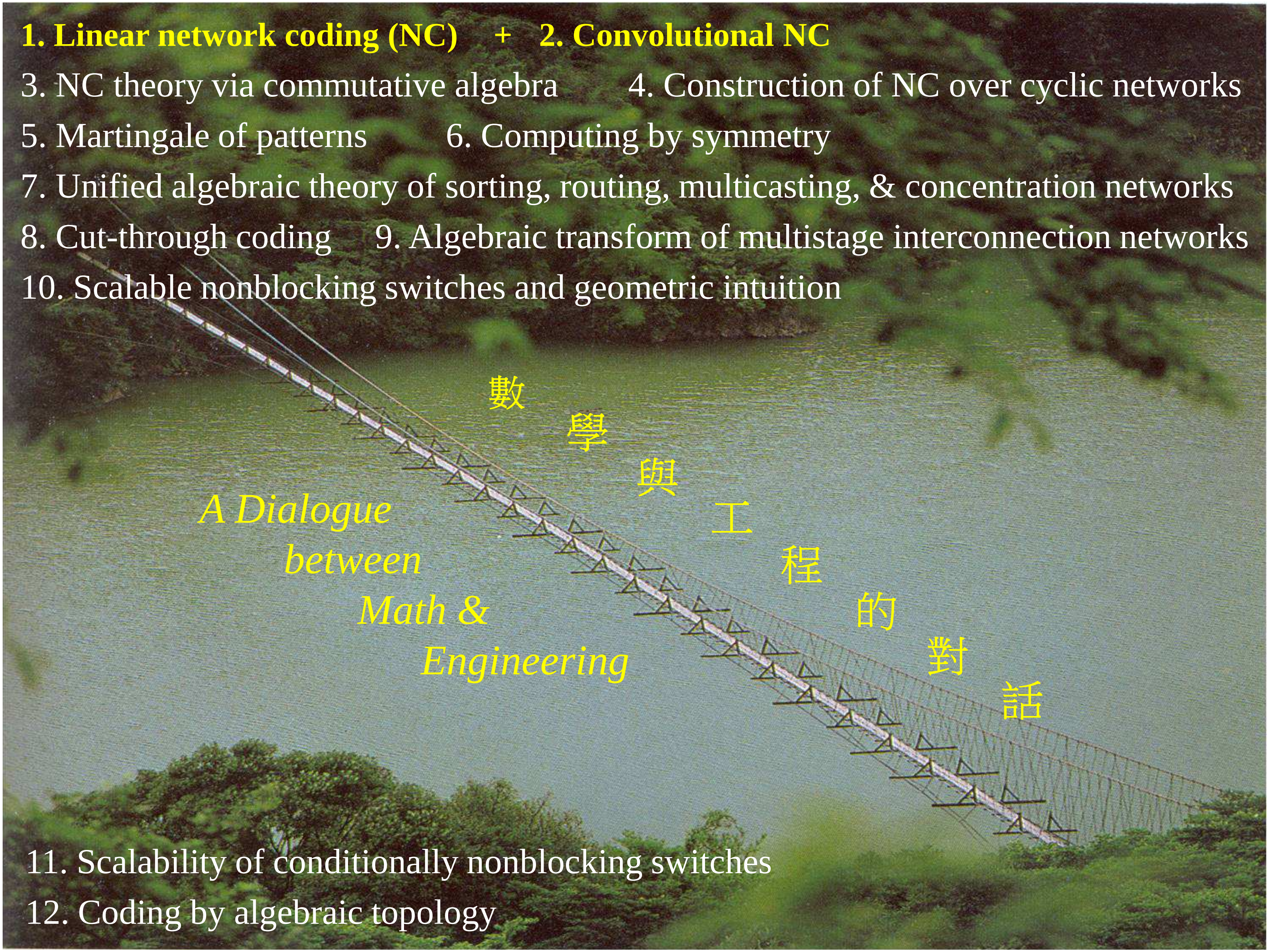
Acyclic vs. cyclic network coding

李碩彥 (Bob Li)
The Chinese University of Hong Kong
www.ie.cuhk.edu.hk/bobli

Playing "*The Butterfly Lovers*" melody


1. Linear network coding (NC) + 2. Convolutional NC
3. NC theory via commutative algebra
4. Construction of NC over cyclic networks
5. Martingale of patterns
6. Computing by symmetry
7. Unified algebraic theory of sorting, routing, multicasting, & concentration networks
8. Cut-through coding
9. Algebraic transform of multistage interconnection networks
10. Scalable nonblocking switches and geometric intuition
數學與工程的對話
A Dialogue between Math & Engineering
11. Scalability of conditionally nonblocking switches
12. Coding by algebraic topology

Store-and-forward

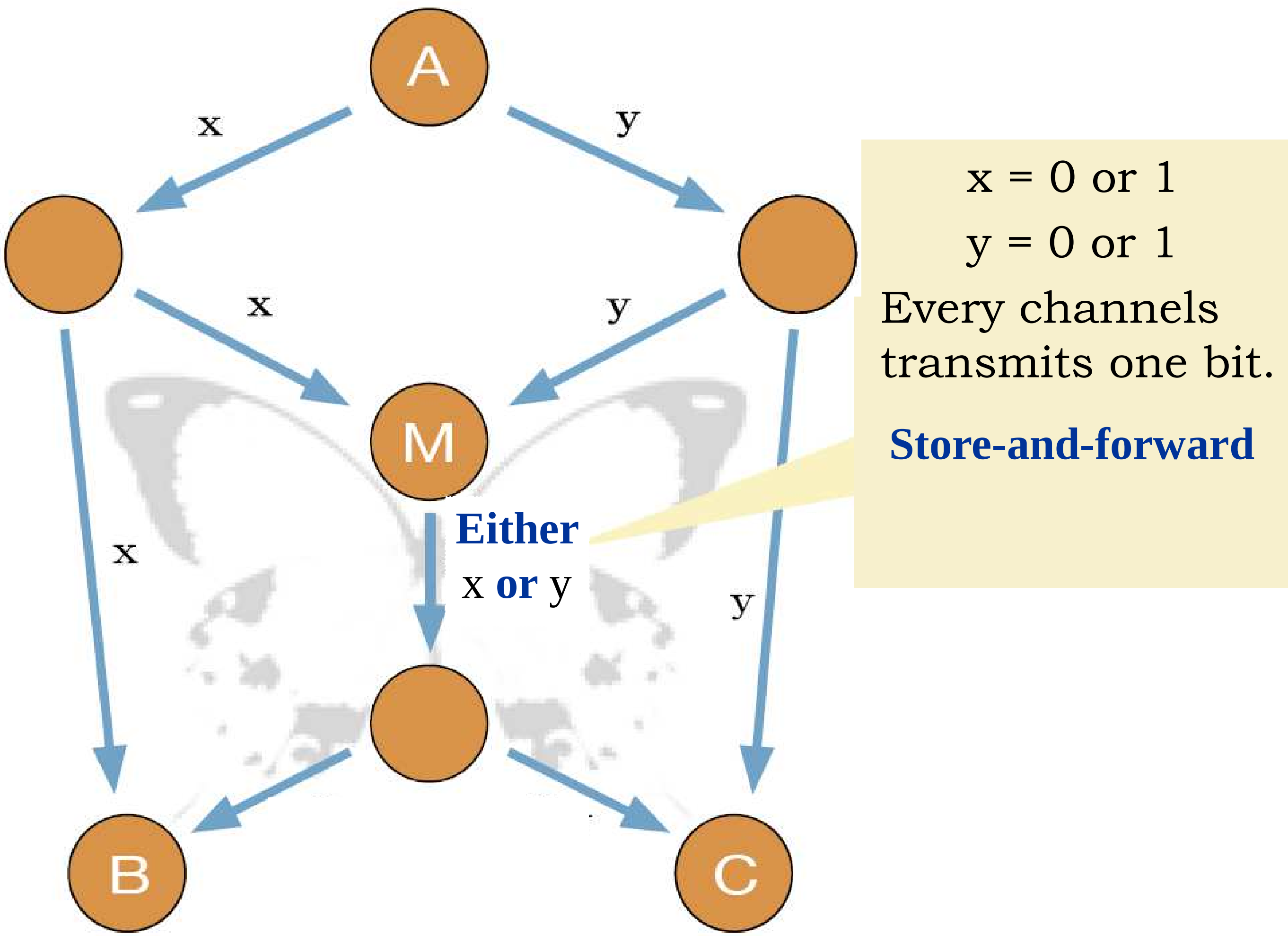


Figure adapted from *Scientific American*, Chinese 7/2007 edition

Store-and-forward

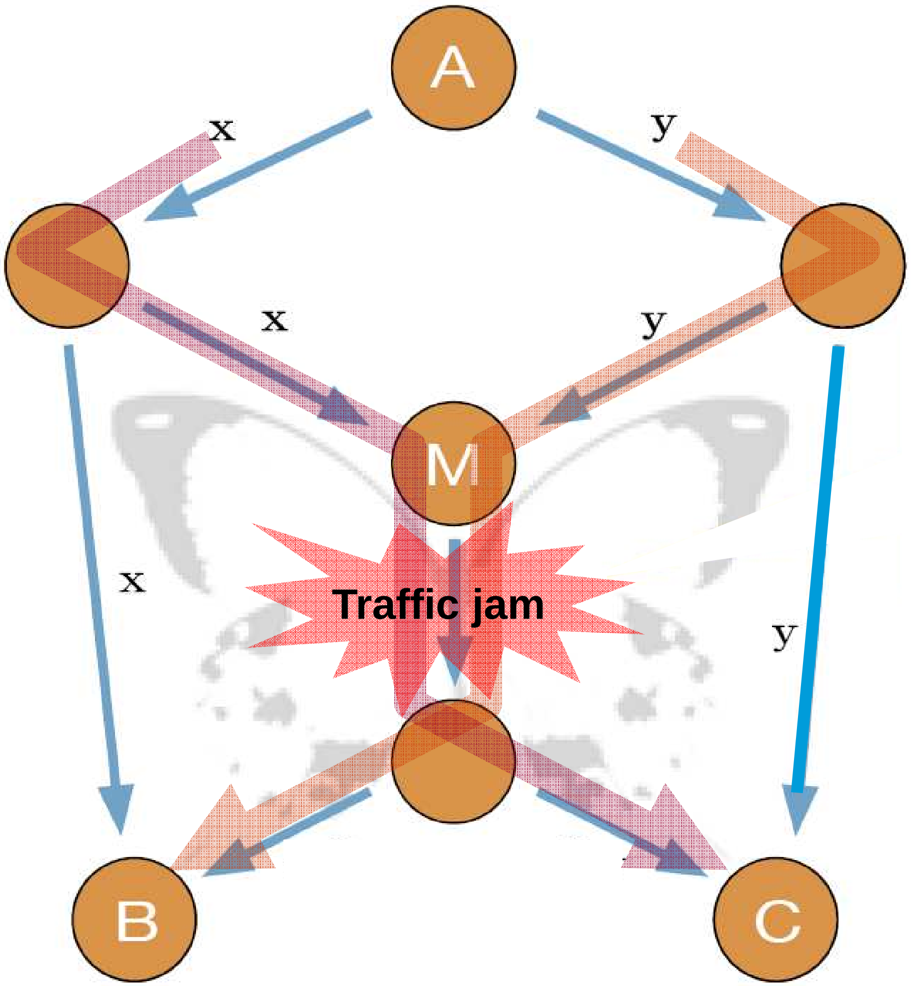


Figure adapted from *Scientific American*, Chinese 7/2007 edition

Network coding (NC)

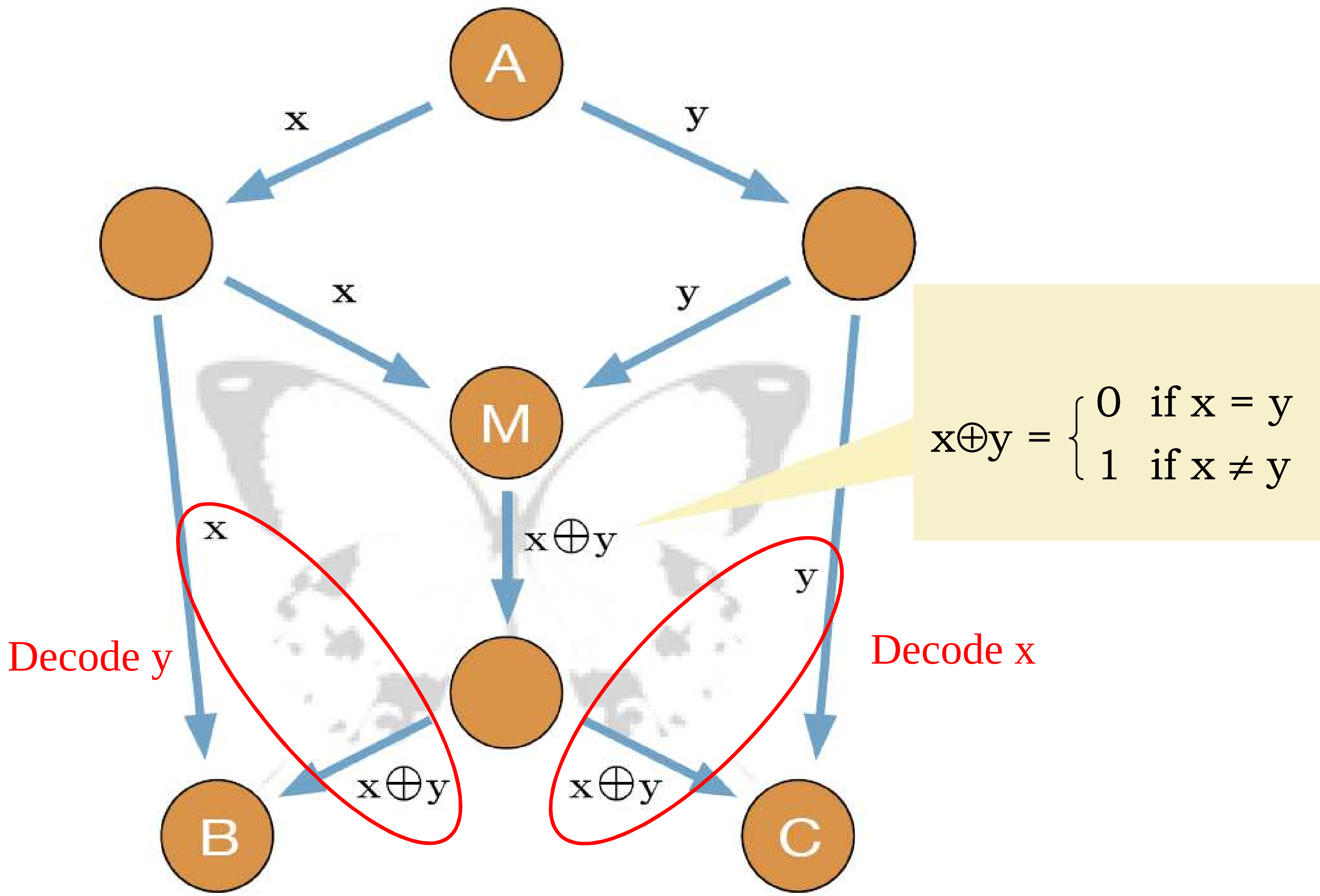
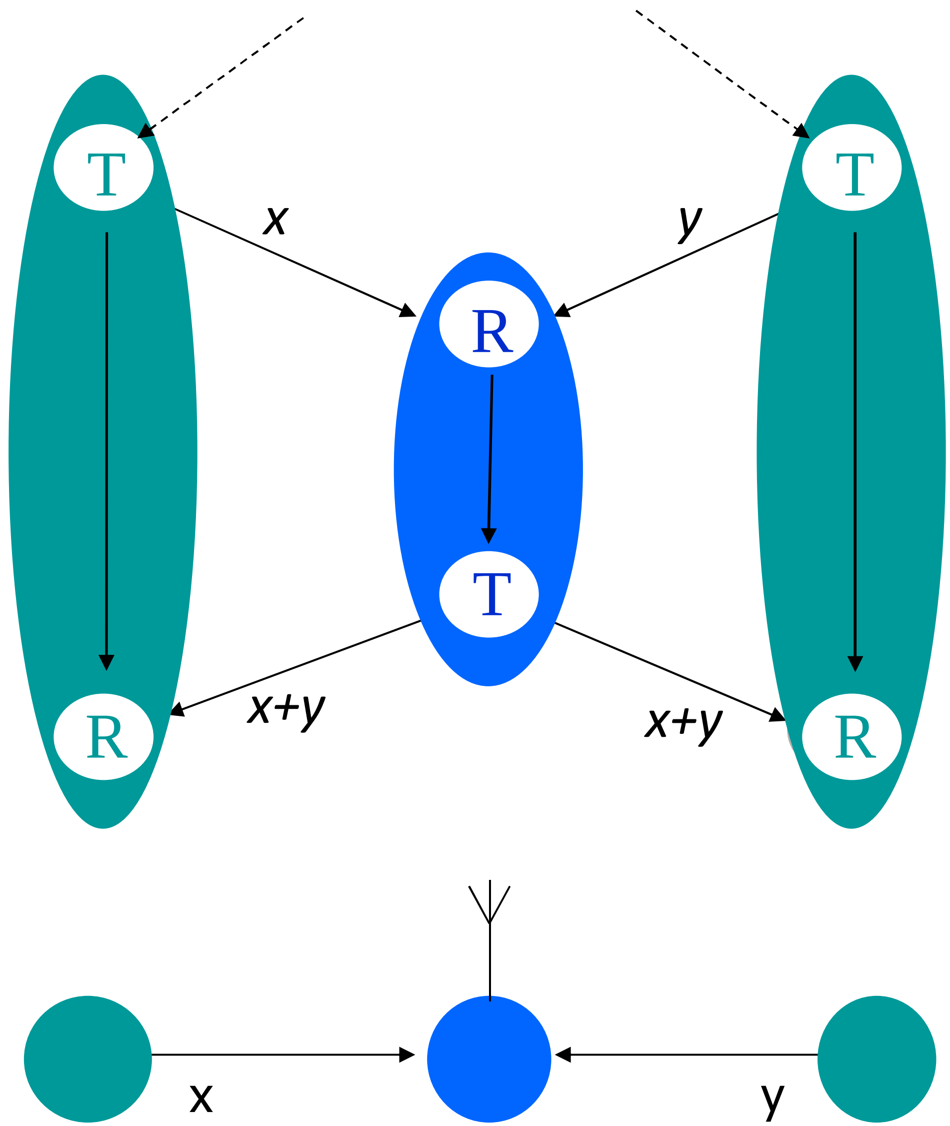


Figure adapted from *Scientific American*, Chinese 7/2007 edition

Applications

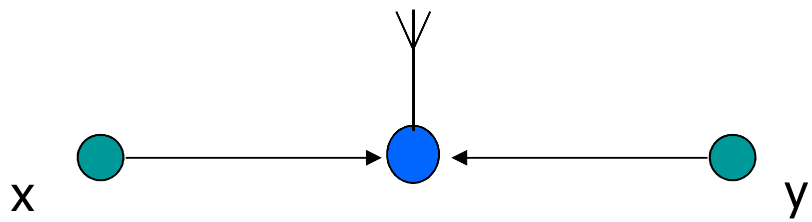
- Wireless communications
- P2P content delivery
- Redundant data storage
- Network security
- Optical multicast
- Network tomography, …

One interpretation of the Butterfly Network



Wireless application

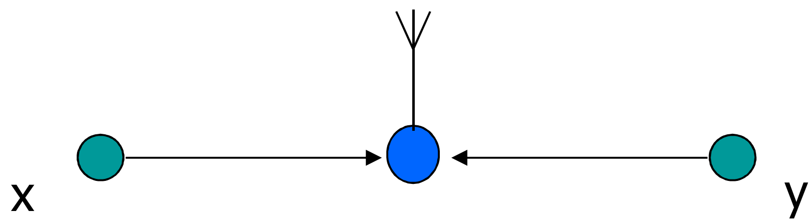
Wireless transmission is multicasting in nature, perfect for applying NC.



Store-and-forward, **4 steps** to exchange a message through the middle relay

Wireless application

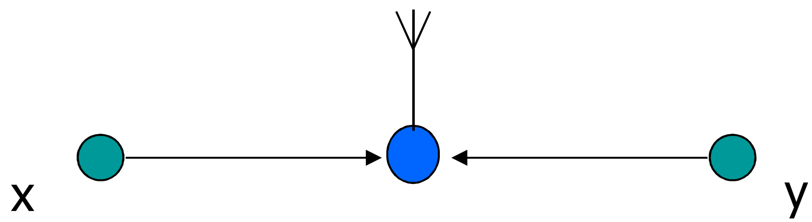
Wireless transmission is multicasting in nature, perfect for applying NC.



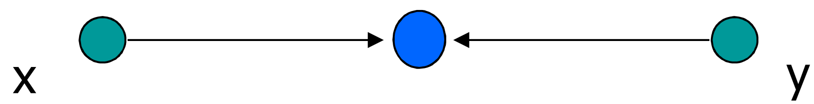
Store-and-forward, **4 steps** to exchange a message through the middle relay

Wireless application

Wireless transmission is multicasting in nature, perfect for applying NC.



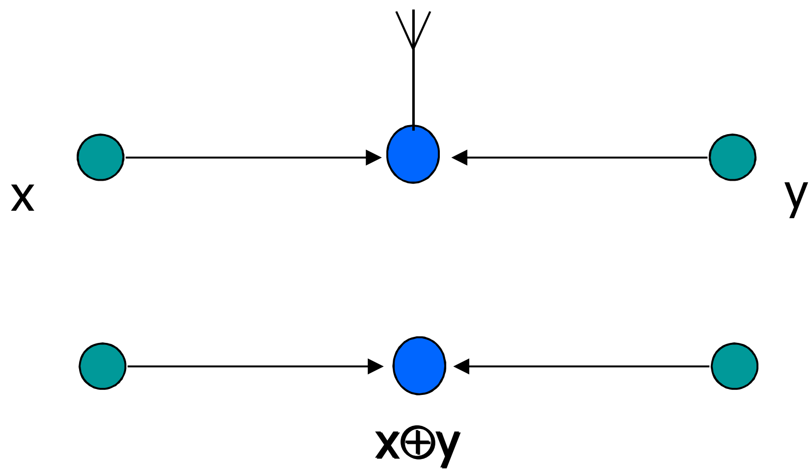
Store-and-forward, **4 steps** to exchange a message through the middle relay



NC, **3 steps**

Wireless application

Wireless transmission is multicasting in nature, perfect for applying NC.



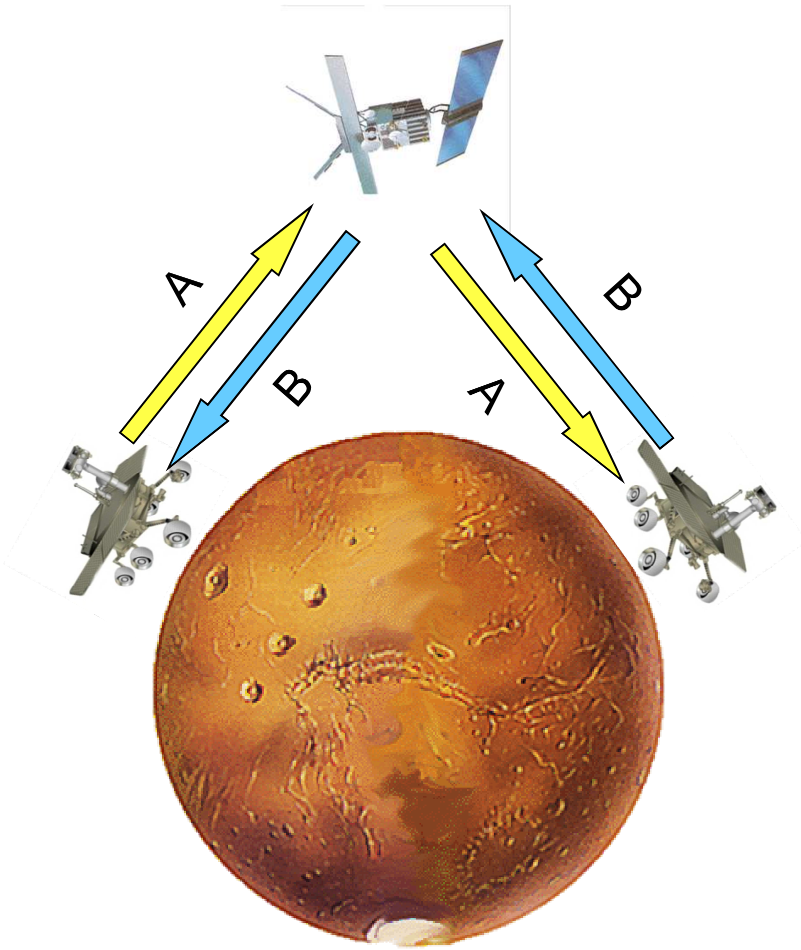
Store-and-forward, **4 steps** to exchange a message through the middle relay

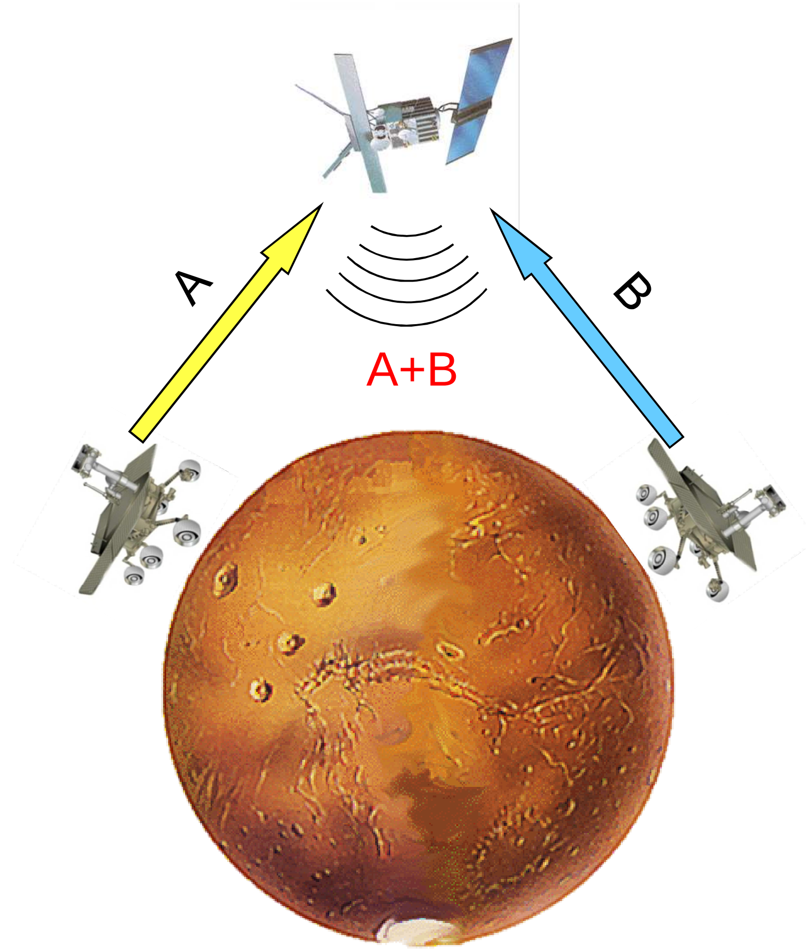
NC, **3 steps**

M.I.T. prototype
→ standard of wireless LAN
(802.11 Wi-Fi)

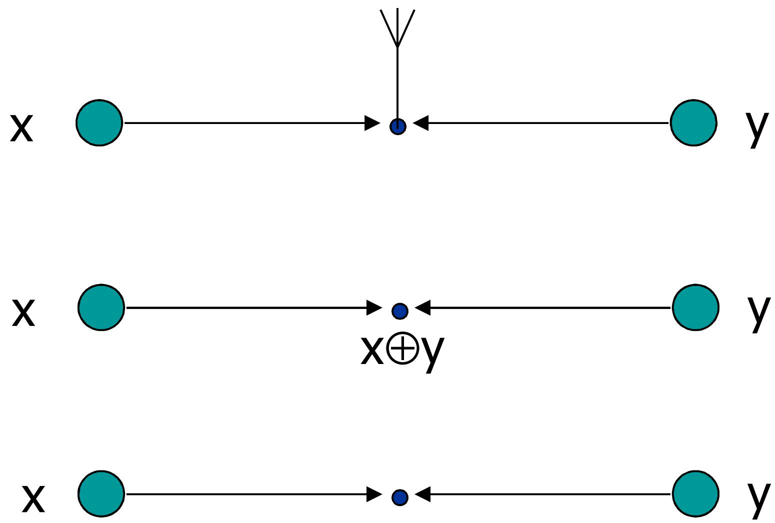
3GPP2 selects NC as its potential technology in 4G wireless systems.

Communications on Mars





Physical-layer NC (PNC)

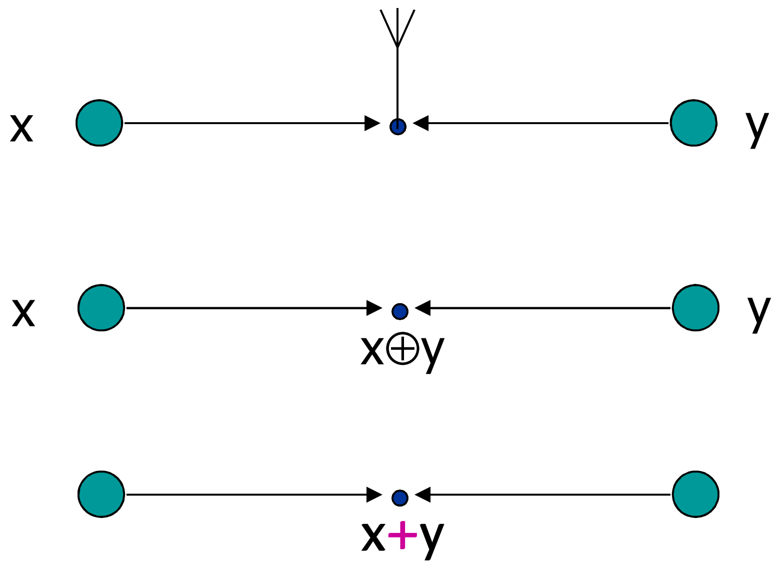


Store-and-forward, 4 steps

NC, 3 steps

2 steps ?

Physical-layer NC (PNC)



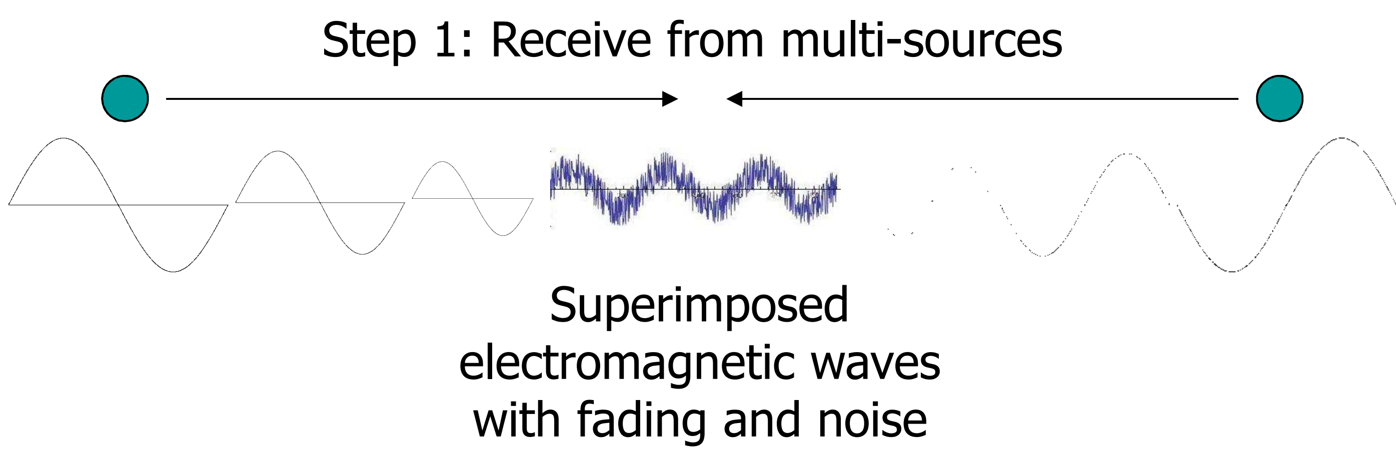
Store-and-forward, 4 steps

NC, 3 steps

Physical NC, 2 steps

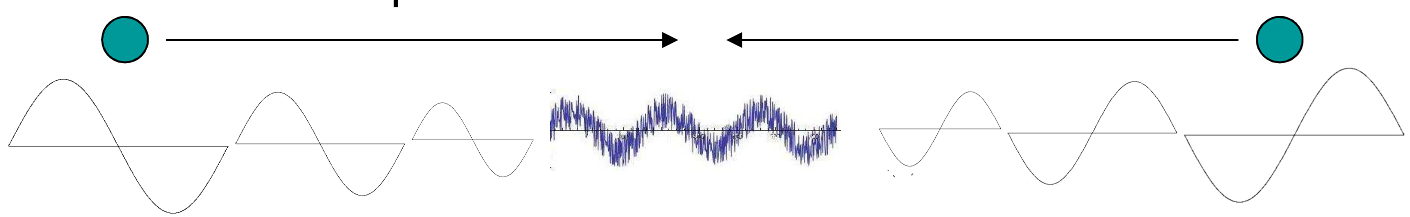
Physical-layer NC (PNC)

Interference is good → free higher throughput



Digital PNC [Zhang-Liew-Lam06]

Step 1: Receive from multi-sources



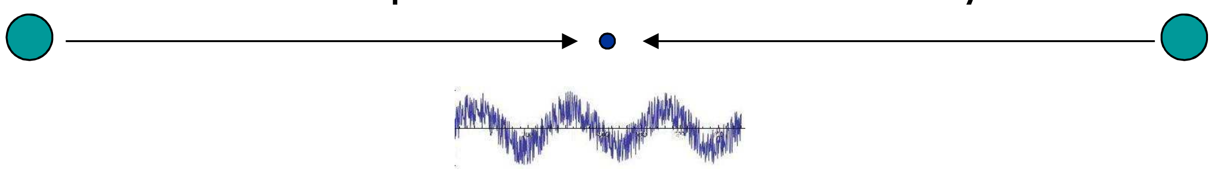
0/1 → ±Cos(ωt)　　　±2 Cos(ωt)　or　0　　　±Cos(ωt) ← 0/1

plus noise and fading

(assuming perfect sync)

Digital PNC [Zhang-Liew-Lam06]

Step 1a: Translation into binary



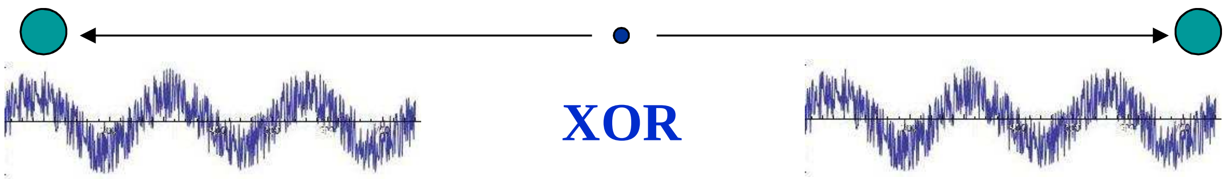
$0/1 \rightarrow \pm\mathrm{Cos}(\omega t)$ $\qquad$ $\pm 2\ \mathrm{Cos}(\omega t)$ or 0 (+ noise, fading) $\qquad$ $\pm\mathrm{Cos}(\omega t) \leftarrow 0/1$

(not easy)

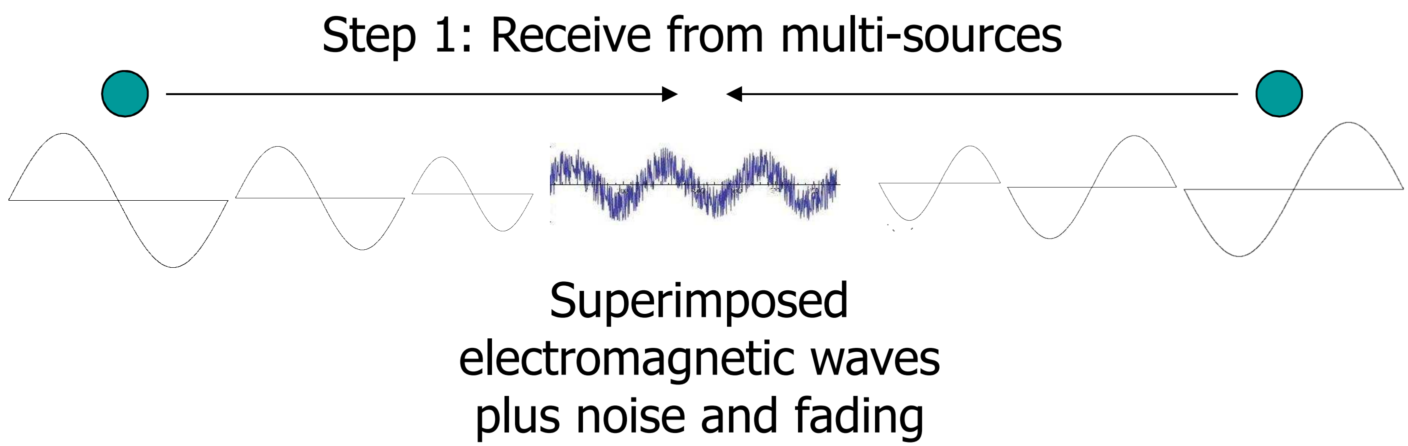
$0 \qquad 1$ (= XOR)

Digital PNC [Zhang-Liew-Lam06]

Step 2: Broadcast

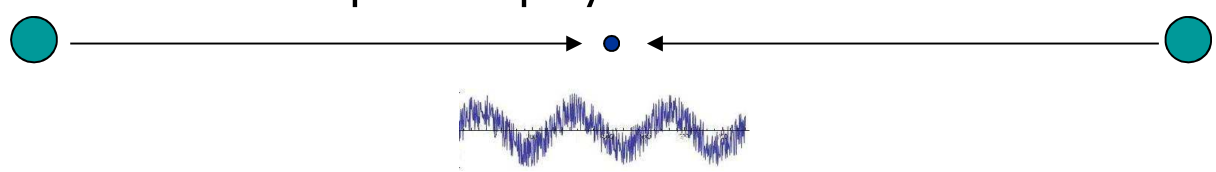


Analog PNC


Step 1: Receive from multi-sources
Superimposed
electromagnetic waves
plus noise and fading

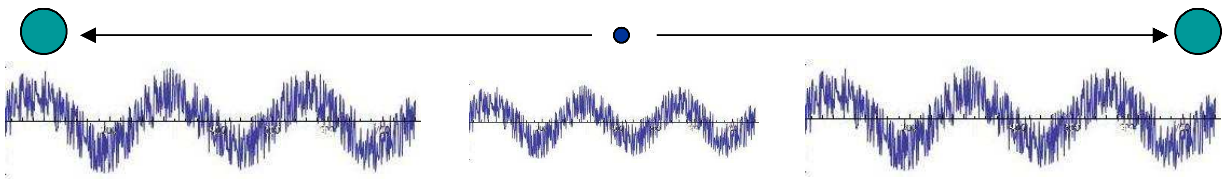
Analog PNC

Step 2: Amplify and broadcast



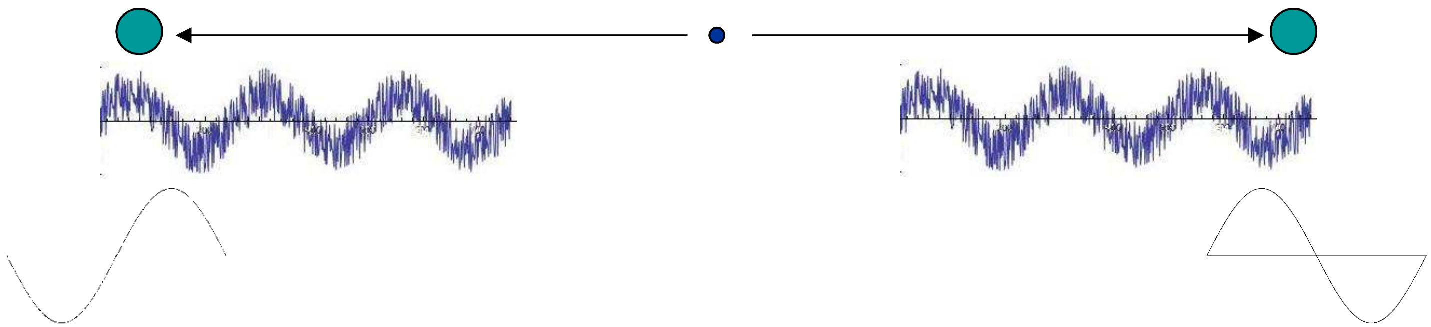
Analog PNC

Step 2: Amplify and broadcast



Analog PNC

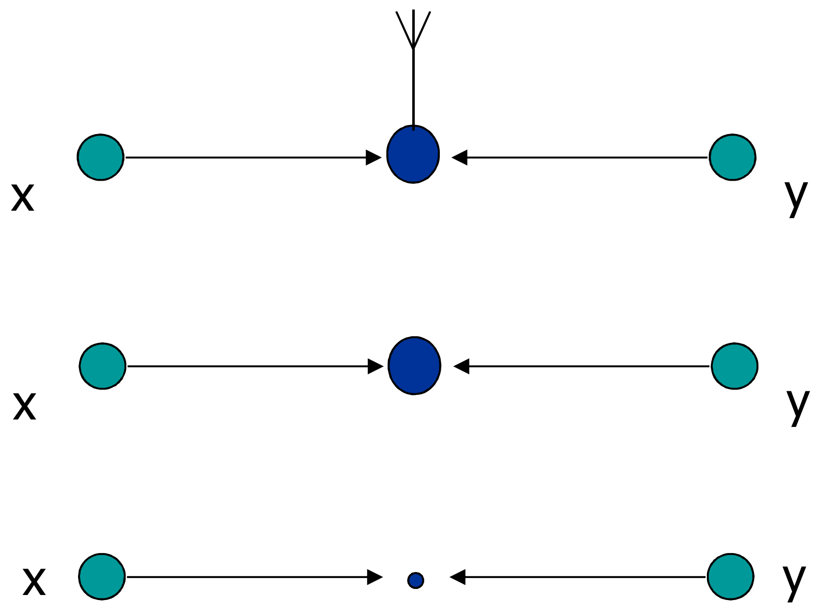
Step 2a : Decoding at each end (not easy)



PNC

7言诗

清明时节雨纷纷...



Store-and-forward, 4 steps

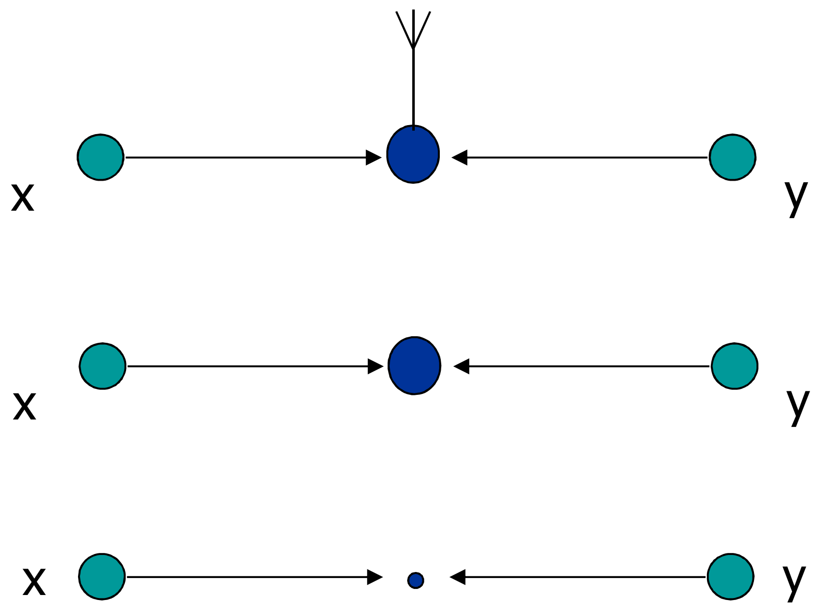
NC, 3 steps

Physical NC, 2 steps Decoding not easy

PNC

6言

清明时节雨纷纷...



Store-and-forward, 4 steps

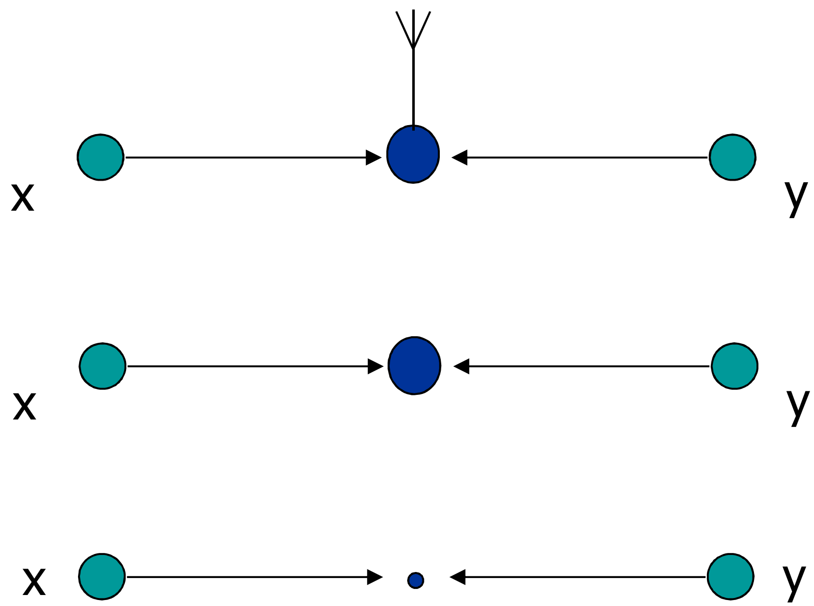
NC, 3 steps

Physical NC, 2 steps Decoding not easy

PNC

6言 5言

清明时节雨纷纷...

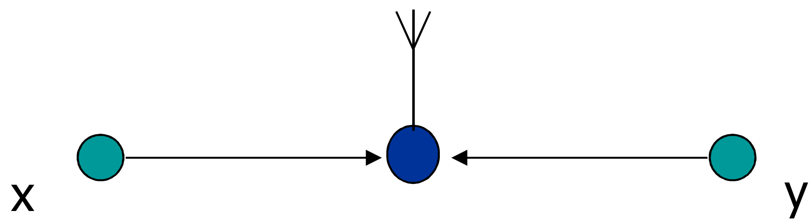


Store-and-forward, 4 steps

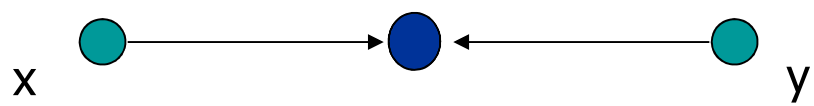
NC, 3 steps

Physical NC, 2 steps Decoding not easy

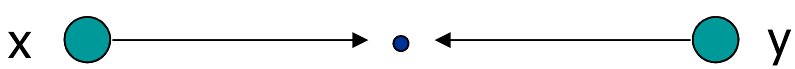
PNC 清明时节雨纷纷



4 清明雨纷，路人断魂。
酒家何处？指杏花村。

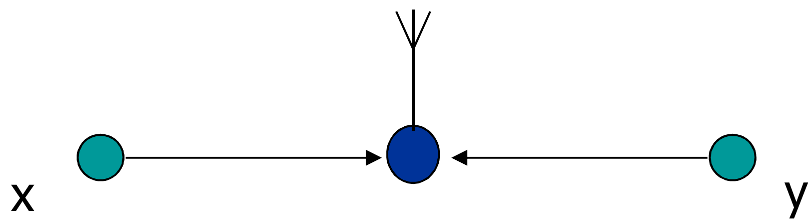


NC, 3 steps

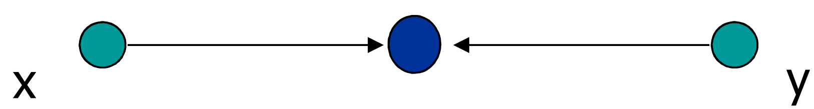


Physical NC, 2 steps Decoding not easy

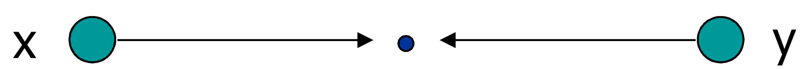
PNC 清明时节雨纷纷



4 清明雨纷，路人断魂。
酒家何处？指杏花村。

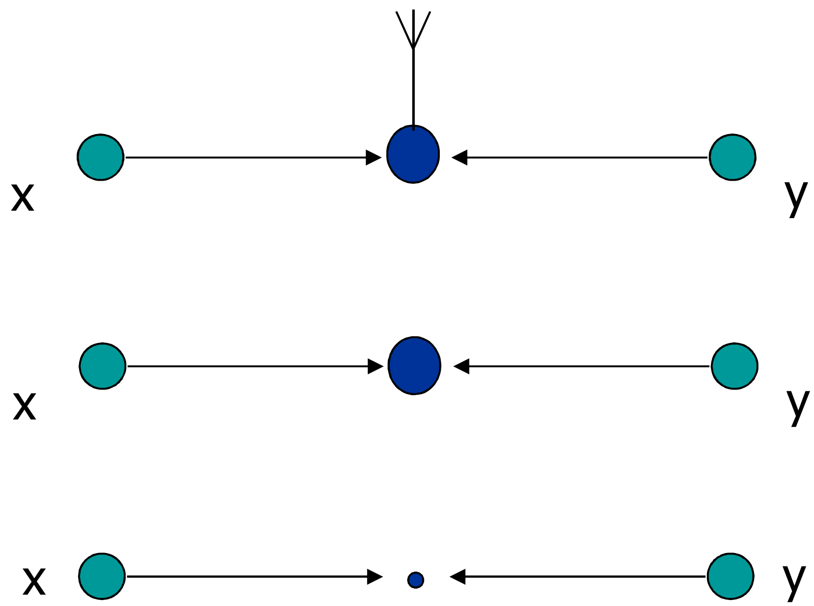


3 清明雨，人断魂。
酒何处？杏花村。



Physical NC, 2 steps Decoding not easy

PNC　　清明时节雨纷纷



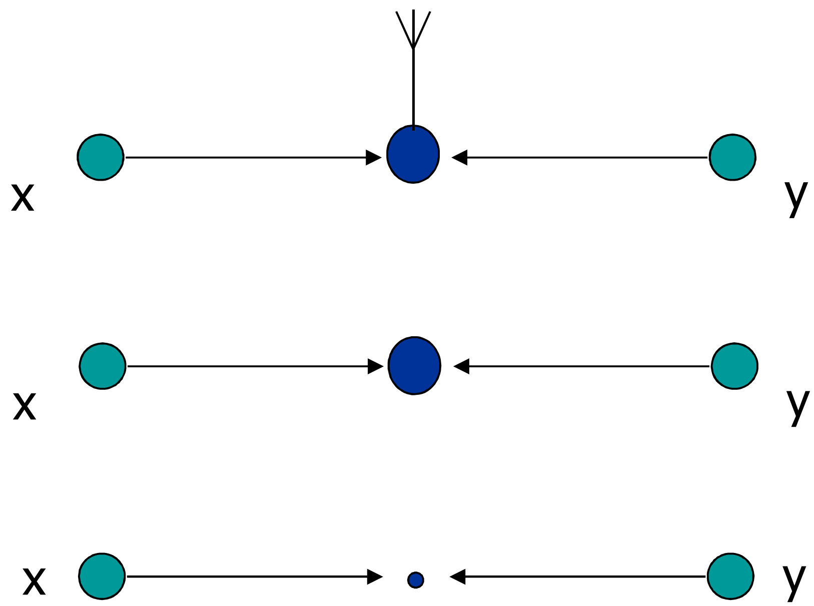
4 清明雨纷，路人断魂。
酒家何处？指杏花村。

3 清明雨，人断魂。
酒何处？杏花村。

2 清明，断魂。
酒处？杏村。

Decoding not easy

PNC　清明时节雨纷纷



4　清明雨纷，路人断魂。
酒家何处？指杏花村。

3　清明雨，人断魂。
酒何处？杏花村。

2　清明，断魂。
酒处？杏村。

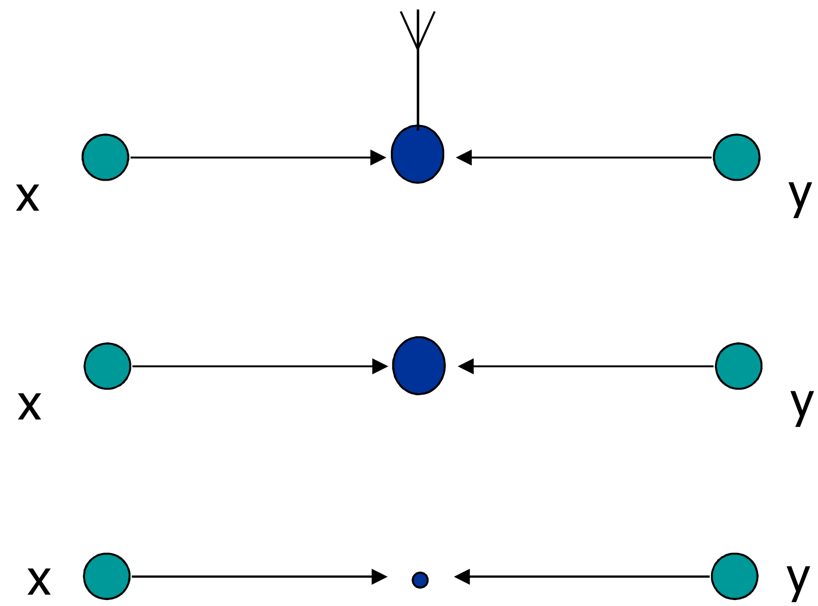
Decoding not easy

1　?

1　X

PNC

清明时节雨纷纷



4 清明雨纷，路人断魂。
酒家何处？指杏花村。

3 清明雨，人断魂。
酒何处？杏花村。

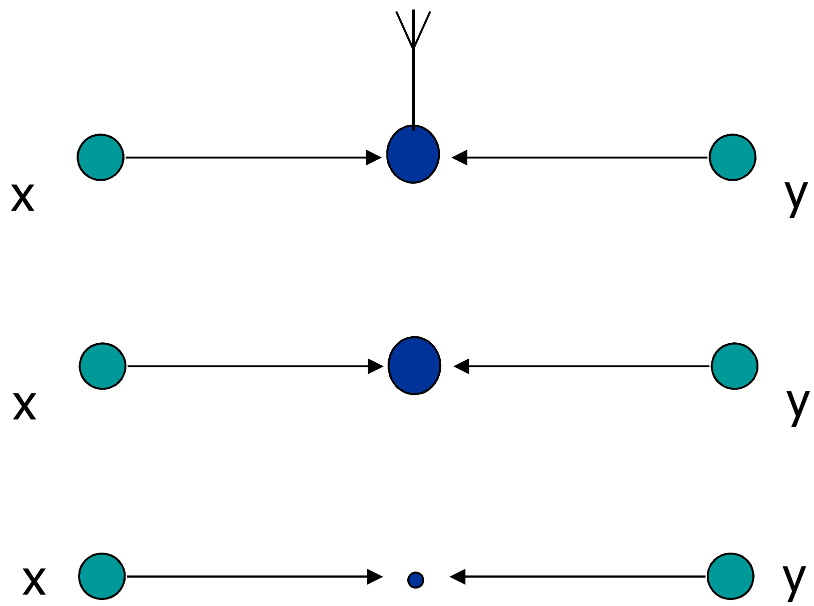
2 清明，断魂。
酒处？杏村。

Decoding not easy

1 Beam forming, MIMO antenna,
full-duplex transmission, …

1 X

PNC 清明时节雨纷纷



4 清明雨纷，路人断魂。
酒家何处？指杏花村。

3 清明雨，人断魂。
酒何处？杏花村。

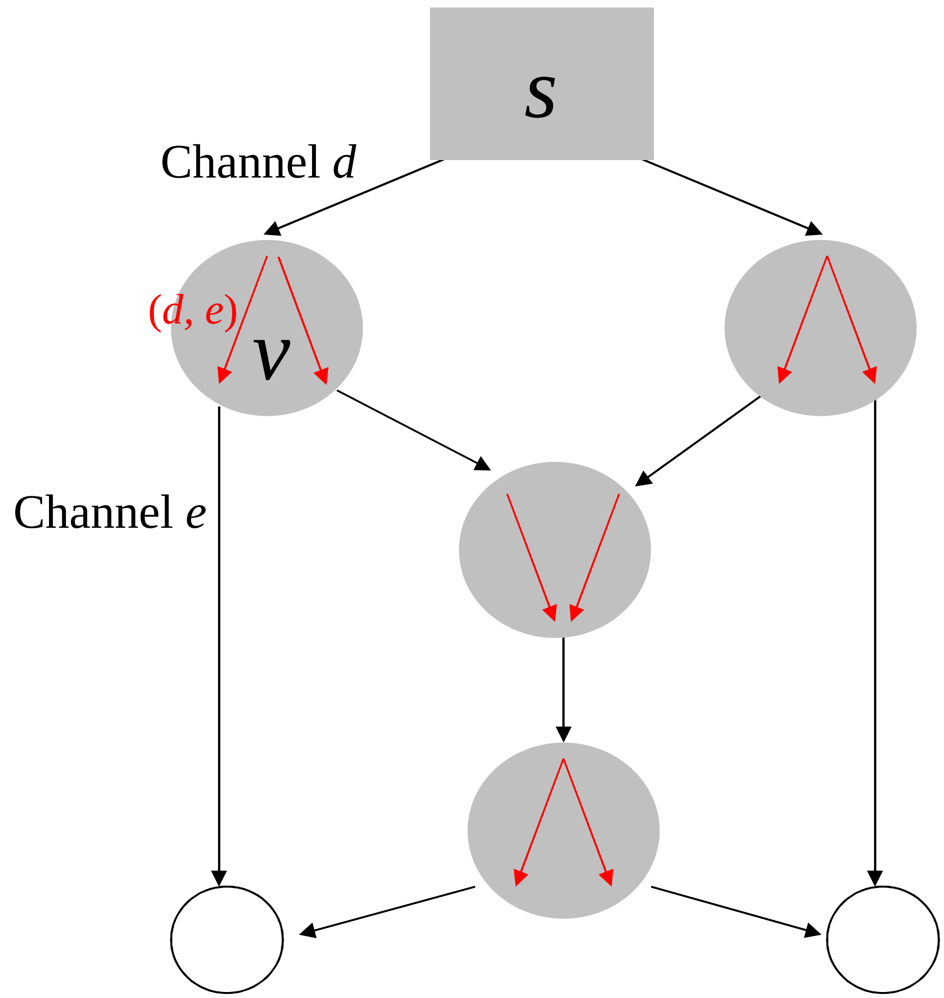
2 清明，断魂。
酒处？杏村。

Decoding not easy

Recall $n \rightarrow 4 \rightarrow 3 \rightarrow 2 \rightarrow 1$ transistors per memory bit.

Formulation of linear NC over a network

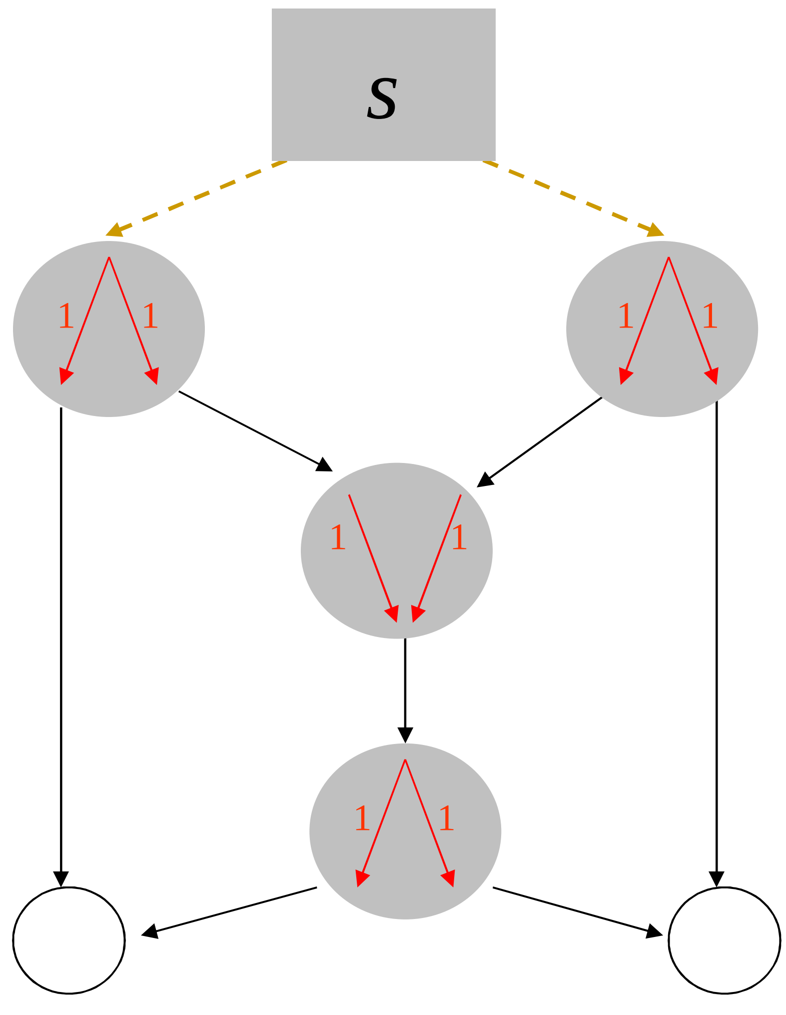
Definition. When channel *d* ends at the node where channel *e* begins, (*d*, *e*) is called an adjacent pair . It corresponds to a red arrow inside the joining node *v*.



Formulation of linear NC on a network

Definition. The data symbol alphabet is algebraically structured as a finite field $\mathbb{F}$.

An $\mathbb{F}$-linear network code assigns a coding coefficient $k_{d,e} \in \mathbb{F}$ to every adjacent pair (d, e).

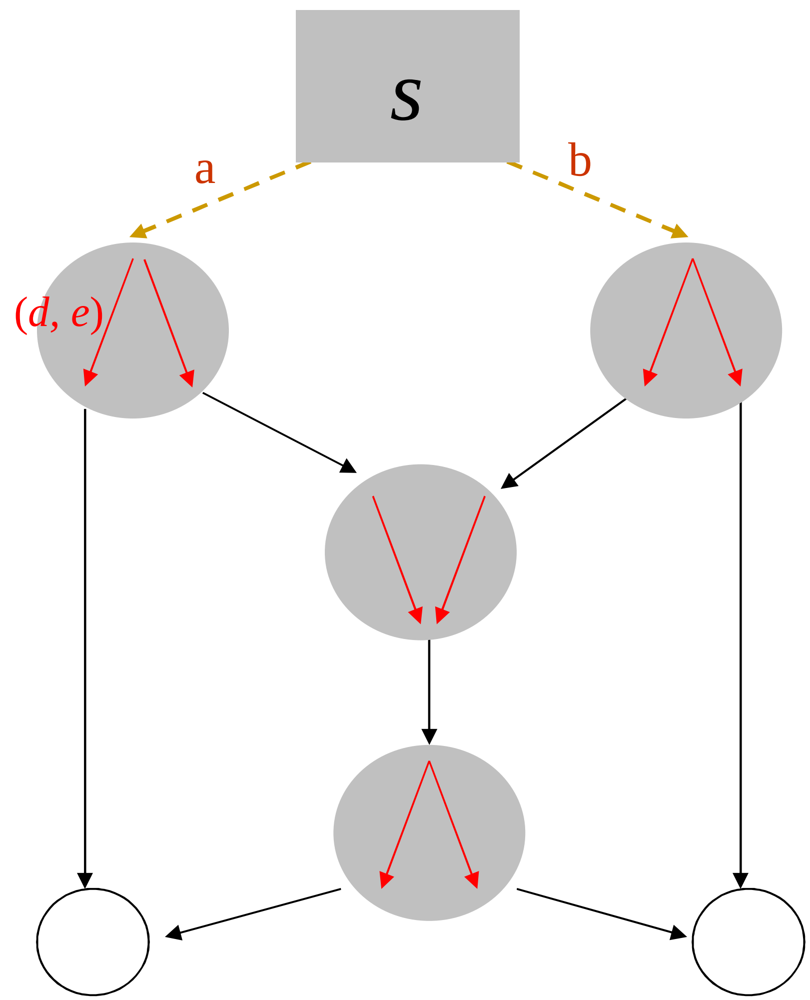


Formulation of linear NC over a network

Terminology. A channel is called an s-channel or a link depending whether its originating end is the source s.

There are ω s-channels.   // ω = 2 here

A message is an ω-dim row vector (a b) over $\mathbb{F}$, in which every data symbol is to be transmitted from a different *s-channel*.

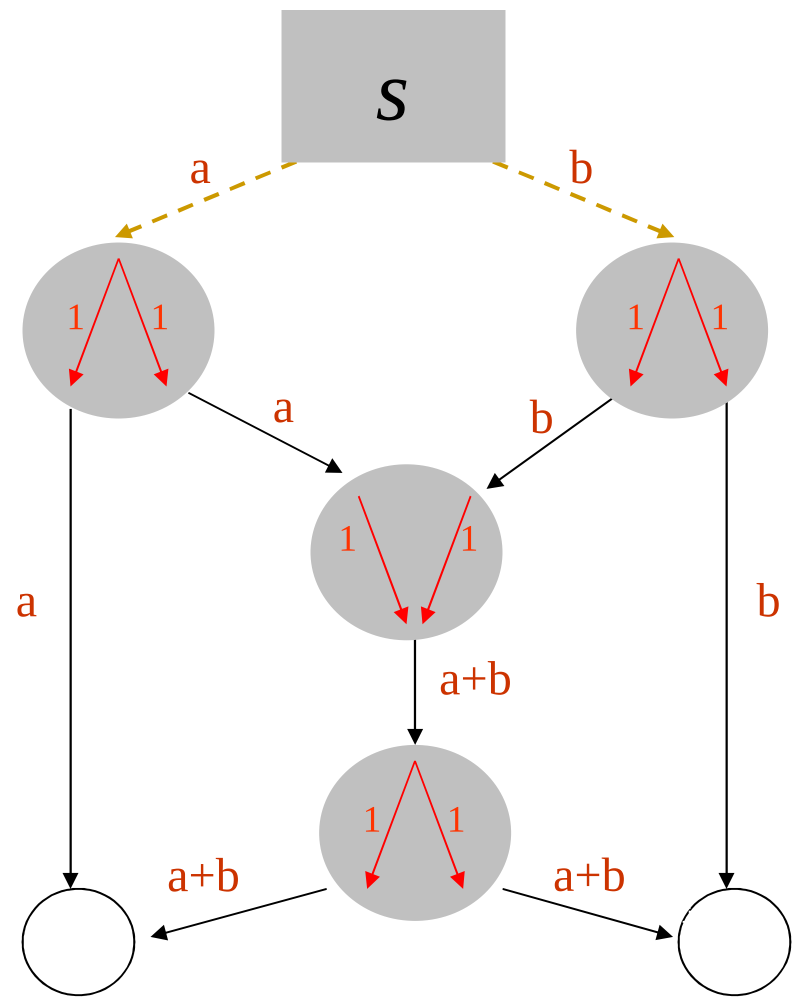


Formulation of LNC over an acyclic network

Assume that the network is acyclic.

Every intermediate node in data propagation makes a linear combination of incoming symbols for each of its outgoing *links*.

Through top-down telescoping, the symbol transmitted over every channel is a linear combination of message symbols.



Transmitted symbol = (message) $\cdot \begin{pmatrix} \text{coding} \\ \text{vector} \end{pmatrix}$

Assume that the network is acyclic.

The symbol transmitted over a channel e, being a linear combination of message symbols, can be written as

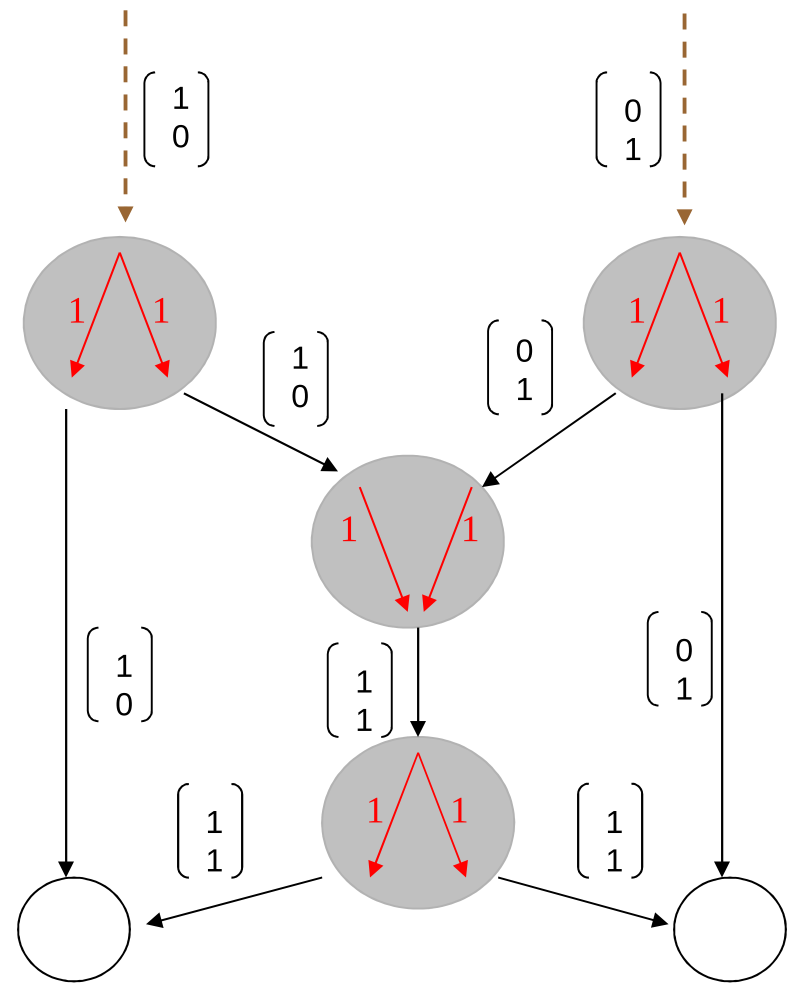
$$(\mathrm{a} \;\; \mathrm{b}) \cdot \boldsymbol{f_e}$$

where $\boldsymbol{f_e}$ is an ω-dim column vector, called the coding vector.

Initialization. Coding vectors of s-channels form the natural basis of $\mathbb{F}^{\omega}$.

Recursion. For an outgoing link e from node v,

$$\boldsymbol{f_e} = \Sigma_{d\in \mathrm{In}(v)}\, k_{d,e}\, \boldsymbol{f_d}$$



Decoding at node v

Juxtapose $\boldsymbol{f}_{tv}$ and $\boldsymbol{f}_{uv}$ into the matrix

$$\boldsymbol{M}_v = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$$

Given the message (a b) from the source, symbols received by node v form the row vector

$$(a \;\; a+b) = (a \;\; b) \cdot \boldsymbol{M}_v$$

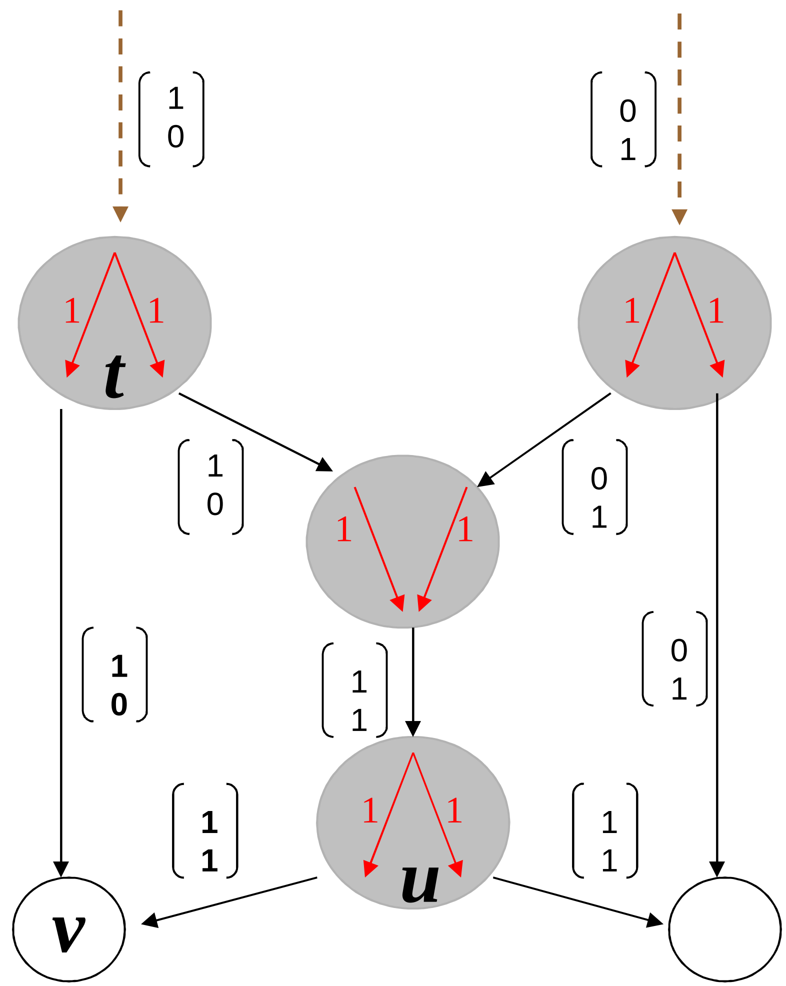
The message is then decoded by:

$$(a \;\; b) = (a \;\; a+b) \cdot \boldsymbol{M}_v^{-1}$$

Conditions for decodability at v:

① Incoming coding vectors span the full rank ω (= 2 here).

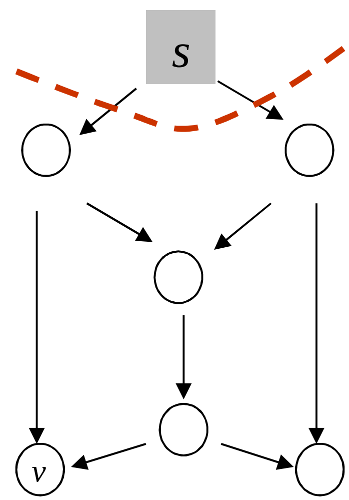
② The node v knows the incoming encoding vectors.



① Do incoming vectors span full rank?

V_v = vector space spanned by incoming coding vectors to node v

$\dim(V_v)$ = Information rate from s to node v
$\leq$ maxflow(v) = the max flow from s to v
= the min cut between s and v
$\leq$ the cut beneath s
= the full rank ω of message



Intrinsic limitation on information rate

Linear dependence among coding vectors are constrained:

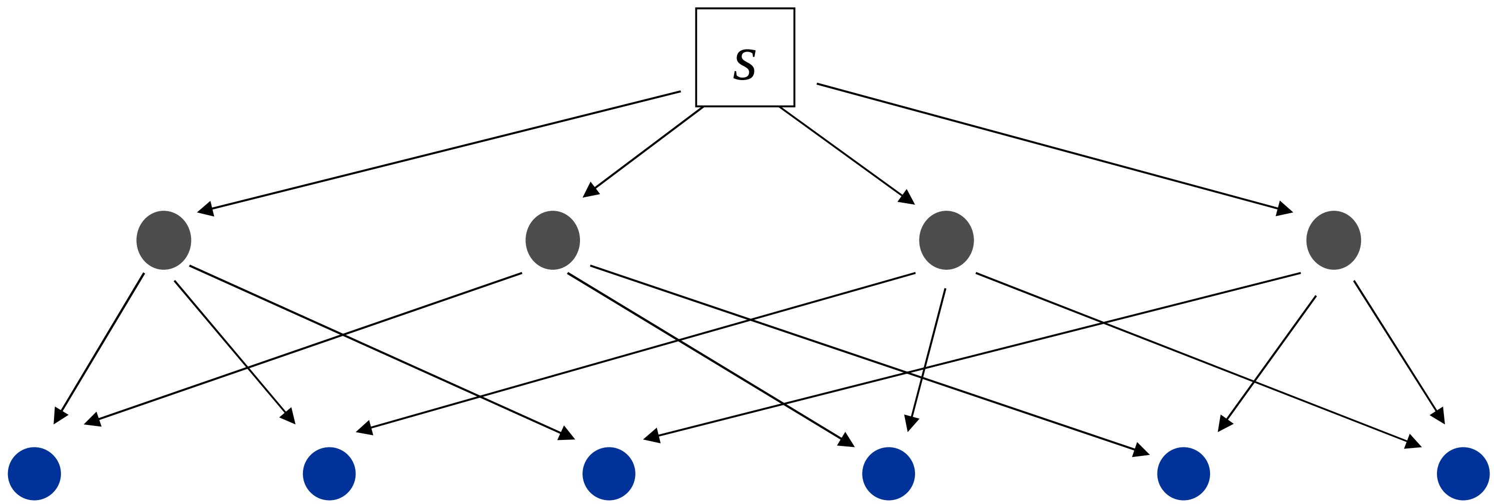
- $\dim(V_v) \le \omega$ ——by the network topology
- A subtle constraint —— by the choice of the symbol field $\mathbb{F}$

Choice of the symbol field $\mathbb{F}$ matters.

Q. Can two symbols be transmitted from *s* to all six receivers?

// Here maxflow of every receiver = 2 = ω.

- **No**, when a symbol = a bit.
- **Yes**, when a symbol = a byte.



Intrinsic limitation on information rate

Linear dependence among coding vectors are:

- $\dim(V_v) \le \omega$ ——— by the network topology
- ~~A subtle constraint ——— by the choice of the symbol field $\mathbb{F}$~~

  The fundamental theorem of linear NC finesses this constraint by assuming large enough $|\mathbb{F}|$ so as to guarantee the existence of an “optimal network code.”

Optimal network codes

Definition. An ω-dim $\mathbb{F}$-valued network code on an acyclic network qualifies as:

- a linear multicast when

$$\text{maxflow}(v) = \omega \quad \Rightarrow \quad \dim(V_v) = \omega$$

Good enough for most applications

- a linear broadcast when $\dim(V_v) = \text{maxflow}(v)$ for every node v

Extra application to scalable video coding (SVC)

Optimal network codes

Definition. An ω-dim $\mathbb{F}$-valued network code on an acyclic network qualifies as:

- a linear multicast when $\dim(V_v) = \omega$ for every node v with maxflow(v) = ω

$$\not\Downarrow \quad \Uparrow$$

- a linear broadcast when $\dim(V_v) = \text{maxflow}(v)$ for every node v

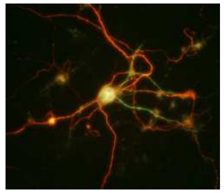
$$\not\Downarrow \quad \Uparrow$$

- a linear dispersion when $\dim\langle \cup_{v\in \wp} V_v \rangle = \text{maxflow}(\wp)$ for every collection $\wp$ of nodes

$$\not\Downarrow \quad \Uparrow$$

- a generic network code

Tremendous impact

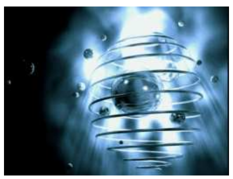


Intracellular Communications

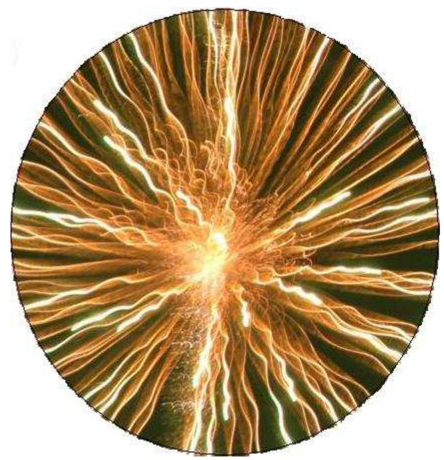
Information theory

Channel coding

Wireless networks



Quantum information theory

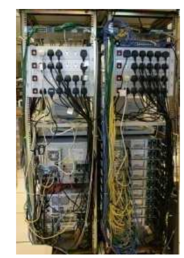


Graph theory

Computer networks

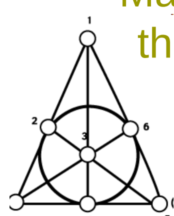
Optimization theory

Switching theory



Game theory

Matroid theory



Cryptography

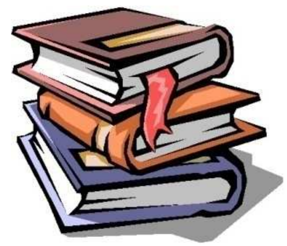
Computer science

Research impact of NC

PUBLICATIONS

Papers (3000+)

Books (4)



Journal special issues (7)

BEST PAPER AWARDS (6)
(2 by AoE members)

ANNUAL CONFERENCES (2)

NETCOD

WiNC

Acyclic networks vs. cyclic networks

Theorem of Linear NC is for acyclic networks only.

But, engineering applications all ignore the acyclic restriction.

How do the 3000+ papers justify 10-pound application out of a 5-pound theorem?

- By sheer engineering instinct. Great!

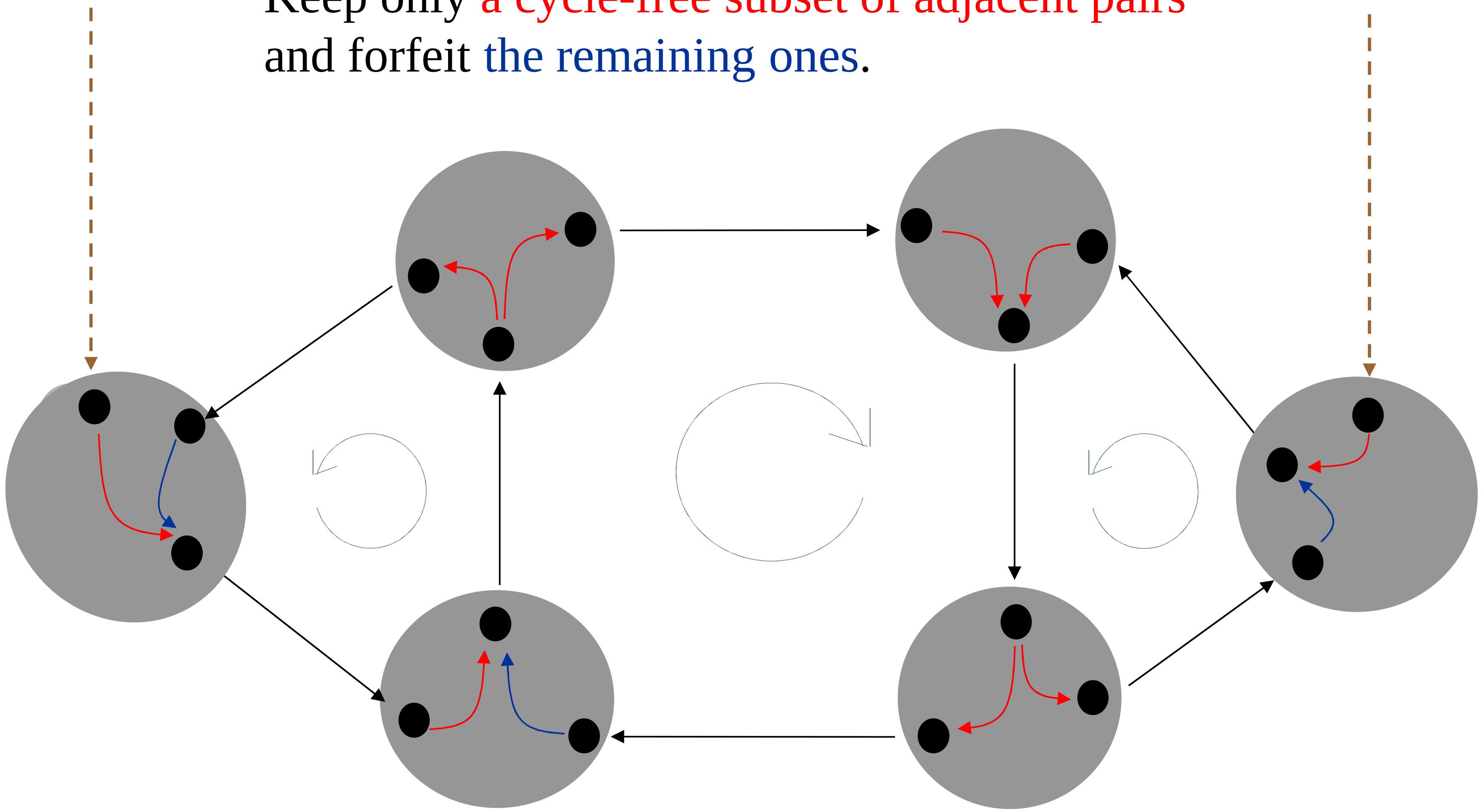
Here we shall investigate mathematically.

Convolutional NC

- **Acyclic networks vs. cyclic networks**
- Finiteness in implementation
- Causality in data propagation
- Existence and uniqueness of coding vectors

Can we amputate some adjacent pairs to make it acyclic?

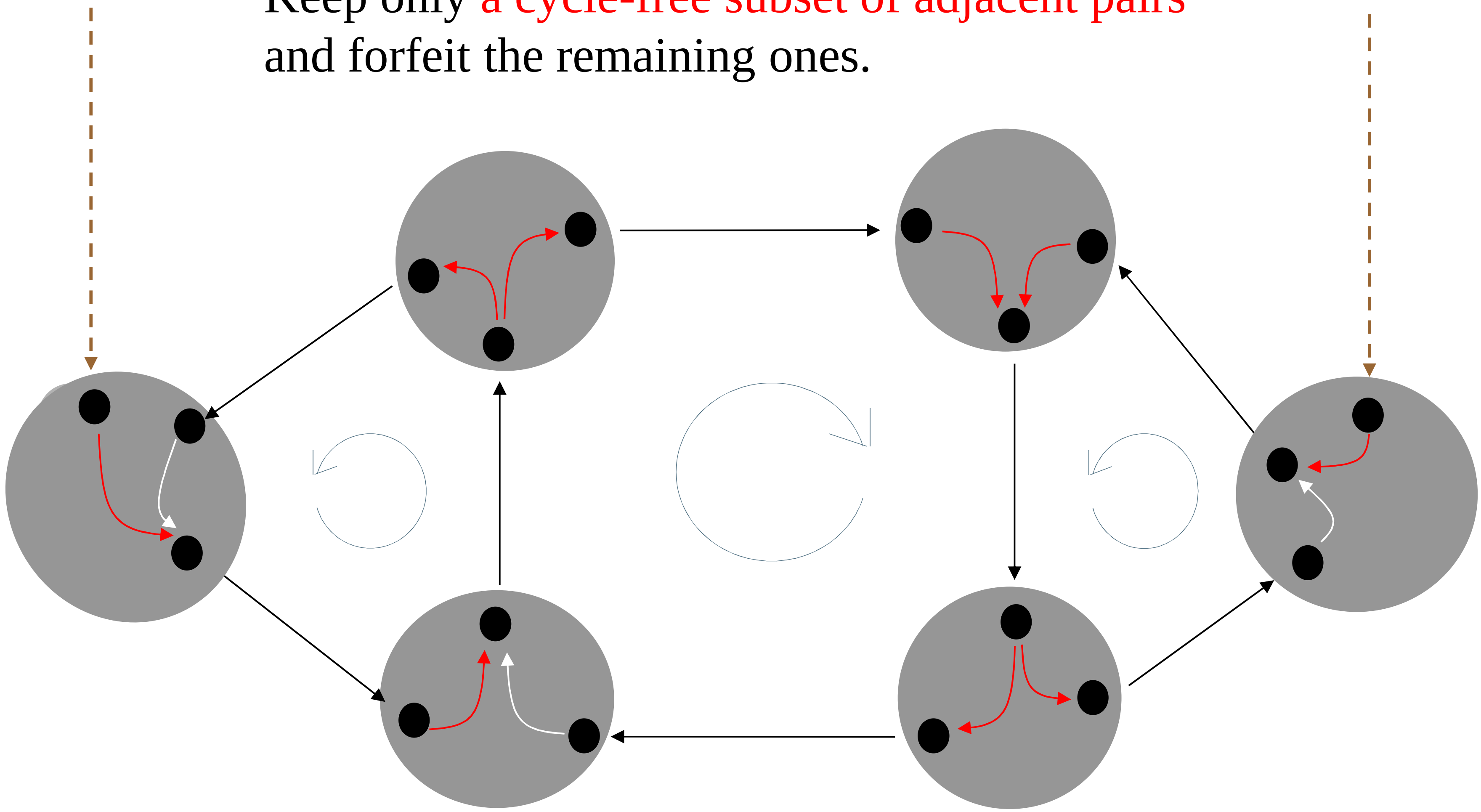
Keep only a cycle-free subset of adjacent pairs
and forfeit the remaining ones.



The *Shuttle Network*

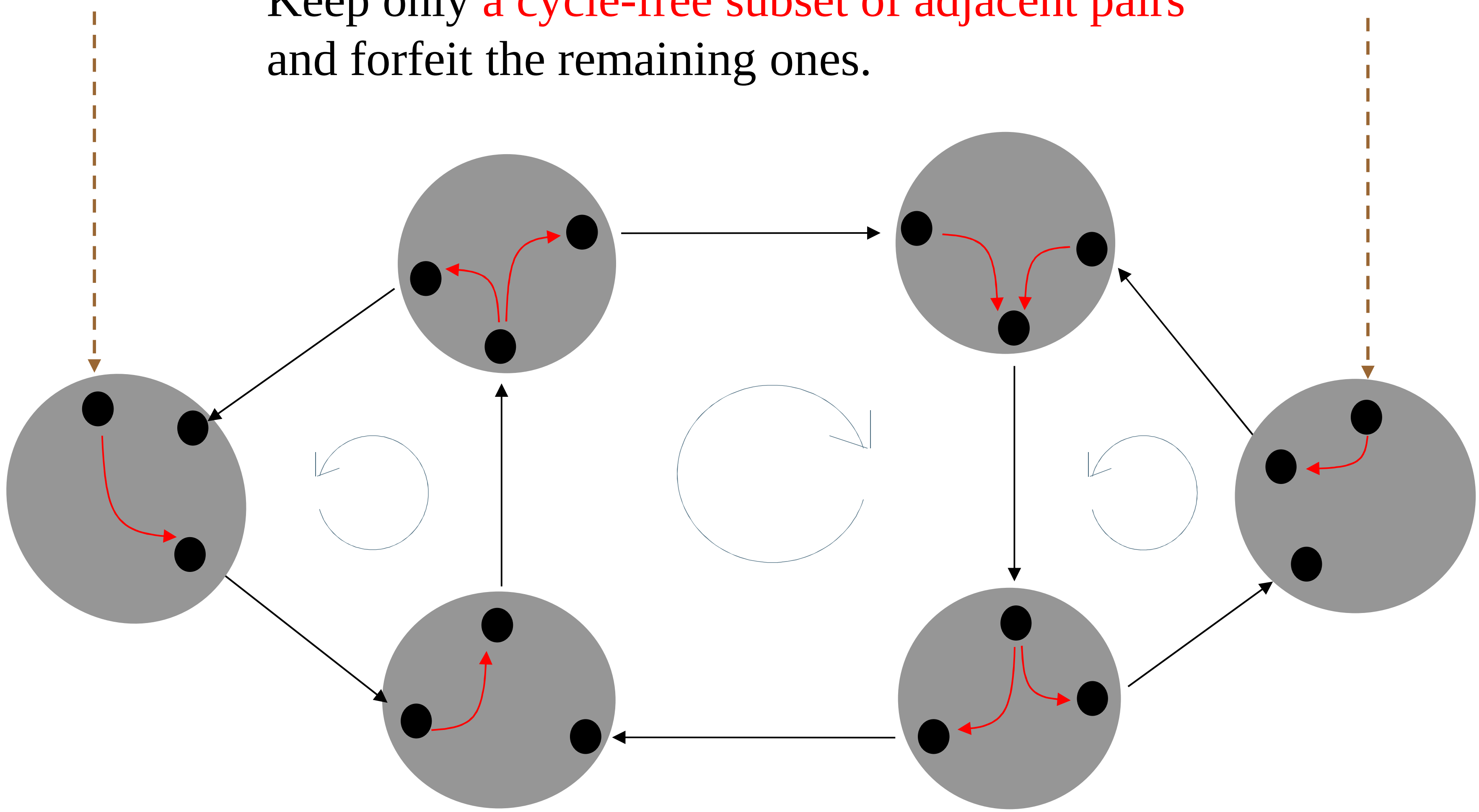
Artificial reduction of 3 adjacent pairs

Keep only a cycle-free subset of adjacent pairs
and forfeit the remaining ones.



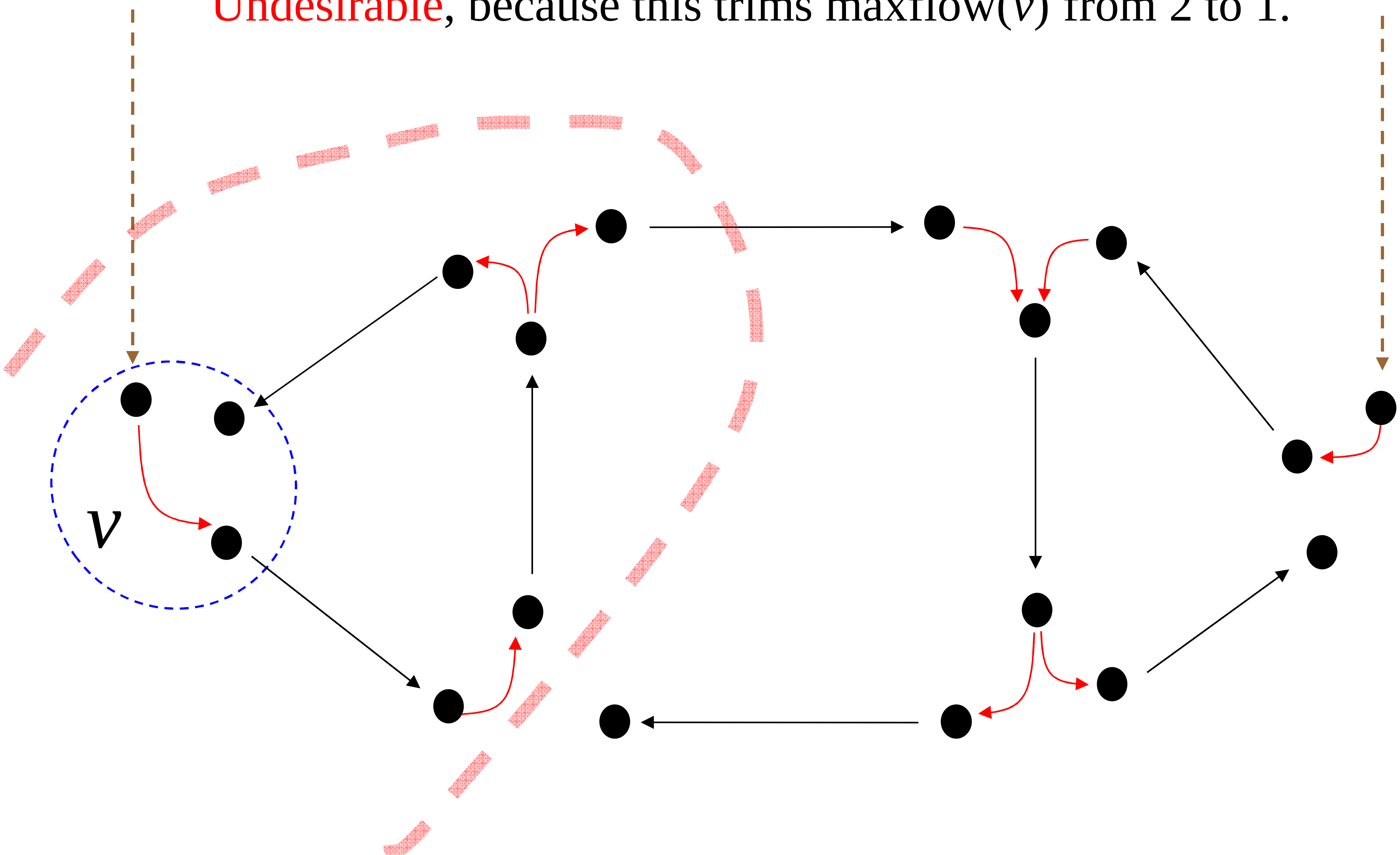
Artificial reduction of 3 adjacent pairs

Keep only a cycle-free subset of adjacent pairs
and forfeit the remaining ones.



Artificial reduction of 3 adjacent pairs

Undesirable, because this trims maxflow(v) from 2 to 1.



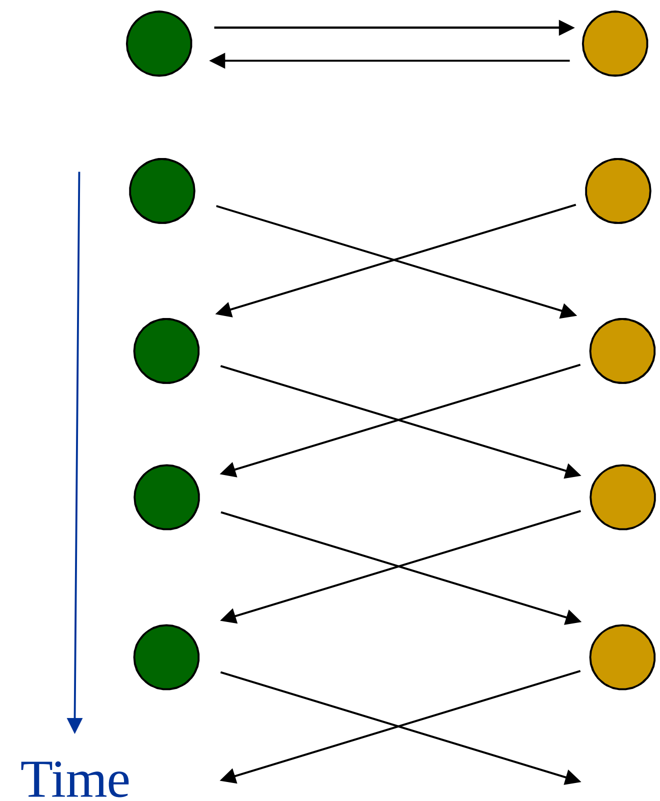
Time-multiplexed deployment of a network

A better justification for applying Thm of NC to cyclic networks

Every time-multiplexed channel carries a pipeline of data symbols.

Unfolding the multiplexing w.r.t. time yields a trellis network in the combined space-time domain.

It is acyclic because time is unidirectional.



Trellis network

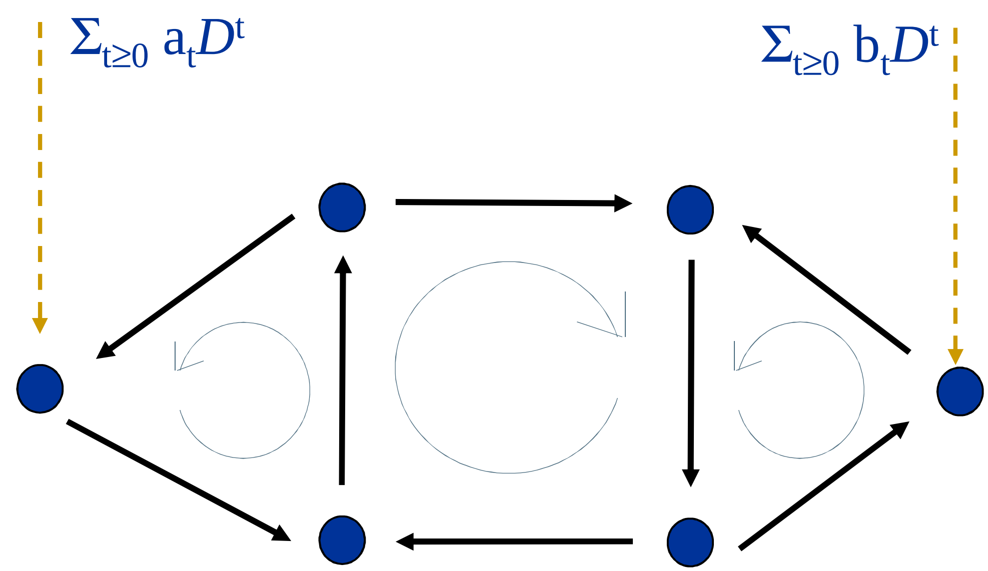
Message pipelining over a network

A pipeline of symbols $(a_0, a_1, \ldots, a_t, \ldots)$

$\leftrightarrow$ A power series $\Sigma_{t\geq 0}\, a_t D^t \in \mathbb{F}[[D]]$ // D = unit-time delay

A pipeline of messages

$\leftrightarrow$ An ω-dim row vector $(\Sigma_{t\geq 0}\, a_t D^t \quad \Sigma_{t\geq 0}\, b_t D^t)$ over $\mathbb{F}[[D]]$

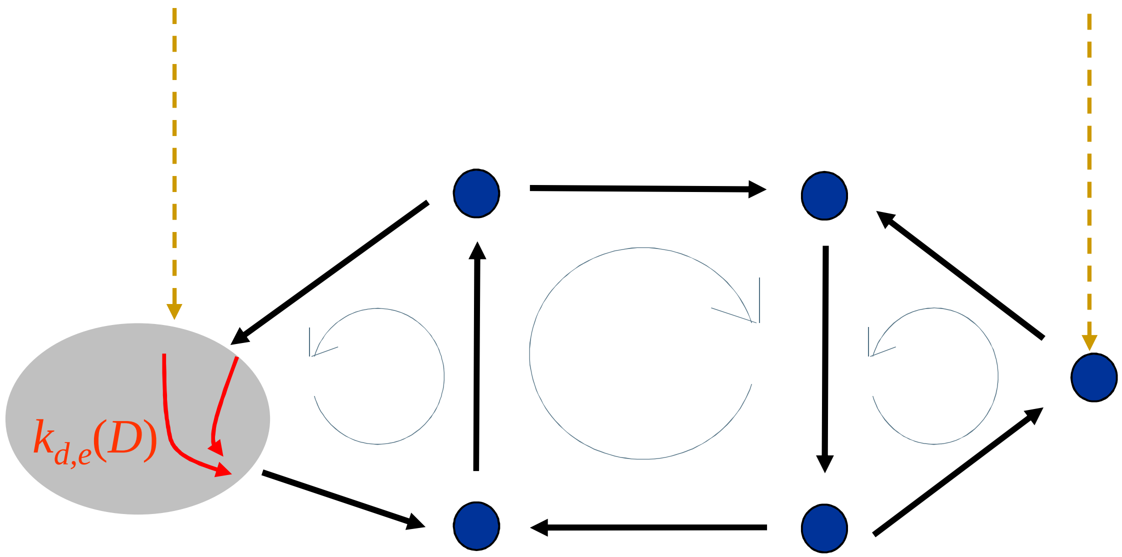


Convolutional network coding (CNC)

Tentative definition. An $\mathbb{F}$-convolutional NC assigns a coding coefficient $k_{d,e}(D) \in \mathbb{F}[[D]]$ to every adjacent pair (d, e).

// Convolution = multiplication between two power series.

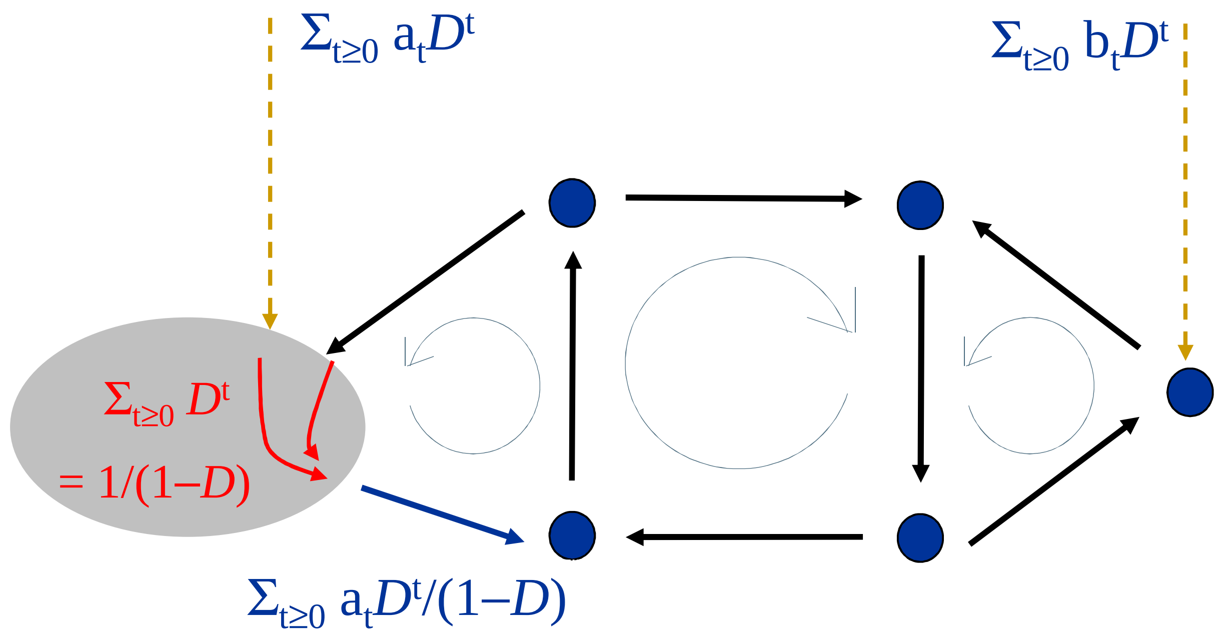
// An adjacent pair (d, e) corresponds to a red arrow.



Convolutional network coding (CNC)

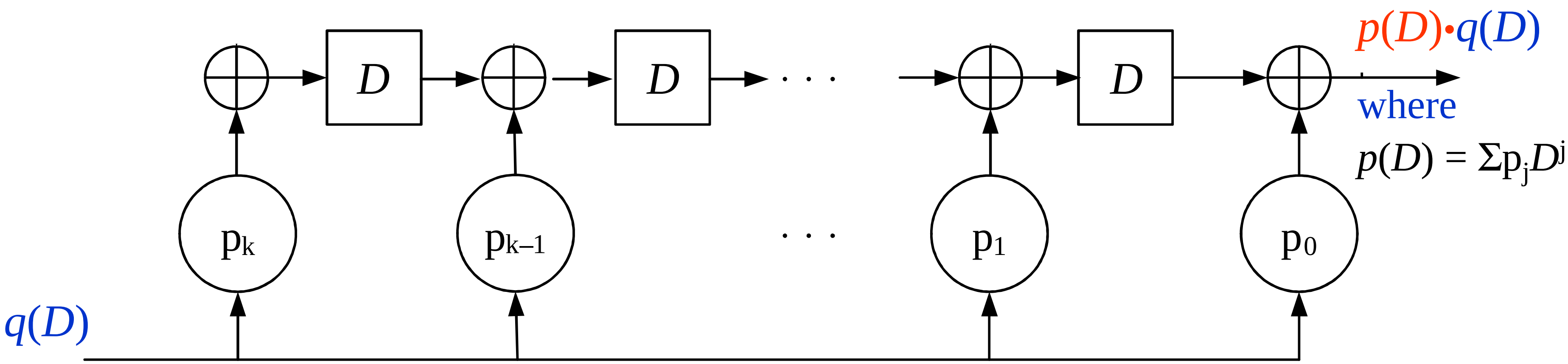
Tentative definition. An $\mathbb{F}$-convolutional NC assigns a coding coefficient $k_{d,e}(D) \in \mathbb{F}[[D]]$ to every adjacent pair (d, e).

$\leftrightarrow$ There is a convolutional encoder at every adjacent pair (d, e), in which the kernel is $k_{d,e}$.



Polynomial kernel in convolutional encoder

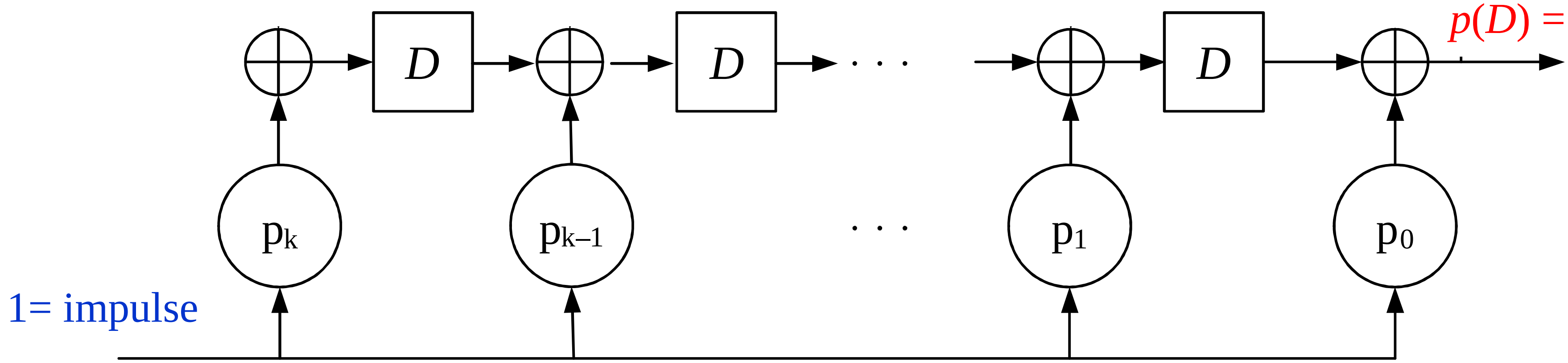
When the kernel is a polynomial $\Sigma p_j D^j \in \mathbb{F}[D]$, a simple implementation of the convolutional encoder is by a shift register:



Polynomial kernel in convolutional encoder

When the kernel is a polynomial $\Sigma p_j D^j \in \mathbb{F}[D]$, a simple implementation of the convolutional encoder is by a shift register:

The impulse response is $p(D) = \Sigma p_j D^j$



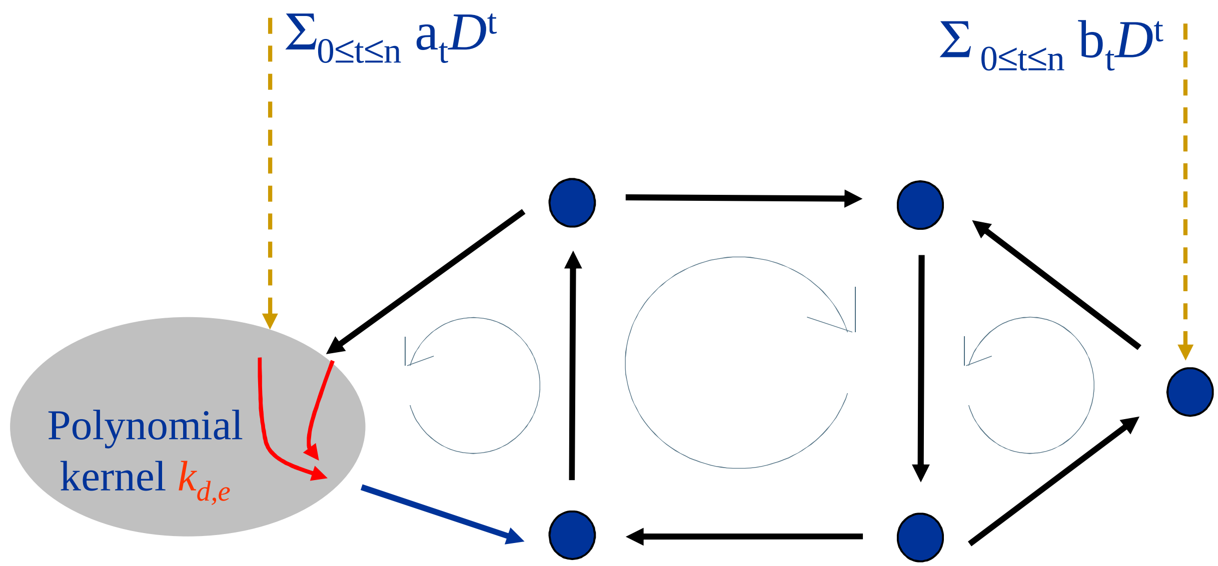
Convolutional NC

- Acyclic networks vs. cyclic networks
- **Finiteness in implementation**
- Causality in data propagation
- Existence and uniqueness of coding vectors

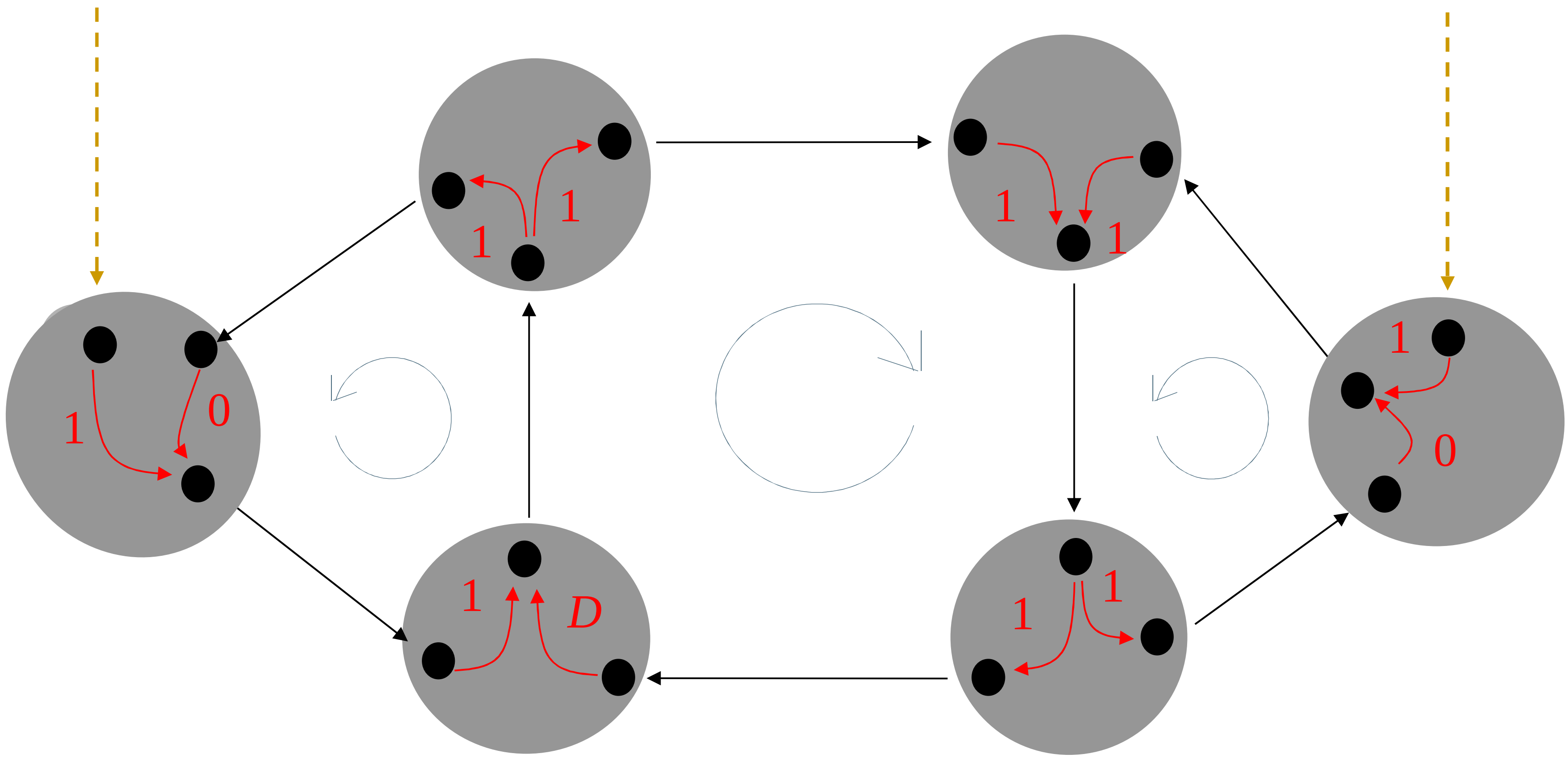
Polynomial kernel in convolutional encoder

Problem. In practice, an input pipeline of symbols is finite and hence represented by a polynomial.

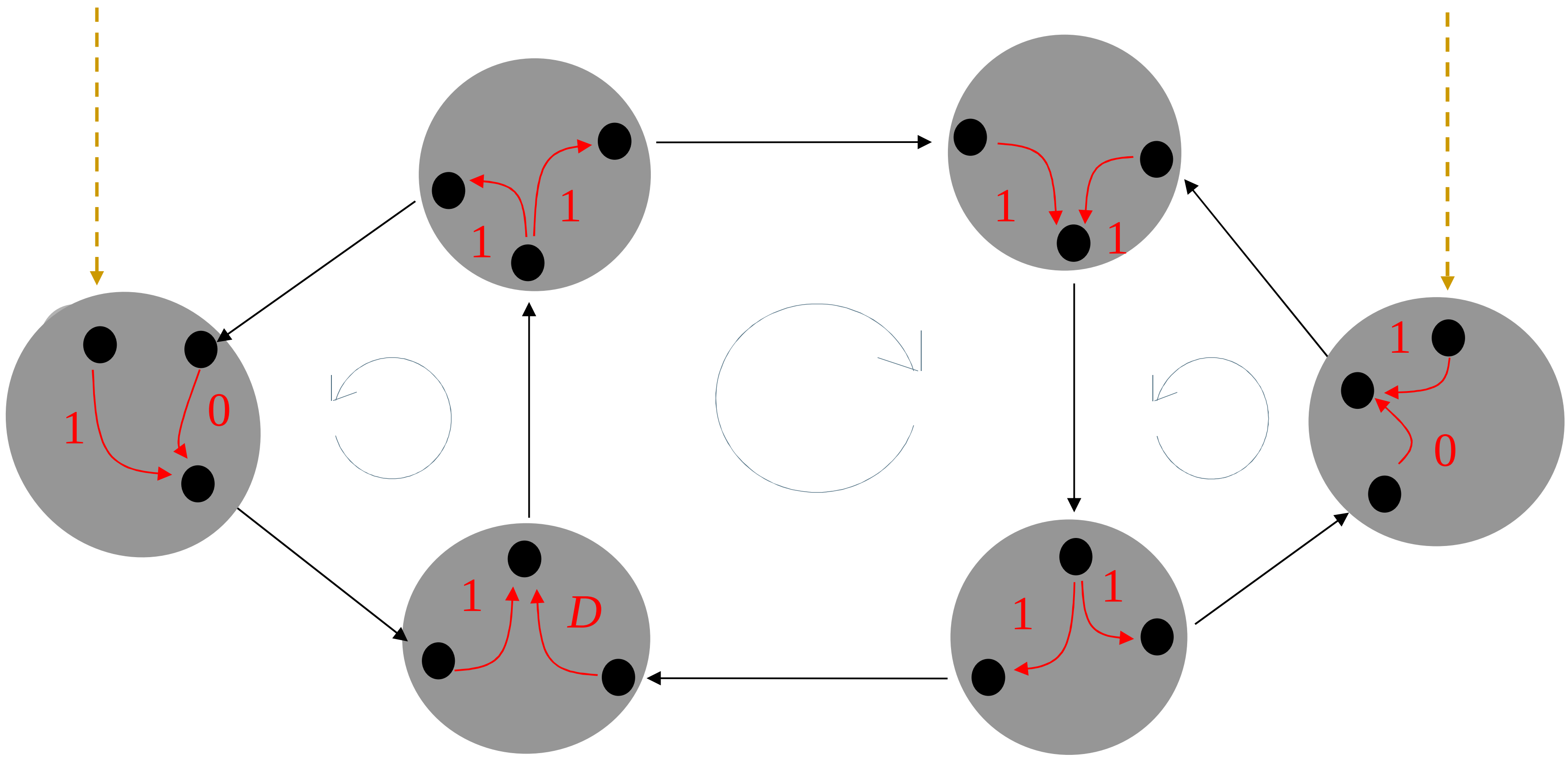
Can we restrict every $k_{d,e}$ to be a polynomial too so that the convolutional encoder can be implemented by finite circuitry/memory/time?



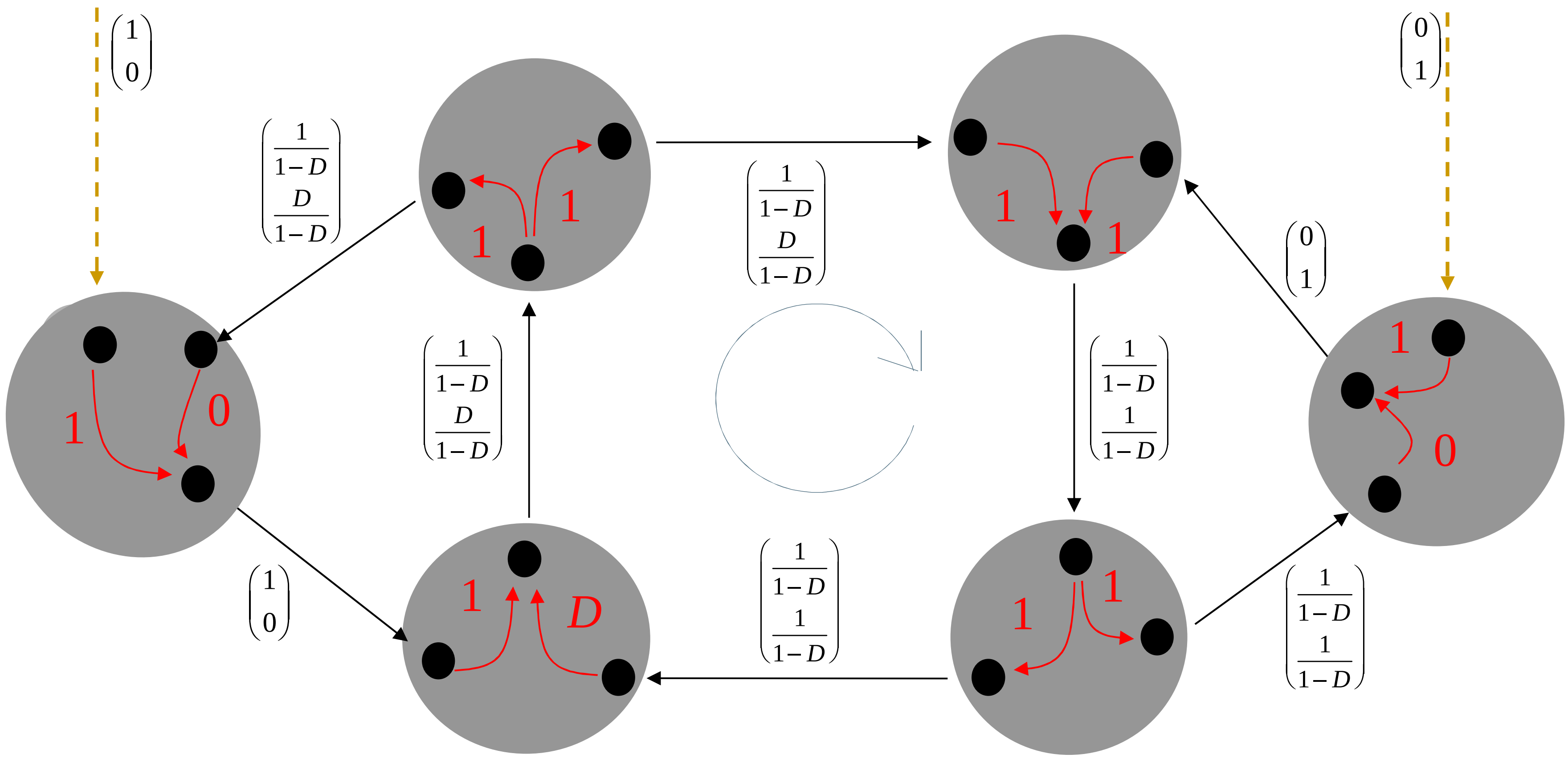
Example of polynomial coding coefficients



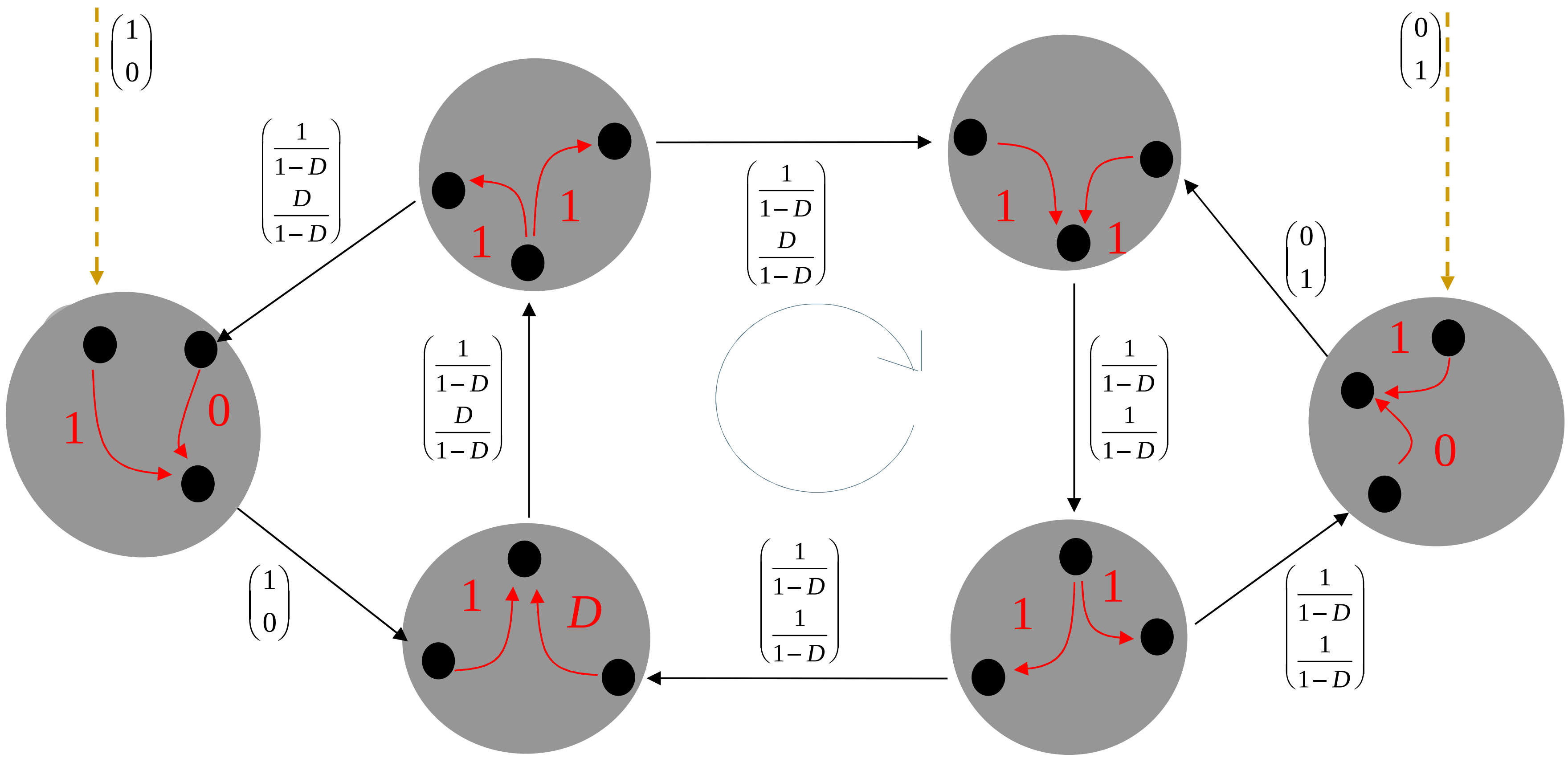
Example of polynomial coding coefficients



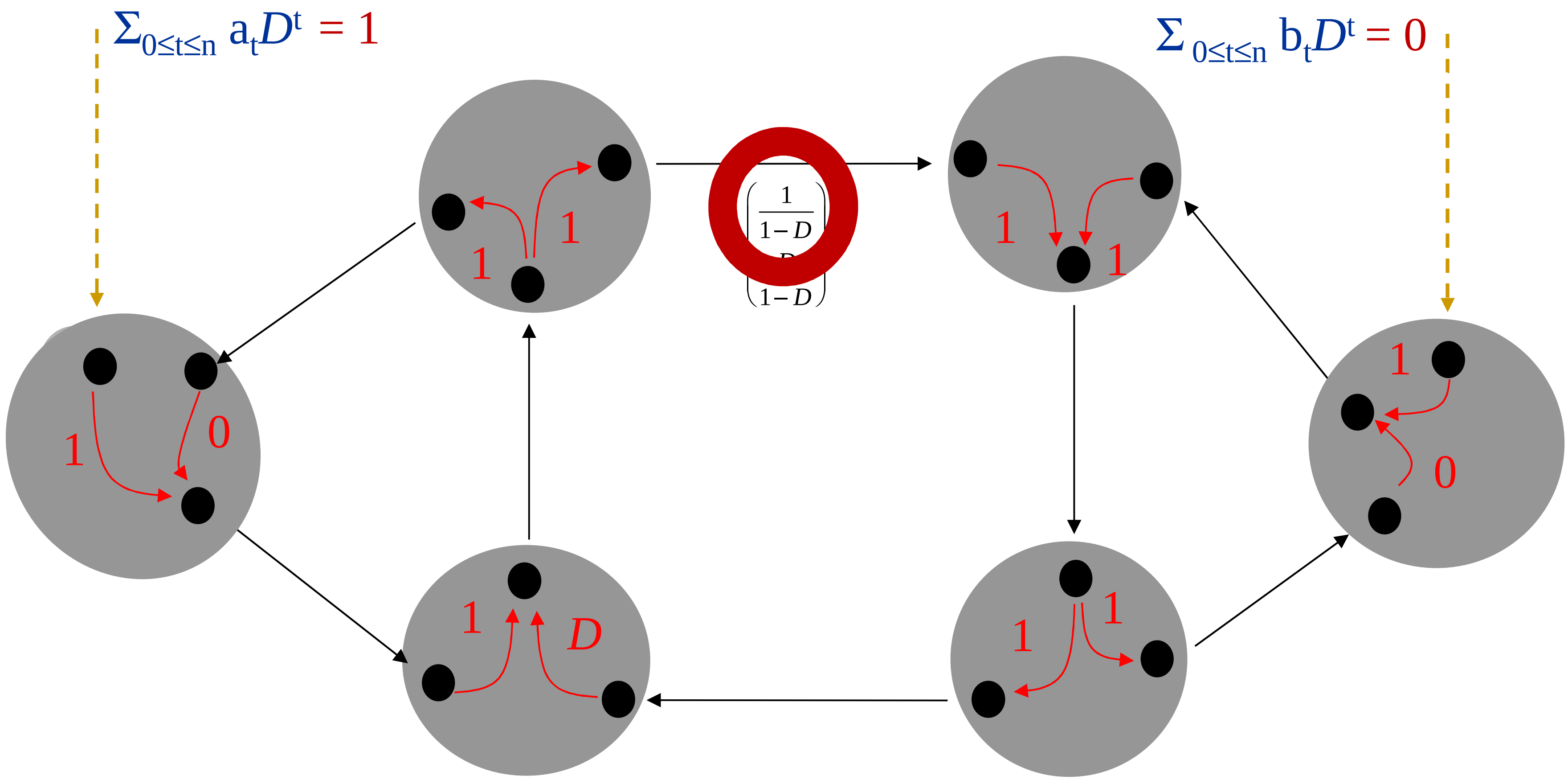
Example of polynomial coding coefficients



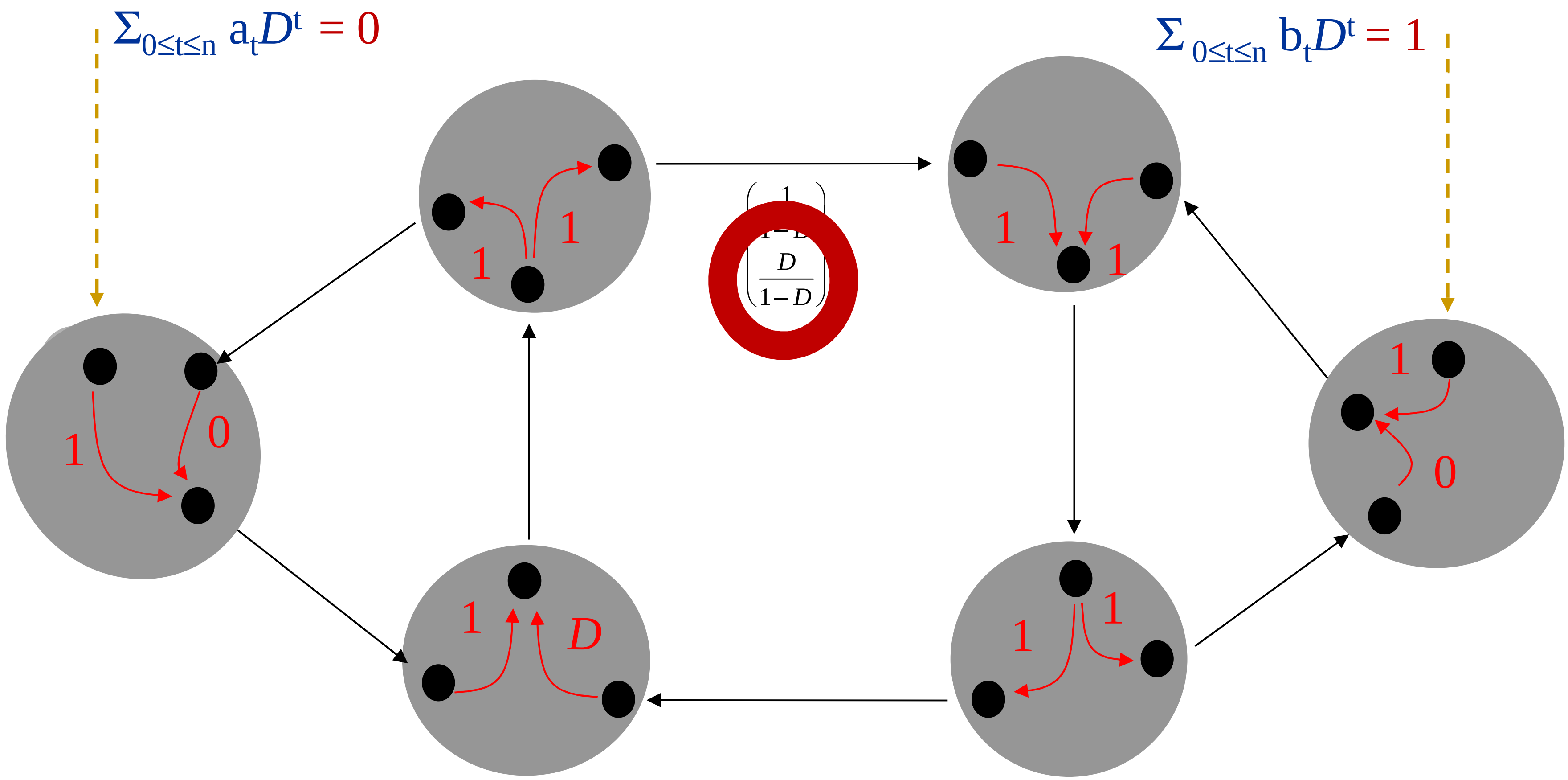
Polynomial coding coefficients $\not\Rightarrow$ polynomial coding vectors



Polynomial coding coefficients $\not\Rightarrow$ polynomial impulse response



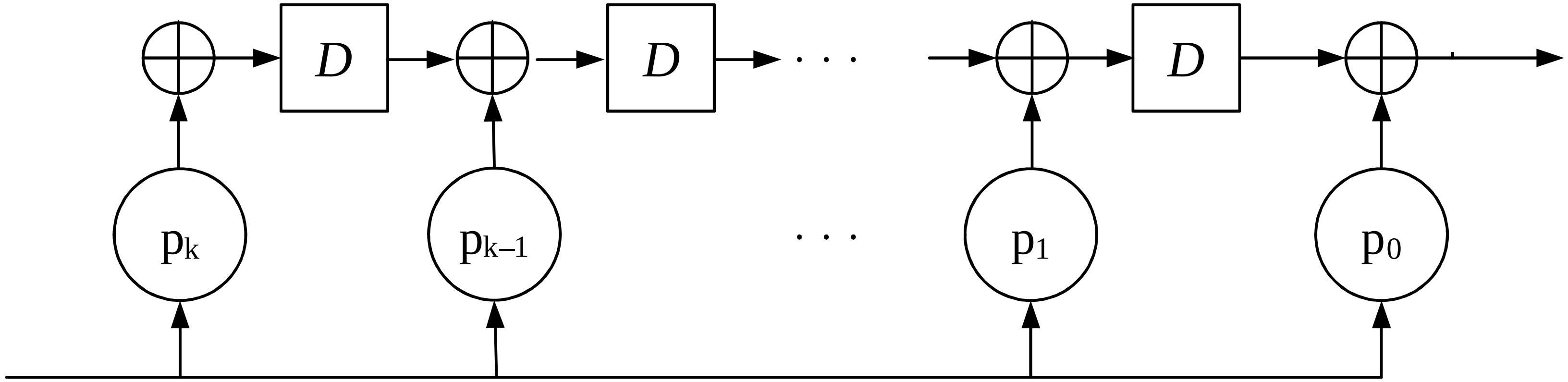
Polynomial coding coefficients $\not\Rightarrow$ polynomial impulse response



Polynomial ring is not a “closed” algebraic structure for CNC.

Explanation. When the kernel in a convolutional encoder is a polynomial $\Sigma p_j D^j \in \mathbb{F}[D]$, there is no feedback channels (i.e., no loop) in the encoder.

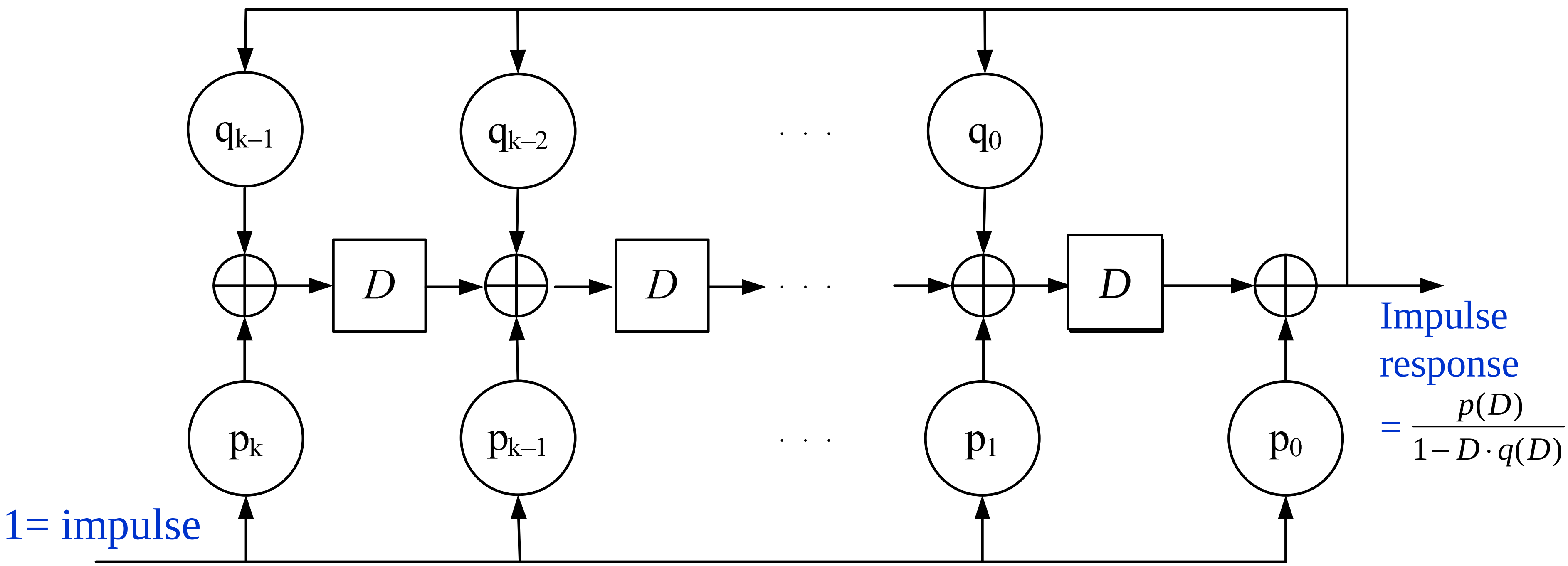
However, even when every node encodes without feedback, a cycle in the network would still create the feedback effect.



Linear encoders with feedback

Finite-state linear *time-invariant* filter

↔ The impulse response = Rational power series [Forney 1970]



where $p(D) = \Sigma p_j D^j$ and $q(D) = \Sigma q_j D^j$

Convolutional NC

- Acyclic networks vs. cyclic networks
- Finiteness in implementation
- **Causality in data propagation**
- Existence and uniqueness of coding vectors

Convolutional encoder $\leftrightarrow$ Algebraic structure

Polynomial ring

Linear encoders without feedback	Impulse response $\longleftrightarrow$	$p(D) \in \mathbb{F}[D]$
Feedback	$\longleftrightarrow$	Algebraic inverse

Convolutional encoder $\leftrightarrow$ Algebraic structure

Linear encoders without feedback

To expand the ensemble of encoders by allowing feedback channels …

Impulse response

$\longleftrightarrow$

Polynomial ring

$p(D) \in \mathbb{F}[D]$

Convolutional encoder $\leftrightarrow$ Algebraic structure

Impulse response

Linear encoders without feedback $\longleftrightarrow$ $p(D) \in \mathbb{F}[D]$

Linear encoders allowing feedback $\longleftrightarrow$ $\dfrac{p(D)}{q(D)} \in \mathbb{F}(D)$

Rational functions
= rational Laurent series

Convolutional encoder $\leftrightarrow$ Algebraic structure

Linear encoders without feedback $\longleftrightarrow$ $p(D) \in \mathbb{F}[D]$ (Impulse response)

Linear encoders that allow feedback for inverting everything **but time** $\longleftrightarrow$ To create an inverse for everything **not divisible by *D***

~~Linear encoders allowing feedback~~ $\longleftrightarrow$ $\cancel{\dfrac{p(D)}{q(D)} \in \mathbb{F}(D)}$

~~Rational functions = rational Laurent series~~

Convolutional encoder $\leftrightarrow$ Algebraic structure

Impulse response

Linear encoders without feedback $\longleftrightarrow$ $p(D) \in \mathbb{F}[D]$

Linear causal encoders $\longleftrightarrow$ $\dfrac{p(D)}{1-D\cdot q(D)} \in \mathbb{F}[(D)]$

Rational power series

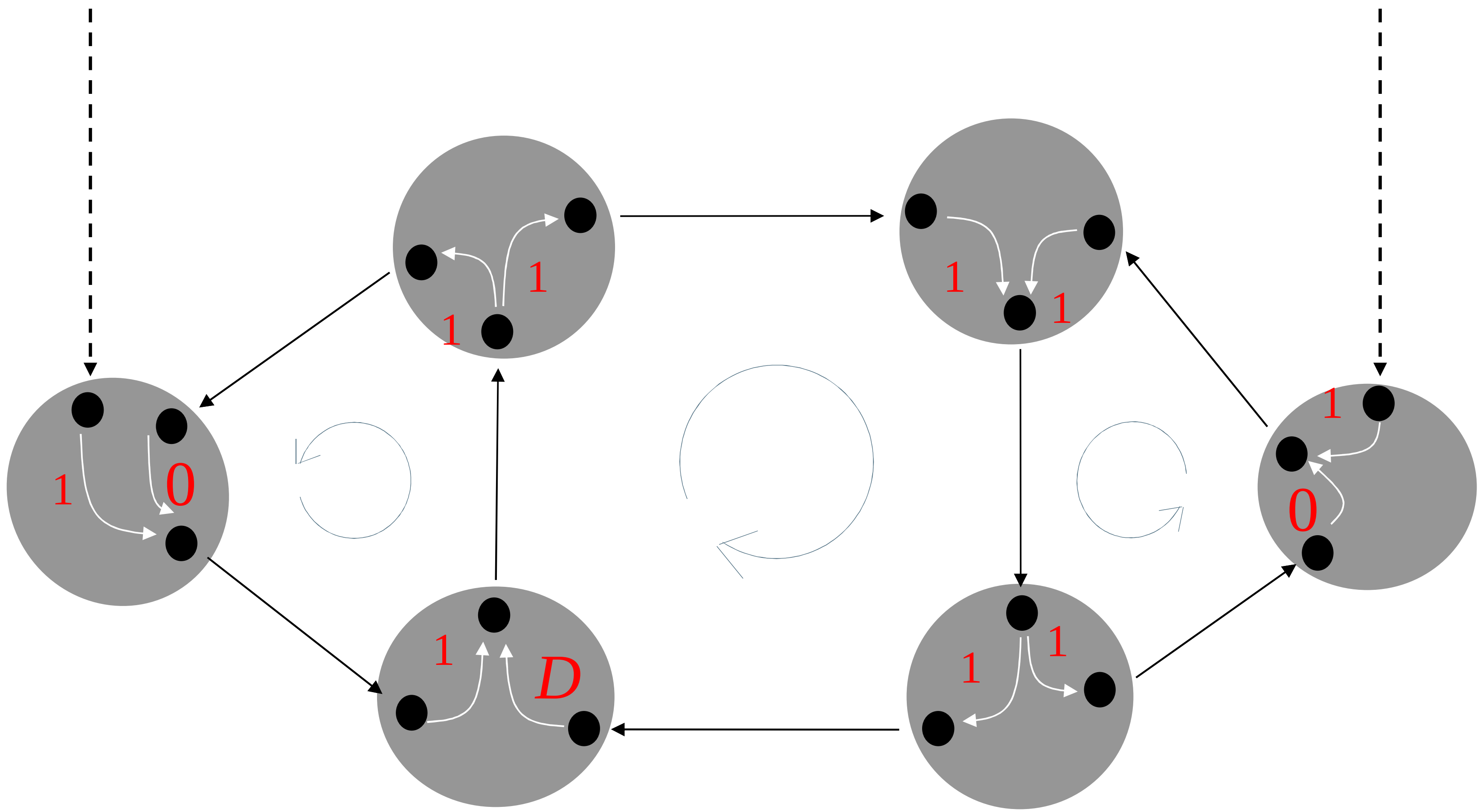
Causal convolutional NC

Definition. An $\mathbb{F}$-convolutional network code assigns a coding coefficient $k_{d,e}(D) \in \mathbb{F}[(D)]$ to every adjacent pair (d, e) of channels.

Moreover, the $\mathbb{F}$-convolutional network code is said to be causal if:

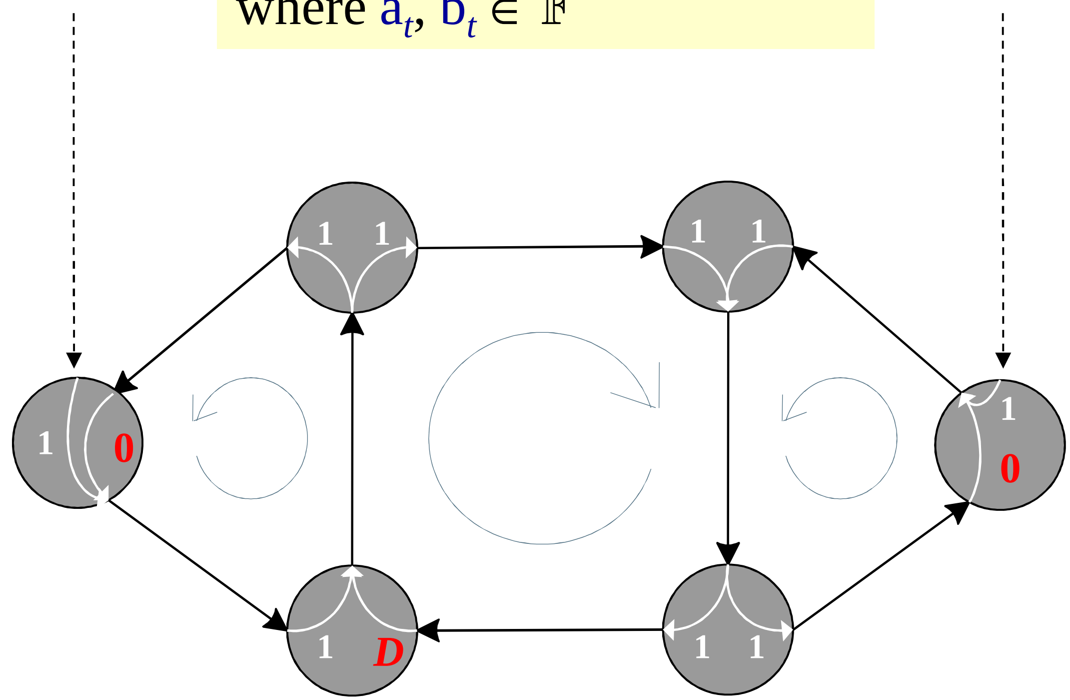
- On every cycle, at least one pair (d, e) is with $k_{d,e}(D)$ divisible by D.

A causal $\mathbb{F}$-convolutional network code



Unidirectional nature of time breaks deadlock in CNC

Message = $(\sum_t a_t D^t \quad \sum_t b_t D^t)$,
where $a_t, b_t \in \mathbb{F}$

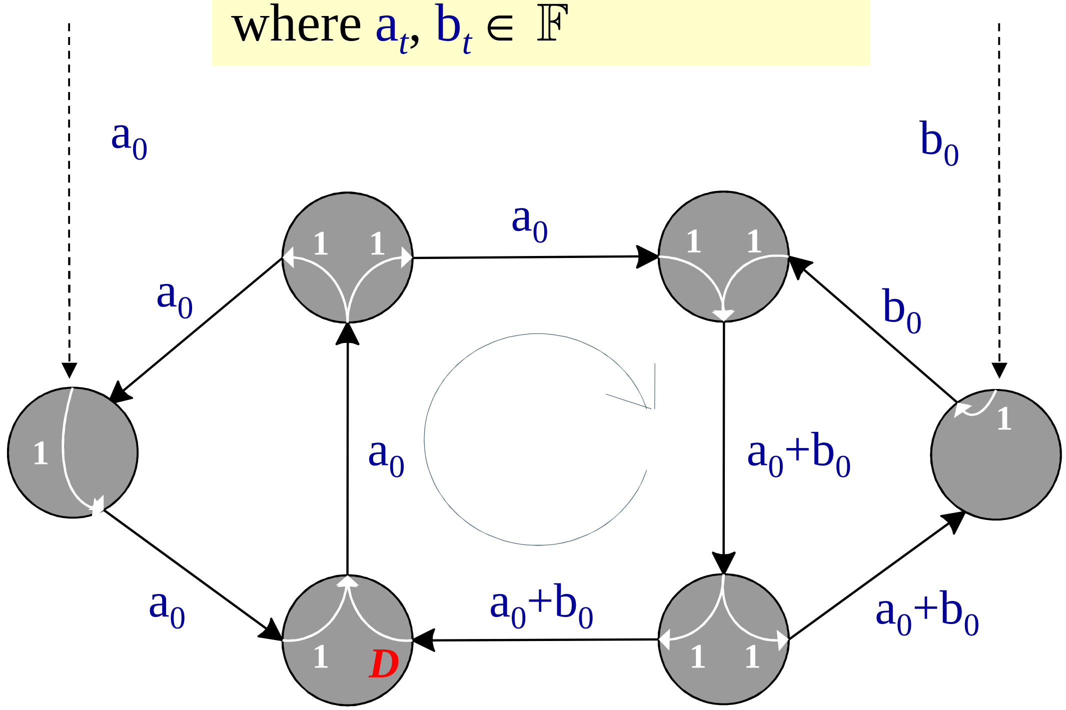


A causal $\mathbb{F}[(D)]$-linear code on the *Shuttle Network*

Timeslot 0

Message = $(\sum_t a_t D^t \quad \sum_t b_t D^t)$,

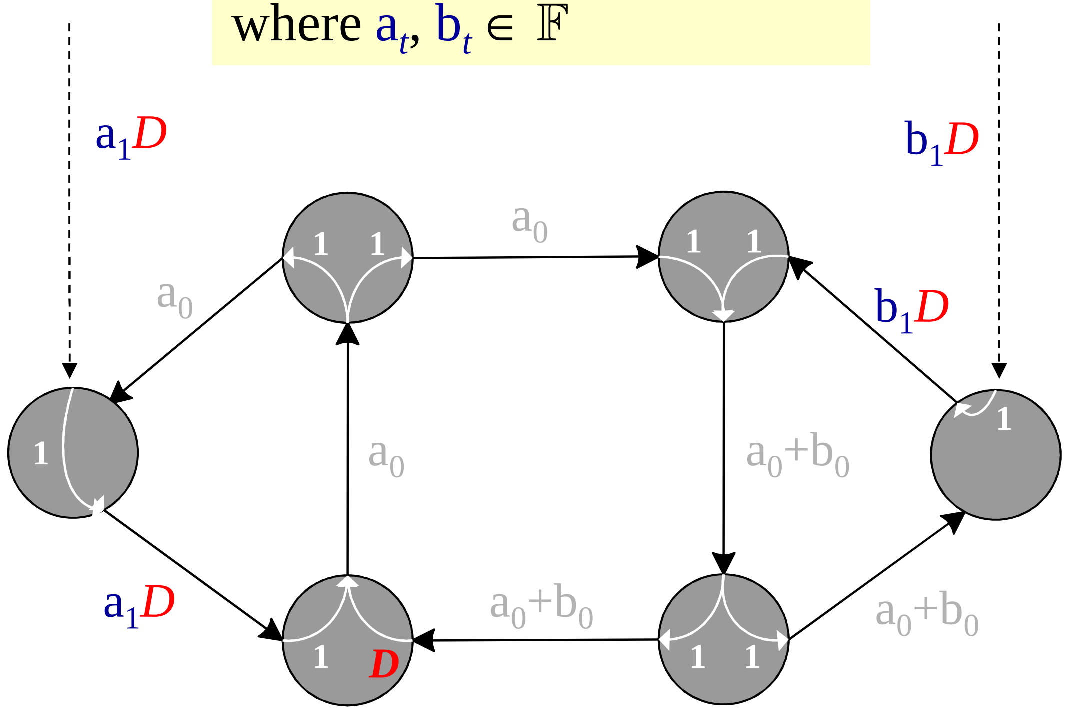
where a_t, $b_t \in \mathbb{F}$



A causal $\mathbb{F}[(D)]$-linear code on the *Shuttle Network*

Timeslot 1

Message = $(\sum_t a_t D^t \quad \sum_t b_t D^t)$,
where a_t, $b_t \in \mathbb{F}$

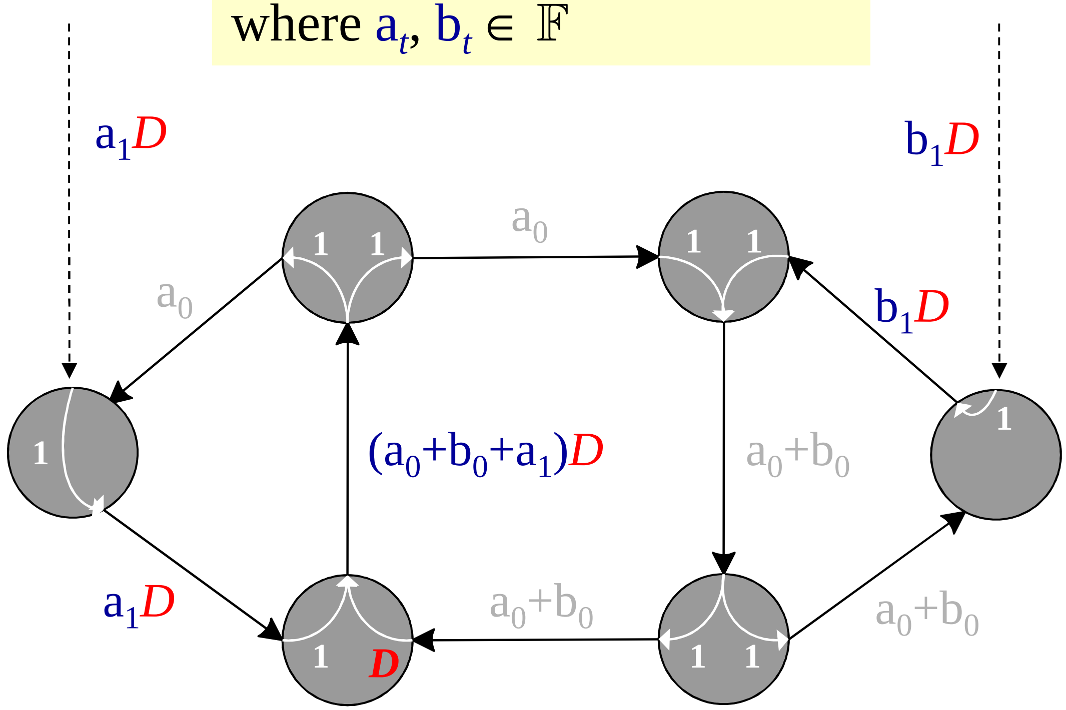


A causal $\mathbb{F}[(D)]$-linear code on the *Shuttle Network*

Timeslot 1

Message = $(\sum_t a_t D^t \quad \sum_t b_t D^t)$,

where a_t, $b_t \in \mathbb{F}$

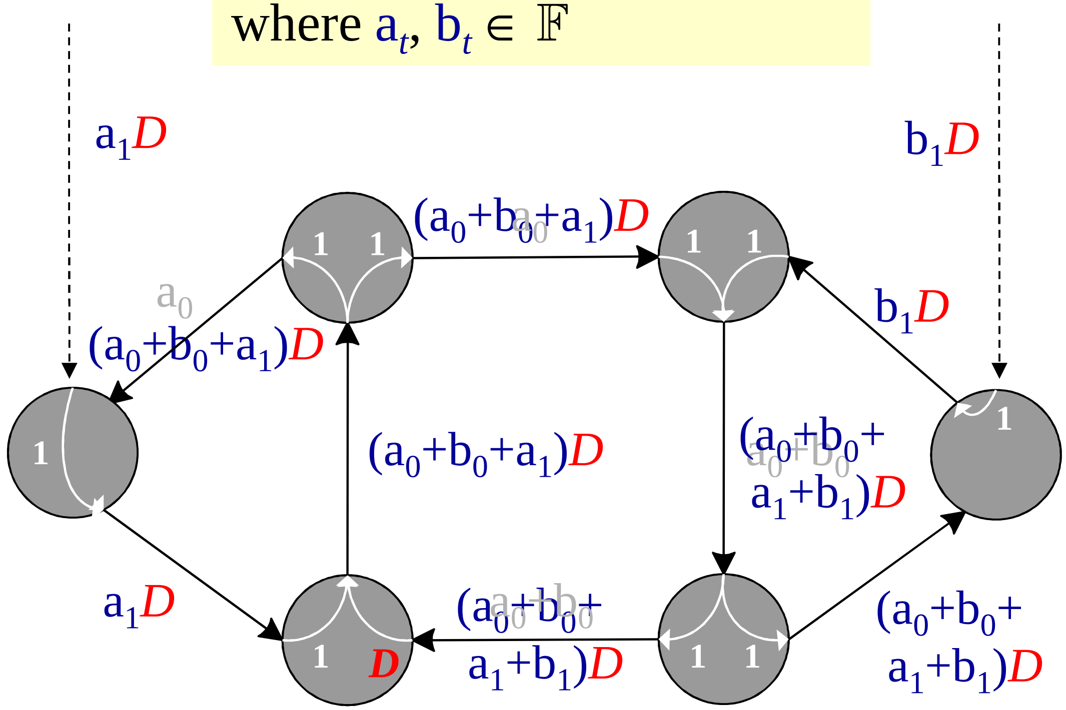


A causal $\mathbb{F}[(D)]$-linear code on the *Shuttle Network*

Timeslot 1

Message = $(\sum_t a_t D^t \quad \sum_t b_t D^t)$,

where a_t, $b_t \in \mathbb{F}$

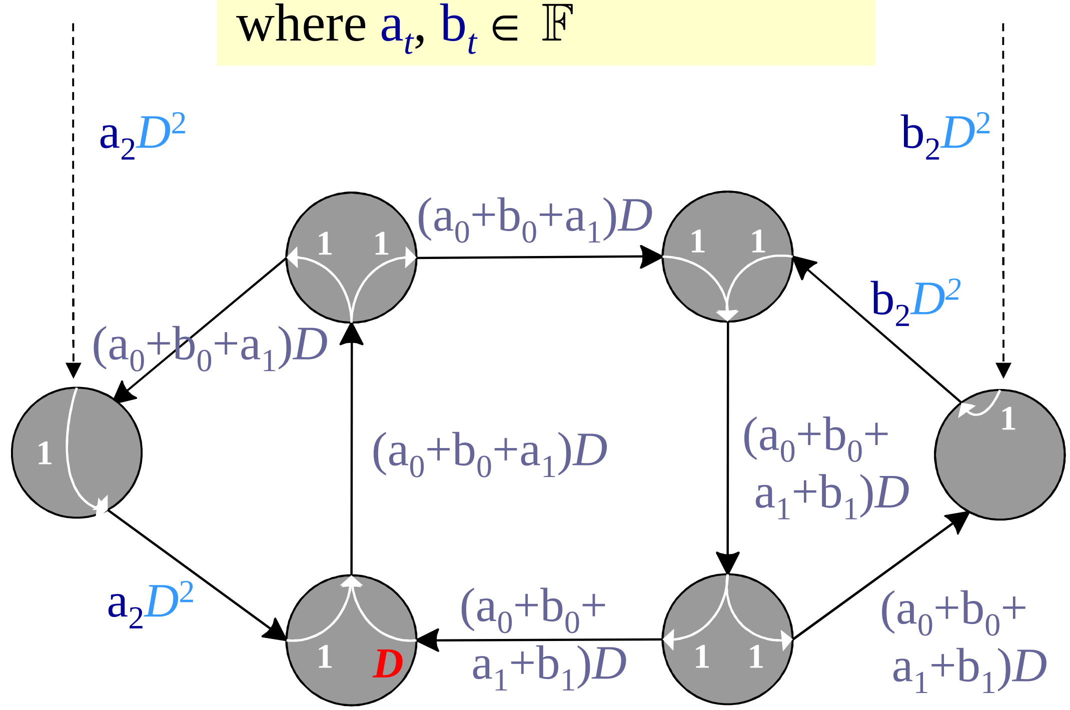


A causal $\mathbb{F}[(D)]$-linear code on the *Shuttle Network*

Timeslot 2

Message = $(\sum_t a_t D^t \quad \sum_t b_t D^t)$,

where a_t, $b_t \in \mathbb{F}$

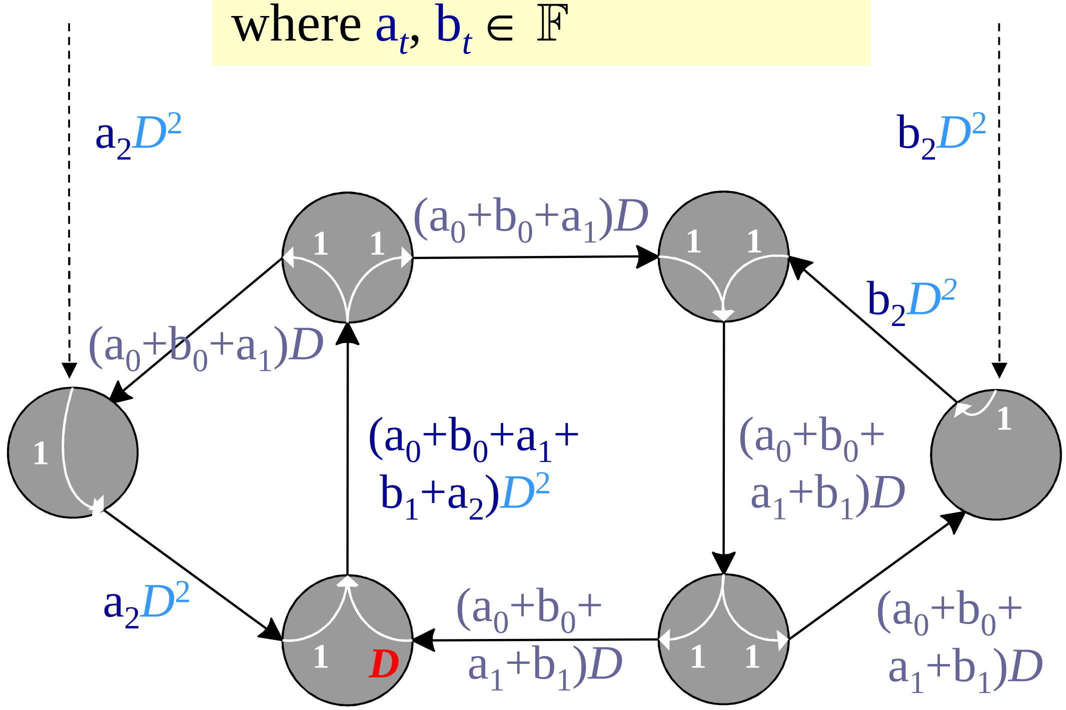


A causal $\mathbb{F}[(D)]$-linear code on the *Shuttle Network*

Timeslot 2

Message = $(\sum_t a_t D^t \quad \sum_t b_t D^t)$,

where a_t, $b_t \in \mathbb{F}$

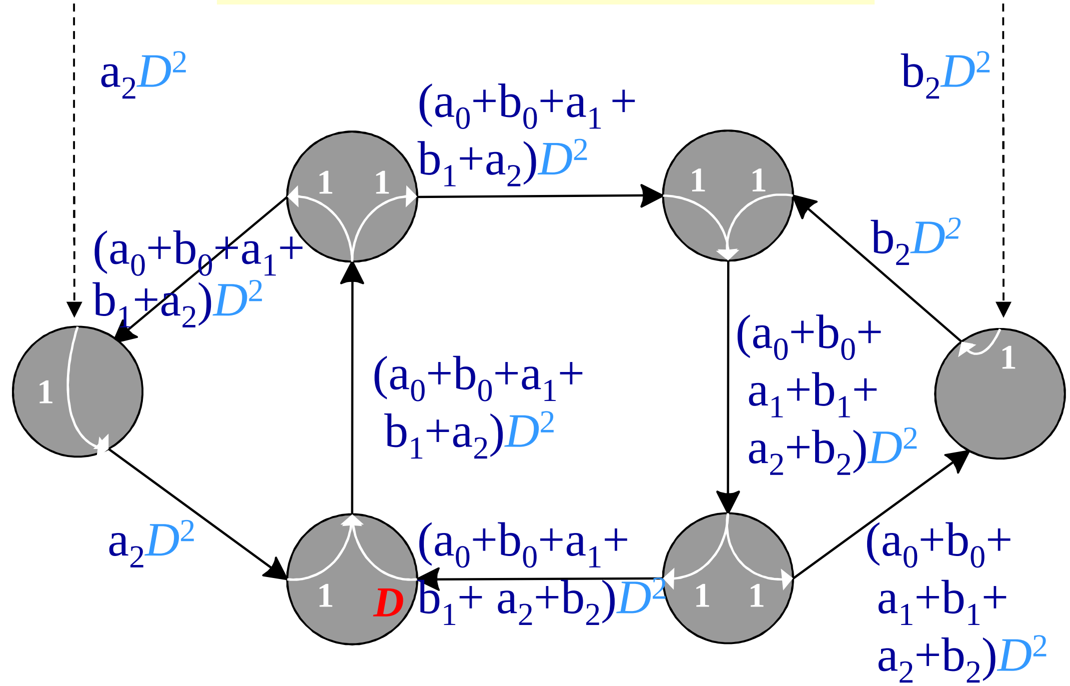


A causal $\mathbb{F}[(D)]$-linear code on the *Shuttle Network*

Timeslot 2

Message = $(\sum_t a_t D^t \quad \sum_t b_t D^t)$,

where $a_t, b_t \in \mathbb{F}$

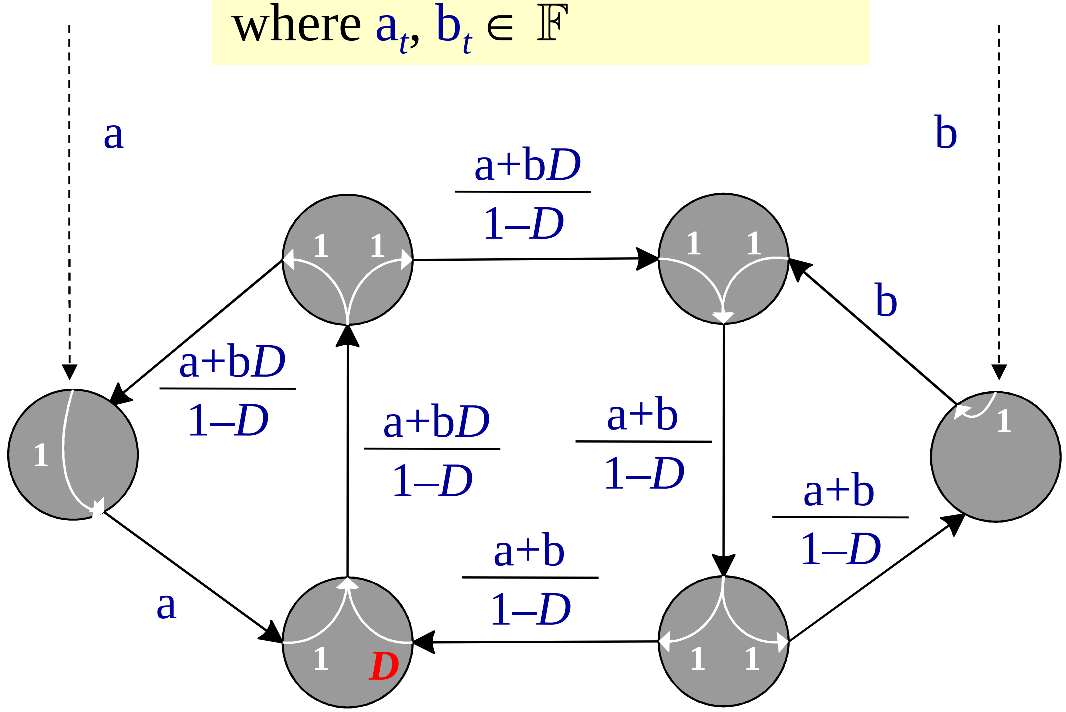


A causal $\mathbb{F}[(D)]$-linear code on the *Shuttle Network*

Cumulative transmission till time = ∞

Message = $(\sum_t a_t D^t \quad \sum_t b_t D^t)$,

where $a_t, b_t \in \mathbb{F}$



A causal $\mathbb{F}[(D)]$-linear code on the *Shuttle Network*

Convolutional NC

- Acyclic networks vs. cyclic networks
- Finiteness
- Causality in data propagation
- **Existence and uniqueness of coding vectors**

Coding vectors of a causal code

Theorem. There exists a unique set of coding vectors $\boldsymbol{f_e(D)}$ over $\mathbb{F}[(D)]$ for every causal $\mathbb{F}$-convolutional network code.

This is crucial for:

- Message propagation — The coding vector $\boldsymbol{f_e(D)}$ unambiguously specifies a rational power series to be carried by a channel e.
- Message reception — A receiving node decodes based on its incoming coding vectors.

Optimal data reception

Definition. For a causal $\mathbb{F}$-convolutional network code, incoming coding vectors to a node v generate a set of ω-dim vectors over $\mathbb{F}[(D)]$. Let this set be called the received free submodule by v.

// A "module" is analogous to a vector space.

The data reception rate of v from s means the rank of this submodule.

// "Rank" here is analogous to "dimension."

// A necessary notion for network **de**coding.

When this rate = maxflow(v) for every v, the causal $\mathbb{F}$-convolutional network code is said to be optimal.

All these will be clarified by **Invariant Factor Theorem of Free Submodule over a PID** in my next set of lecture notes.

Existence of optimal convolutional NC

Theorem 1. There exists an optimal $\mathbb{F}$-convolutional NC.

Proof.

> **Lemma 1.** There exists an optimal $\mathbb{F}$-linear NC with all coding coefficients in any sufficiently large subset of $\mathbb{F}$.

So we may assume that $\mathbb{P}$ is *not* a field and hence is an infinite PID.

Applying the Lemma 1 to $\mathbb{F}$ = quotient field of $\mathbb{P}$, there exists

- An optimal $\mathbb{F}$-linear NC C with coding coefficients in $\mathbb{P}$.

Thus C qualifies as

- a nonsingular $\mathbb{P}$-linear network code.

Normalize C into an optimal $\mathbb{P}$-linear NC.

Practical concern with CNC

Long-distance symbol-level synchronization is difficult, e.g.,

- when a symbol b misses the synchronization by a unit time (say, a μs), it becomes bD.

LNC	**Acyclic network**	**Data unit** $\in \mathbb{F}$
CNC	**Any network**	**Data unit** $\in \mathbb{F}[(D)]$
Abstract generalization	**Any network**	**Data unit** $\in$ **Some algebraic structure that shares the key property of** $\mathbb{F}[(D)]$

Algebraic properties of the ring $\mathbb{F}[(D)]$

Inside the ring $\mathbb{F}[(D)]$, the subset $\langle D^t \rangle$ represents things that happen from the time t onward.

Key property: Ideals in the ring $\mathbb{F}[(D)]$ form a strictly descending chain

$$\langle D \rangle \supset \langle D^2 \rangle \supset \ldots \supset \langle D^t \rangle \supset \ldots \to \{0\}$$

// Anything infinitely delayed is null.

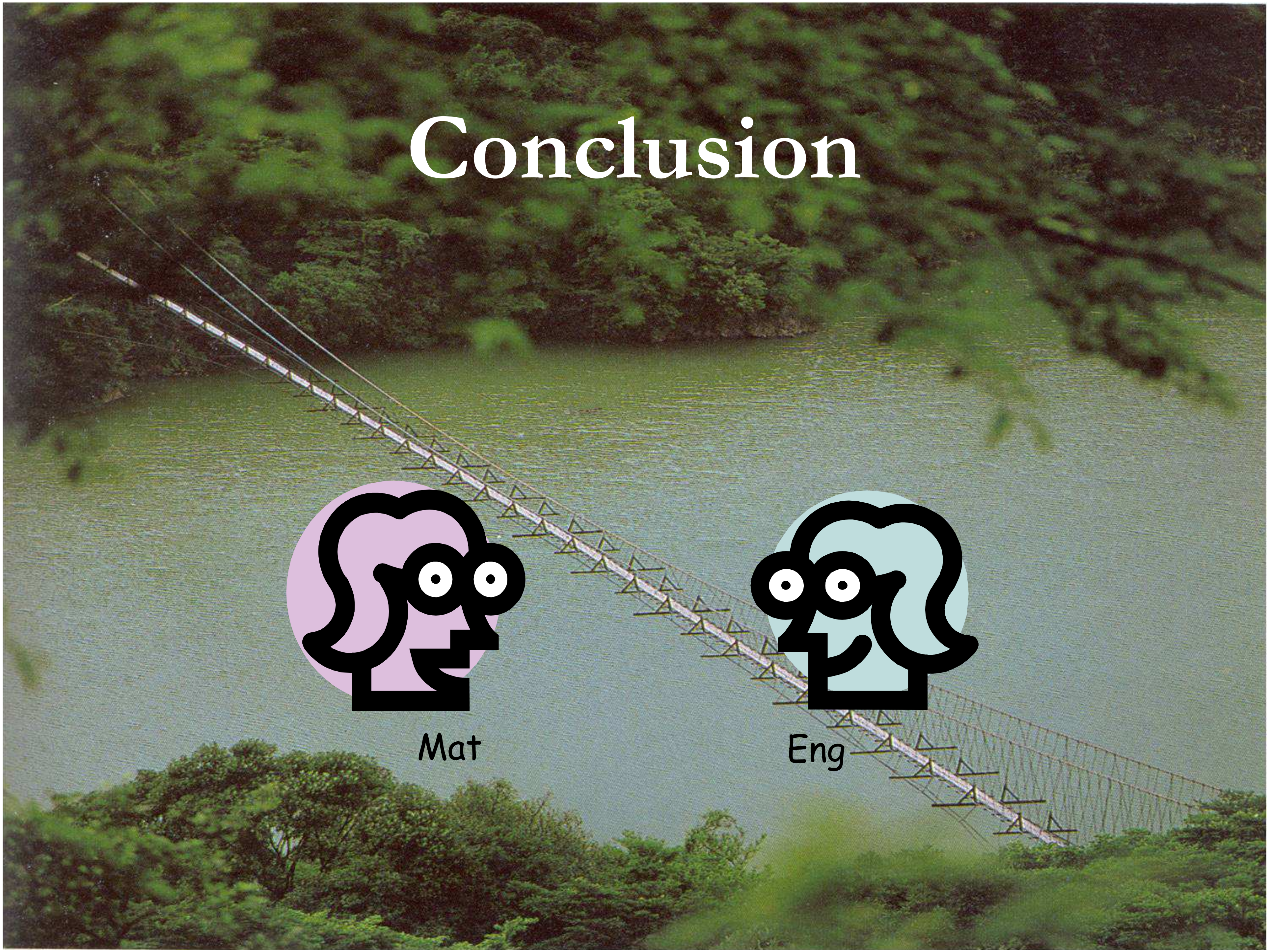
To identify other algebraic structures that share this key property.

CNC expands the polynomial ring into rational power series by making everything, except time, invertible.

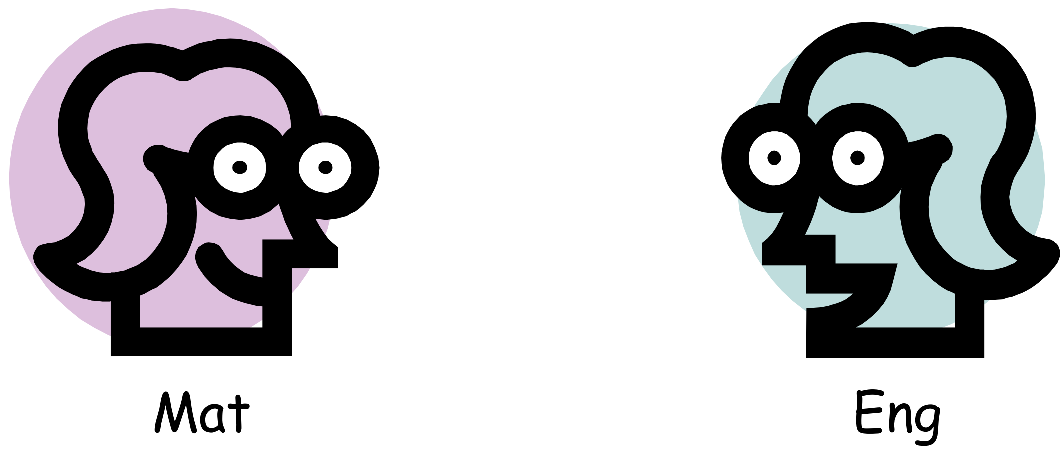
In algebra, this expansion is the localization of a PID at a prime ideal.

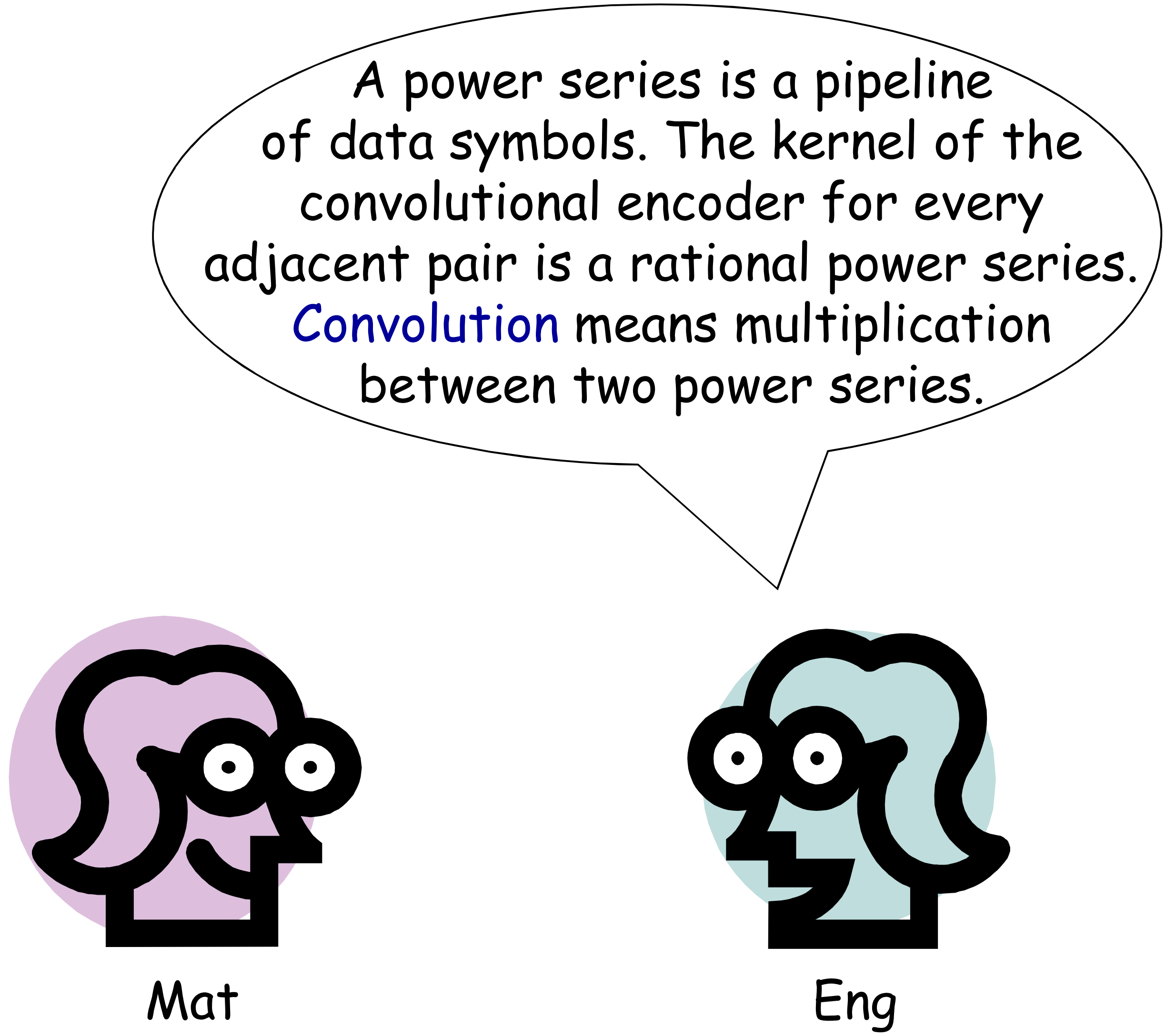
This results in a local ring.

…


Conclusion
Mat
Eng

Linear NC combines
linear algebra with concepts in
network flow to achieve simultaneous
optimal data rates of multiple receivers.
Thus information flow can do better than
the store-and-forward mode of
commodity flow.



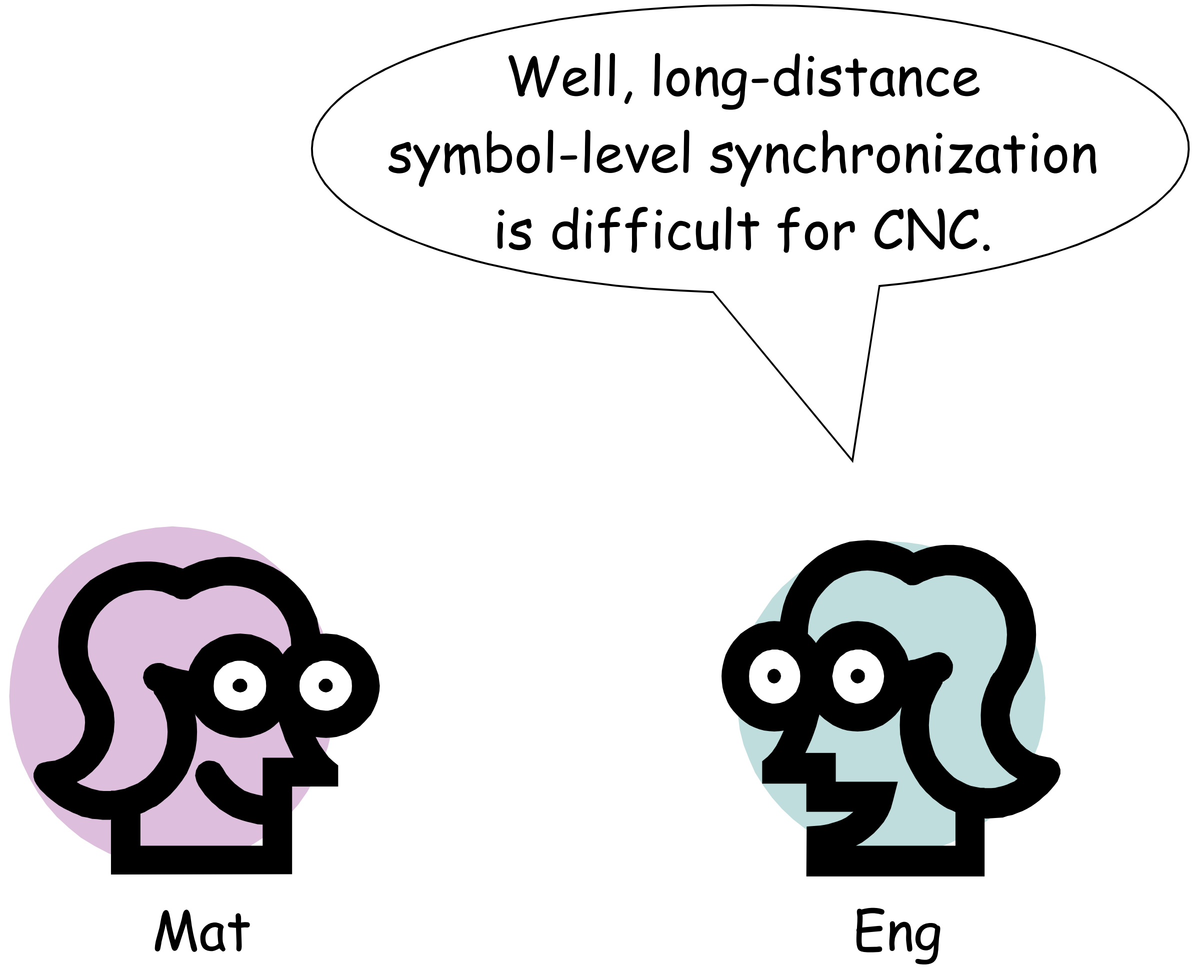

A power series is a pipeline of data symbols. The kernel of the convolutional encoder for every adjacent pair is a rational power series. Convolution means multiplication between two power series.
Mat
Eng

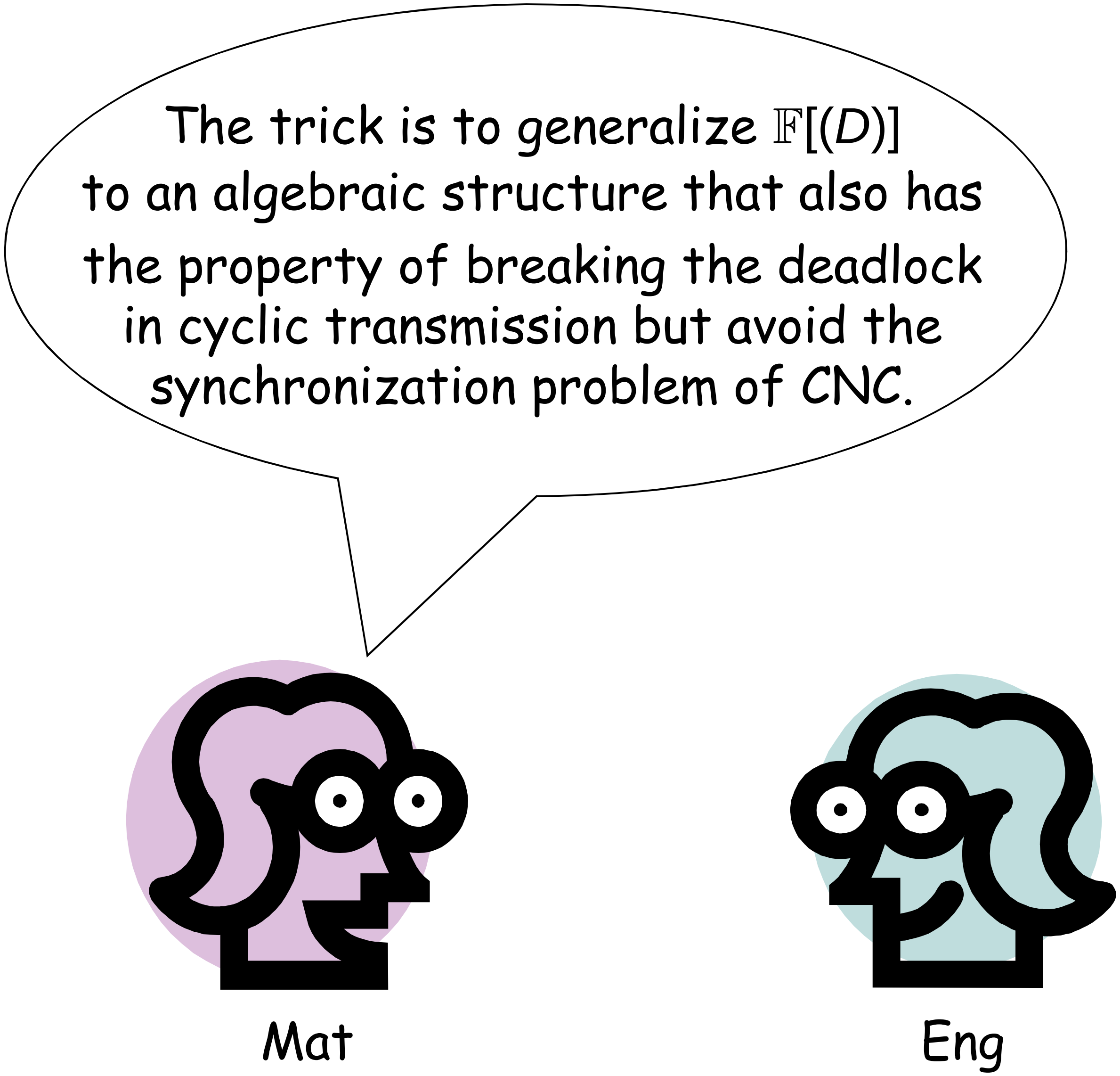
LNC is linear w.r.t. the symbol field $\mathbb{F}$.

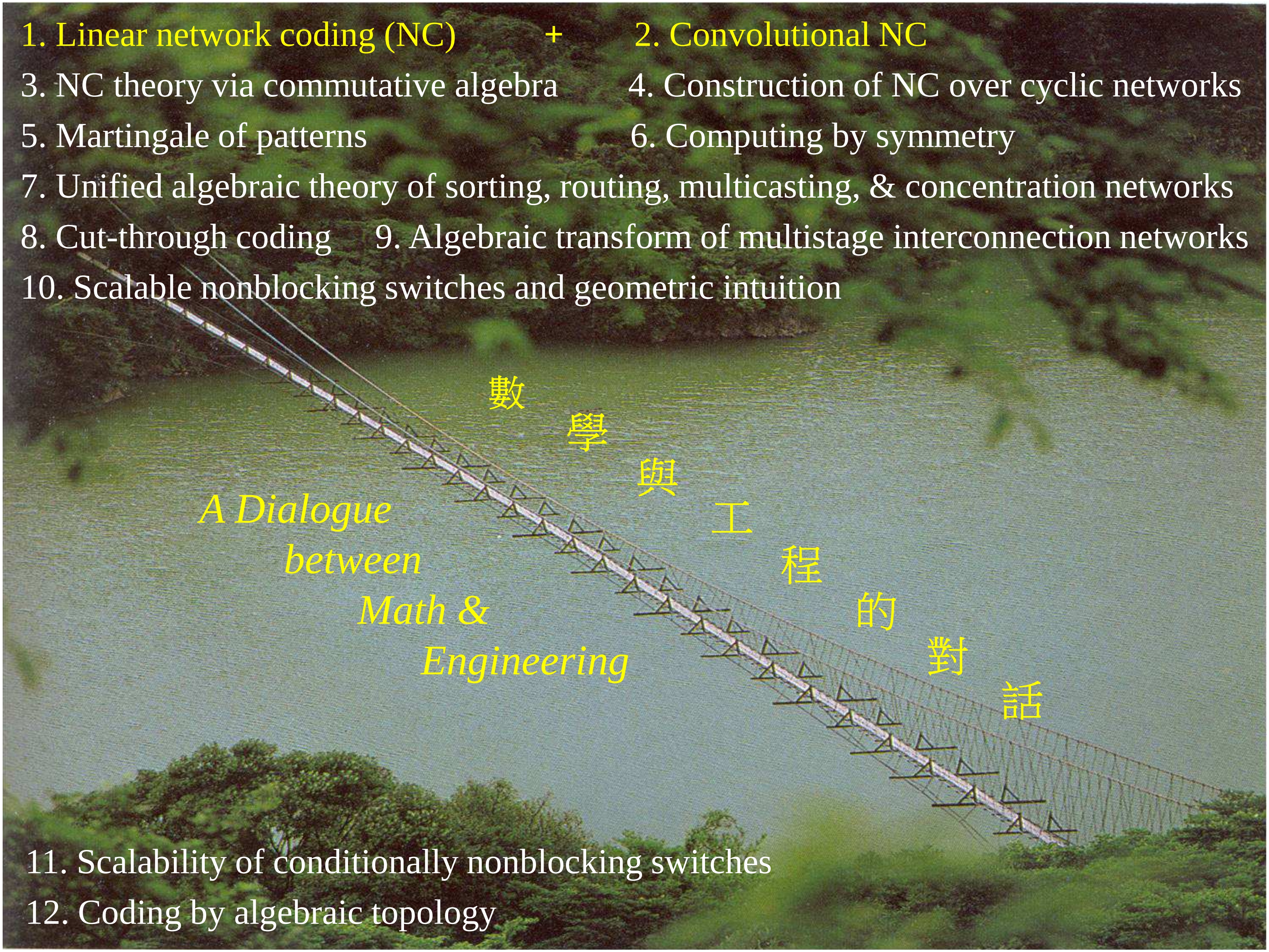
CNC is also linear, but it is w.r.t. the ring $\mathbb{F}[(D)]$ of rational power series.

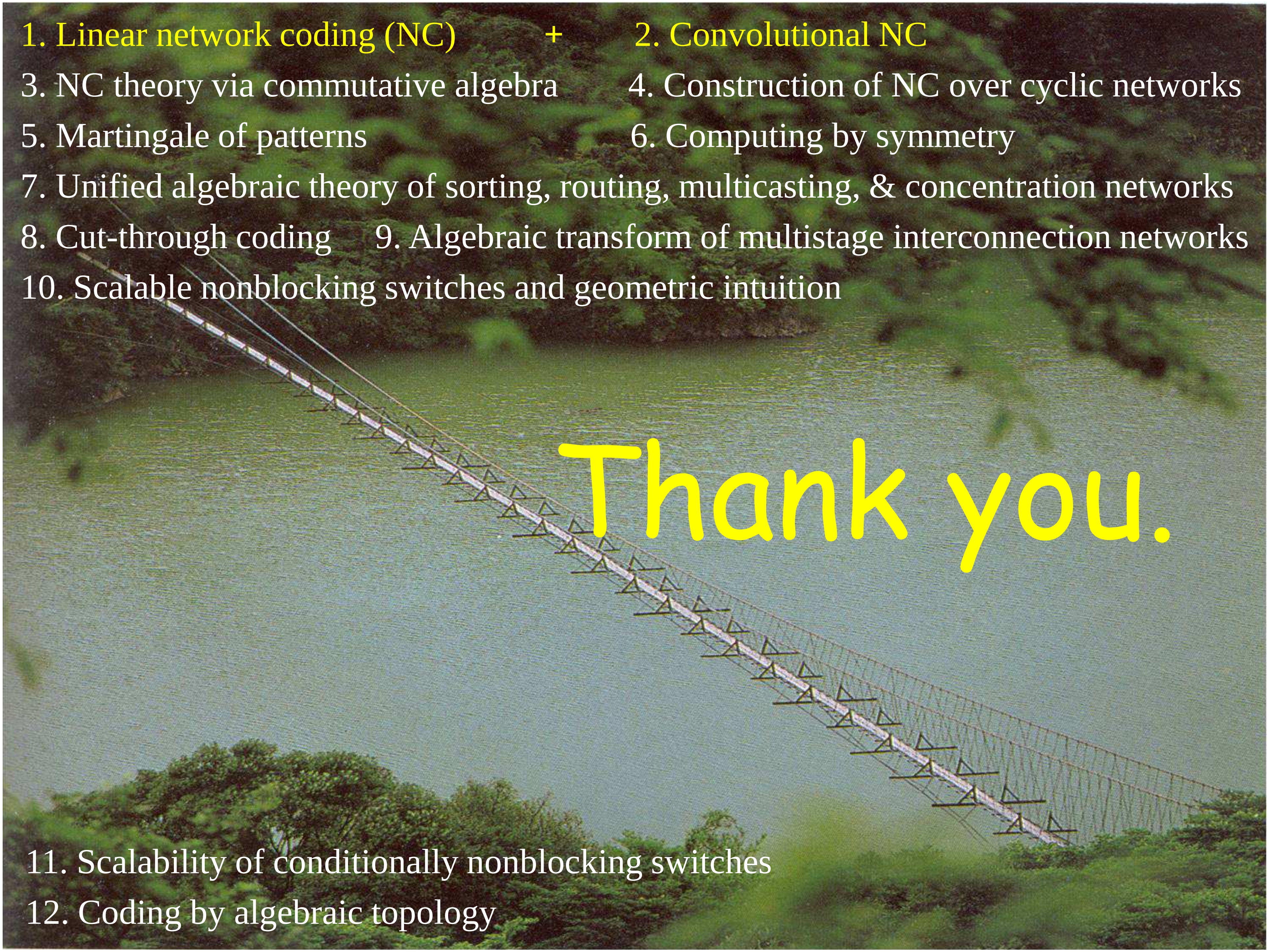
Mat

Eng


Well, long-distance
symbol-level synchronization
is difficult for CNC.
Mat
Eng


The trick is to generalize $\mathbb{F}[(D)]$ to an algebraic structure that also has the property of breaking the deadlock in cyclic transmission but avoid the synchronization problem of CNC.
Mat
Eng


1. Linear network coding (NC) + 2. Convolutional NC
3. NC theory via commutative algebra
4. Construction of NC over cyclic networks
5. Martingale of patterns
6. Computing by symmetry
7. Unified algebraic theory of sorting, routing, multicasting, & concentration networks
8. Cut-through coding
9. Algebraic transform of multistage interconnection networks
10. Scalable nonblocking switches and geometric intuition
數學與工程的對話
A Dialogue between Math & Engineering
11. Scalability of conditionally nonblocking switches
12. Coding by algebraic topology


1. Linear network coding (NC) + 2. Convolutional NC
3. NC theory via commutative algebra
4. Construction of NC over cyclic networks
5. Martingale of patterns
6. Computing by symmetry
7. Unified algebraic theory of sorting, routing, multicasting, & concentration networks
8. Cut-through coding
9. Algebraic transform of multistage interconnection networks
10. Scalable nonblocking switches and geometric intuition
Thank you.
11. Scalability of conditionally nonblocking switches
12. Coding by algebraic topology