Abstract Cryptography

and secure MPC

Ueli Maurer

ETH Zurich

joint work with **Renato Renner**

Secure Multi-Party Computation Workshop, Aarhus, June 2012.

Abstraction

Abstraction

Algebraic abstraction:

- group $\langle G, \star, e, \widehat{\ } \rangle$

Abstraction

Algebraic abstraction:

- group $\langle G, \star, e, \hat{}\ \rangle$
- field $\langle F, +, \cdot, 0, 1 \rangle$

Abstraction

Algebraic abstraction:

- group $\langle G, \star, e, \hat{} \rangle$
- field $\langle F, +, \cdot, 0, 1 \rangle$
- vector space, graph, data structures, OSI layers, ...

Abstraction

Algebraic abstraction:

- group $\langle G, \star, e, \hat{\ } \rangle$
- field $\langle F, +, \cdot, 0, 1 \rangle$
- vector space, graph, data structures, OSI layers, ...

Levels of abstraction:

1. Abstract group: $\langle G, \star, e, \hat{\ } \rangle$
2. Instantiations: Integers, real number, elliptic curves
3. Representations: e.g. projective coordinates for ECs

Abstraction

Algebraic abstraction:

- group $\langle G, \star, e, \hat{} \rangle$
- field $\langle F, +, \cdot, 0, 1 \rangle$
- vector space, graph, data structures, OSI layers, ...

What is the abstraction of:

- cryptosystem ?
- digital signature scheme ?
- MPC protocol ?
- zero-knowledge proof ?
- algorithm, distinguisher, hybrid argument, ...?

Abstraction

Algebraic abstraction:

- group $\langle G, \star, e, \hat{\ } \rangle$
- field $\langle F, +, \cdot, 0, 1 \rangle$
- vector space, graph, data structures, OSI layers, ...

Goals of abstraction:

- eliminate irrelevant details, minimality
- simpler definitions
- generality of results
- simpler proofs
- elegance
- didactic suitability, understanding

Levels of abstraction in AC

#	main concept	concepts treated at this level
0.	Constructions	composability, construction trees
0'.	Games	isomorphism
1.	Abstract systems	cryptographic algebras
2.	Discrete systems	indistinguishability proofs
3.	System implem.	complexity, efficiency, asymptotics

The construction paradigm

The construction paradigm

Many scientific disciplines can be seen as being **constructive**.

The construction paradigm

Many scientific disciplines can be seen as being **constructive**.

Construct an object **S** from another object **R** via construction α.

The construction paradigm

Many scientific disciplines can be seen as being **constructive**.

Construct an object **S** from another object **R** via construction α.

Notation: $\mathbf{R} \xrightarrow{\alpha} \mathbf{S}$

The construction paradigm

Many scientific disciplines can be seen as being **constructive**.

Construct an object **S** from another object **R** via construction α.

Notation: $\mathbf{R} \xrightarrow{\alpha} \mathbf{S}$

Examples:

- A (k, n)-error-correcting code constructs a reliable channel from a noisy channel:

The construction paradigm

Many scientific disciplines can be seen as being **constructive**.

Construct an object **S** from another object **R** via construction α.

Notation: $\mathbf{R} \xrightarrow{\alpha} \mathbf{S}$

Examples:

- A (k, n)-error-correcting code constructs a reliable channel from a noisy channel:

$$\mathrm{NC}_n \xrightarrow{\text{(enc,dec)}} \mathrm{RC}_k$$

The construction paradigm

Many scientific disciplines can be seen as being **constructive**.

Construct an object **S** from another object **R** via construction α.

Notation: $\mathbf{R} \xrightarrow{\alpha} \mathbf{S}$

Examples:

- A (k, n)-error-correcting code constructs a reliable channel from a noisy channel:

$$\mathrm{NC}_n \xrightarrow{\text{(enc,dec)}} \mathrm{RC}_k$$

- An extractor constructs a uniform m-bit string U_m from any RV X with min-entropy $> m + c$ and U_s:

$$(\mathrm{X},\mathrm{U}_s) \xrightarrow{\text{ext}} \mathrm{U}_m$$

The construction paradigm

Many scientific disciplines can be seen as being **constructive**.

Construct an object **S** from another object **R** via construction α.

Notation: $\mathbf{R} \xrightarrow{\alpha} \mathbf{S}$

Examples:

- A (k, m)-PRG constructs a uniform m-bit string from a uniform k-bit string:

$$\mathsf{U}_k \xrightarrow{\text{PRG}} \mathsf{U}_m$$

The construction paradigm

Many scientific disciplines can be seen as being **constructive**.

Construct an object **S** from another object **R** via construction α.

Notation: $\mathbf{R} \xrightarrow{\alpha} \mathbf{S}$

Examples:

- A (k, m)-PRG constructs a uniform m-bit string from a uniform k-bit string:

$$\mathsf{U}_k \xrightarrow{\text{PRG}} \mathsf{U}_m$$

- A key agreement protocol (KAP) constructs a shared secret n-bit key from ???:

$$??? \xrightarrow{\text{KAP}} \mathsf{KEY}_n$$

The construction paradigm

Many scientific disciplines can be seen as being **constructive**.

Construct an object **S** from another object **R** via construction α.

Notation: $\mathbf{R} \xrightarrow{\alpha} \mathbf{S}$

Involved types:

- **resource**
- **constructor**
- **construction notion**

The construction paradigm

Many scientific disciplines can be seen as being **constructive**.

Construct an object **S** from another object **R** via construction α.

Notation: $\mathbf{R} \xrightarrow{\alpha,\epsilon} \mathbf{S}$

Involved types:

- **resource**
- **constructor**
- **construction notion**
- **metric** **on space of resources**

$$\mathbf{d}(\mathbf{R},\mathbf{S}) \leq \epsilon \iff \mathbf{R} \approx_{\epsilon} \mathbf{S}$$

The construction paradigm

Many scientific disciplines can be seen as being **constructive**.

Construct an object **S** from another object **R** via construction α.

Notation: $\mathbf{R} \xrightarrow{\alpha,\epsilon} \mathbf{S}$

Examples:

- A (k,n)-error-correcting code constructs a reliable channel from a noisy channel:

$$\mathrm{NC}_n \xrightarrow{\text{(enc,dec)}} \mathrm{RC}_k$$

The construction paradigm

Many scientific disciplines can be seen as being **constructive**.

Construct an object **S** from another object **R** via construction α.

Notation: $\mathbf{R} \xrightarrow{\alpha,\epsilon} \mathbf{S}$

Examples:

- A (k, n)-error-correcting code constructs a reliable channel from a noisy channel:

$$\mathrm{NC}_n \xrightarrow{\text{(enc,dec)}} \mathrm{RC}_k$$

$$\mathbf{R} \xrightarrow{(\beta,\gamma),\epsilon} \mathbf{S} \quad :\Longleftrightarrow \quad \mathbf{S} \approx_\epsilon \beta\,\mathbf{R}\,\gamma$$

The construction paradigm

Many scientific disciplines can be seen as being **constructive**.

Construct an object **S** from another object **R** via construction α.

Notation: $\mathbf{R} \xrightarrow{\alpha,\epsilon} \mathbf{S}$

Examples:

- A (k,n)-error-correcting code constructs a reliable channel from a noisy channel:

$$\mathrm{NC}_n \xrightarrow{\text{(enc,dec)}} \mathrm{RC}_k$$

$$\mathbf{R} \xrightarrow{(\beta,\gamma),\epsilon} \mathbf{S} \quad :\Longleftrightarrow \quad \mathbf{S} \approx_\epsilon \beta\, \mathbf{R}\, \gamma$$

$$\mathrm{NC}_n \xrightarrow{\text{(enc,dec)},\epsilon} \mathrm{RC}_k \quad \Longleftrightarrow \quad \mathrm{RC}_k \approx_\epsilon \text{enc}\ \mathrm{NC}_n\ \text{dec}$$

Levels of abstraction in AC

#	main concept	concepts treated at this level
0.	Constructions	composability, construction trees
0'.	Games	isomorphism
1.	Abstract systems	cryptographic algebras
2.	Discrete systems	indistinguishability proofs
3.	System implem.	complexity, efficiency, asymptotics

Constructions and composability

resource set $\langle \Omega, \; || \rangle$, **constructor set** $\langle \Gamma, \; \circ, \; | \; , \mathrm{id} \rangle$

Constructions and composability

resource set $\langle \Omega, \;||\rangle$, **constructor set** $\langle \Gamma, \;\circ, \;|\;, \mathrm{id}\rangle$

Construction = subset of $\Omega \times \Gamma \times \Omega$

Possible notation: $\mathbf{R} \xrightarrow{\alpha} \mathbf{S}$

Constructions and composability

resource set $\langle \Omega, \; || \rangle$, **constructor set** $\langle \Gamma, \; \circ, \; | \; , \mathrm{id} \rangle$

Construction = subset of $\Omega \times \Gamma \times \Omega$

Possible notation: $\mathbf{R} \xrightarrow{\alpha} \mathbf{S}$

Definition: A construction is **serially composable** if

1. $\mathbf{R} \xrightarrow{\alpha} \mathbf{S} \;\wedge\; \mathbf{S} \xrightarrow{\beta} \mathbf{T} \;\Rightarrow\; \mathbf{R} \xrightarrow{\alpha\circ\beta} \mathbf{T}$

Constructions and composability

resource set $\langle \Omega, \ || \rangle$, **constructor set** $\langle \Gamma, \ \circ, \ | \ , \mathrm{id} \rangle$

Construction = subset of $\Omega \times \Gamma \times \Omega$

Possible notation: $\mathbf{R} \xrightarrow{\alpha} \mathbf{S}$

Definition: A construction is serially composable if

1. $\mathbf{R} \xrightarrow{\alpha} \mathbf{S} \ \wedge \ \mathbf{S} \xrightarrow{\beta} \mathbf{T} \ \Rightarrow \ \mathbf{R} \xrightarrow{\alpha \circ \beta} \mathbf{T}$

and generally composable if also:

2. $\mathbf{R} \xrightarrow{\mathrm{id}} \mathbf{R}$
3. $\mathbf{R} \xrightarrow{\alpha} \mathbf{S} \ \Rightarrow \ \mathbf{R} \| \mathbf{T} \xrightarrow{\alpha | \mathrm{id}} \mathbf{S} \| \mathbf{T}$

Constructions and composability

resource set $\langle \Omega, \; \| \rangle$, **constructor set** $\langle \Gamma, \; \circ, \; | \; , \mathrm{id} \rangle$

Construction = subset of $\Omega \times \Gamma \times \Omega$

Possible notation: $\mathbf{R} \xrightarrow{\alpha} \mathbf{S}$

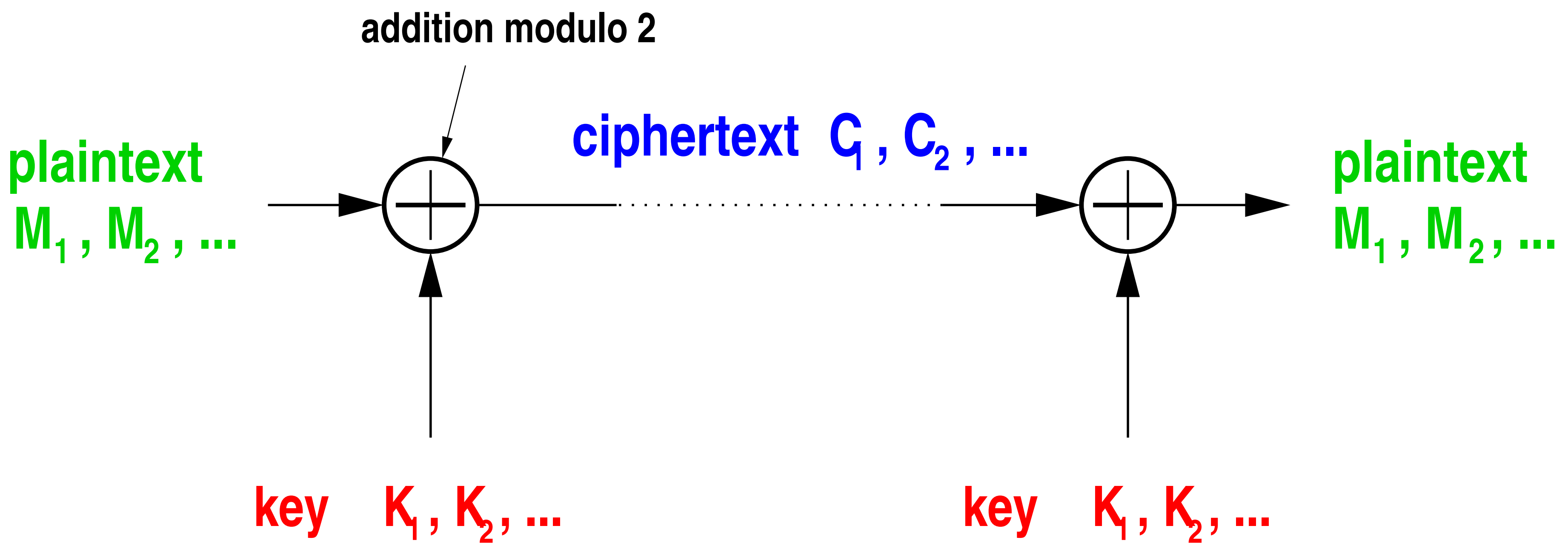
Definition: **A construction is serially composable if**

1. $\mathbf{R} \xrightarrow{\alpha} \mathbf{S} \;\wedge\; \mathbf{S} \xrightarrow{\beta} \mathbf{T} \;\Rightarrow\; \mathbf{R} \xrightarrow{\alpha\circ\beta} \mathbf{T}$

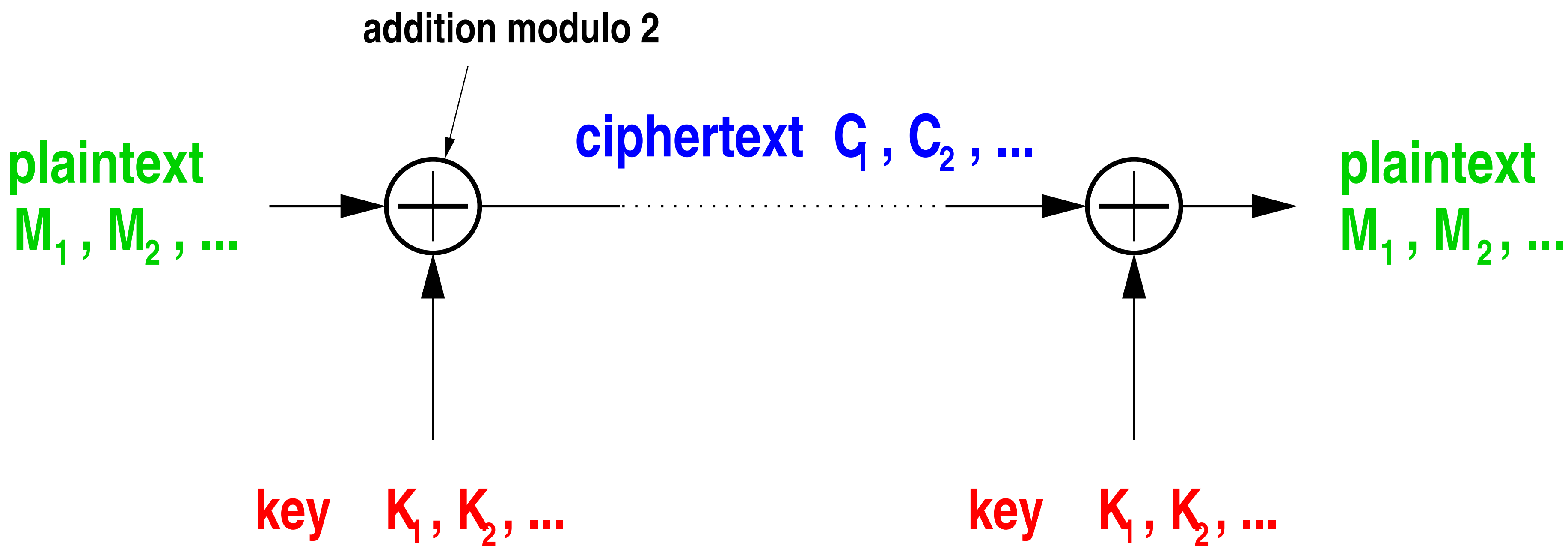
and generally composable if also:

2. $\mathbf{R} \xrightarrow{\mathrm{id}} \mathbf{R}$
3. $\mathbf{R} \xrightarrow{\alpha} \mathbf{S} \;\Rightarrow\; \mathbf{R}\|\mathbf{T} \xrightarrow{\alpha|\mathrm{id}} \mathbf{S}\|\mathbf{T} \;\wedge\; \mathbf{T}\|\mathbf{R} \xrightarrow{\mathrm{id}|\alpha} \mathbf{T}\|\mathbf{S}$

One-time pad: A constructive perspective



One-time pad: A constructive perspective

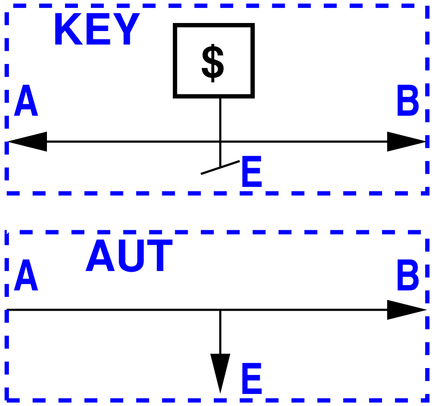


Perfect secrecy (Shannon):

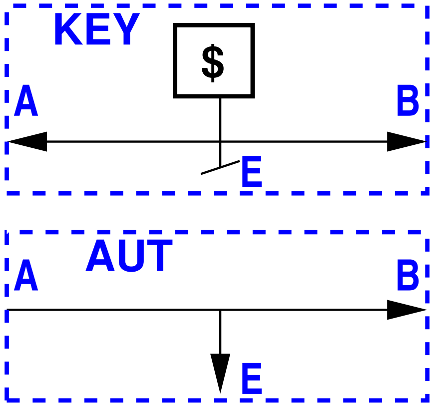
C and **M** are statistically independent.

One-time pad in constructive cryptography

One-time pad in constructive cryptography

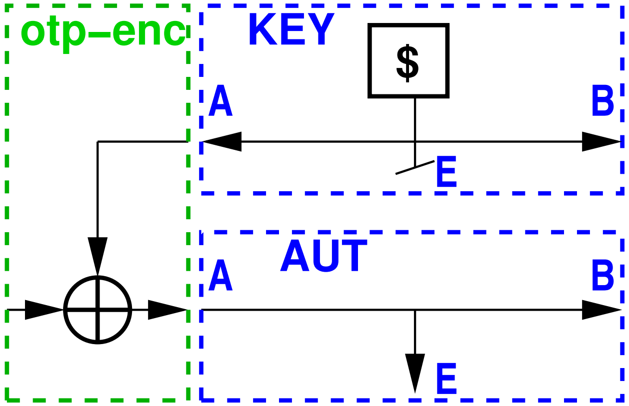


One-time pad in constructive cryptography



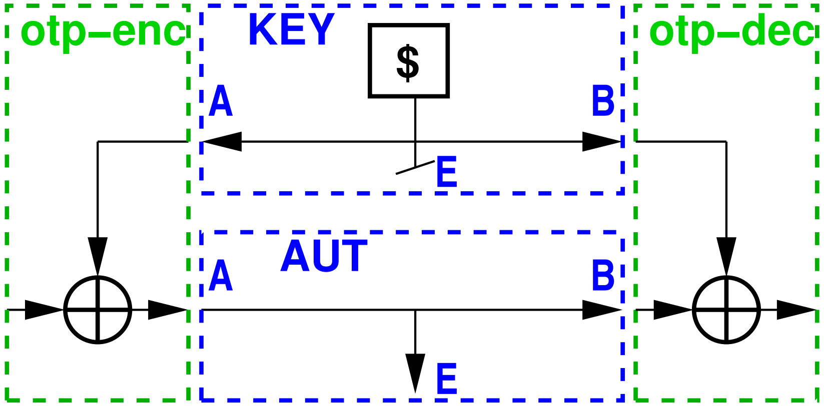
[**KEY**,**AUT**]

One-time pad in constructive cryptography



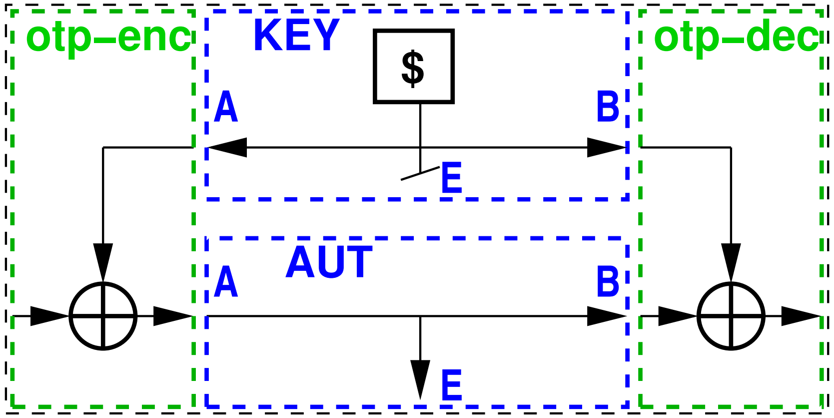
otp-enc^{A} [KEY,AUT]

One-time pad in constructive cryptography



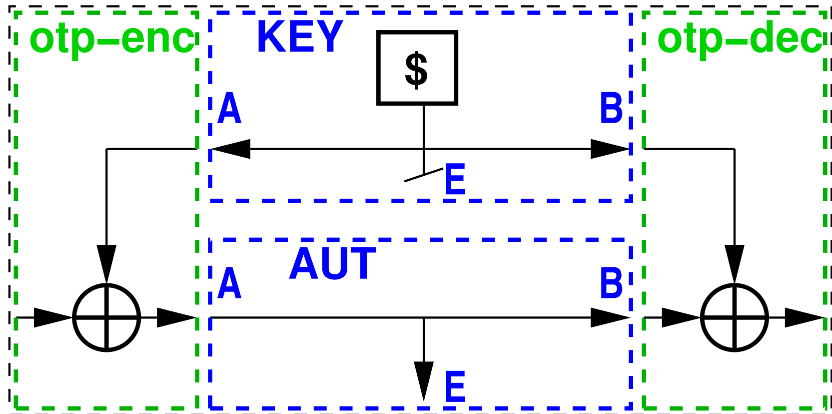
otp-dec^B otp-enc^A [KEY, AUT]

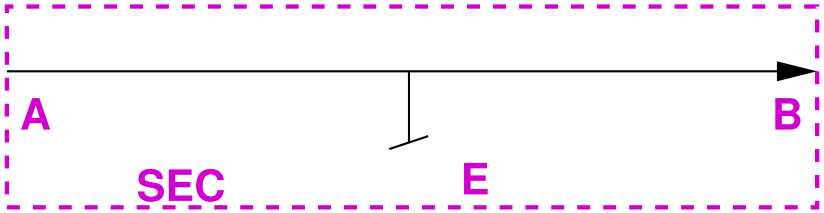
One-time pad in constructive cryptography



$\textbf{otp-dec}^{B}\ \textbf{otp-enc}^{A}\ [\textbf{KEY}, \textbf{AUT}]$

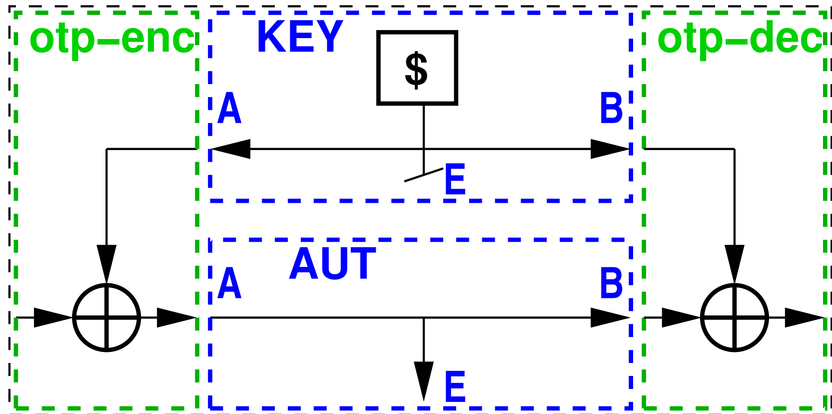
One-time pad in constructive cryptography

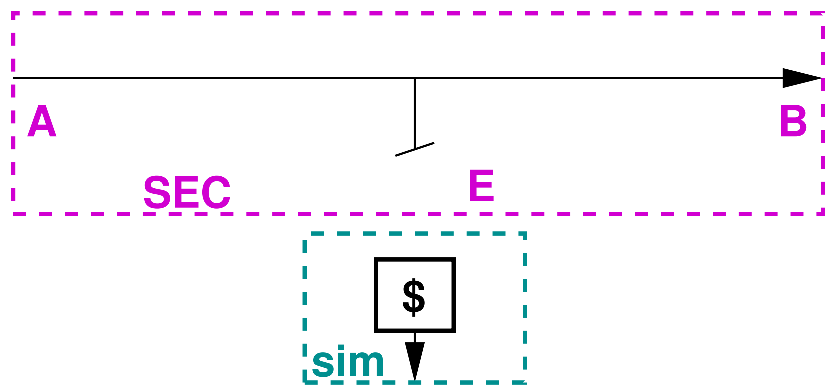




$\text{otp-dec}^{B}\ \text{otp-enc}^{A}\ [\text{KEY}, \text{AUT}]$ SEC

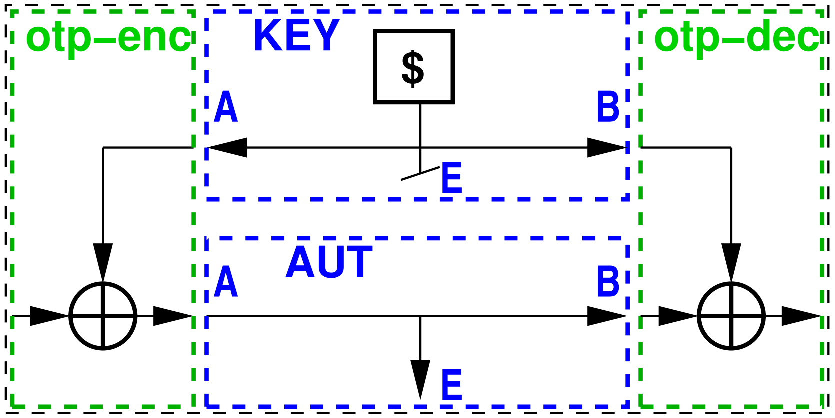
One-time pad in constructive cryptography

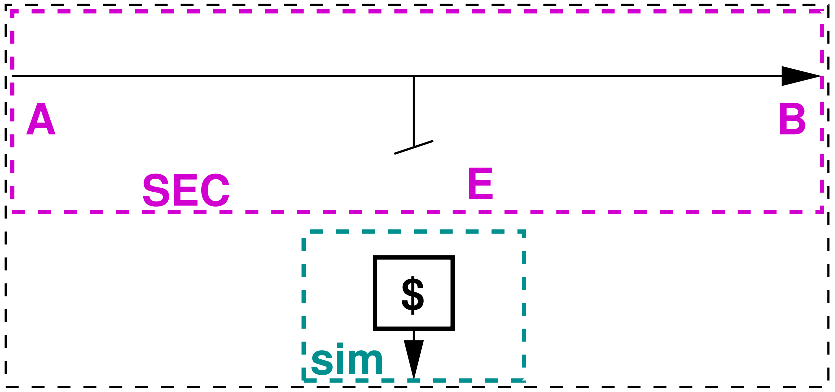




$\text{otp-dec}^{B}\ \text{otp-enc}^{A}\ [\mathbf{KEY},\mathbf{AUT}]$ $\qquad$ $\text{sim}^{E}\ \mathbf{SEC}$

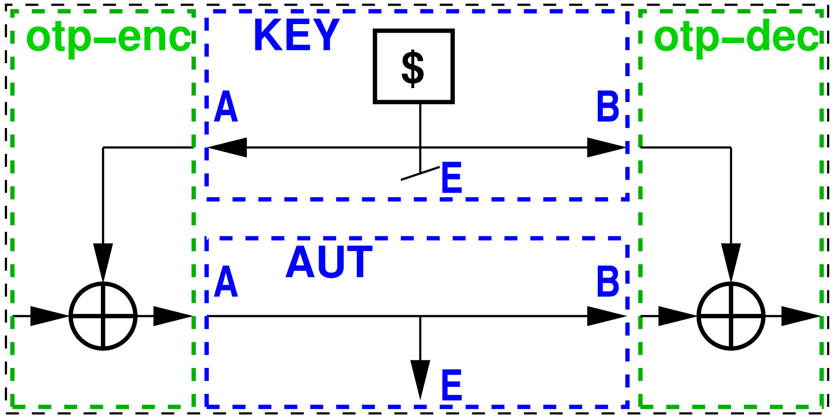
One-time pad in constructive cryptography

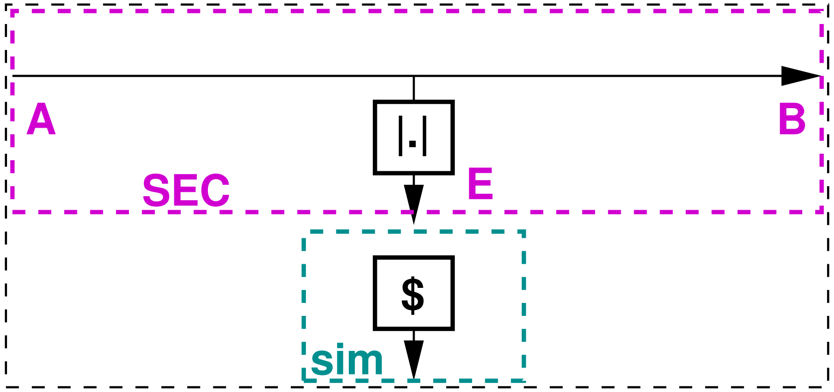




$$\text{otp-dec}^{B}\ \text{otp-enc}^{A}\,[\text{KEY},\text{AUT}] \equiv \text{sim}^{E}\ \text{SEC}$$

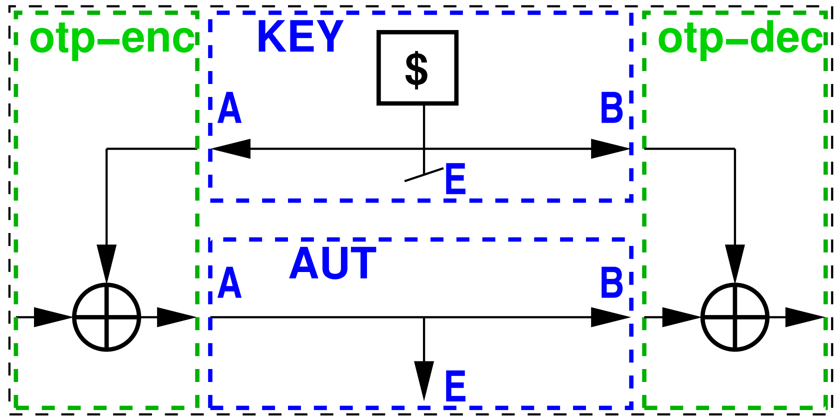
One-time pad in constructive cryptography

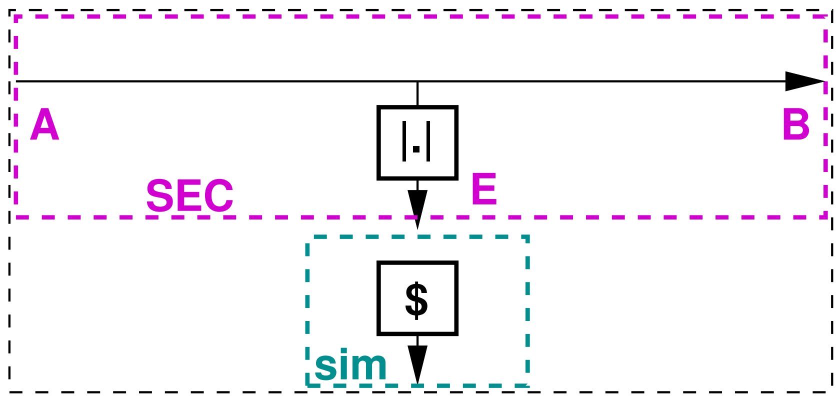




$$\text{otp-dec}^{B}\ \text{otp-enc}^{A}\,[\text{KEY},\text{AUT}] \equiv \text{sim}^{E}\ \text{SEC}$$

One-time pad in constructive cryptography

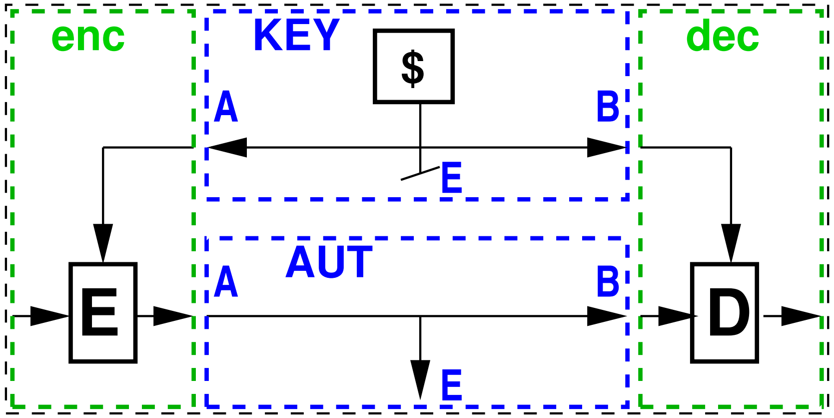


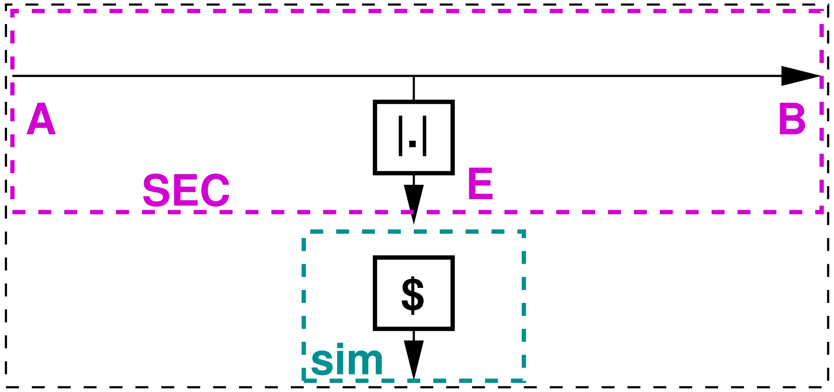


$$\text{otp-dec}^{B}\ \text{otp-enc}^{A}\ [\text{KEY}, \text{AUT}] \equiv \text{sim}^{E}\ \text{SEC}$$

written as a construction: $[\text{KEY}, \text{AUT}] \xrightarrow{\text{OTP}} \text{SEC}$

Symmetric encryption in CC





$$\mathrm{dec}^B\,\mathrm{enc}^A\,[\mathrm{KEY},\mathrm{AUT}] \quad \approx \quad \mathrm{sim}^E\,\mathrm{SEC}$$

written as a construction: $[\mathrm{KEY},\mathrm{AUT}] \xrightarrow{\mathrm{SYM}} \mathrm{SEC}$

Levels of abstraction in AC

#	main concept	concepts treated at this level
0.	Constructions	composability, construction trees
0'.	Games	isomorphism
1.	Abstract systems	cryptographic algebras
2.	Discrete systems	indistinguishability proofs
3.	System implem.	complexity, efficiency, asymptotics

Levels of abstraction in AC

#	main concept	concepts treated at this level
0.	Constructions	composability, construction trees
0'.	Games	isomorphism
1.	Abstract systems	cryptographic algebras
2.	Discrete systems	indistinguishability proofs
3.	System implem.	complexity, efficiency, asymptotics

Constructive cryptography

Resource **S** is constructed from **R** by protocol π:

$$\mathbf{R} \xrightarrow{\pi} \mathbf{S}$$

Constructive cryptography

Resource **S** is constructed from **R** by protocol π:

$$\mathbf{R} \xrightarrow{\pi} \mathbf{S}$$

Example: Alice-Bob-Eve setting, $\pi = (\pi_1, \pi_2)$

Constructive cryptography

Resource **S** is constructed from **R** by protocol π:

$$\mathbf{R} \xrightarrow{\pi} \mathbf{S}$$

Example: Alice-Bob-Eve setting, $\pi = (\pi_1, \pi_2)$

$$\mathbf{R} \xrightarrow{\pi} \mathbf{S} \quad :\Leftrightarrow \quad \exists \sigma : \ \pi_1^{\mathsf{A}} \pi_2^{\mathsf{B}} \mathbf{R} \approx \sigma^{\mathsf{E}} \mathbf{S}$$

Constructive cryptography

Resource $\mathbf{S}$ is constructed from $\mathbf{R}$ by protocol π:

$$\mathbf{R} \xrightarrow{\pi} \mathbf{S}$$

Example: Alice-Bob-Eve setting, $\pi = (\pi_1, \pi_2)$

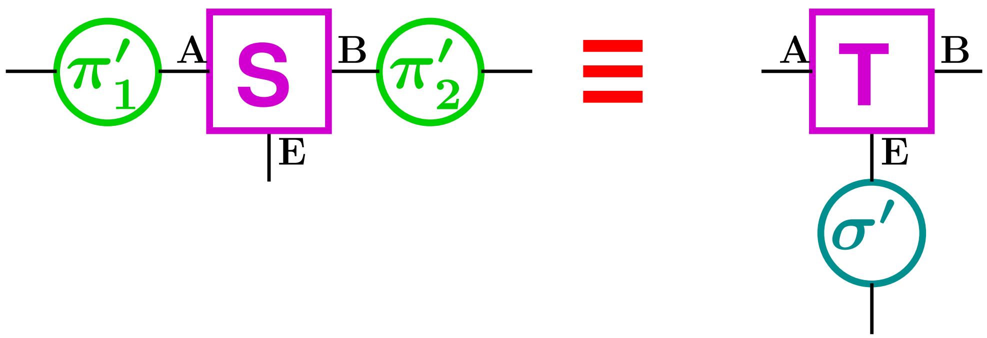
$$\mathbf{R} \xrightarrow{\pi} \mathbf{S} \quad :\Leftrightarrow \quad \exists \sigma : \ \pi_1^{\mathsf{A}} \pi_2^{\mathsf{B}} \mathbf{R} \approx \sigma^{\mathsf{E}} \mathbf{S}$$

and

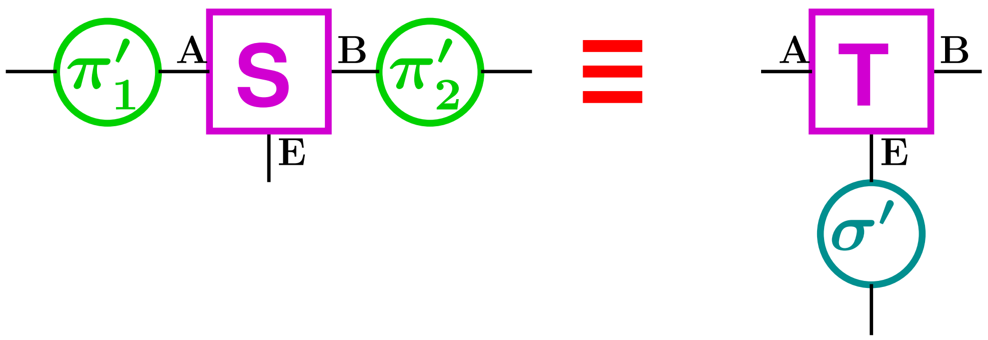
$$\pi_1^{\mathsf{A}} \pi_2^{\mathsf{B}} \perp^{\mathsf{E}} \mathbf{R} \approx \perp^{\mathsf{E}} \mathbf{S}$$

Proof of composition theorem for ABE-setting

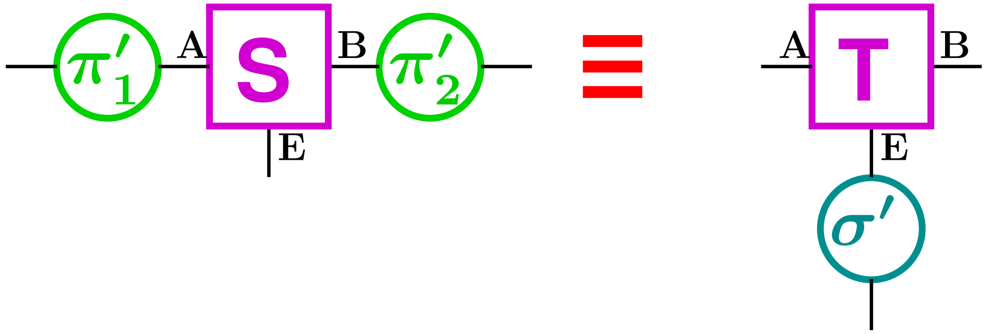
Proof of composition theorem for ABE-setting



Proof of composition theorem for ABE-setting

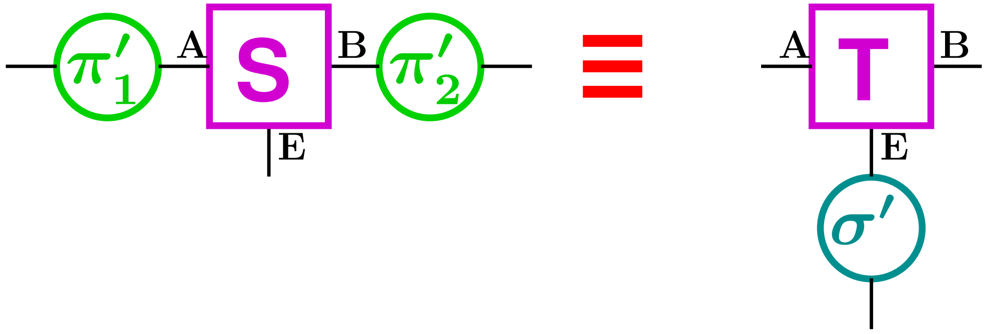


Proof of composition theorem for ABE-setting



Definition: **d** non-expanding: $\mathbf{d}(\alpha^i \mathbf{R}, \alpha^i \mathbf{S}) \leq \mathbf{d}(\mathbf{R}, \mathbf{S})$

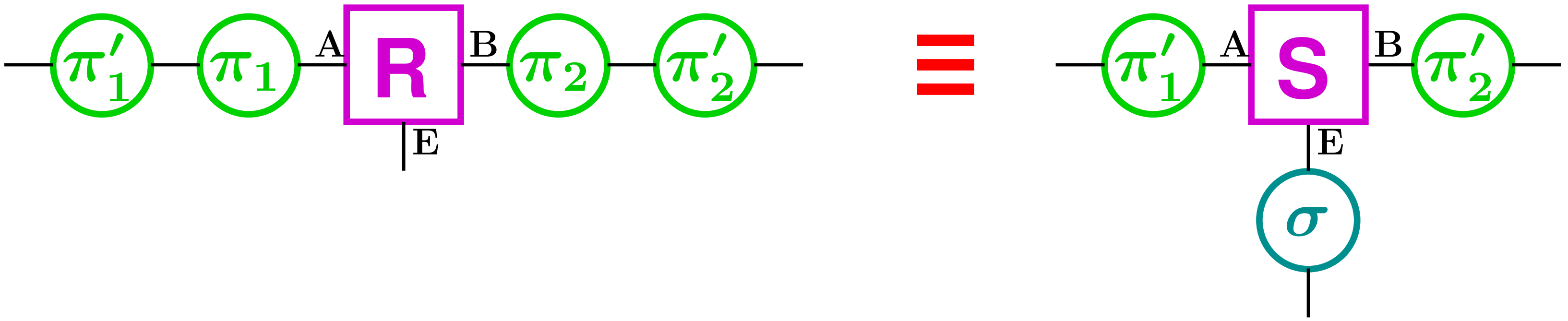
Proof of composition theorem for ABE-setting

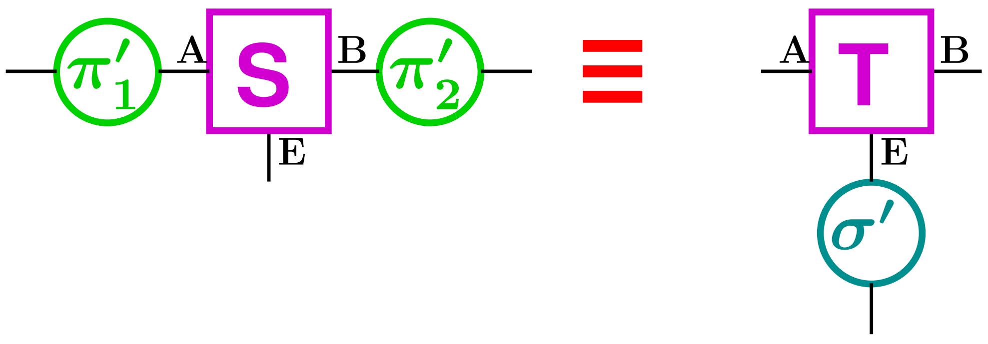


Definition: **d** non-expanding: **d**$(\alpha^i$**R**$,\alpha^i$**S**$) \leq$ **d**(**R**,**S**)

Example: efficient (e.g. poly-time) implementable systems

Proof of composition theorem for ABE-setting

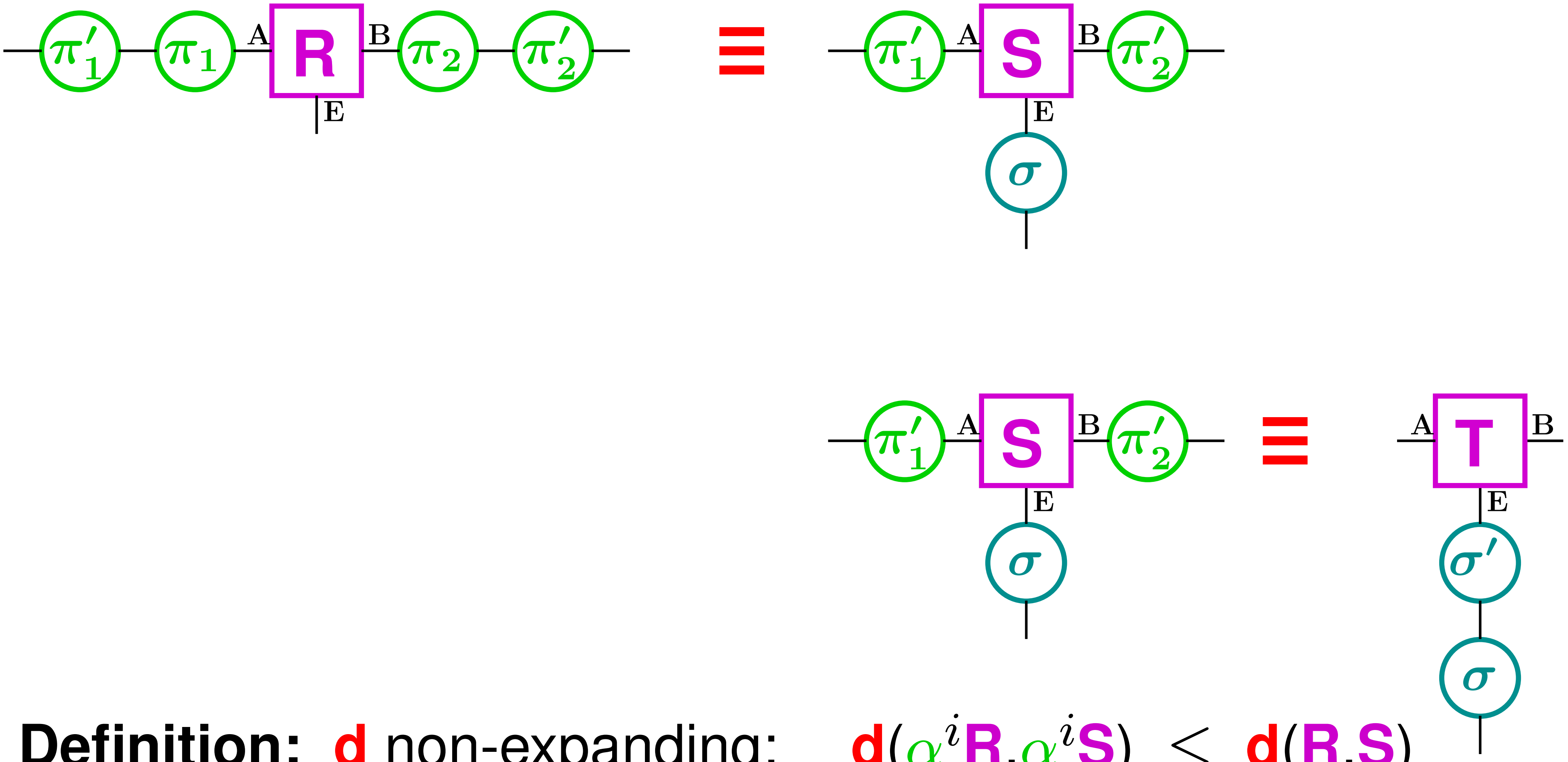




Definition: **d** non-expanding: $\mathbf{d}(\alpha^i \mathbf{R}, \alpha^i \mathbf{S}) \leq \mathbf{d}(\mathbf{R}, \mathbf{S})$

Example: efficient (e.g. poly-time) implementable systems

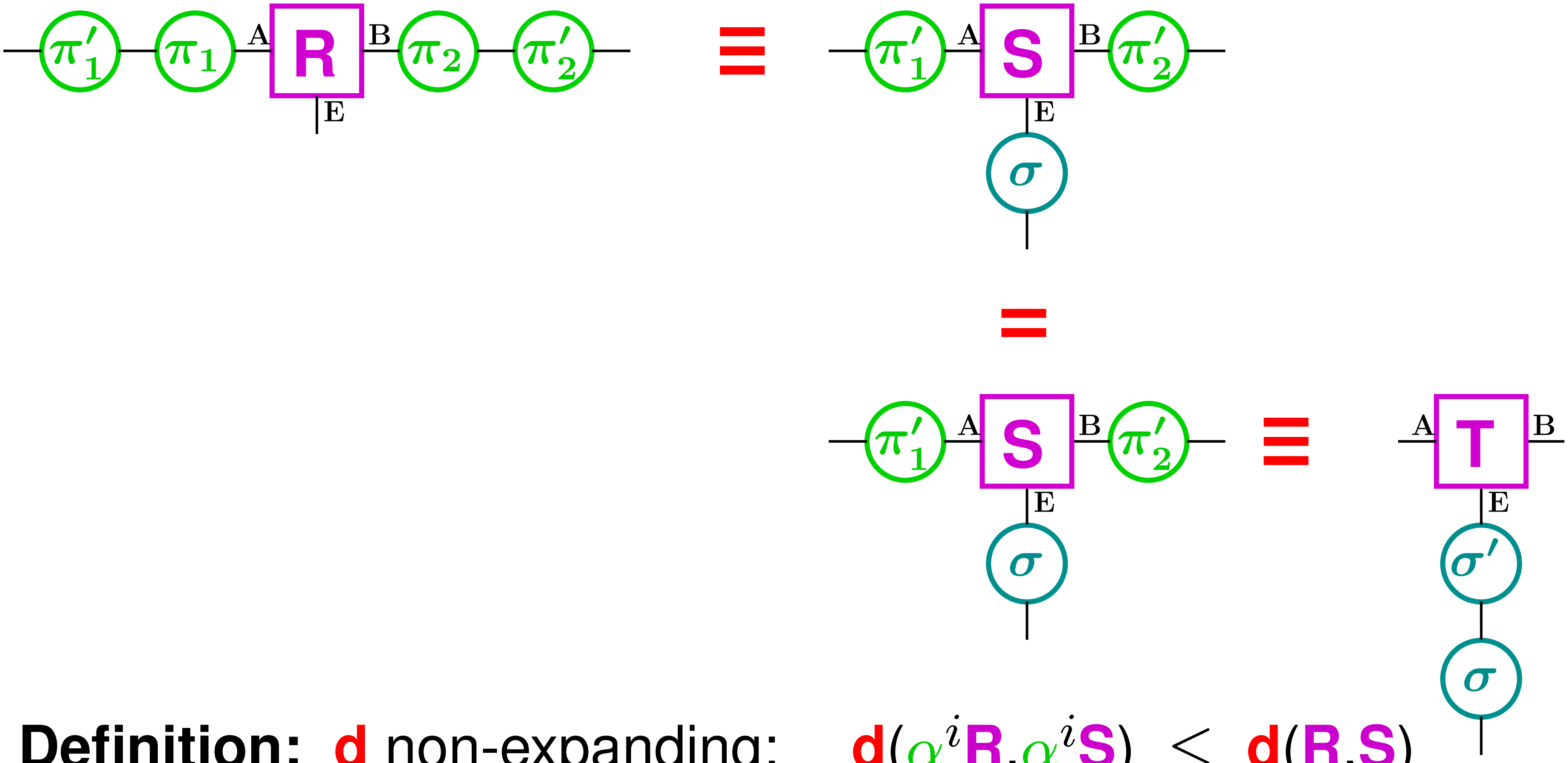
Proof of composition theorem for ABE-setting



Definition: **d** non-expanding: $\mathbf{d}(\alpha^i\mathbf{R},\alpha^i\mathbf{S}) \leq \mathbf{d}(\mathbf{R},\mathbf{S})$

Example: efficient (e.g. poly-time) implementable systems

Proof of composition theorem for ABE-setting



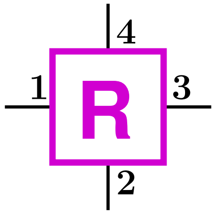
Definition: **d** non-expanding: $\mathbf{d}(\alpha^i\mathbf{R},\alpha^i\mathbf{S}) \leq \mathbf{d}(\mathbf{R},\mathbf{S})$

Example: efficient (e.g. poly-time) implementable systems

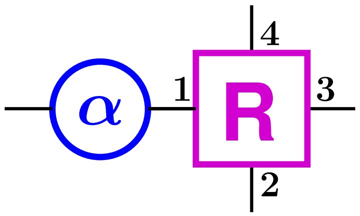
Levels of abstraction in AC

#	main concept	concepts treated at this level
0.	Constructions	composability, construction trees
0’.	Games	isomorphism
1.	Abstract systems	cryptographic algebras
2.	Discrete systems	indistinguishability proofs
3.	System implem.	complexity, efficiency, asymptotics

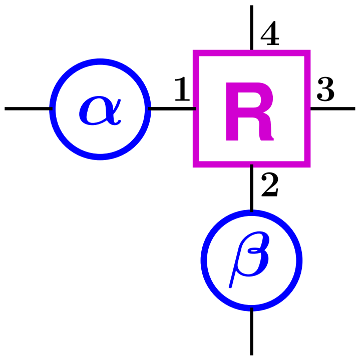
Cryptographic algebra $\langle \Phi, \Sigma \rangle$



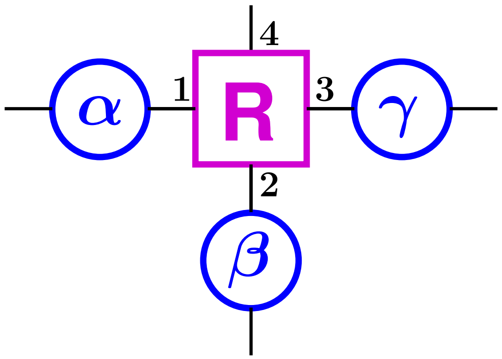
Cryptographic algebra $\langle \Phi, \Sigma \rangle$



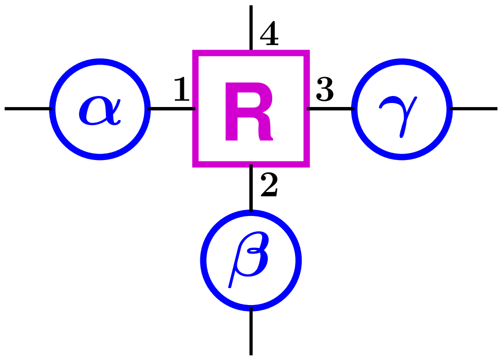
Cryptographic algebra $\langle \Phi, \Sigma \rangle$



Cryptographic algebra $\langle \Phi, \Sigma \rangle$



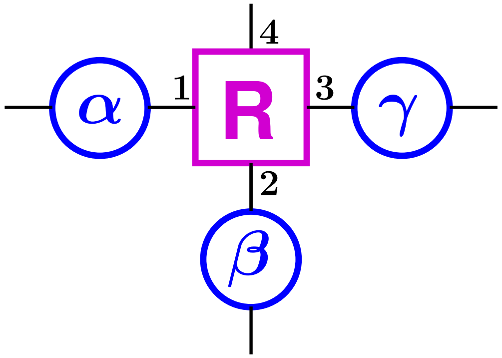
Cryptographic algebra $\langle \Phi, \Sigma \rangle$



Resource set Φ (here for interface set $\mathcal{I} = \{1, 2, 3, 4\}$)

Converter set Σ

Cryptographic algebra $\langle \Phi, \Sigma \rangle$



Resource set Φ (here for interface set $\mathcal{I} = \{1, 2, 3, 4\}$)

Converter set Σ

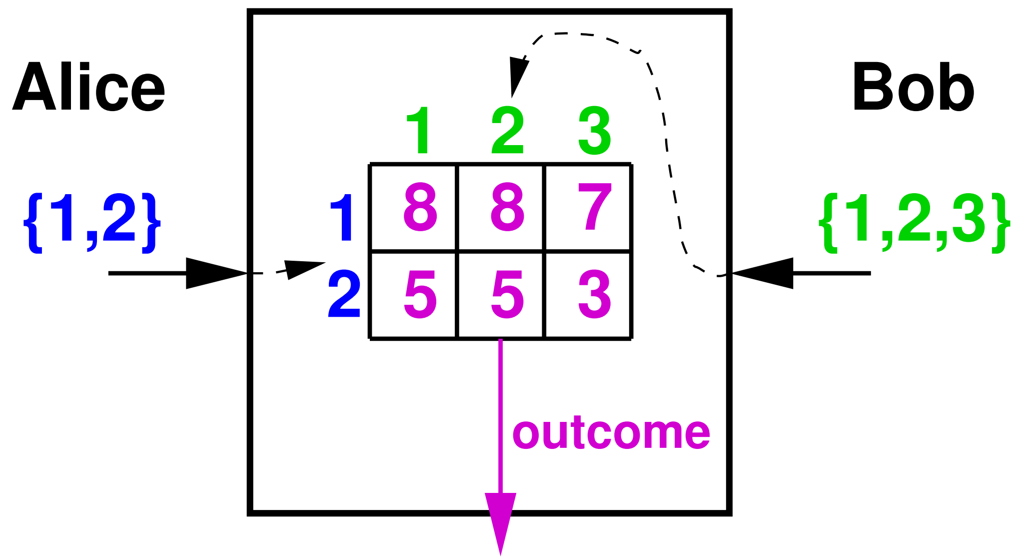
Algebraic laws:

- $\alpha^i \mathbf{R} \in \Phi$ for all $\mathbf{R} \in \Phi$, $\alpha \in \Sigma$, $i \in \mathcal{I}$
- $\alpha^i \beta^j \mathbf{R} = \beta^j \alpha^i \mathbf{R}$ for all $i \neq j$
- $1^i \mathbf{R} = \mathbf{R}$ for all i

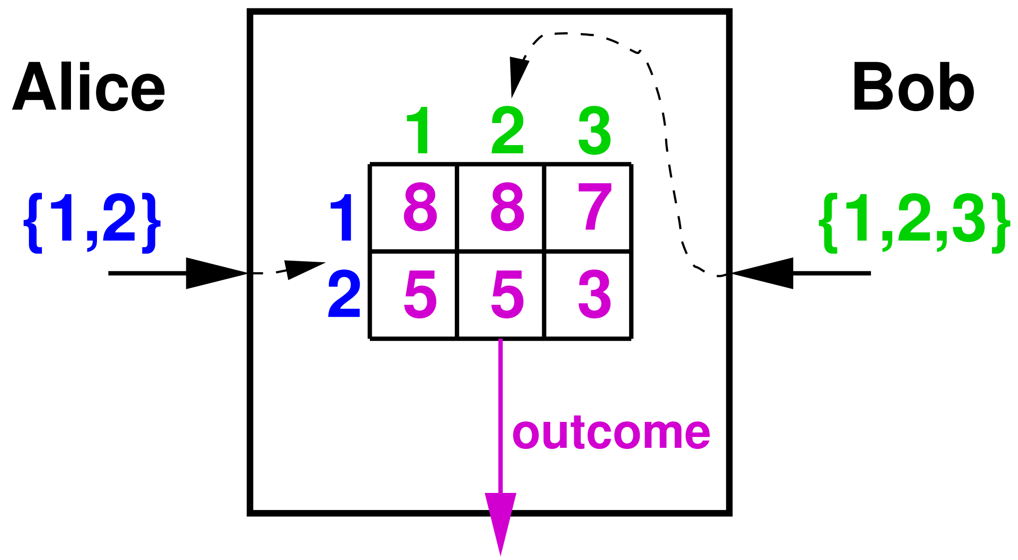
Levels of abstraction in AC

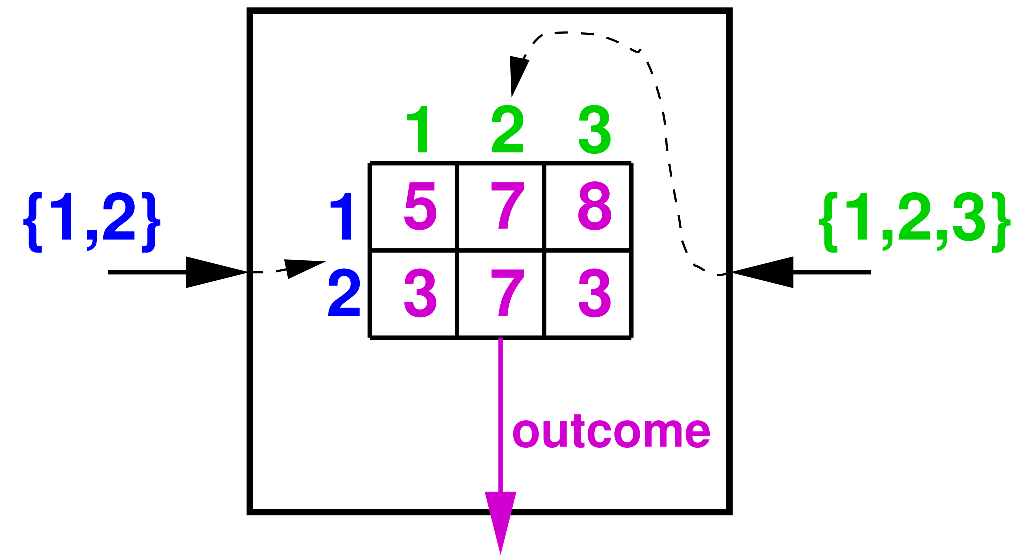
#	main concept	concepts treated at this level
0.	Constructions	composability, construction trees
0'.	Games	isomorphism
1.	Abstract systems	cryptographic algebras
2.	Discrete systems	indistinguishability proofs
3.	System implem.	complexity, efficiency, asymptotics

Games and isomorphisms

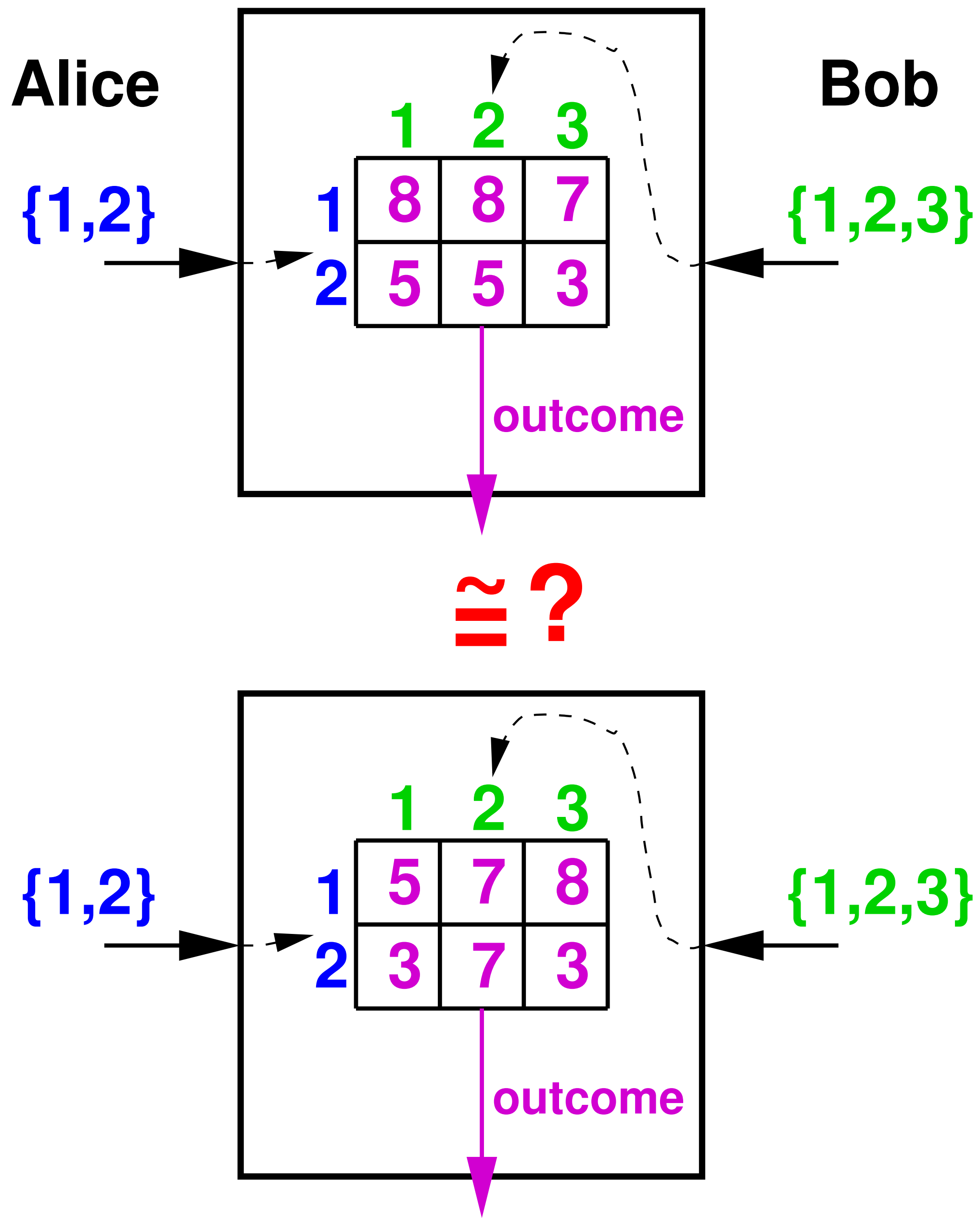


Games and isomorphisms

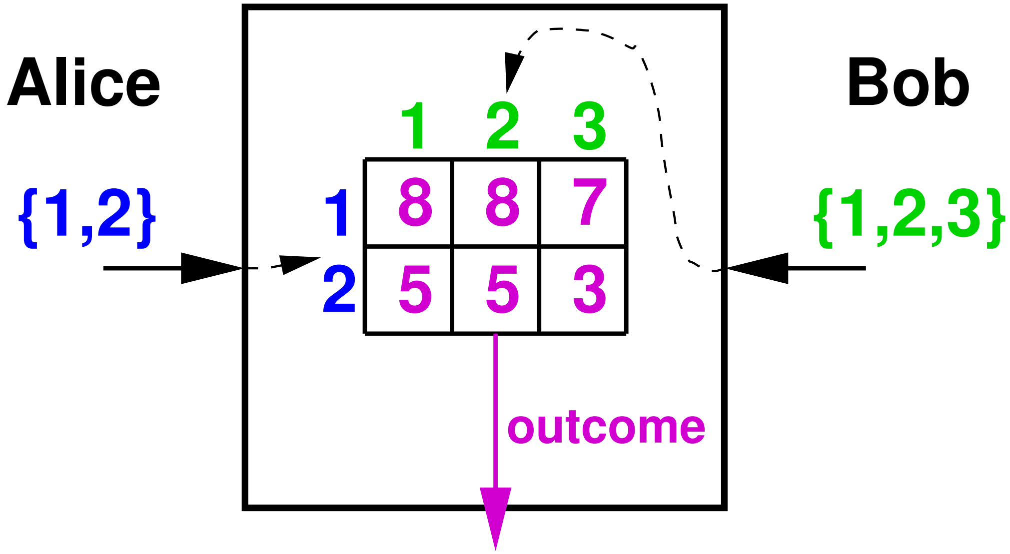




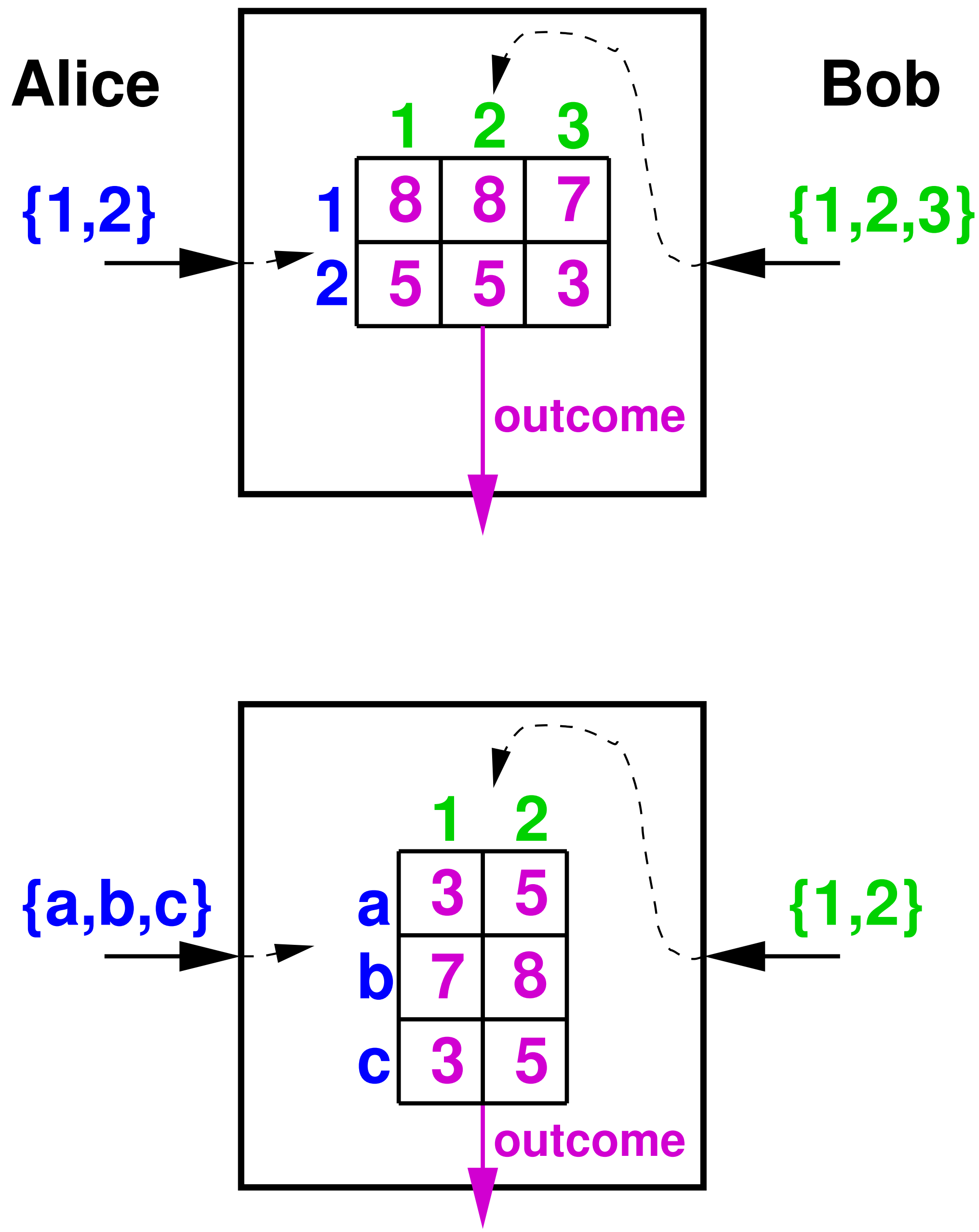
Games and isomorphisms



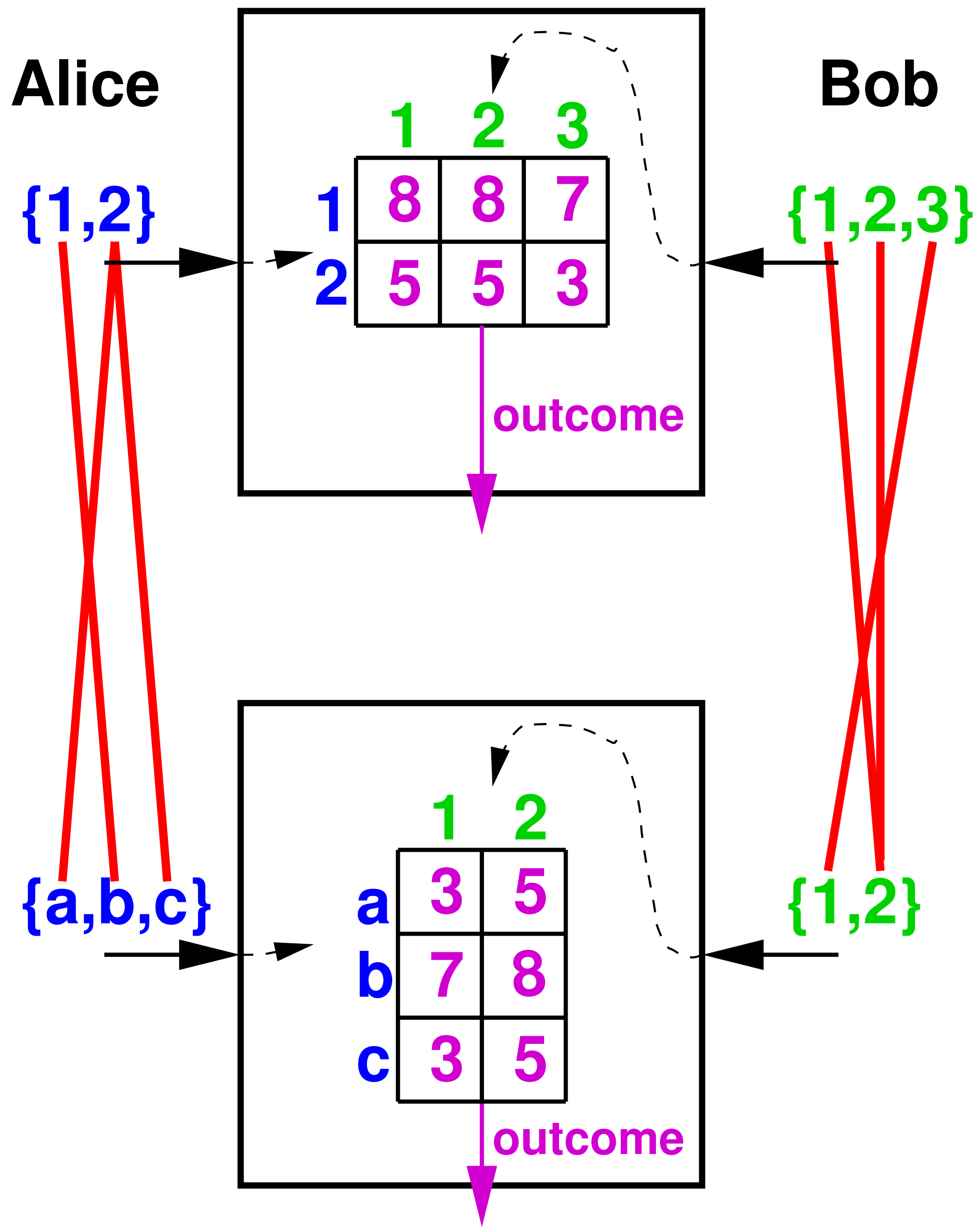
Games and isomorphisms



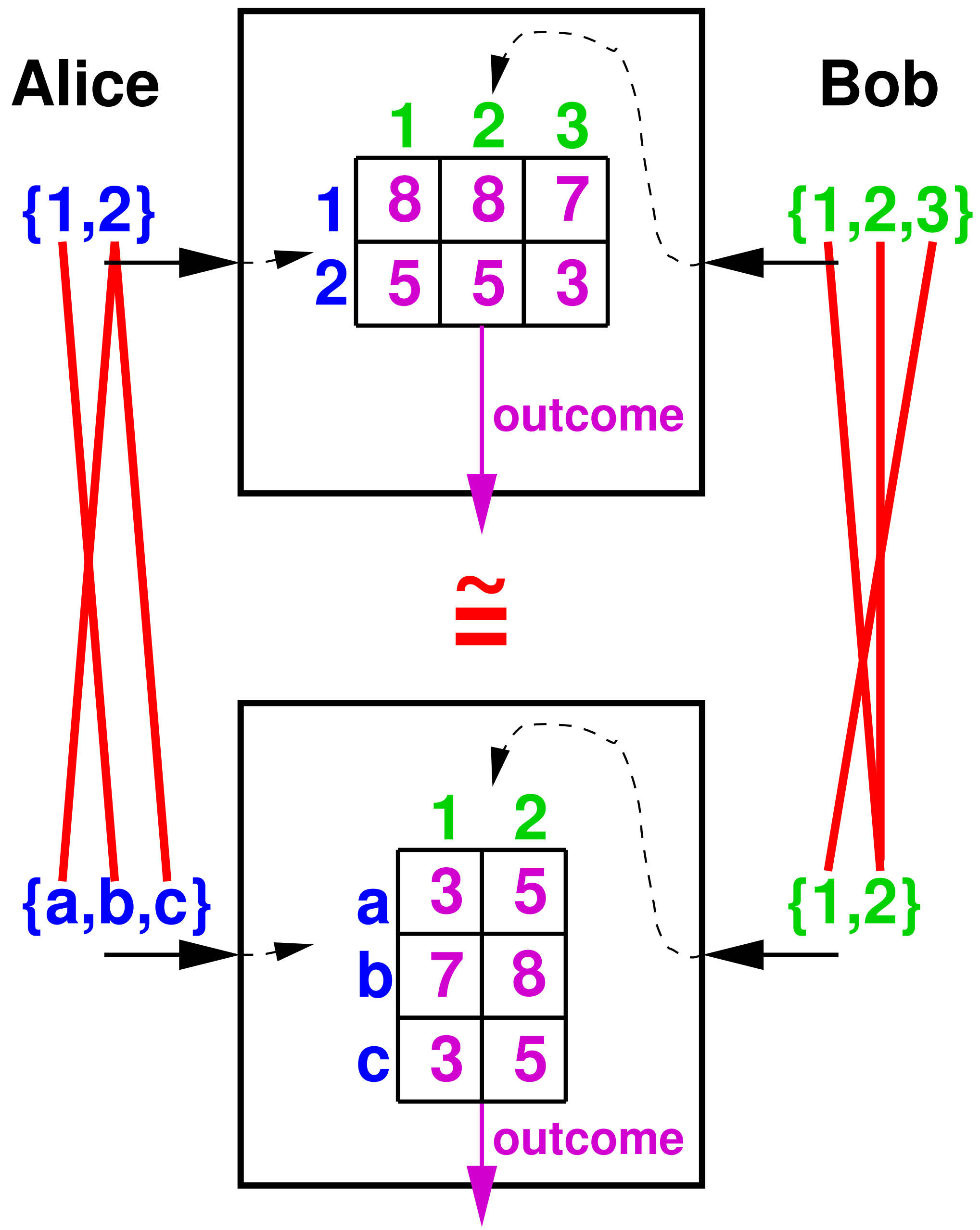
Games and isomorphisms



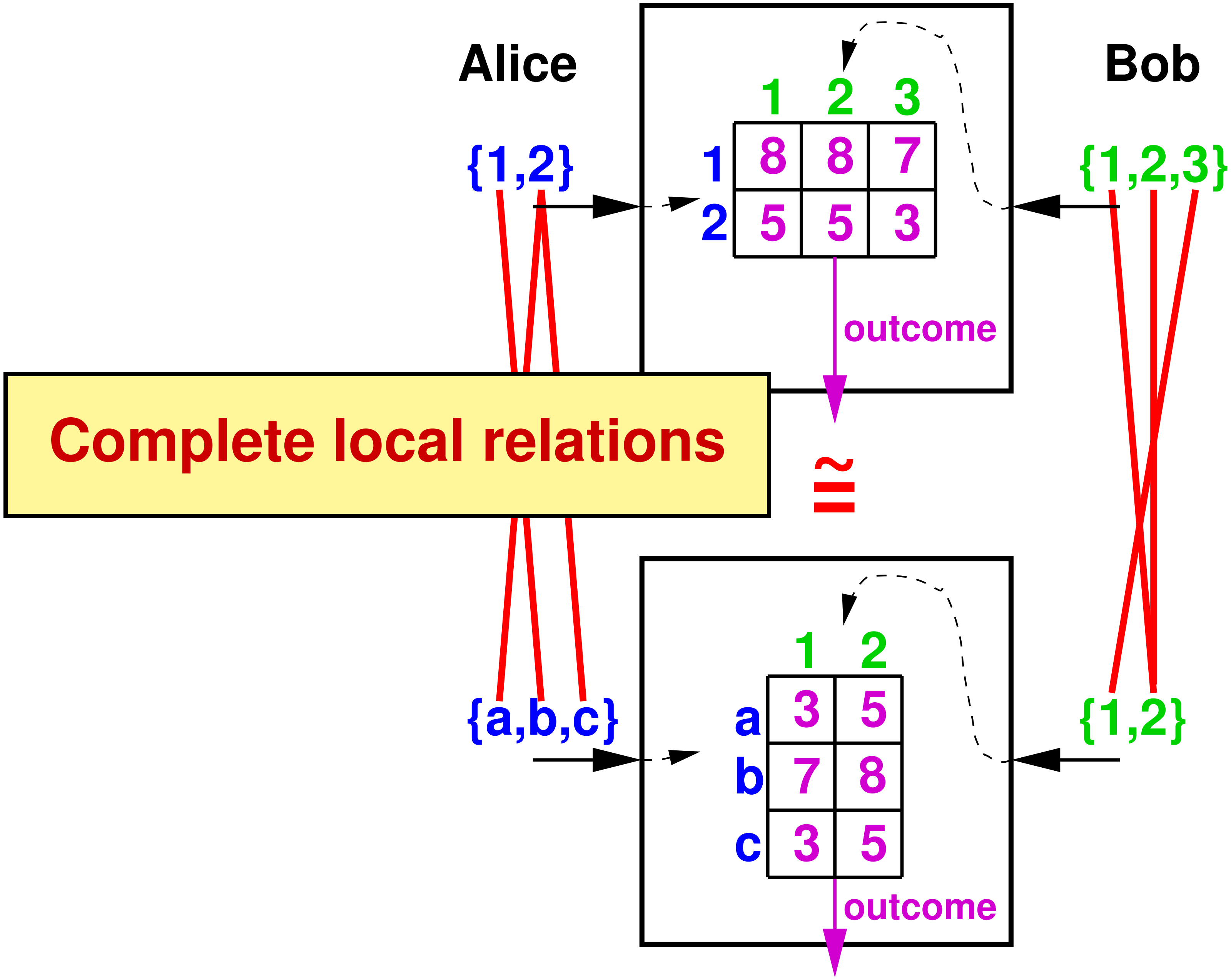
Games and isomorphisms



Games and isomorphisms



Games and isomorphisms



Levels of abstraction in AC

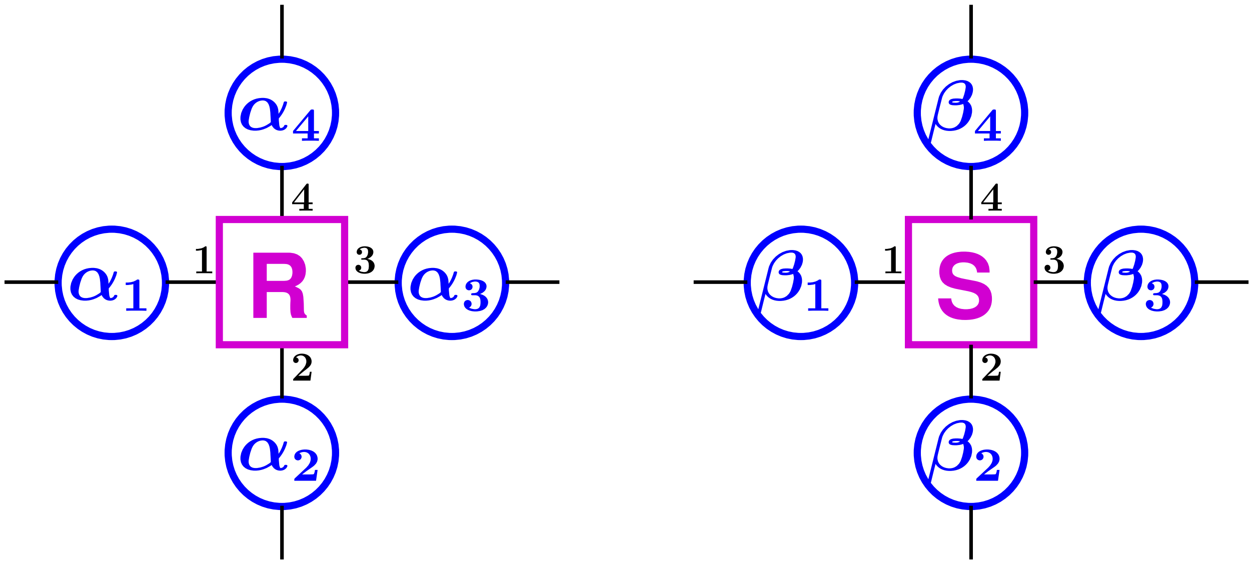
#	main concept	concepts treated at this level
0.	Constructions	composability, construction trees
0’.	Games	isomorphism
1.	Abstract systems	cryptographic algebras
2.	Discrete systems	indistinguishability proofs
3.	System implem.	complexity, efficiency, asymptotics

Levels of abstraction in AC

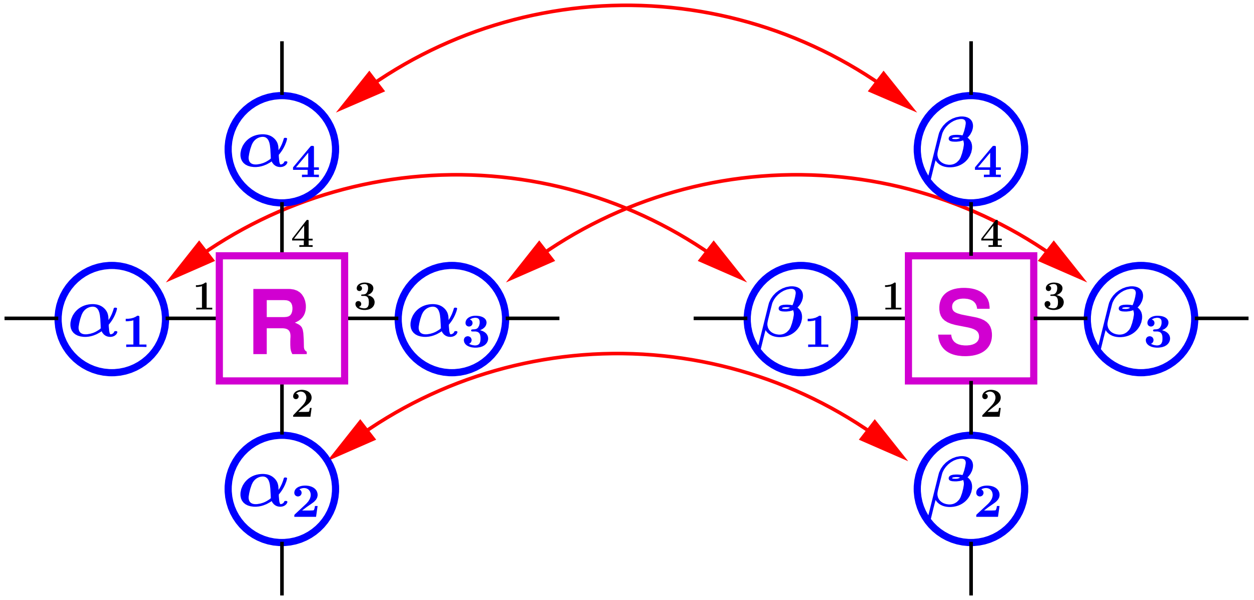
#	main concept	concepts treated at this level
0.	Constructions	composability, construction trees
0'.	Games	isomorphism
1.	Abstract systems	cryptographic algebras
2.	Discrete systems	indistinguishability proofs
3.	System implem.	complexity, efficiency, asymptotics

Resource isomorphisms

Resource isomorphisms

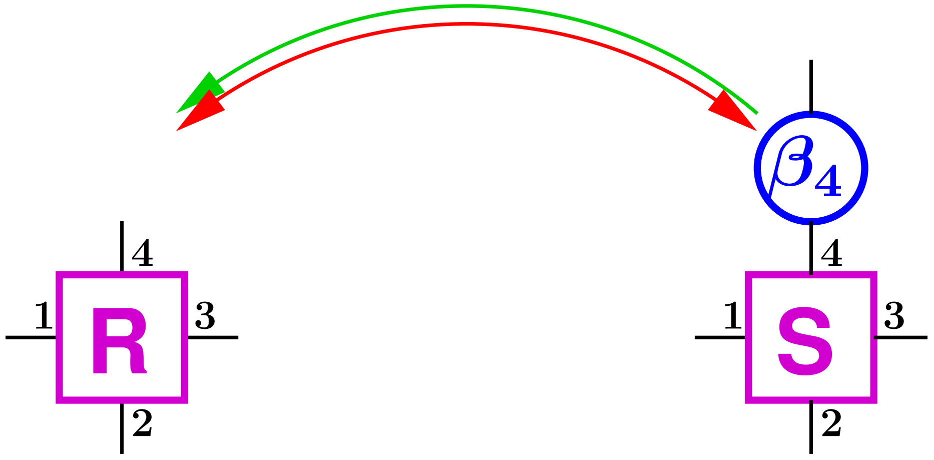


Resource isomorphisms



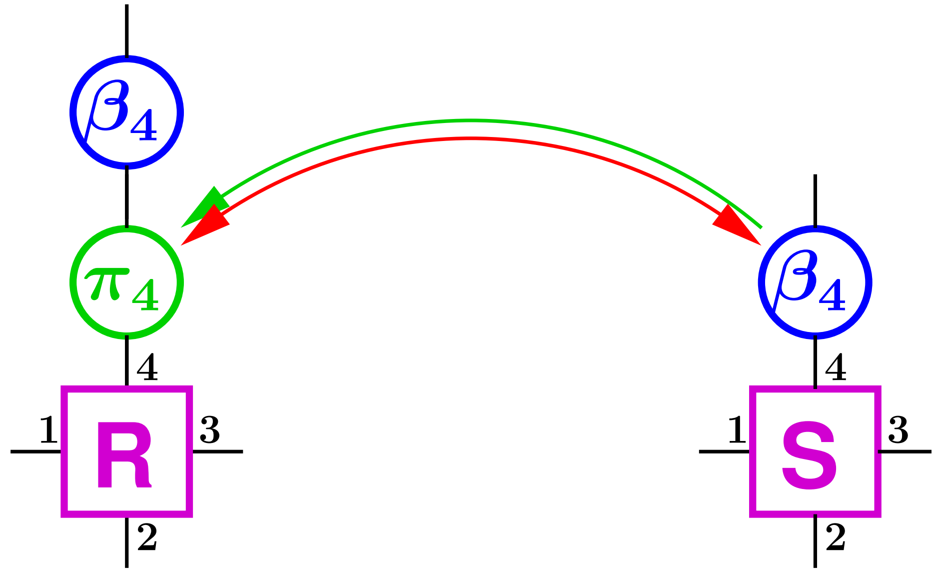
Theorem: **R** is isomorphic to **S** ....

Resource isomorphisms



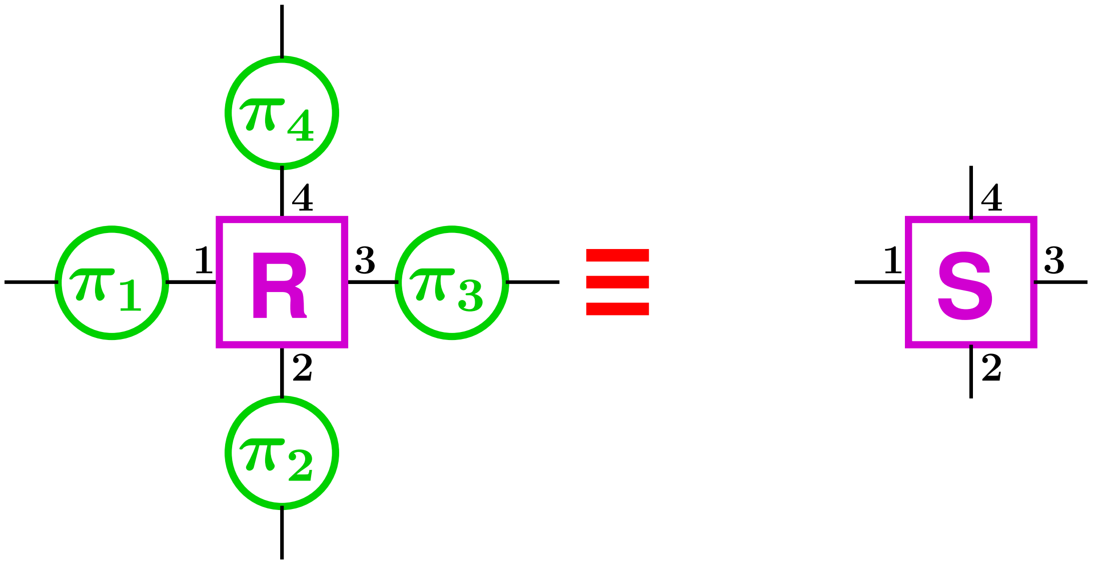
Theorem: **R** is isomorphic to **S** ....

Resource isomorphisms



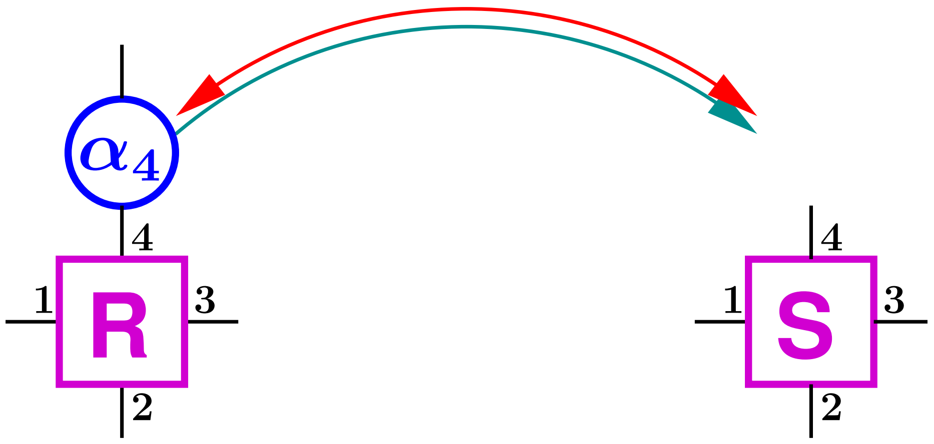
Theorem: **R** is isomorphic to **S** ....

Resource isomorphisms



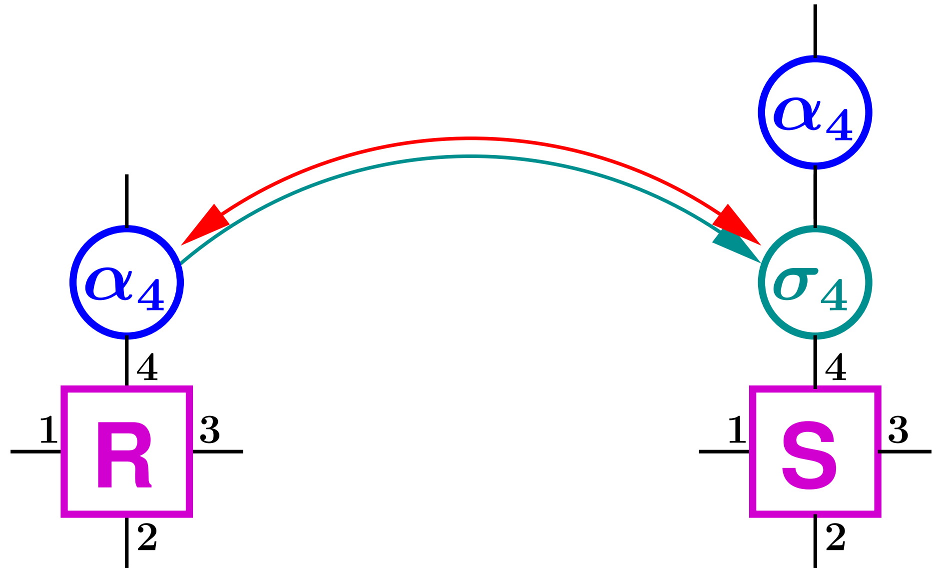
Theorem: **R** is isomorphic to **S** ....

Resource isomorphisms



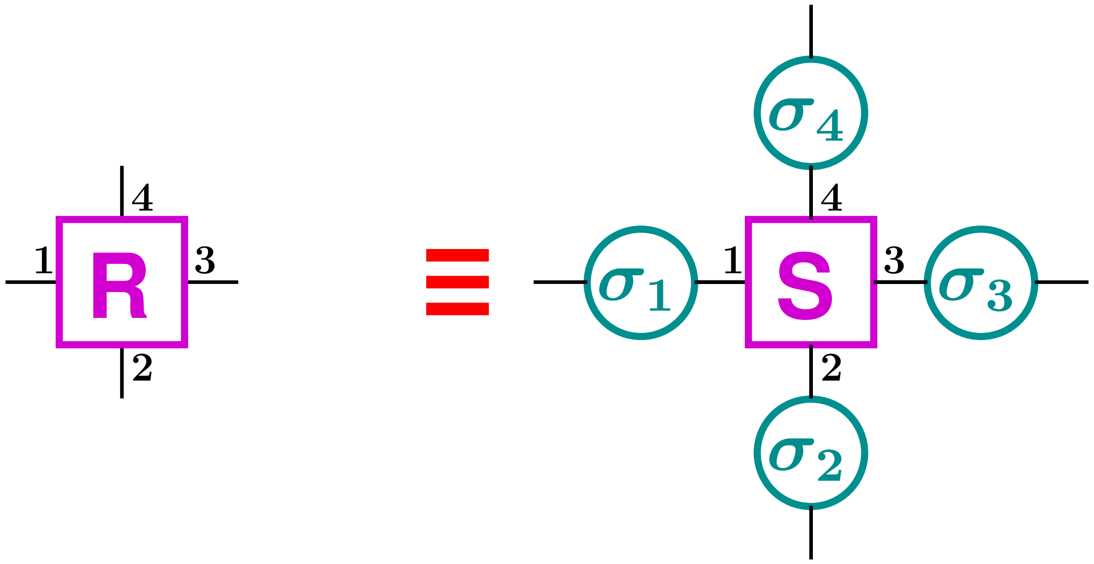
Theorem: **R** is isomorphic to **S** ....

Resource isomorphisms



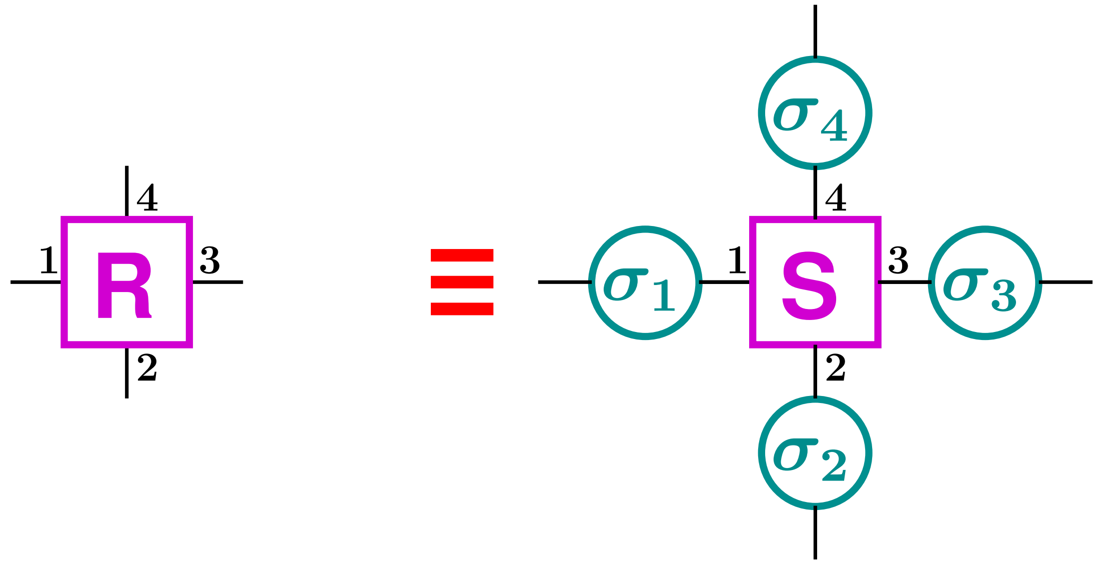
Theorem: **R** is isomorphic to **S** ....

Resource isomorphisms



Theorem: **R** is isomorphic to **S** ....

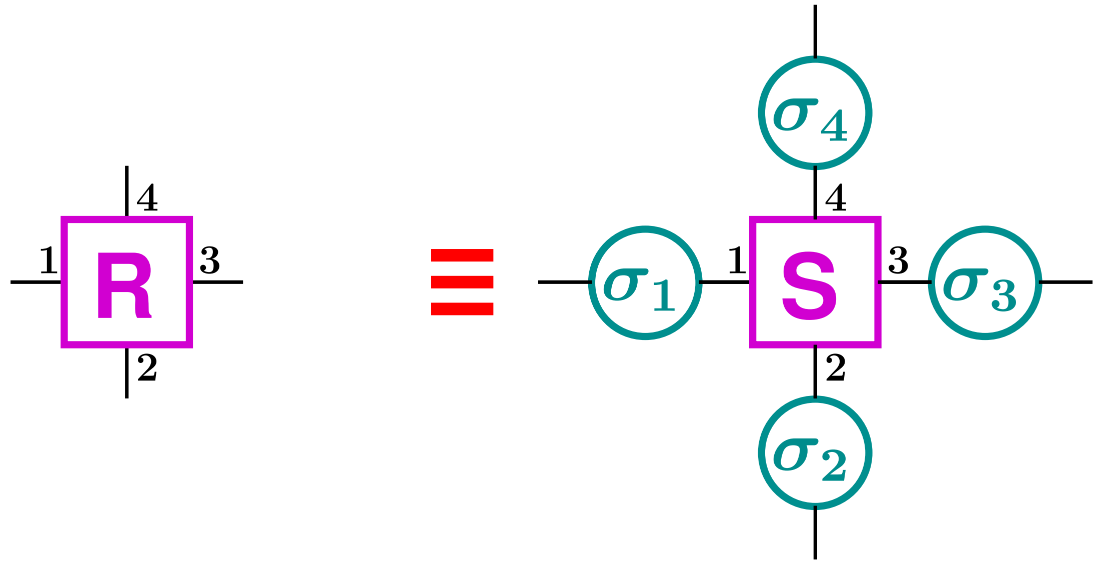
Resource isomorphisms



$$\pi = (\pi_1, \ldots, \pi_n) \qquad\qquad \sigma = (\sigma_1, \ldots, \sigma_n)$$

Theorem: **R** is isomorphic to **S** ....

Resource isomorphisms

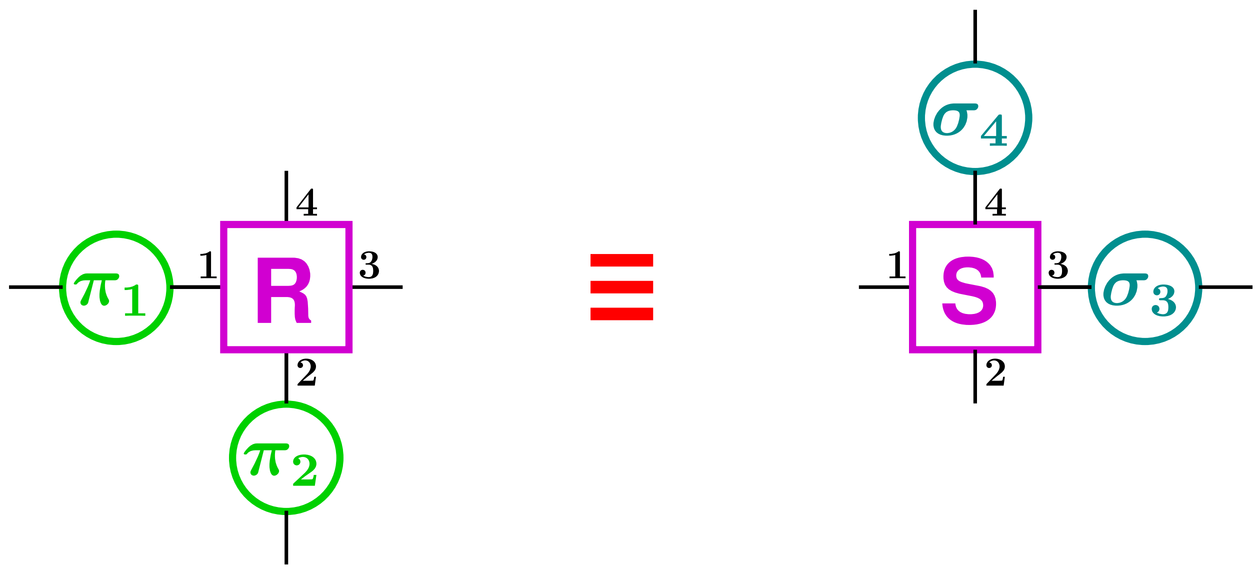


$$\pi = (\pi_1, \ldots, \pi_n) \qquad\qquad \sigma = (\sigma_1, \ldots, \sigma_n)$$

Theorem: **R** is isomorphic to **S** via π, denoted **R** $\cong^{\pi}$ **S**, if

$$\exists \boldsymbol{\sigma}\ \forall \mathcal{P} \subseteq \mathcal{I} : \qquad \boldsymbol{\pi}_{\mathcal{P}}\, \mathbf{R} \equiv \boldsymbol{\sigma}_{\overline{\mathcal{P}}}\, \mathbf{S}$$

Resource isomorphisms

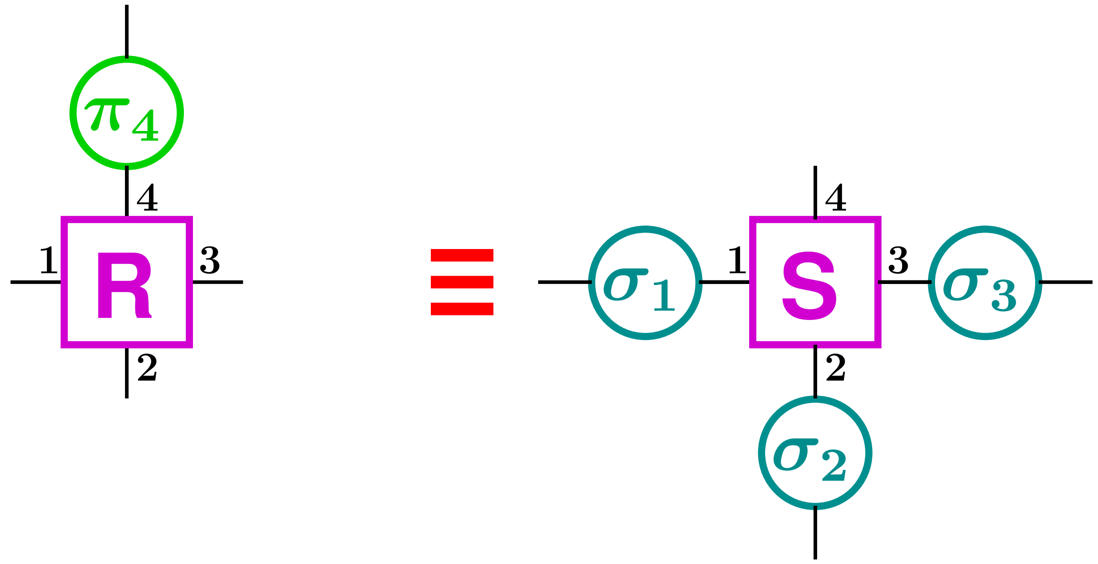


$$\pi = (\pi_1, \ldots, \pi_n) \qquad \sigma = (\sigma_1, \ldots, \sigma_n)$$

Theorem: **R** is isomorphic to **S** via π, denoted **R** $\cong^{\pi}$ **S**, if

$$\exists \boldsymbol{\sigma}\ \forall \mathcal{P} \subseteq \mathcal{I} : \qquad \boldsymbol{\pi}_{\mathcal{P}}\, \mathbf{R} \equiv \boldsymbol{\sigma}_{\overline{\mathcal{P}}}\, \mathbf{S}$$

Resource isomorphisms

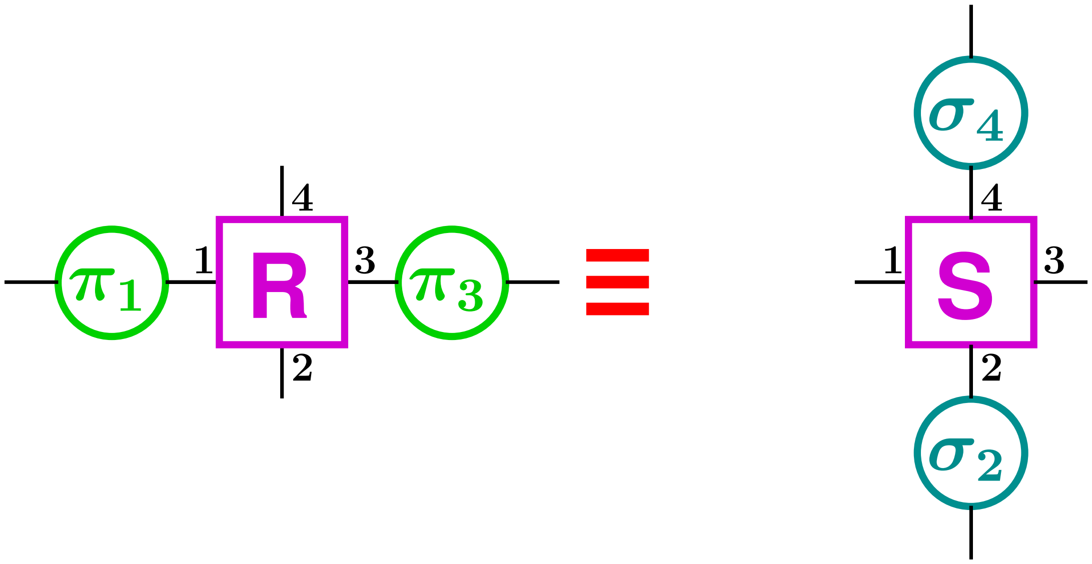


$$\pi = (\pi_1, \ldots, \pi_n) \qquad \sigma = (\sigma_1, \ldots, \sigma_n)$$

Theorem: **R** is isomorphic to **S** via π, denoted **R** $\cong^{\pi}$ **S**, if

$$\exists \boldsymbol{\sigma}\ \forall \mathcal{P} \subseteq \mathcal{I}: \qquad \boldsymbol{\pi}_{\mathcal{P}}\, \mathbf{R} \equiv \boldsymbol{\sigma}_{\overline{\mathcal{P}}}\, \mathbf{S}$$

Resource isomorphisms



$$\pi = (\pi_1, \ldots, \pi_n) \qquad \sigma = (\sigma_1, \ldots, \sigma_n)$$

Theorem: **R** is isomorphic to **S** via π, denoted **R** $\cong^{\pi}$ **S**, if

$$\exists \sigma\ \forall \mathcal{P} \subseteq \mathcal{I}: \qquad \pi_{\mathcal{P}}\,\mathbf{R} \equiv \sigma_{\overline{\mathcal{P}}}\,\mathbf{S}$$

Example: 2-party resources

$$\mathbf{R} \cong^{\pi} \mathbf{S} :\Longleftrightarrow \begin{cases} \pi_1 \mathbf{R} \pi_2 \approx \mathbf{S} \\ \pi_1 \mathbf{R} \approx \mathbf{S} \sigma_2 \\ \mathbf{R} \pi_2 \approx \sigma_1 \mathbf{S} \\ \mathbf{R} \approx \sigma_1 \mathbf{S} \sigma_2 \end{cases}$$

Example: 2-party resources

$$\mathbf{R} \cong^{\pi} \mathbf{S} :\Longleftrightarrow \left\{ \begin{array}{rcl} \pi_1 \mathbf{R} \pi_2 & \approx & \mathbf{S} \\ \pi_1 \mathbf{R} & \approx & \mathbf{S} \sigma_2 \\ \mathbf{R} \pi_2 & \approx & \sigma_1 \mathbf{S} \\ \mathbf{R} & \approx & \sigma_1 \mathbf{S} \sigma_2 \end{array} \right.$$

$$\left. \begin{array}{c} \\ \\ \\ \end{array} \right\} \Leftrightarrow \text{abstract UC}$$

Example: 2-party resources

$$\mathbf{R} \cong^{\pi} \mathbf{S} :\Longleftrightarrow \begin{cases} \pi_1 \mathbf{R} \pi_2 \approx \mathbf{S} \\ \pi_1 \mathbf{R} \approx \mathbf{S} \sigma_2 \\ \mathbf{R} \pi_2 \approx \sigma_1 \mathbf{S} \\ \mathbf{R} \approx \sigma_1 \mathbf{S} \sigma_2 \end{cases}$$

Special case: $\mathbf{R}$ = channel (neutral element, e.g. $\pi_1 \mathbf{R} = \pi_1$)

Example: 2-party resources

$$\mathbf{R} \cong^{\pi} \mathbf{S} :\Longleftrightarrow \begin{cases} \pi_1 \quad \pi_2 \approx \quad \mathbf{S} \\ \pi_1 \qquad\quad \approx \quad \mathbf{S}\sigma_2 \\ \qquad \pi_2 \approx \sigma_1\mathbf{S} \\ \qquad\qquad \approx \sigma_1\mathbf{S}\sigma_2 \end{cases}$$

Special case: **R** = channel (neutral element, e.g. $\pi_1\mathbf{R} = \pi_1$)

Example: 2-party resources

$$\mathbf{R} \cong^{\pi} \mathbf{S} :\Longleftrightarrow \left\{ \begin{array}{lll} \pi_1 & \pi_2 \approx & \mathbf{S} \\ \pi_1 & \approx & \mathbf{S}\sigma_2 \\ & \pi_2 \approx & \sigma_1\mathbf{S} \\ & \approx & \sigma_1\mathbf{S}\sigma_2 \end{array} \right.$$

$$\boxed{\Big\} \Rightarrow \pi_1\pi_2 \approx \mathbf{S}\sigma_2\sigma_1\mathbf{S} \approx \mathbf{S}}$$

Special case: **R** = channel (neutral element, e.g. $\pi_1\mathbf{R} = \pi_1$)

Example: 2-party resources

$$\mathbf{R} \cong^{\pi} \mathbf{S} :\Longleftrightarrow \left\{ \begin{array}{llll} \pi_1 & \pi_2 & \approx & \mathbf{S} \\ \pi_1 & & \approx & \mathbf{S}\sigma_2 \\ & \pi_2 & \approx & \sigma_1\mathbf{S} \\ & & \approx & \sigma_1\mathbf{S}\sigma_2 \end{array} \right.$$

$$\boxed{\left.\right\} \Rightarrow \pi_1\pi_2 \approx \mathbf{S}\sigma_2\sigma_1\mathbf{S} \approx \mathbf{S}}$$

Special case: **R** = channel (neutral element, e.g. $\pi_1\mathbf{R} = \pi_1$)

Theorem: A resource **S** such that $\mathbf{S}\alpha\mathbf{S} \not\approx \mathbf{S}$ for all α cannot be constructed from a communication channel.

Example: 2-party resources

$$\mathbf{R} \cong^{\pi} \mathbf{S} :\Longleftrightarrow \left\{ \begin{array}{llll} \pi_1 & \pi_2 & \approx & \mathbf{S} \\ \pi_1 & & \approx & \mathbf{S}\sigma_2 \\ & \pi_2 & \approx & \sigma_1\mathbf{S} \\ & & \approx & \sigma_1\mathbf{S}\sigma_2 \end{array} \right\} \Rightarrow \pi_1\pi_2 \approx \mathbf{S}\sigma_2\sigma_1\mathbf{S} \approx \mathbf{S}$$

Special case: $\mathbf{R}$ = channel (neutral element, e.g. $\pi_1\mathbf{R} = \pi_1$)

Theorem: A resource $\mathbf{S}$ such that $\mathbf{S}\alpha\mathbf{S} \not\approx \mathbf{S}$ for all α cannot be constructed from a communication channel.

Corollary [CF01]: Commitment cannot be constructed (from a communication channel).

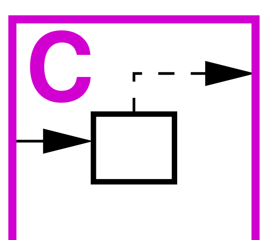
Example: 2-party resources

$$\mathbf{R} \cong^{\pi} \mathbf{S} :\iff \left\{ \begin{array}{rrcl} \pi_1 & \pi_2 & \approx & \mathbf{S} \\ \pi_1 & & \approx & \mathbf{S}\sigma_2 \\ & \pi_2 & \approx & \sigma_1\mathbf{S} \\ & & \approx & \sigma_1\mathbf{S}\sigma_2 \end{array} \right\} \Rightarrow \pi_1\pi_2 \approx \mathbf{S}\sigma_2\sigma_1\mathbf{S} \approx \mathbf{S}$$

Special case: **R** = channel (neutral element, e.g. $\pi_1\mathbf{R} = \pi_1$)

Theorem: A resource **S** such that $\mathbf{S}\alpha\mathbf{S} \not\approx \mathbf{S}$ for all α cannot be constructed from a communication channel.

Corollary [CF01]: Commitment cannot be constructed (from a communication channel).



Example: 2-party resources

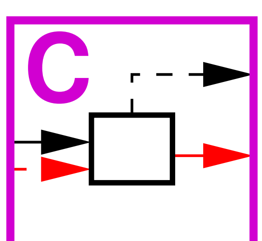
$$\mathbf{R} \cong^{\pi} \mathbf{S} :\Longleftrightarrow \left\{ \begin{array}{lllr} \pi_1 & \pi_2 & \approx & \mathbf{S} \\ \pi_1 & & \approx & \mathbf{S}\sigma_2 \\ & \pi_2 & \approx & \sigma_1\mathbf{S} \\ & & \approx & \sigma_1\mathbf{S}\sigma_2 \end{array} \right.$$

$$\left. \right\} \Rightarrow \pi_1\pi_2 \approx \mathbf{S}\sigma_2\sigma_1\mathbf{S} \approx \mathbf{S}$$

Special case: $\mathbf{R}$ = channel (neutral element, e.g. $\pi_1\mathbf{R} = \pi_1$)

Theorem: A resource $\mathbf{S}$ such that $\mathbf{S}\alpha\mathbf{S} \not\approx \mathbf{S}$ for all α cannot be constructed from a communication channel.

Corollary [CF01]: Commitment cannot be constructed (from a communication channel).



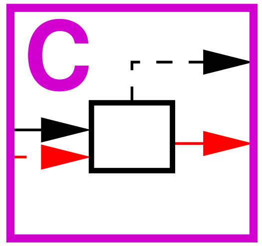
Example: 2-party resources

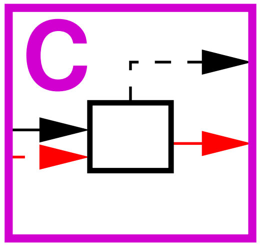
$$\mathbf{R} \cong^{\pi} \mathbf{S} :\Longleftrightarrow \left\{ \begin{array}{llrl} \pi_1 & \pi_2 & \approx & \mathbf{S} \\ \pi_1 & & \approx & \mathbf{S}\sigma_2 \\ & \pi_2 & \approx & \sigma_1\mathbf{S} \\ & & \approx & \sigma_1\mathbf{S}\sigma_2 \end{array} \right\} \Rightarrow \pi_1\pi_2 \approx \mathbf{S}\sigma_2\sigma_1\mathbf{S} \approx \mathbf{S}$$

Special case: $\mathbf{R}$ = channel (neutral element, e.g. $\pi_1\mathbf{R} = \pi_1$)

Theorem: A resource $\mathbf{S}$ such that $\mathbf{S}\alpha\mathbf{S} \not\approx \mathbf{S}$ for all α cannot be constructed from a communication channel.

Corollary [CF01]: Commitment cannot be constructed (from a communication channel).





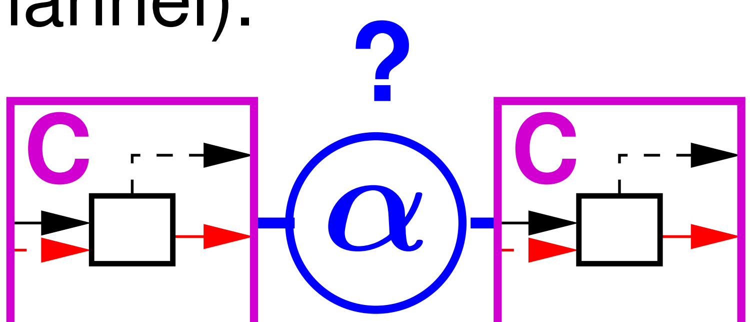
Example: 2-party resources

$$\mathbf{R} \cong^{\pi} \mathbf{S} :\Longleftrightarrow \left\{ \begin{array}{llcr} \pi_1 & \pi_2 & \approx & \mathbf{S} \\ \pi_1 & & \approx & \mathbf{S}\sigma_2 \\ & \pi_2 & \approx & \sigma_1\mathbf{S} \\ & & \approx & \sigma_1\mathbf{S}\sigma_2 \end{array} \right\} \Rightarrow \pi_1\pi_2 \approx \mathbf{S}\sigma_2\sigma_1\mathbf{S} \approx \mathbf{S}$$

Special case: **R** = channel (neutral element, e.g. $\pi_1\mathbf{R} = \pi_1$)

Theorem: A resource **S** such that $\mathbf{S}\alpha\mathbf{S} \not\approx \mathbf{S}$ for all α cannot be constructed from a communication channel.

Corollary [CF01]: Commitment cannot be constructed (from a communication channel).



Example: 2-party resources

$$\mathbf{R} \cong^{\pi} \mathbf{S} :\Longleftrightarrow \left\{ \begin{array}{llll} \pi_1 & \pi_2 \approx & \mathbf{S} & \\ \pi_1 & \approx & \mathbf{S}\sigma_2 & \\ & \pi_2 \approx & \sigma_1\mathbf{S} & \\ & \approx & \sigma_1\mathbf{S}\sigma_2 & \end{array} \right.$$

$$\boxed{\Big\} \Rightarrow \pi_1\pi_2 \approx \mathbf{S}\sigma_2\sigma_1\mathbf{S} \approx \mathbf{S}}$$

Special case: **R** = channel (neutral element, e.g. $\pi_1\mathbf{R} = \pi_1$)

Theorem: A resource **S** such that $\mathbf{S}\alpha\mathbf{S} \not\approx \mathbf{S}$ for all α cannot be constructed from a communication channel.

Corollary [CF01]: Commitment cannot be constructed (from a communication channel).

Corollary: A delayed communication channel cannot be constructed (from a communication channel).

Abstraction by sets of resources

Definition: A **specification** is a set $\mathcal{R}$ of resources.

Abstraction by sets of resources

Definition: A **specification** is a set $\mathcal{R}$ of resources.

Of special interest: Specifications with (for each party)

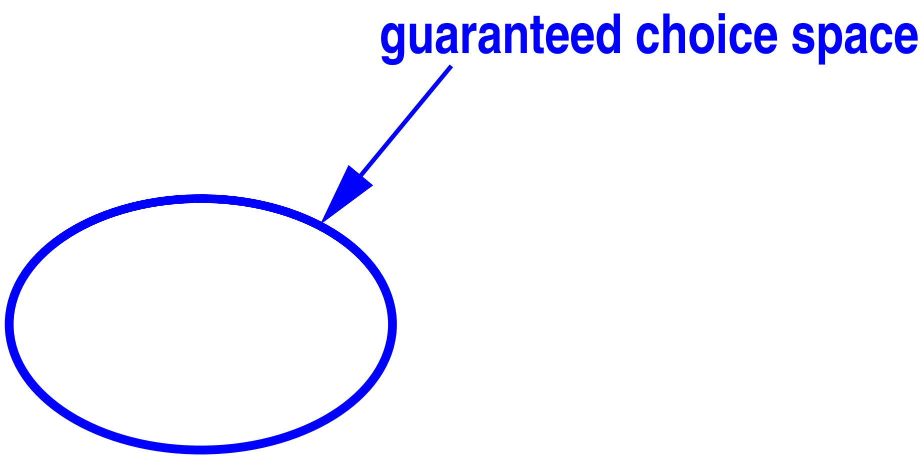
- a guaranteed choice space
- a possible choice space

Abstraction by sets of resources

Definition: A **specification** is a set $\mathcal{R}$ of resources.

Of special interest: Specifications with (for each party)

- a guaranteed choice space
- a possible choice space

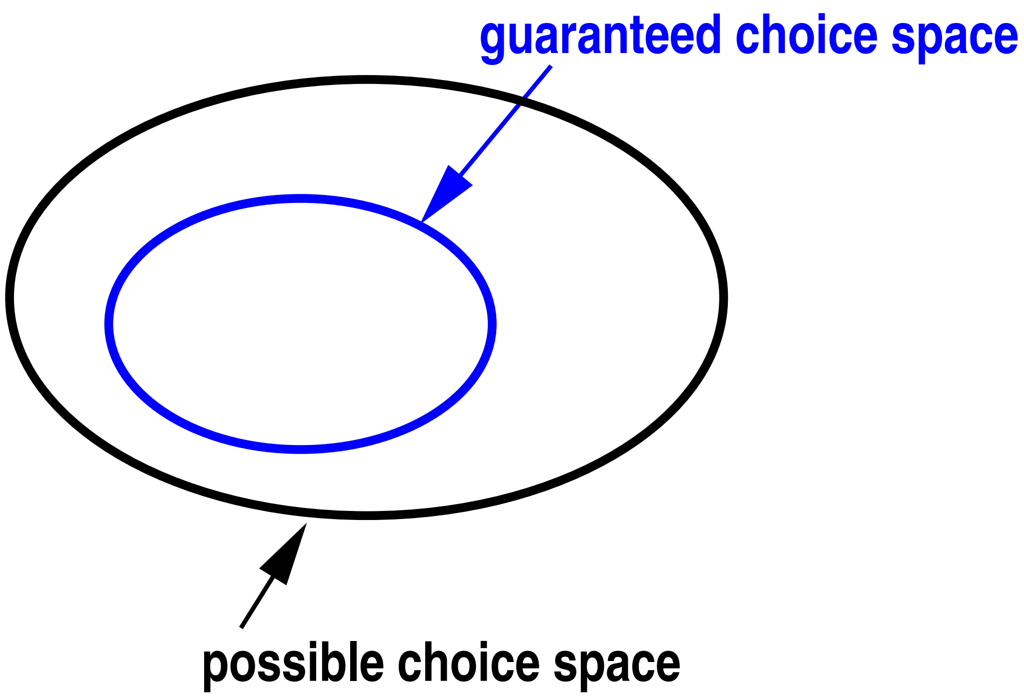


Abstraction by sets of resources

Definition: A **specification** is a set $\mathcal{R}$ of resources.

Of special interest: Specifications with (for each party)

- a guaranteed choice space
- a possible choice space

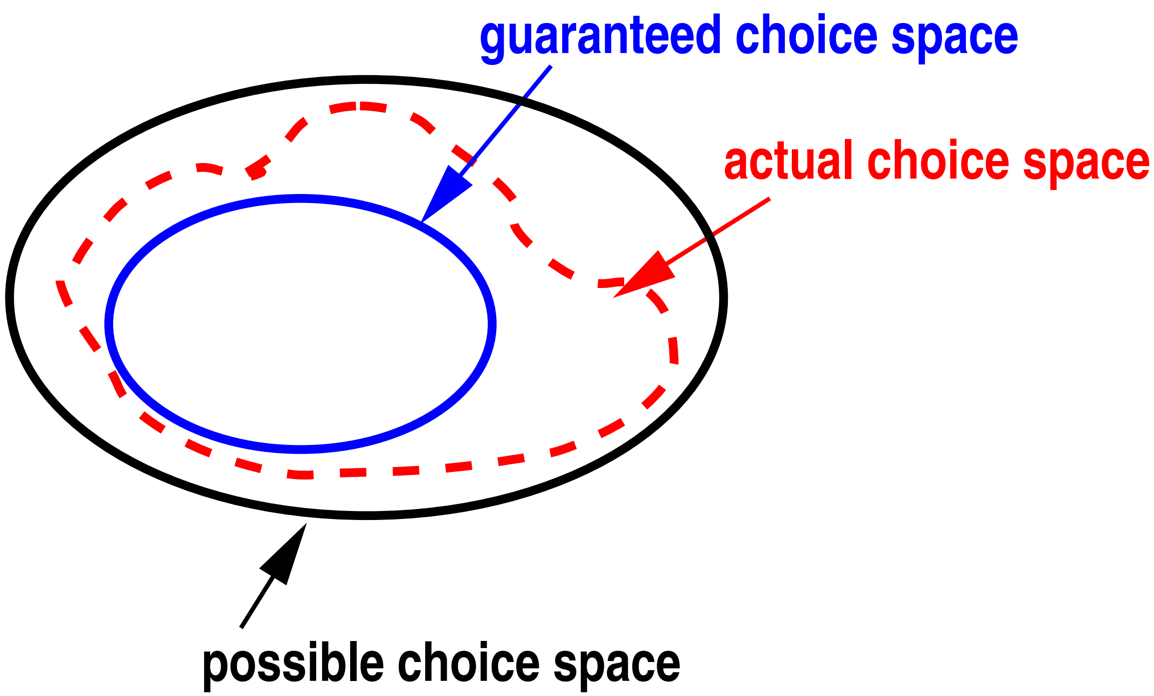


Abstraction by sets of resources

Definition: A **specification** is a set $\mathcal{R}$ of resources.

Of special interest: Specifications with (for each party)

- a guaranteed choice space
- a possible choice space

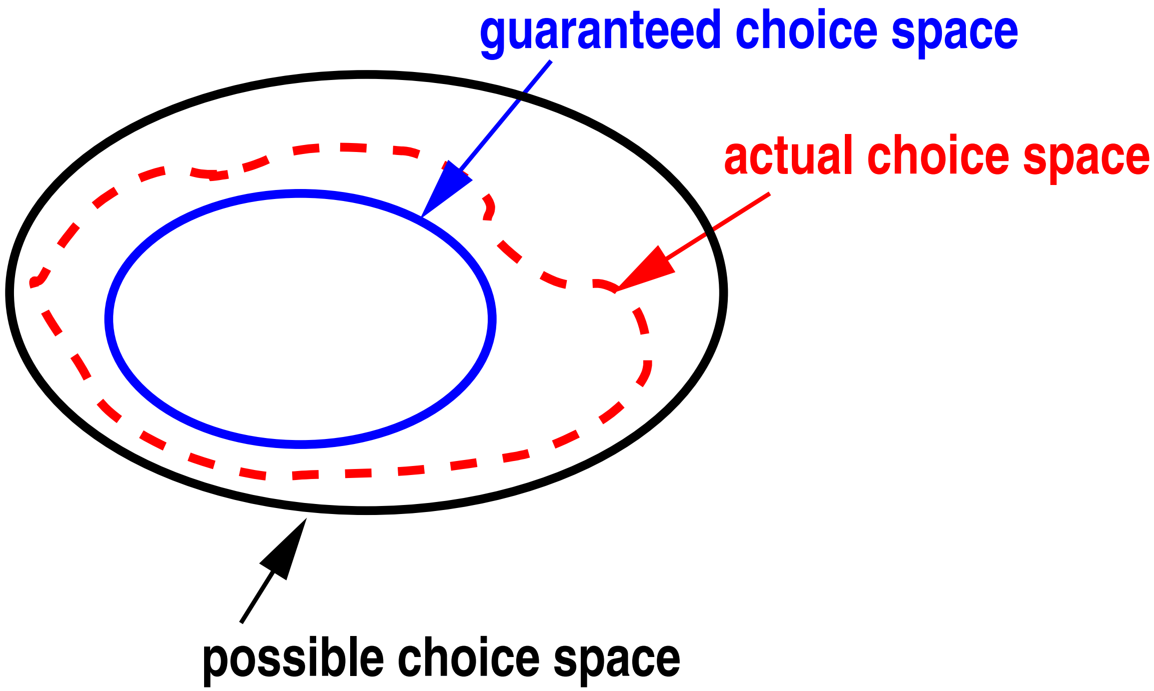


Abstraction by sets of resources

Definition: A **specification** is a set $\mathcal{R}$ of resources.

Of special interest: Specifications with (for each party)

- a guaranteed choice space
- a possible choice space

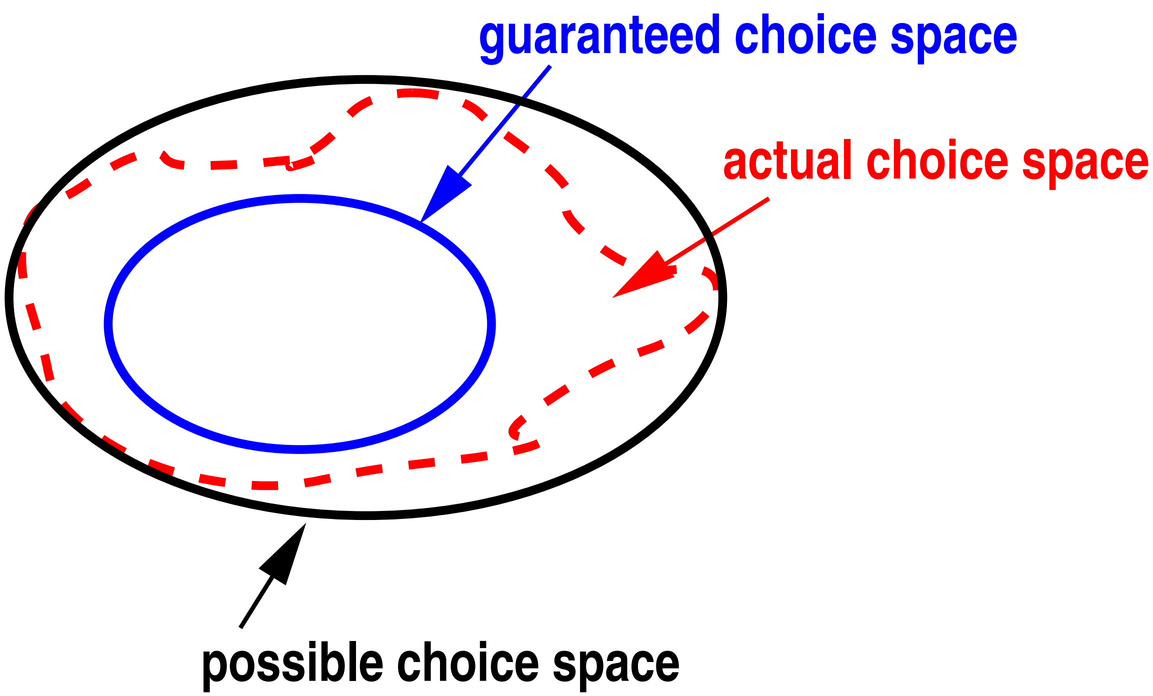


Abstraction by sets of resources

Definition: A **specification** is a set $\mathcal{R}$ of resources.

Of special interest: Specifications with (for each party)

- a guaranteed choice space
- a possible choice space

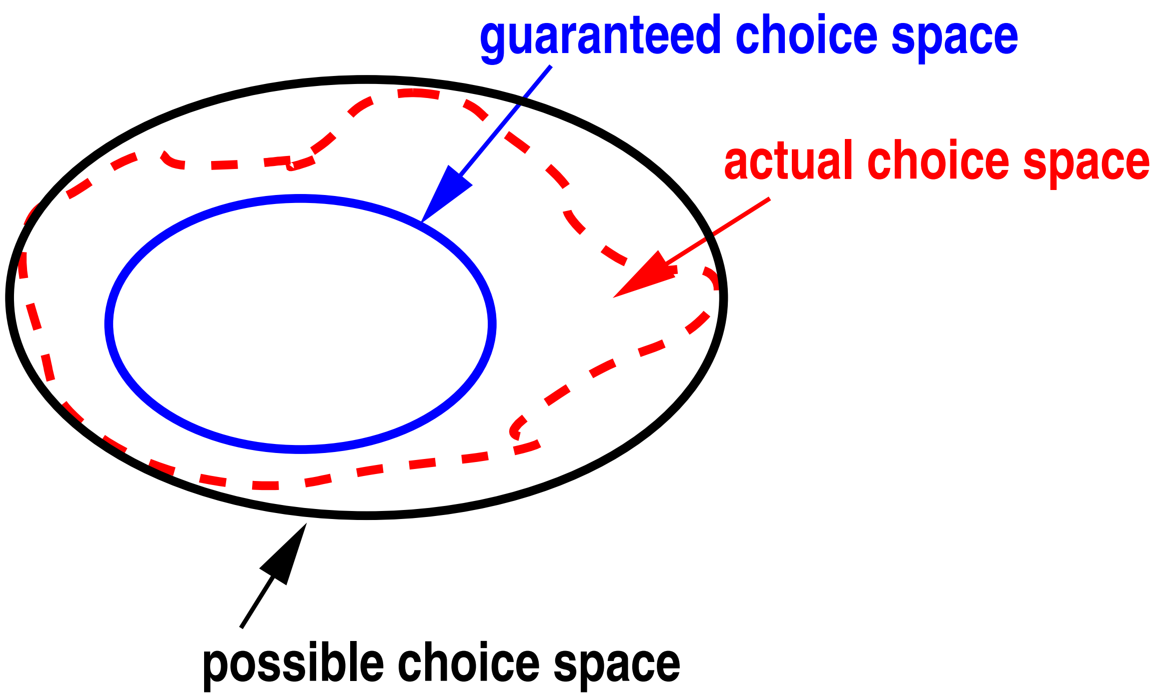


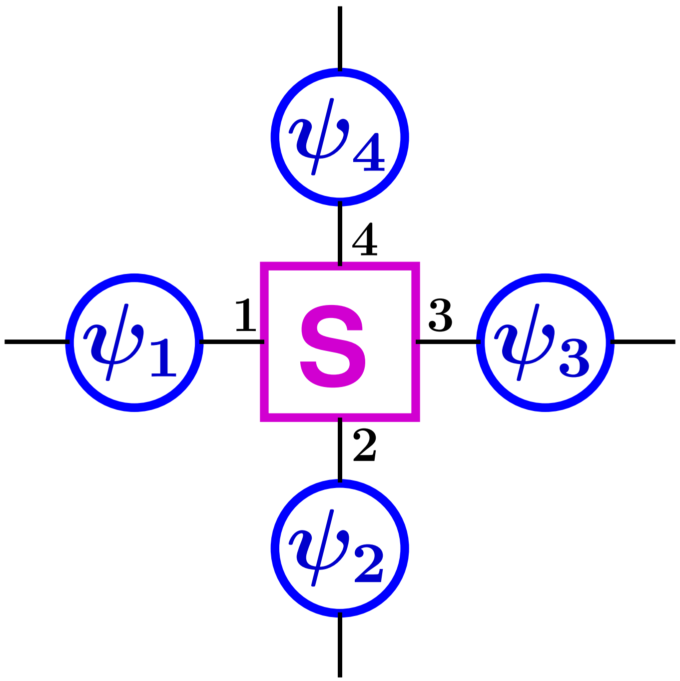
Abstraction by sets of resources

Definition: A **specification** is a set $\mathcal{R}$ of resources.

Of special interest: Specifications with (for each party)

- a guaranteed choice space
- a possible choice space





Abstraction by sets of resources

Definition: A **specification** is a set $\mathcal{R}$ of resources.

Of special interest: Specifications with (for each party)

- a guaranteed choice space
- a possible choice space

Definition: $\mathcal{S}$ **is an abstraction of** $\mathcal{R}$ via π:

$$\mathcal{R} \sqsubseteq^{\pi} \mathcal{S} \quad :\Longleftrightarrow \quad \forall \mathbf{R} \in \mathcal{R} \ \ \exists \mathbf{S} \in \mathcal{S} : \ \mathbf{R} \cong^{\pi} \mathbf{S}$$

Abstraction by sets of resources

Definition: A **specification** is a set $\mathcal{R}$ of resources.

Of special interest: Specifications with (for each party)

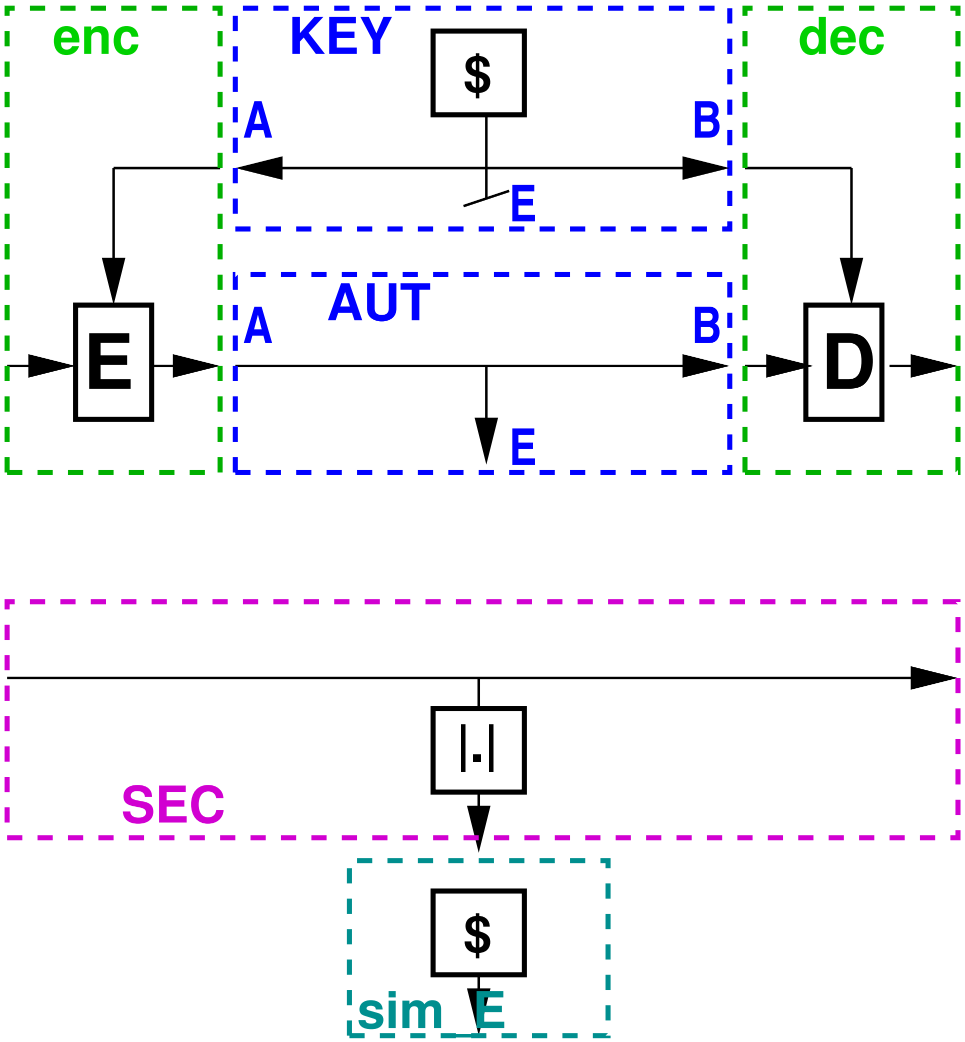
- a guaranteed choice space
- a possible choice space

Definition: $\mathcal{S}$ **is an abstraction of** $\mathcal{R}$ via π:

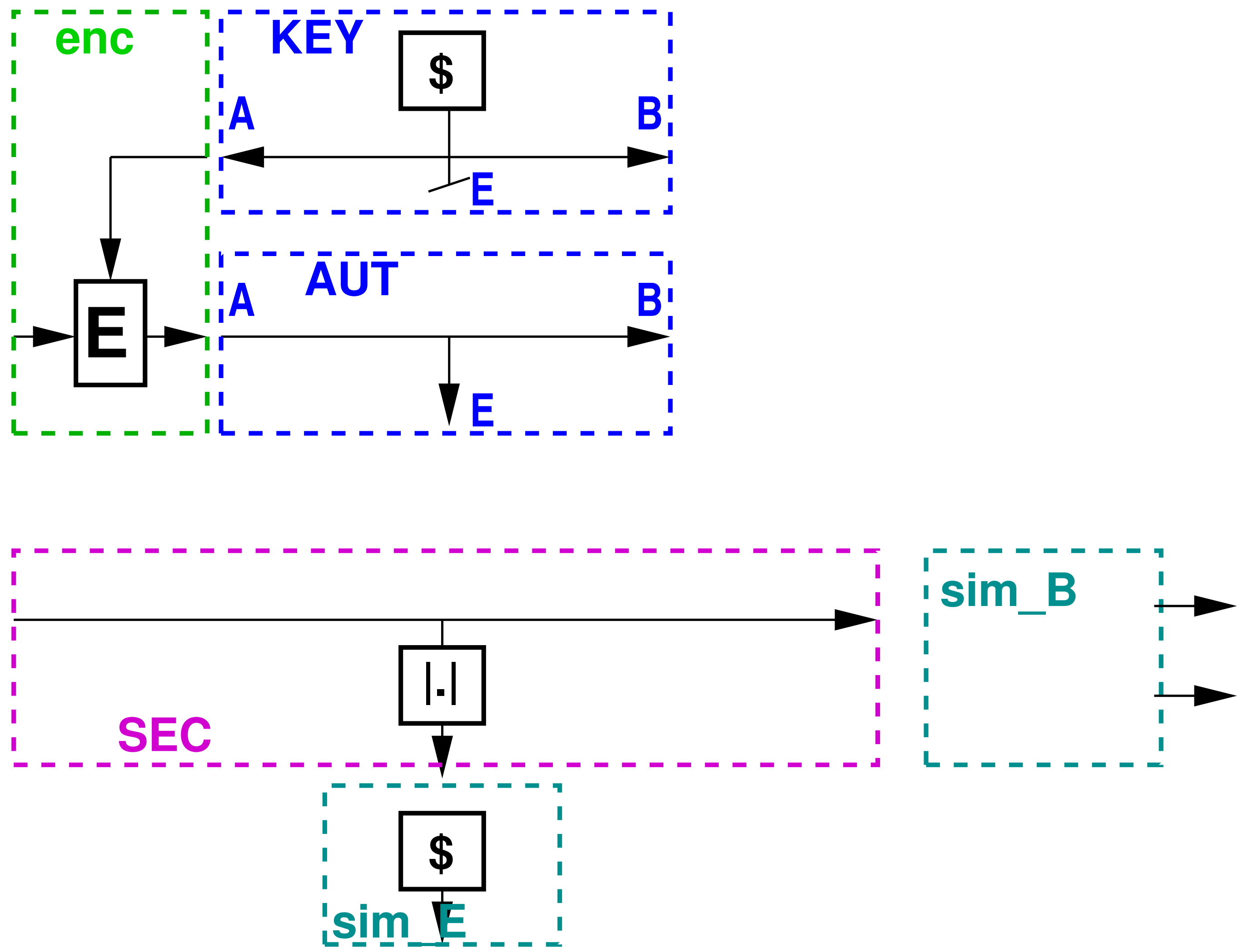
$$\mathcal{R} \sqsubseteq^{\pi} \mathcal{S} \quad :\Longleftrightarrow \quad \forall \mathbf{R} \in \mathcal{R} \;\; \exists \mathbf{S} \in \mathcal{S} : \;\; \mathbf{R} \cong^{\pi} \mathbf{S}$$

Theorem: $\mathcal{R} \sqsubseteq^{\pi} \mathcal{S}$ is generally composable.

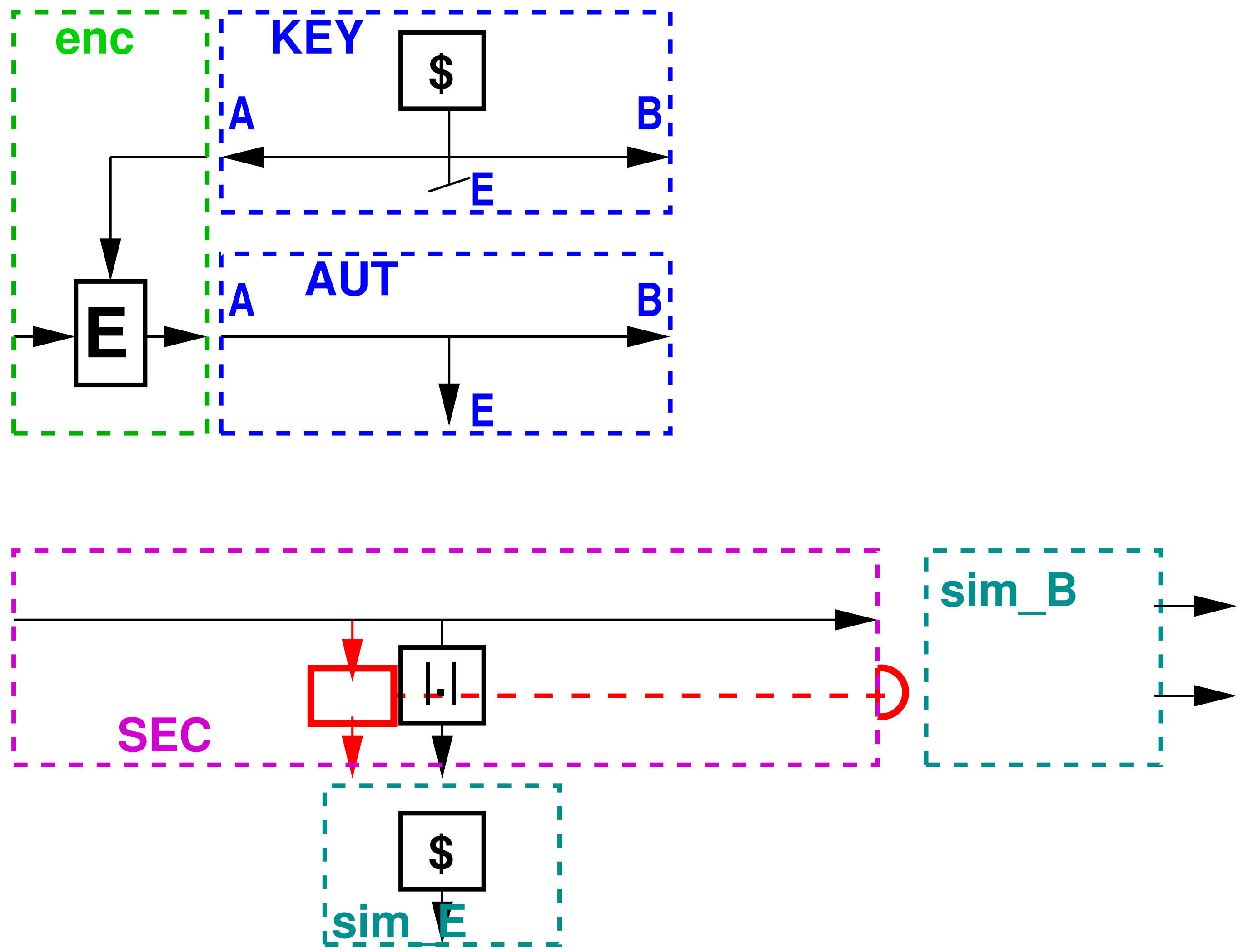
Example: Encryption, capturing coercibility



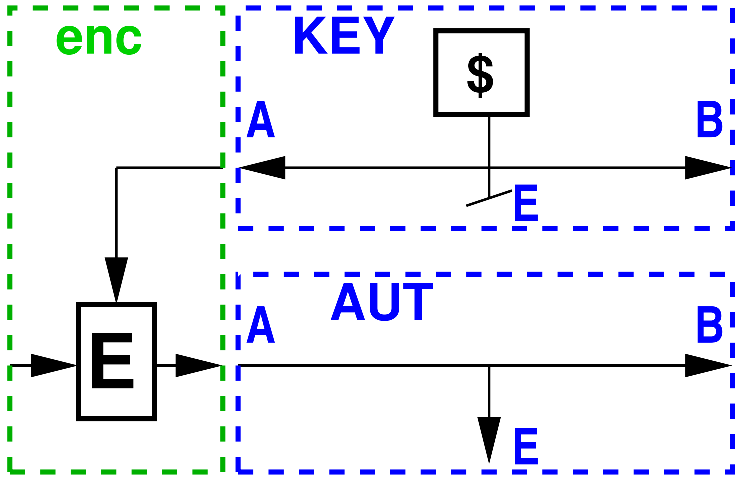
Example: Encryption, capturing coercibility

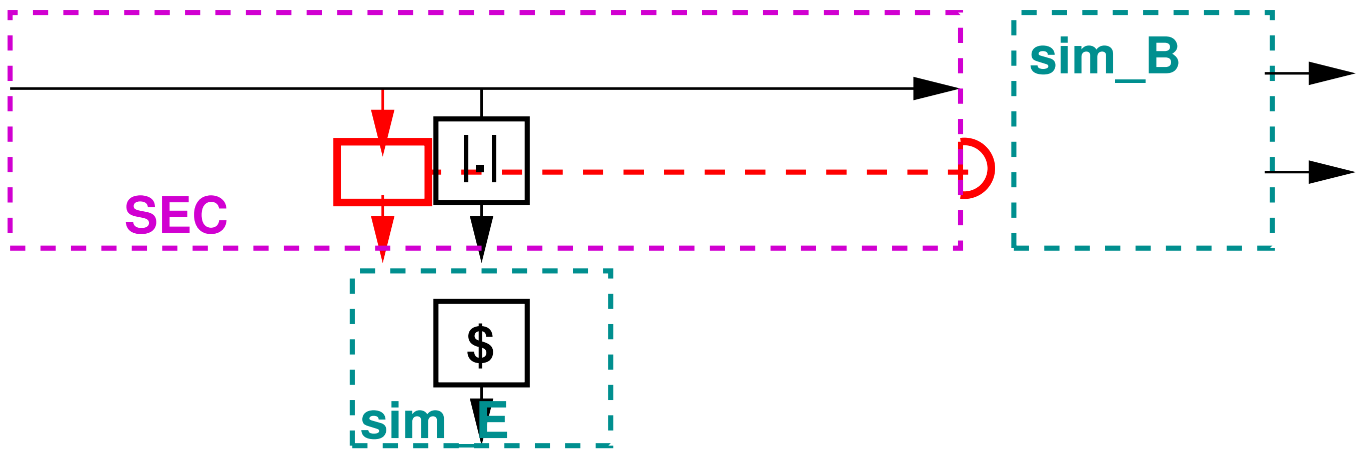


Example: Encryption, capturing coercibility



Example: Encryption, capturing coercibility





Theorem: An unleakable (uncoercible) secure communication channel cannot be constructed from an authenticated channel and a secret key.

Some Features of Abstract Cryptography

Some Features of Abstract Cryptography

- top-down abstraction

Some Features of Abstract Cryptography

- top-down abstraction
- isomorphism: exact relation between ideal and real

Some Features of Abstract Cryptography

- top-down abstraction
- isomorphism: exact relation between ideal and real
- no central adversary; local simulators

Some Features of Abstract Cryptography

- top-down abstraction
- isomorphism: exact relation between ideal and real
- no central adversary; local simulators
- all resources captured
  - communication model not hard-wired into the model
  - absence can be modeled

Some Features of Abstract Cryptography

- top-down abstraction
- isomorphism: exact relation between ideal and real
- no central adversary; local simulators
- all resources captured
  - communication model not hard-wired into the model
  - absence can be modeled
- feasibility/efficiency notions free; not hard-wired

Some Features of Abstract Cryptography

- top-down abstraction
- isomorphism: exact relation between ideal and real
- no central adversary; local simulators
- all resources captured
  - communication model not hard-wired into the model
  - absence can be modeled
- feasibility/efficiency notions free; not hard-wired
- specifications: guaranteed/possible choice domains

Some Features of Abstract Cryptography

- top-down abstraction
- isomorphism: exact relation between ideal and real
- no central adversary; local simulators
- all resources captured
  - communication model not hard-wired into the model
  - absence can be modeled
- feasibility/efficiency notions free; not hard-wired
- specifications: guaranteed/possible choice domains
- existing frameworks can be captured as special cases
  - universal composability (UC) by Canetti
  - reactive simulatability by Pfitzmann/Waidner/Backes
  - indifferentiability [MRH04]
  - collusion-preserving computation [AKMZ12]

Thank you!

Constructing channels and keys: •-calculus

$A \longrightarrow B$ (insecure) channel from A to B

The symbol "•" stands for exclusive access to the channel.

"•" at output: receiver is exclusive $\longrightarrow$ confidentiality

"•" at input: sender is exclusive $\longrightarrow$ authenticity

$A \longrightarrow\!\bullet\ B$ secret channel from A to B

$A\ \bullet\!\!\longrightarrow B$ authentic channel from A to B

$A\ \bullet\!\!\longrightarrow\!\bullet\ B$ secure channel from A to B (secret and authentic)

$A\ \bullet\!\!=\!=\!=\!\!\bullet\ B$ secret key shared by A and B

$A\ =\!=\!=\!\!\bullet\ B$ one-sided key: A knows that at most B knows the key, but B does not know who holds the key.

Key transport in CC

$$A \bullet\!\xrightarrow{t_1 t_2}\!\bullet\ B \quad \xrightarrow{\textbf{KT}} \quad A\ \bullet\!\overset{t_2}{=\!=\!=}\!\bullet\ B$$

Key transport in CC

$$A \bullet \xrightarrow{t_1 t_2} \bullet B \qquad \xrightarrow{\textbf{KT}} \qquad A \bullet \overset{t_2}{=\!=\!=} \bullet B$$

$$A \xrightarrow{t_1 t_2} \bullet B \qquad \xrightarrow{\textbf{KT}} \qquad A \overset{t_2}{=\!=\!=} \bullet B$$

Key transport in CC

$$A \bullet\!\xrightarrow{t_1 t_2}\!\bullet B \qquad \xrightarrow{\textbf{KT}} \qquad A \bullet\!\overset{t_2}{=\!=\!=}\!\bullet B$$

$$A \;\xrightarrow{t_1 t_2}\!\bullet B \qquad \xrightarrow{\textbf{KT}} \qquad A \;\overset{t_2}{=\!=\!=}\!\bullet B$$

Attention: $$A \bullet\!\xrightarrow{t_1 t_2} B \qquad \xrightarrow{\cancel{\textbf{KT}}} \qquad A \bullet\!\overset{t_2}{=\!=\!=}\; B$$

Symmetric cryptosystem in CC

$$\left.\begin{array}{c} A \overset{t_1}{\Longrightarrow\!\bullet} B \\ A \xrightarrow{t_2 t_3} B \\ t_2 \geq t_1 \end{array}\right\} \xrightarrow{\textbf{SYM}} A \overset{t_2 t_3}{\longrightarrow\!\bullet} B$$

Symmetric cryptosystem in CC

$$\left.\begin{array}{c} A \overset{t_1}{=\!=\!=\!=\!\bullet} B \\ A \xrightarrow{t_2 t_3} B \\ t_2 \geq t_1 \end{array}\right\} \xrightarrow{\textbf{SYM}} A \xrightarrow{t_2 t_3}\bullet\, B$$

$$\left.\begin{array}{c} A \overset{t_1}{=\!=\!=\!=\!\bullet} B \\ A \bullet\!\xrightarrow{t_2 t_3} B \\ t_2 \geq t_1 \end{array}\right\} \xrightarrow{\textbf{SYM}} A \bullet\!\xrightarrow{t_2 t_3}\bullet\, B$$

MACs in CC

$$\left.\begin{array}{c} A \;\overset{t_1}{\bullet\!\!=\!\!=\!\!=}\; B \\ A \xrightarrow{t_2 t_3} B \\ t_2 \geq t_1 \end{array}\right\} \xrightarrow{\textbf{MAC}} A \bullet\!\!\xrightarrow{t_2 t_3} B$$

MACs in CC

$$\left.\begin{array}{c} A \;\bullet\!\overset{t_1}{=\!=\!=\!=}\; B \\ A \xrightarrow{t_2 t_3} B \\ t_2 \geq t_1 \end{array}\right\} \quad \xrightarrow{\textbf{MAC}} \quad A \;\bullet\!\xrightarrow{t_2 t_3} B$$

$$\left.\begin{array}{c} A \;\bullet\!\overset{t_1}{=\!=\!=\!=}\; B \\ A \xrightarrow{t_2 t_3}\!\bullet\; B \\ t_2 \geq t_1 \end{array}\right\} \quad \xrightarrow{\textbf{MAC}} \quad A \;\bullet\!\xrightarrow{t_2 t_3}\!\bullet\; B$$

Combining Encryption and MAC

Goal:

$$\left.\begin{array}{c} A \;\bullet\!\overset{t_1}{=\!=\!=\!=}\!\bullet\; B \\ A \xrightarrow{t_2 t_3} B \\ t_2 \geq t_1 \end{array}\right\} \xrightarrow{???} A \;\bullet\!\xrightarrow{t_2 t_3}\!\bullet\; B$$

Combining Encryption and MAC

Goal:

$$\left.\begin{array}{c} A \;\; \bullet\!\overset{t_1}{=\!=\!=\!=}\!\bullet \;\; B \\ A \xrightarrow{t_2 t_3} B \\ t_2 \geq t_1 \end{array}\right\} \xrightarrow{???} A \;\bullet\!\xrightarrow{t_2 t_3}\!\bullet\; B$$

Key expansion:

$$A \;\; \bullet\!\overset{t_1}{=\!=\!=\!=}\!\bullet \;\; B \xrightarrow{\text{PRG}} \left\{\begin{array}{c} A \;\; \bullet\!\overset{t_1}{=\!=\!=\!=}\!\bullet \;\; B \\ A \;\; \bullet\!\overset{t_1}{=\!=\!=\!=}\!\bullet \;\; B \end{array}\right.$$

Combining Encryption and MAC

Goal:

$$\left.\begin{array}{c} A \;\bullet\!\overset{t_1}{=\!=\!=\!=}\!\bullet\; B \\ A \xrightarrow{t_2 t_3} B \\ t_2 \geq t_1 \end{array}\right\} \xrightarrow{\text{???}} A \;\bullet\!\xrightarrow{t_2 t_3}\!\bullet\; B$$

Key expansion:

$$A \;\bullet\!\overset{t_1}{=\!=\!=\!=}\!\bullet\; B \xrightarrow{\text{PRG}} \left\{\begin{array}{c} A \;\bullet\!\overset{t_1}{=\!=\!=\!=}\!\bullet\; B \\ A \;\bullet\!\overset{t_1}{=\!=\!=\!=}\!\bullet\; B \end{array}\right.$$

Encrypt-then-MAC:

$$\left.\begin{array}{c} A \;\bullet\!\overset{t_1}{=\!=\!=\!=}\; B \\ A \xrightarrow{t_2 t_3} B \\ t_2 \geq t_1 \end{array}\right\} \xrightarrow{\text{MAC}} A \;\bullet\!\xrightarrow{t_2 t_3}\; B$$

$$\left.\begin{array}{c} A \;\overset{t_1}{=\!=\!=\!=}\!\bullet\; B \\ A \;\bullet\!\xrightarrow{t_2 t_3} B \\ t_2 \geq t_1 \end{array}\right\} \xrightarrow{\text{SYM}} A \;\bullet\!\xrightarrow{t_2 t_3}\!\bullet\; B$$

Combining Encryption and MAC

Goal:

$$\left.\begin{array}{c} A \;\; \bullet\!\!\overset{t_1}{=\!=\!=\!=}\!\!\bullet \;\; B \\ A \xrightarrow{t_2 t_3} B \\ t_2 \geq t_1 \end{array}\right\} \xrightarrow{\textbf{???}} A \;\bullet\!\!\xrightarrow{t_2 t_3}\!\!\bullet\; B$$

Key expansion:

$$A \;\; \bullet\!\!\overset{t_1}{=\!=\!=\!=}\!\!\bullet \;\; B \xrightarrow{\textbf{PRG}} \left\{\begin{array}{c} A \;\; \bullet\!\!\overset{t_1}{=\!=\!=\!=}\!\!\bullet \;\; B \\ A \;\; \bullet\!\!\overset{t_1}{=\!=\!=\!=}\!\!\bullet \;\; B \end{array}\right.$$

MAC-then-encrypt:

$$\left.\begin{array}{c} A \;\; \overset{t_1}{=\!=\!=\!=}\!\!\bullet \;\; B \\ A \xrightarrow{t_2 t_3} B \\ t_2 \geq t_1 \end{array}\right\} \xrightarrow{\textbf{SYM}} A \xrightarrow{t_2 t_3}\!\!\bullet\; B$$

$$\left.\begin{array}{c} A \;\; \bullet\!\!\overset{t_1}{=\!=\!=\!=} \;\; B \\ A \xrightarrow{t_2 t_3}\!\!\bullet\; B \\ t_2 \geq t_1 \end{array}\right\} \xrightarrow{\textbf{MAC}} A \;\bullet\!\!\xrightarrow{t_2 t_3}\!\!\bullet\; B$$

Public-key cryptosystems in CC

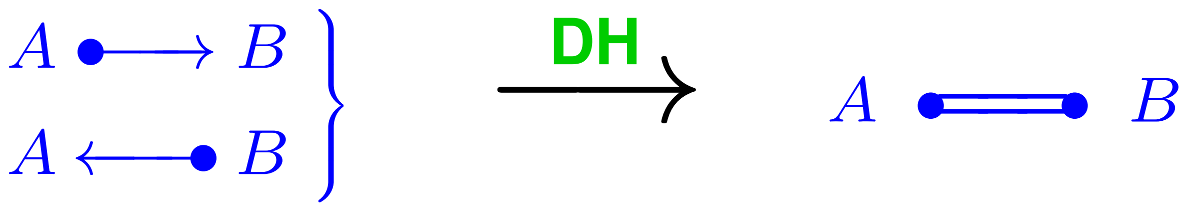
$$\left.\begin{array}{c} A \bullet\!\xrightarrow{t_1 t_2} B \\ A \xleftarrow{t_4 t_3} B \\ t_3 > t_2 \end{array}\right\} \xrightarrow{\textbf{PKC}} A \bullet\!\xleftarrow{t_4 t_3} B$$

Public-key cryptosystems in CC

$$\left.\begin{array}{c} A \bullet\!\xrightarrow{t_1 t_2} B \\ A \xleftarrow{t_4 t_3} B \\ t_3 > t_2 \end{array}\right\} \xrightarrow{\textbf{PKC}} A \bullet\!\xleftarrow{t_4 t_3} B$$

$$\left.\begin{array}{c} A \bullet\!\xrightarrow{t_1 t_2} B \\ A \xleftarrow{t_4 t_3}\!\bullet B \\ t_3 > t_2 \end{array}\right\} \xrightarrow{\textbf{PKC}} A \bullet\!\xleftarrow{t_4 t_3}\!\bullet B$$

Diffie-Hellman key agreement in CC



Diffie-Hellman key agreement in CC

$$\left.\begin{array}{l} A \bullet\!\!\longrightarrow B \\ A \longleftarrow\!\!\bullet B \end{array}\right\} \xrightarrow{\text{DH}} A \bullet\!\!=\!\!\bullet B$$

$$\left.\begin{array}{l} A \bullet\!\!\longrightarrow B \\ A \longleftarrow B \end{array}\right\} \not\xrightarrow{\text{DH}} A \bullet\!\!= B$$

Digital signature schemes in CC

$$\left.\begin{array}{c} A \bullet\!\xrightarrow{t_1 t_2} B \\ A \xrightarrow{t_3 t_4} B \\ t_4 \geq t_2 \end{array}\right\} \xrightarrow{\textsf{DSS}} A \bullet\!\xrightarrow{t_3 t_4} B$$

Digital signature schemes in CC

$$\left.\begin{array}{c} A \bullet\!\xrightarrow{t_1 t_2} B \\ A \xrightarrow{t_3 t_4} B \\ t_4 \geq t_2 \end{array}\right\} \xrightarrow{\textbf{DSS}} A \bullet\!\xrightarrow{t_3 t_4} B$$

$$\left.\begin{array}{c} A \bullet\!\xrightarrow{t_1 t_2} B \\ A \xrightarrow{t_3 t_4}\!\bullet B \\ t_4 \geq t_2 \end{array}\right\} \xrightarrow{\textbf{DSS}} A \bullet\!\xrightarrow{t_3 t_4}\!\bullet B$$

Digital signature schemes in CC

$$\left.\begin{array}{c} A \bullet\!\!\xrightarrow{t_1t_2} B \\ A \xrightarrow{t_3t_4} B \\ t_4 \geq t_2 \end{array}\right\} \xrightarrow{\textbf{DSS}} A \bullet\!\!\xrightarrow{t_3t_4} B$$

$$\left.\begin{array}{c} A \bullet\!\!\xrightarrow{t_1t_2} B \\ A \xrightarrow{t_3t_4}\!\!\bullet B \\ t_4 \geq t_2 \end{array}\right\} \xrightarrow{\textbf{DSS}} A \bullet\!\!\xrightarrow{t_3t_4}\!\!\bullet B$$

Note: Conservation law of the •-calculus.

Digital signature schemes in CC

$$\left.\begin{array}{c} A \bullet\!\xrightarrow{t_1 t_2}\; B \\ A \xrightarrow{t_3 t_4} B \\ t_4 \geq t_2 \end{array}\right\} \xrightarrow{\textbf{DSS}} A \bullet\!\xrightarrow{t_3 t_4}\; B$$

$$\left.\begin{array}{c} A \bullet\!\xrightarrow{t_1 t_2}\; B \\ A \xrightarrow{t_3 t_4}\!\bullet B \\ t_4 \geq t_2 \end{array}\right\} \xrightarrow{\textbf{DSS}} A \bullet\!\xrightarrow{t_3 t_4}\!\bullet B$$

Note: Conservation law of the •-calculus.

Are there any other cryptographic transformations?