Universal composability

(Ran Canetti ja teised)

Simulatability

(Birgit Pfitzmann ja teised)

Peeter

Cyberi teadusseminar

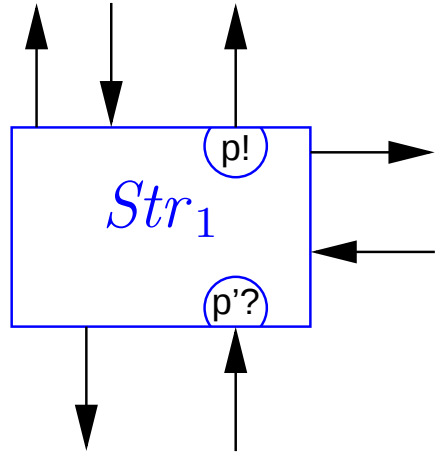
Tartu, 03.06.2004

Krüptosüsteemide turvalisusdefinitsioonid

- Alguses oli: nõuame, et primitiiv rahuldaks teatavaid omadusi.
  - Krüptimine — semantiline turvalisus.
    - Kõike, mida on avateksti kohta võimalik leida krüptotekstist, on võimalik leida ka ilma krüptotekstita.
  - Signatuurid — ei tohi olla võimalik anda signatuuri mingile bitijadale, teadmata seejuures salajast võtit.
  - Ajatemplisüsteemid — dokumentide *back-date*'imine peab olema võimatu.
  - jne.
- Probleem: raske või võimatu süsteeme turvaliselt komponeerida.

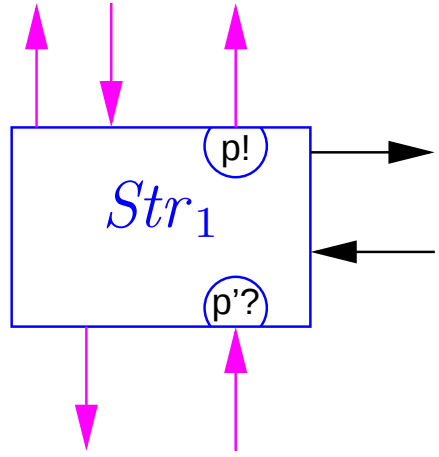
Krüptograafilise struktuuri turvalisus...

Olgu meil antud mingi krüptograafiline struktuur Str_1...



Krüptograafilise struktuuri turvalisus...

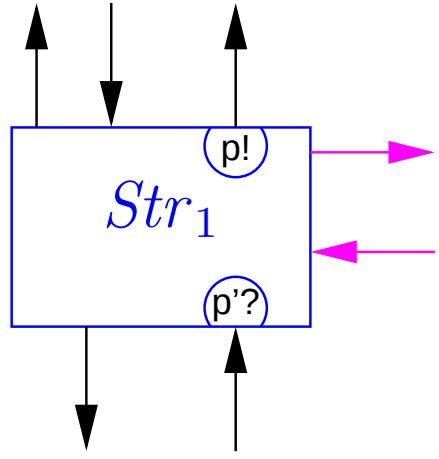
Olgu meil antud mingi krüptograafiline struktuur Str_1...



- mis teatud portidel pakub mingit teenust...

Krüptograafilise struktuuri turvalisus...

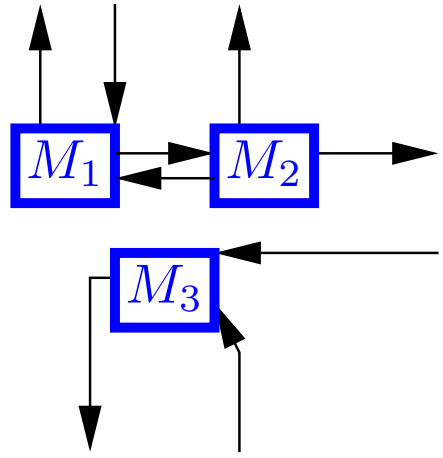
Olgu meil antud mingi krüptograafiline struktuur Str_1...



- mis teatud portidel pakub mingit teenust...
- ning mille toimimist on võimalik veel mingite täiendavate portide kaudu mõjutada,...

Krüptograafilise struktuuri turvalisus...

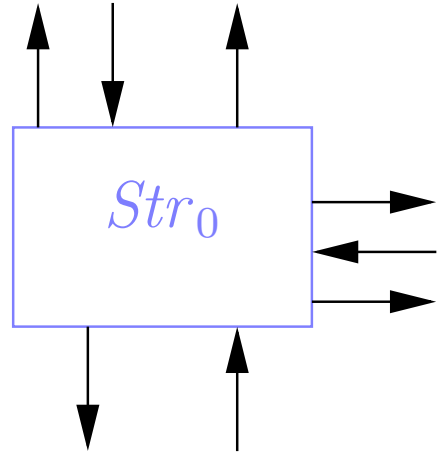
Olgu meil antud mingi krüptograafiline struktuur Str_1...



- mis teatud portidel pakub mingit teenust...
- ning mille toimimist on võimalik veel mingite täiendavate portide kaudu mõjutada,...
- ja mis on realiseeritud interaktiivsete Turingi masinate komplektina.

Krüptograafilise struktuuri turvalisus...

Olgu antud ka mingi teine struktuur Str_0...



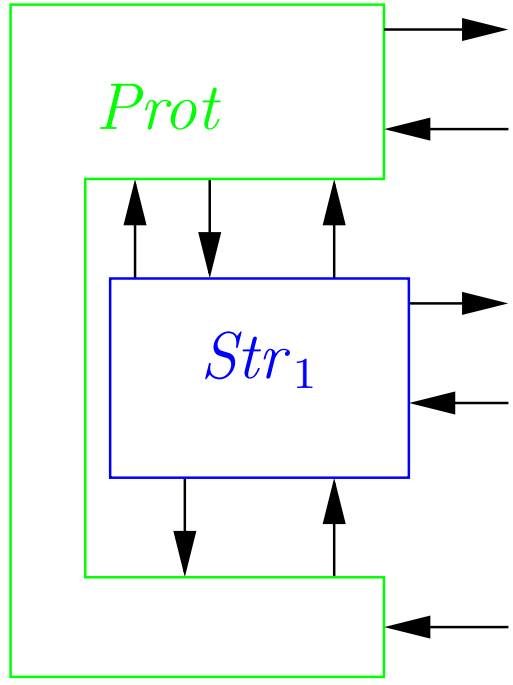
millel on samad pordid, millel teenust pakutakse.

- Kujutame ette, et Str_0 on idealiseeritud variant Str_1-st.
- Realiseeritud tavaliselt üheainsa ITM-na.

Millal on Str_1 kasutamine vähemalt sama turvaline kui Str_0 kasutamine?

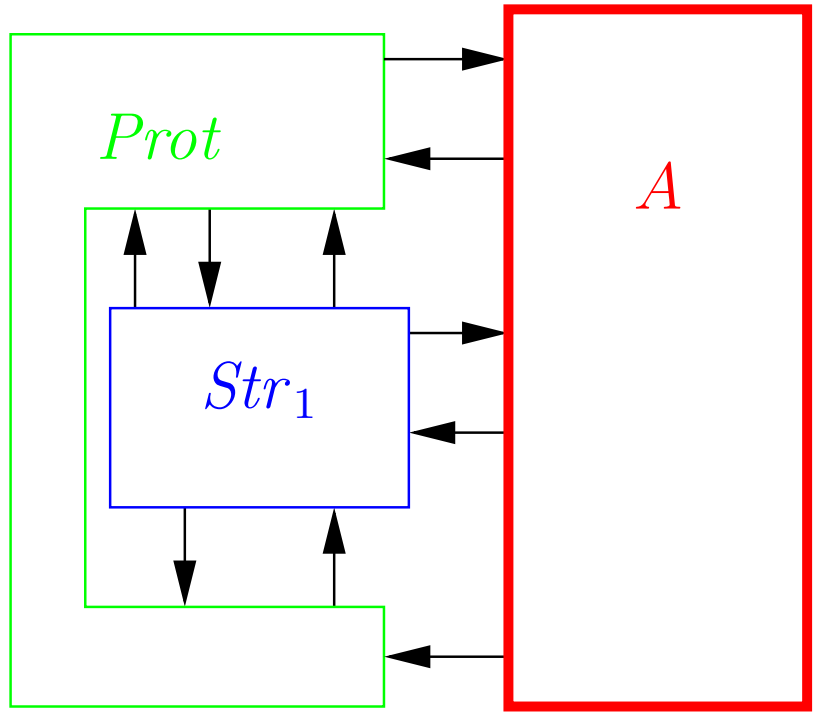
Krüptograafilise struktuuri turvalisus...

Str_1-i kasutatakse osana suuremast struktuurist, olgu $Prot$ selle struktuuri ülejäänud osa.



Krüptograafilise struktuuri turvalisus...

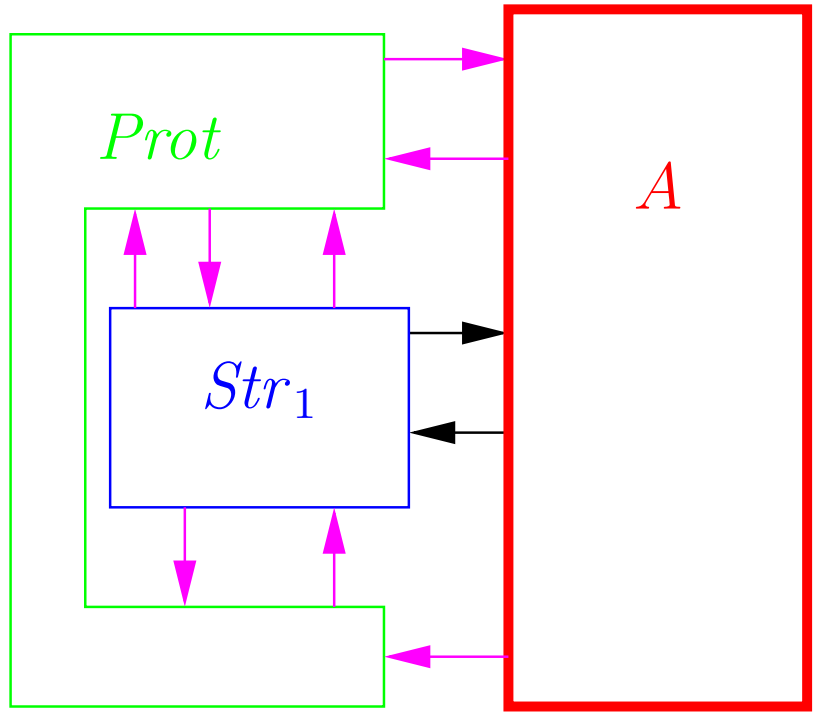
Str_1-i kasutatakse osana suuremast struktuurist, olgu $Prot$ selle struktuuri ülejäänud osa.



Selle struktuuri vabade portide otsa ühendub ründaja.

Krüptograafilise struktuuri turvalisus...

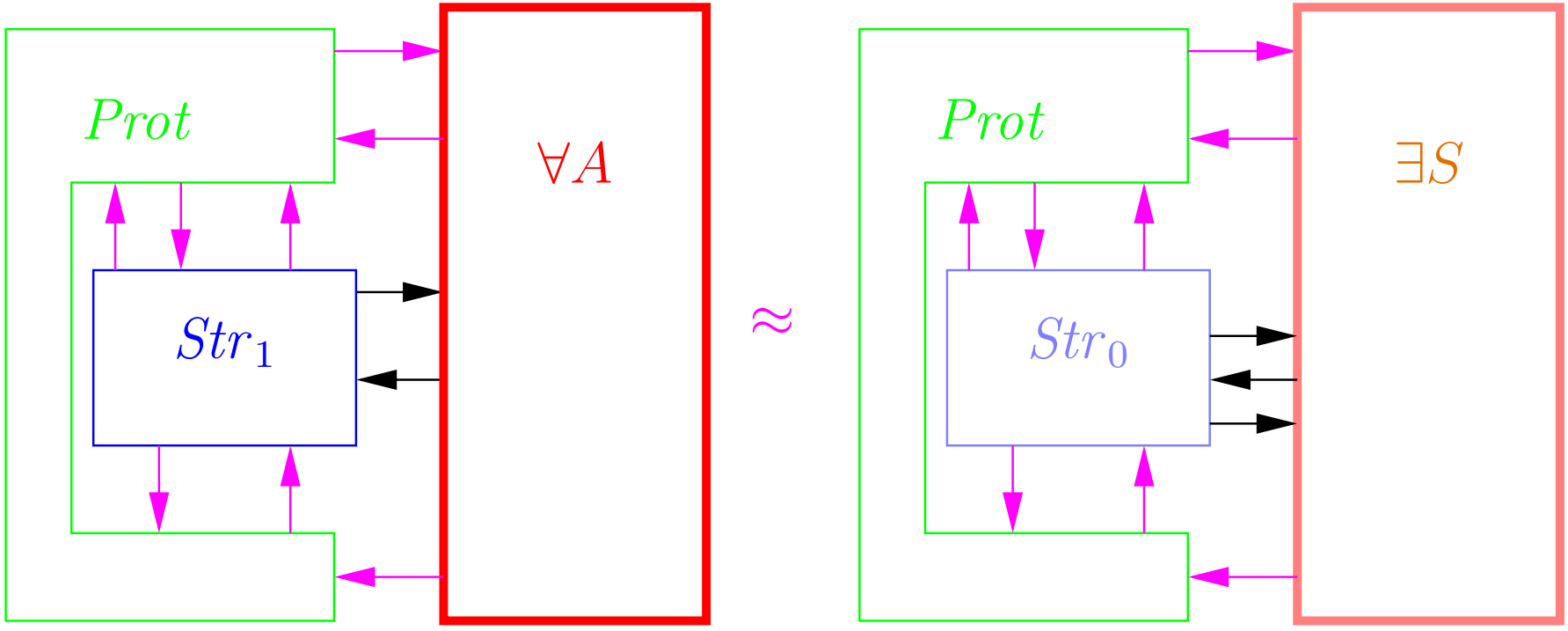
Str_1-i kasutatakse osana suuremast struktuurist, olgu $Prot$ selle struktuuri ülejäänud osa.



Selle struktuuri vabade portide otsa ühendub ründaja.
Liiklus neil portidel on $Prot$-i *vaade*.

Krüptograafilise struktuuri turvalisus...

Str_1 kasutamine koos $Prot$-iga on vähemalt sama turvaline kui Str_0 kasutamine koos $Prot$-iga, kui esimesel juhul saab $Prot$ kogeda ainult seda, mida teisel juhulgi.



Kui see kehtib iga $Prot$ jaoks, siis ütleme, et Str_1 *simuleerib* Str_0-i.

Definitsioonid

Str_1 simuleerib Str_0-i, kui

$$\forall P \, \forall A \, \exists S : view_{(P,Str_1,A)}(P) \approx view_{(P,Str_0,S)}(P) \ .$$

Str_1 simuleerib Str_0-i *universaalselt*, kui

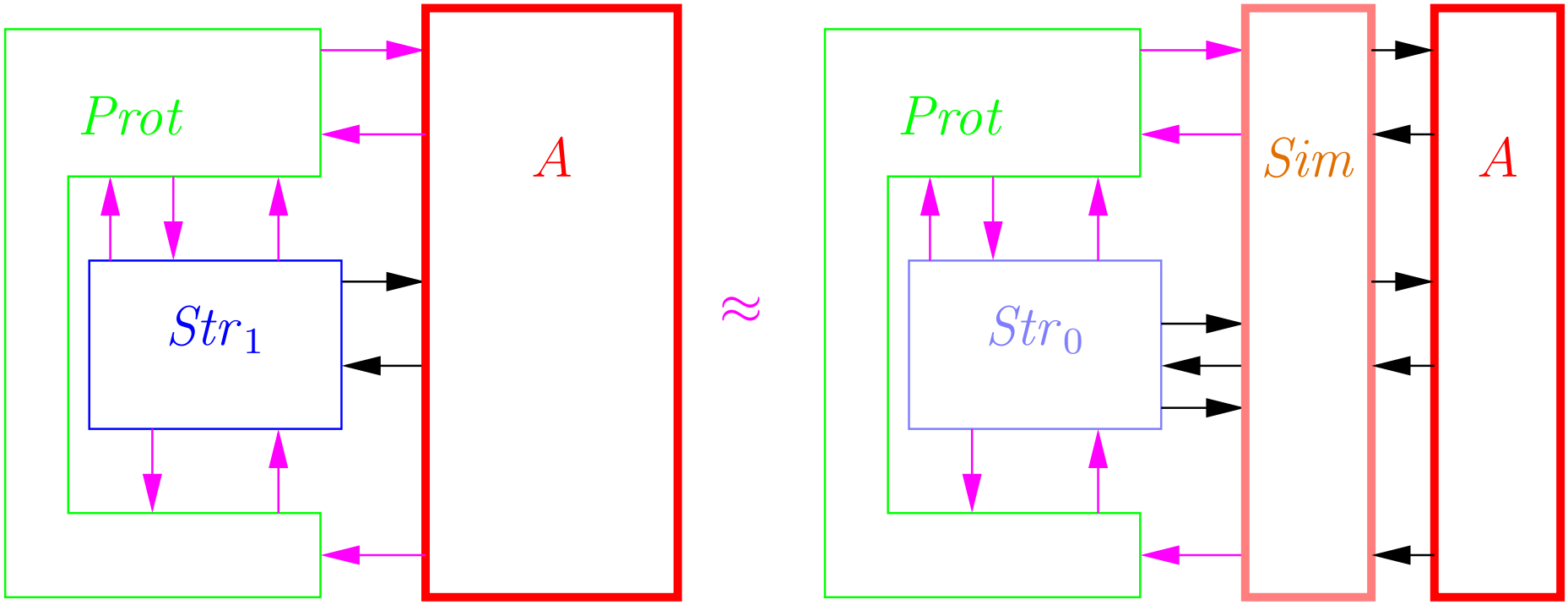
$$\forall A \, \exists S \, \forall P : view_{(P,Str_1,A)}(P) \approx view_{(P,Str_0,S)}(P) \ .$$

Str_1 simuleerib Str_0-i *musta kastina*, kui

$$\exists Sim \, \forall A \, \forall P : view_{(P,Str_1,A)}(P) \approx view_{(P,Str_0,Sim(A))}(P) \ .$$

Tähistame $Str_1 \geq Str_0$. Seda, kas simuleerimine on üldine, universaalne või musta kastina, täpsustame sõnades.

Simuleerimine musta kastina



$$\exists Sim\, \forall A\, \forall P : view_{(P, Str_1, A)}(P) \approx view_{(P, Str_0, Sim(A))}(P)$$

Ebatäpsus: Sim võib sõltuda A *portidest*.

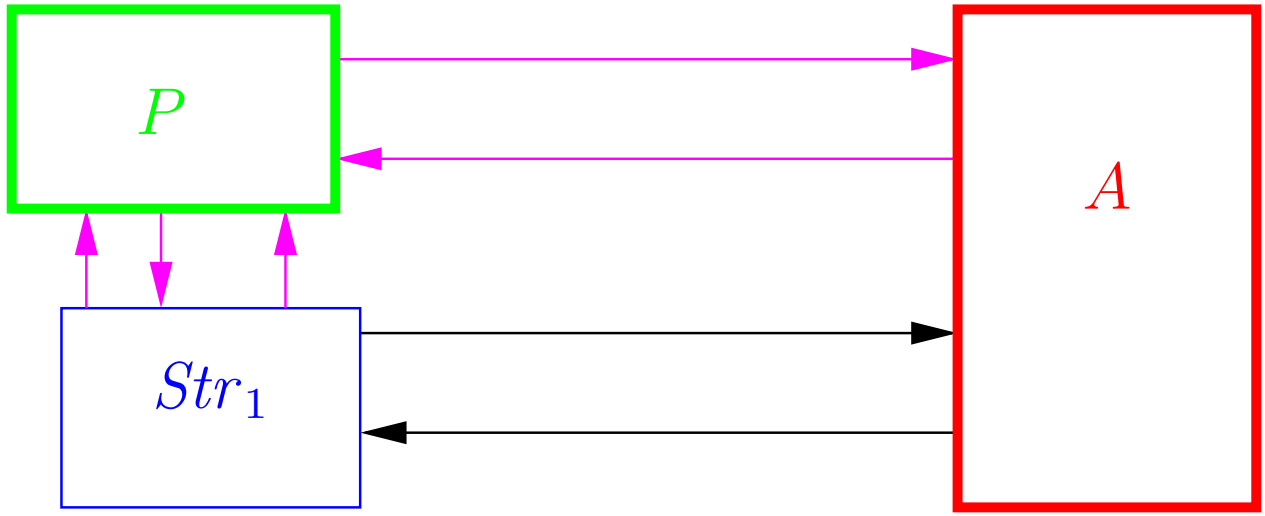
$$\forall(\text{pordihulk } \mathcal{P}) \exists Sim\, \forall A : \mathsf{ports}(A) = \mathcal{P} \Rightarrow \forall P : \ldots$$

Definitsioonid

Loeme veel, et

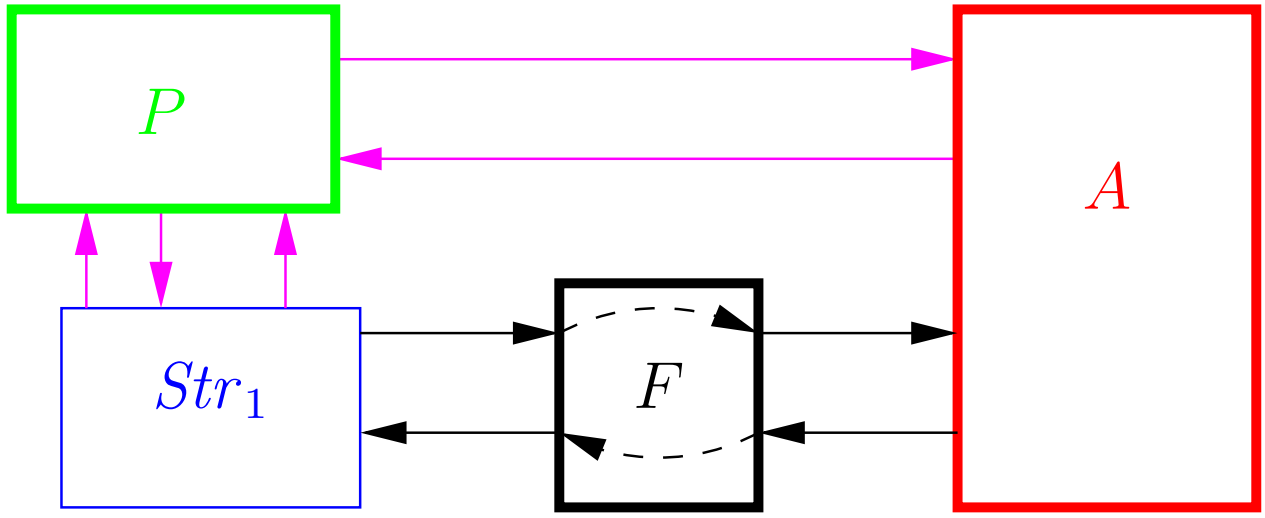
- $Prot$ on üksainus ITM;
- $view(M)$ on tõenäosusjaotus üle järjendite paaridest (p, x), kus
  - p on mõni masina M portidest;
  - x on teade, mis selle pordi kaudu saadeti või vastu võeti;
  - järjestus on ajaline.

Universaalne simuleerimine $\Rightarrow$ musta kastina simuleerimine



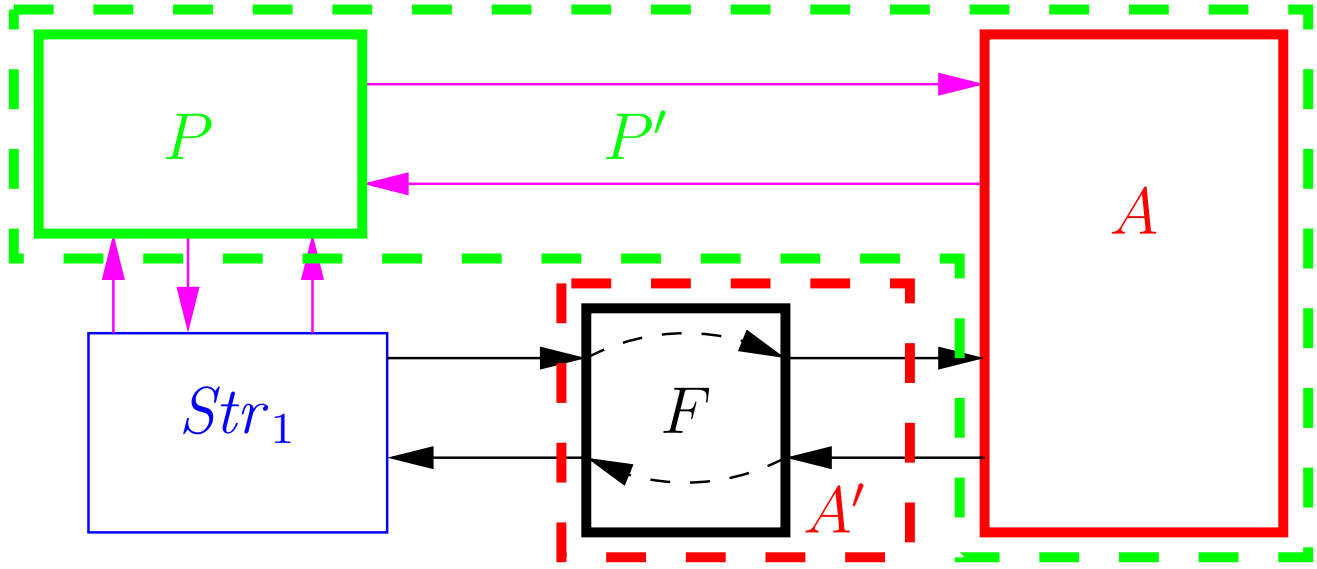
Vaja leida Sim nii, et $Sim(A)$ oleks sobiv ründaja Str_0-le.

Universaalne simuleerimine $\Rightarrow$ musta kastina simuleerimine



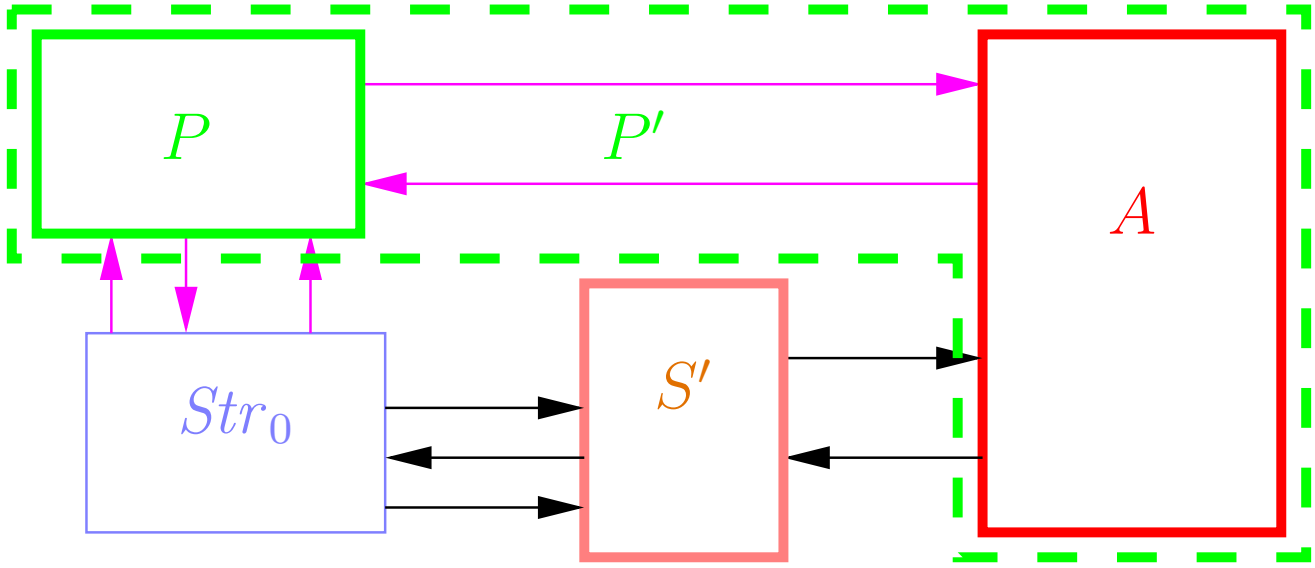
Lisame masina F, mis tegeleb ainult ühenduste edastamisega.

Universaalne simuleerimine $\Rightarrow$ musta kastina simuleerimine



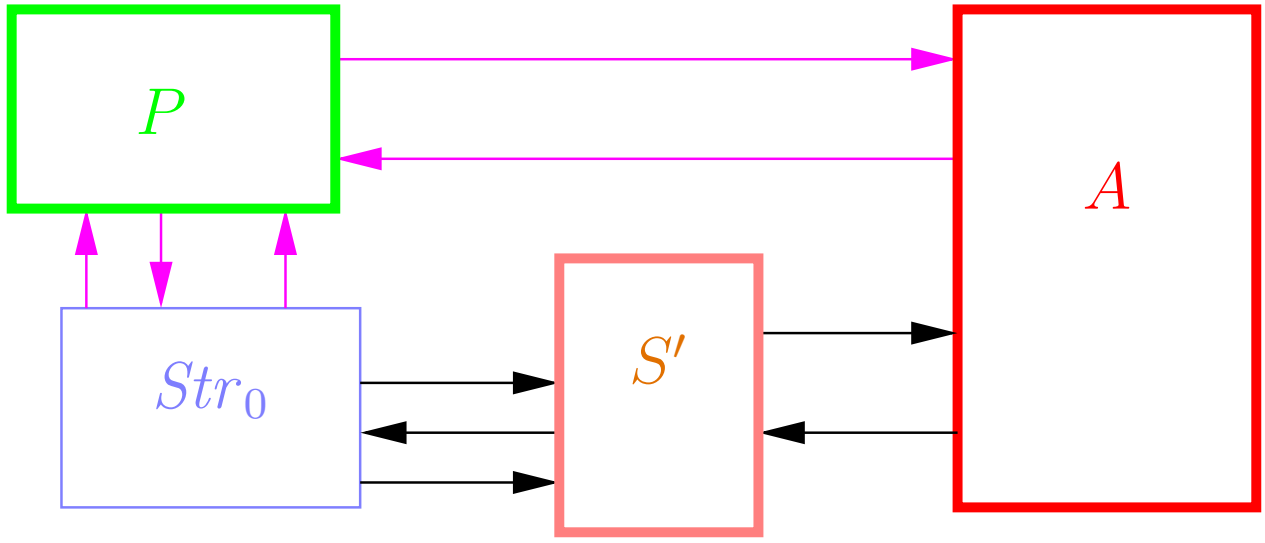
Defineerime uue kasutaja ja ründaja.

Universaalne simuleerimine $\Rightarrow$ musta kastina simuleerimine



Kasutame $Str_1 \geq^{\text{uni.}} Str_0$-i.
S' sõltub ainult A'-st, mis ei sõltu eriti millestki.

Universaalne simuleerimine $\Rightarrow$ musta kastina simuleerimine



S' ongi otsitav simulaator.

Süsteemid

- Süsteem on struktuuride hulk.
- Süsteemi erinevatel elementidel võivad olla erinevad pordid, millel teenust pakutakse.
  - See on vajalik juhul, kui tahame modelleerida midagi, kus on palju osapooli.
  - Erinevad struktuurid vastavad erinevatele võimalikele ausate osapoolte hulkadele.

Süsteemide simuleerimine

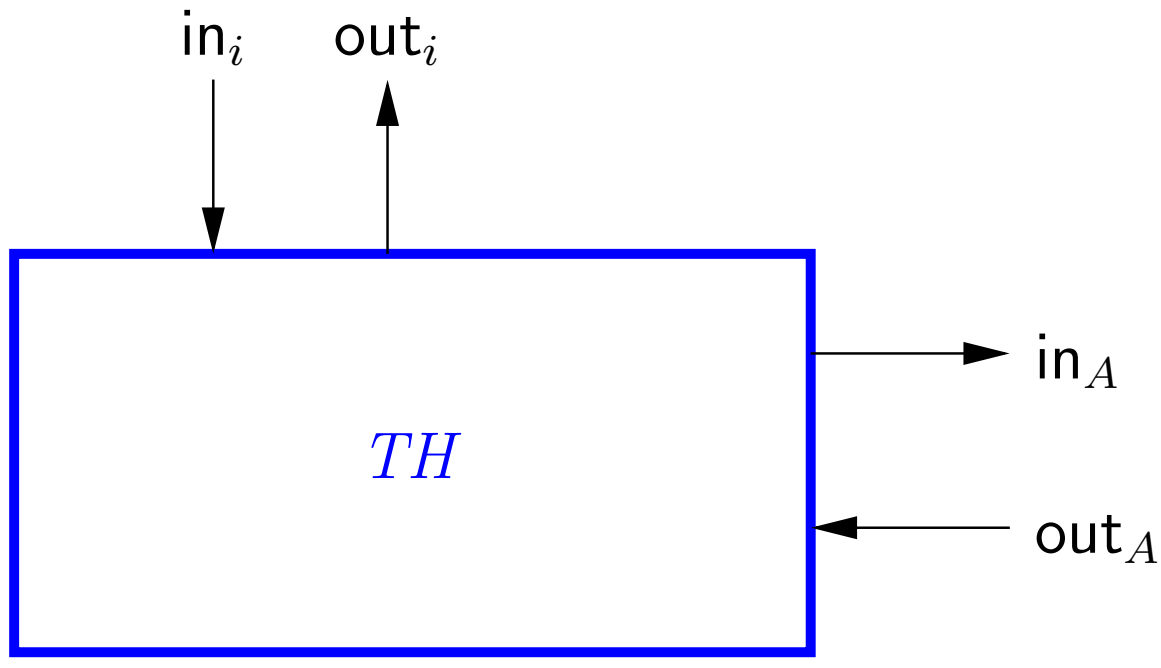
- Sys_1 *simuleerib* Sys_0-i [omadusega], kui iga $Str_1 \in Sys_1$ jaoks leidub $Str_0 \in Sys_0$ nii, et Str_1 simuleerib Str_0-i [omadusega].
  - Üldisemalt: võib olla antud funktsioon $f : Sys_1 \to \mathcal{P}(Sys_0)$ (sobivad struktuurid).
    - See funktsioon peab säilitama pordid, millel teenust pakutakse.
  - Siis nõutakse, et Str_0 eelmises punktis kuuluks hulka $f(Str_1)$.
- Simuleerimist tähistame jällegi $\geq$-ga.
  - Üldisemalt: $\geq^f$-ga.

Näide: turvaline teateedastus

- Tahame järgmist funktsionaalsust:
  - n osapoolt (nimedega $1, \ldots, n$);
  - iga osapool saab igale teisele osapoolele teateid saata;
  - saatmine on konfidentsiaalne ja autentne.
- Meil tuleb lubada teatud "lõtke":
  - teated ei pruugi kohale jõuda;
  - teada on, kes kellega räägib;
    - "mõistliku" realisatsiooni korral lekib ka teate pikkus.
  - Saadetud teadete pikkus ja arv on polünomiaalne turvalisusparameetri suhtes.
- Järgnevas näites lubame ka teadete kordamist.

Ideaalne struktuur

- Koosneb ühestainsast masinast TH.



Ideaalne struktuur

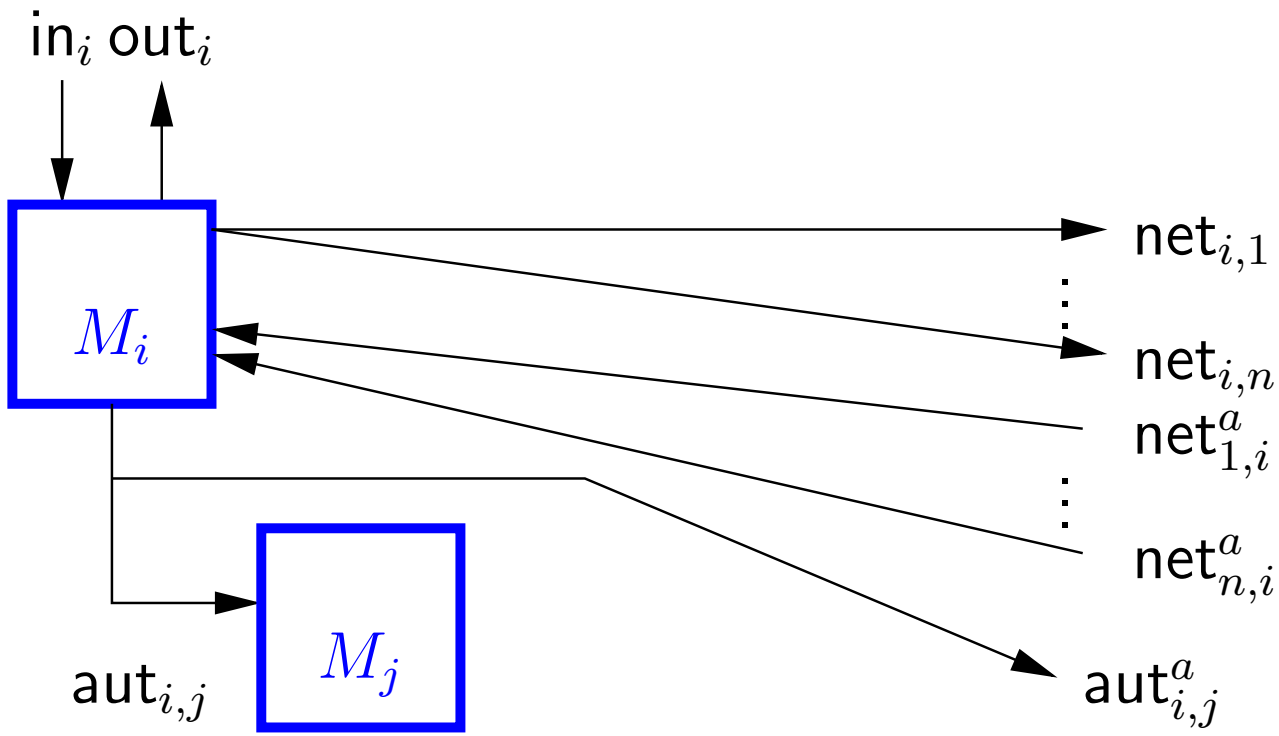
- TH olek koosneb järgmistest osadest:
  - iga $i \in \{1, \ldots, n\}$ jaoks: $init_i \in \{0, 1\}$.
    - "'kas i on omale võtmed genereerinud?"'
  - iga $i, j \in \{1, \ldots, n\}$ jaoks: $init_{i,j} \in \{0, 1\}$.
    - "'kas j on i võtmed kätte saanud?"'
  - iga $i, j \in \{1, \ldots, n\}$ jaoks: sõnumite list $D_{i,j}$, mis osapool i osapoolele j saatnud on.
- TH käsud: initsialiseerimine.
  - Kui in_i-st tuleb (init) ja $init = 0$, siis võtta $init := 1$ ja saata (init, i) out_A-sse.
  - Kui in_A-st tuleb (init, i, j) ja $init_{i,j} = 0$, siis võtta $init_{i,j} = 1$ ja saata (init, i) out_j-i.

Ideaalne struktuur

- TH käsud: saatmine ja vastuvõtmine.
  - Kui in_i-st tuleb (saada, j, m) ja i saab j-le saata ja i pole veel liiga palju teateid saatnud, siis
    - olgu $l := |D_{i,j} + 1|$;
    - olgu $D_{i,j}[l] := m$;
    - saada $(\mathsf{saadeti}, i, j, l, |m|)$ in_A-sse.
  - Kui in_A-st tuleb $(\mathsf{saadeti}, i, j, l)$ ja j suudab i teateid vastu võtta ja $|D_{i,j}| \geq l$, siis saada $(\mathsf{saadeti}, i, D_{i,j}[l])$ out_j-i.

Reaalne struktuur

- Koosneb n-st masinast $M_1, \ldots, M_n$ (igale kasutajale üks).
- Kasutab avaliku võtmega krüptimist ja signeerimist.



Reaalne struktuur

- M_i olek koosneb järgmistest osadest:
  - omaenda salajased võtmed K_i^{-s} ka K_i^{-e} signeerimiseks ja dekrüptimiseks;
  - teiste osapoolte avalikud võtmed K_j^{+s} ja K_j^{+e}.
  - juba saadetud teadete arv.
- M_i käsud: initsialiseerimine.
  - Kui in_i-st tuleb (init) ja K_i^{-s} ja K_i^{-e} pole veel defineeritud, siis genereeri võtmepaarid (K_i^{+s}, K_i^{-s}) ja (K_i^{+e}, K_i^{-e}) ning saada (K_i^{+s}, K_i^{+e}) kanalitesse $\mathsf{aut}_{i,j}$ ja $\mathsf{aut}^a_{i,j}$.
  - Saades esmakordselt kanalist $\mathsf{aut}_{j,i}$ võtmed (K_j^{+s}, K_j^{+e}), salvesta need ning saada (init, j) out_i-sse.

Reaalne struktuur

- M_i käsud: saatmine ja vastuvõtmine.
  - Kui in_i-st tuleb (saada, j, m) ja i saab j-le saata ja pole veel liiga palju teateid saatnud, siis
    - Saata $enc(K_j^{+e}, sign(K_i^{-s}, [i, m, j]))$ $\mathsf{net}_{i,j}$-i.
    - Suurendada juba saadetud teadete arvu.
  - Kui $\mathsf{net}_{j,i}^{a}$-st tuleb midagi ja i saab j-lt teateid vastu võtta, siis
    - parseldada saadut kui $enc(K_i^{+e}, sign(K_j^{-s}, [j, m, i]))$, s.t. dekrüpti ja kontrolli signatuuri ja teate kuju.
    - Kui parseldus, siis saata $(\mathsf{saadeti}, j, m)$ out_i-sse.

Reaalne $\geq^{bb}$ Ideaalne

- Sedasorti tõestused on mahukad.
- Tõestuse ülesehitusest räägime natukese aja pärast.

Struktuuride kompositsioon

- Struktuuride $Str_1, \ldots, Str_n$ *kompositsioon* on struktuur Str, kus
  - masinateks on kõik struktuuridesse $Str_1, \ldots, Str_n$ kuuluvad masinad;
  - ühendused on kõigi sama nimega sisend- ja väljundportide vahel.
  - Ei ole lubatud, et
    - oleks mitu sama nime ja suunaga porti;
    - ühendatakse ära port, mis oli $Str_1, \ldots, Str_n$-s ründaja jaoks mõeldud.
  - Kui mõni mittelubatud asi juhtub, siis need struktuurid ei ole komponeeritavad.
- Tähistame $Str = Str_1 \times \cdots \times Str_n$.

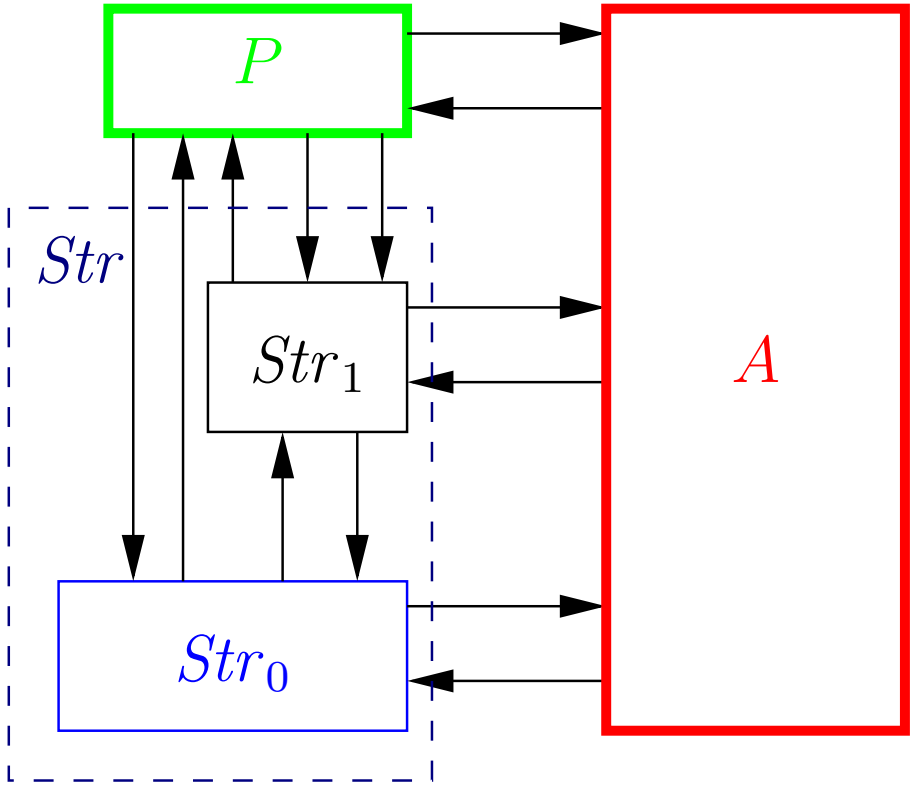
Süsteemide kompositsioon

- Sys on süsteemide $Sys_1, \ldots, Sys_n$ *kompositsioon*, kui iga tema struktuur esitub ühesel viisil mingite $Sys_1, \ldots, Sys_n$-i kuuluvate struktuuride kompositsioonina.
  - Sys ei pea sisaldama kõikvõimalikke selliseid kompositsioone.
- Tähistame $Sys \in Sys_1 \times \cdots \times Sys_n$.

Kompositsiooniteoreem struktuuridele

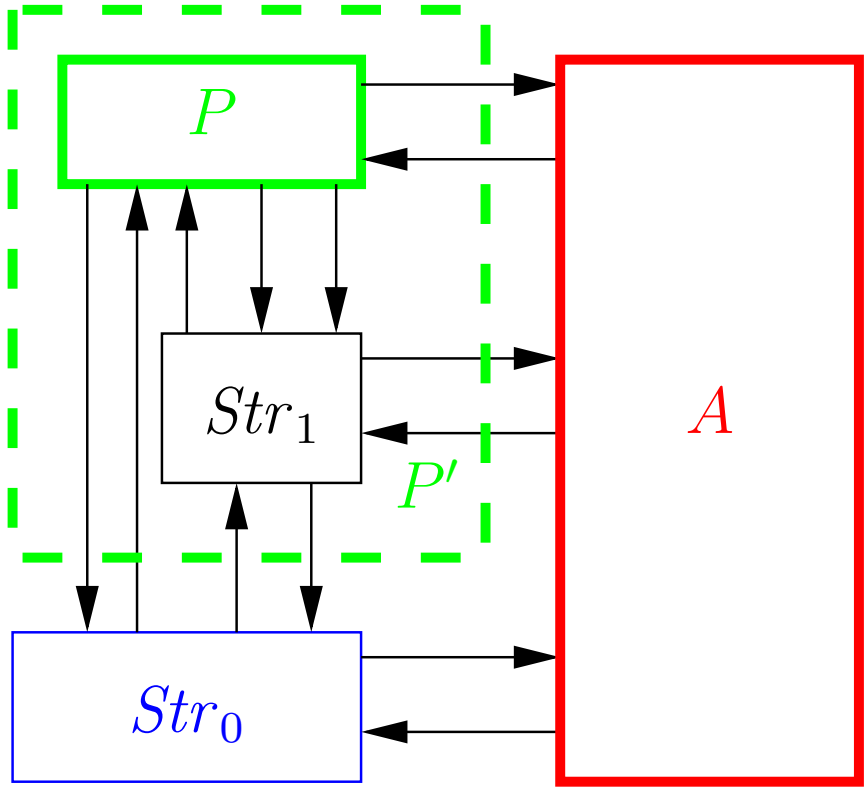
- Olgu $\mathcal{S}tr = \mathcal{S}tr_1 \times \mathcal{S}tr_0$ ja $\mathcal{S}tr' = \mathcal{S}tr_1 \times \mathcal{S}tr'_0$.
- Olgu $\mathcal{S}tr_0 \geq \mathcal{S}tr'_0$ [omadusega].
- Siis $\mathcal{S}tr \geq \mathcal{S}tr'$ [omadusega].

Tõestus



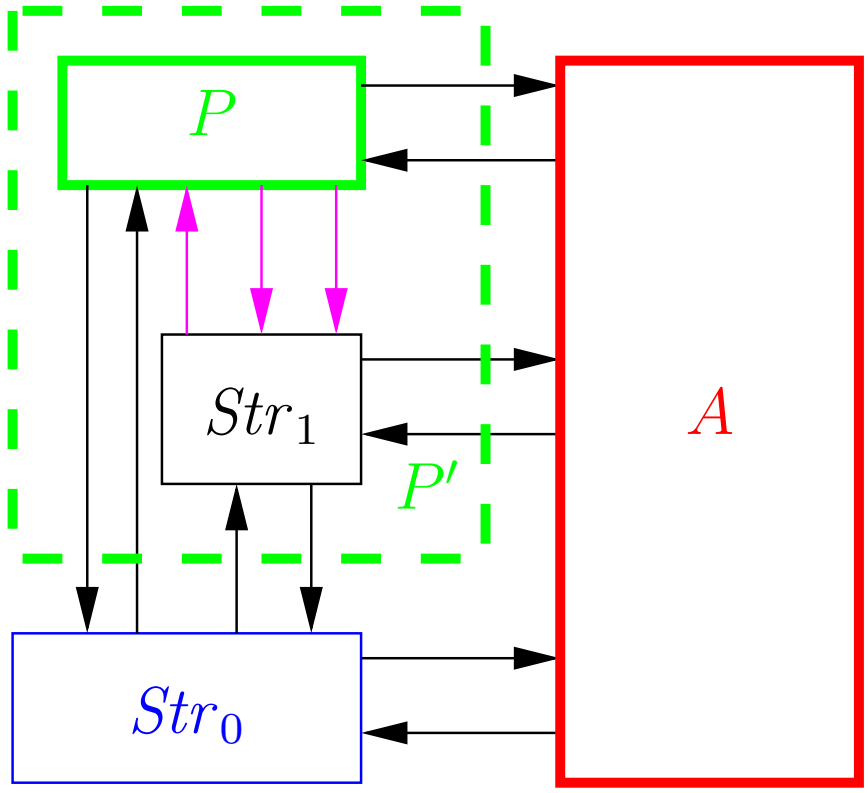
Algseis.

Tõestus



Olgu P' masin, mis on P ja Str_1 kombinatsioon.

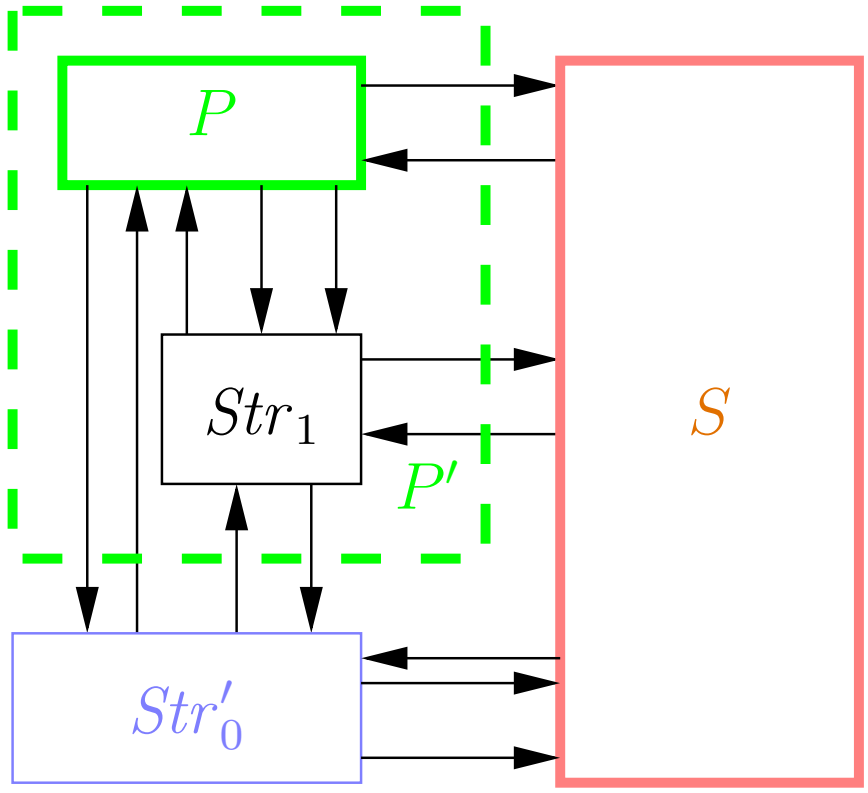
Tõestus



P ja Str_1-e vahel olnud ühendused on nüüd P' ja P' vahel.

P'-i vaade on vähemalt sama rikas kui P oma.

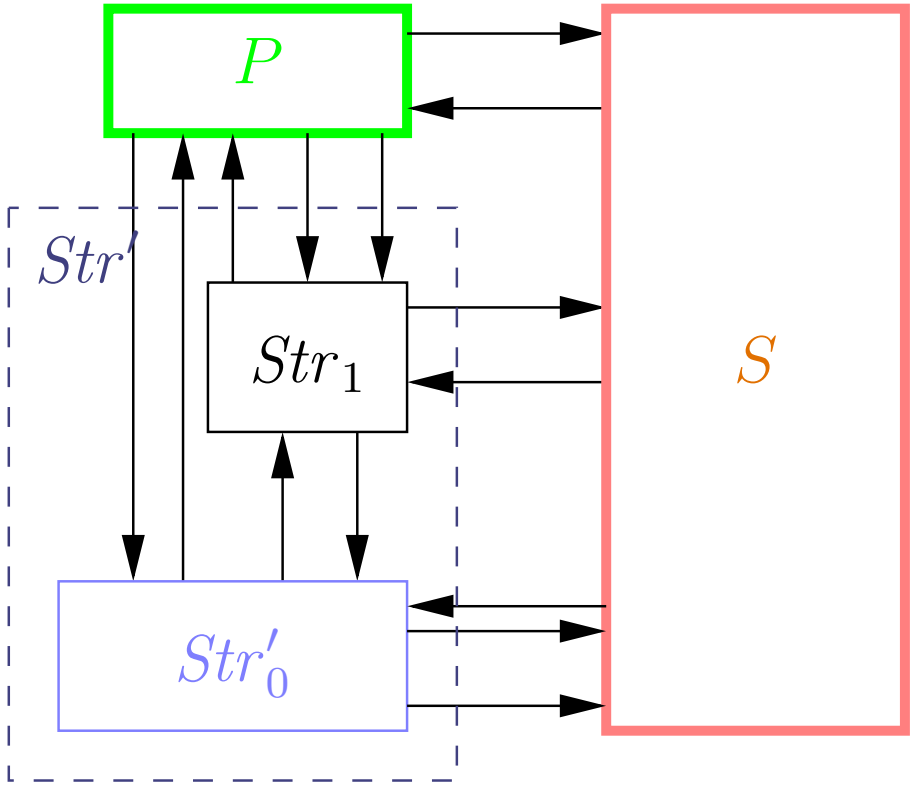
Tõestus



Et $Str_0 \geq Str'_0$, siis leidub S nii, et P' vaade ei muutu.

Siis ka P vaade ei muutu.

Tõestus



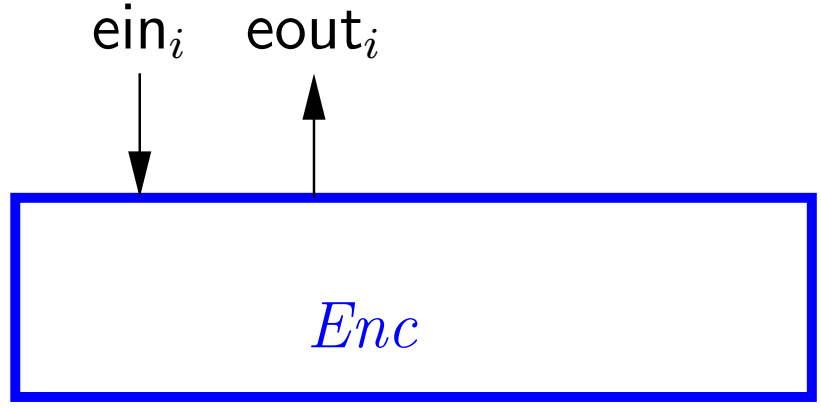
Dekombineerime masina P'.

Kompositsiooniteoreem süsteemidele

- Olgu $Sys \in Sys_1 \times Sys_0$ ja $Sys' \in Sys_1 \times Sys'_0$.
- Olgu $Sys_0 \geq Sys'_0$.
- Sisaldagu Sys' piisavalt palju struktuure (igale Sys-i struktuurile peab vastama Sys' struktuur).
- Siis $Sys \geq Sys'$.

Turvalise teateedastuse tõestusest

1. Asenda krüptimine ja dekrüptimine masinates M_i pöördumistega spetsiaalse masina poole.



Siin M_i ühendub ein$_i$ ja eout$_i$ külge.

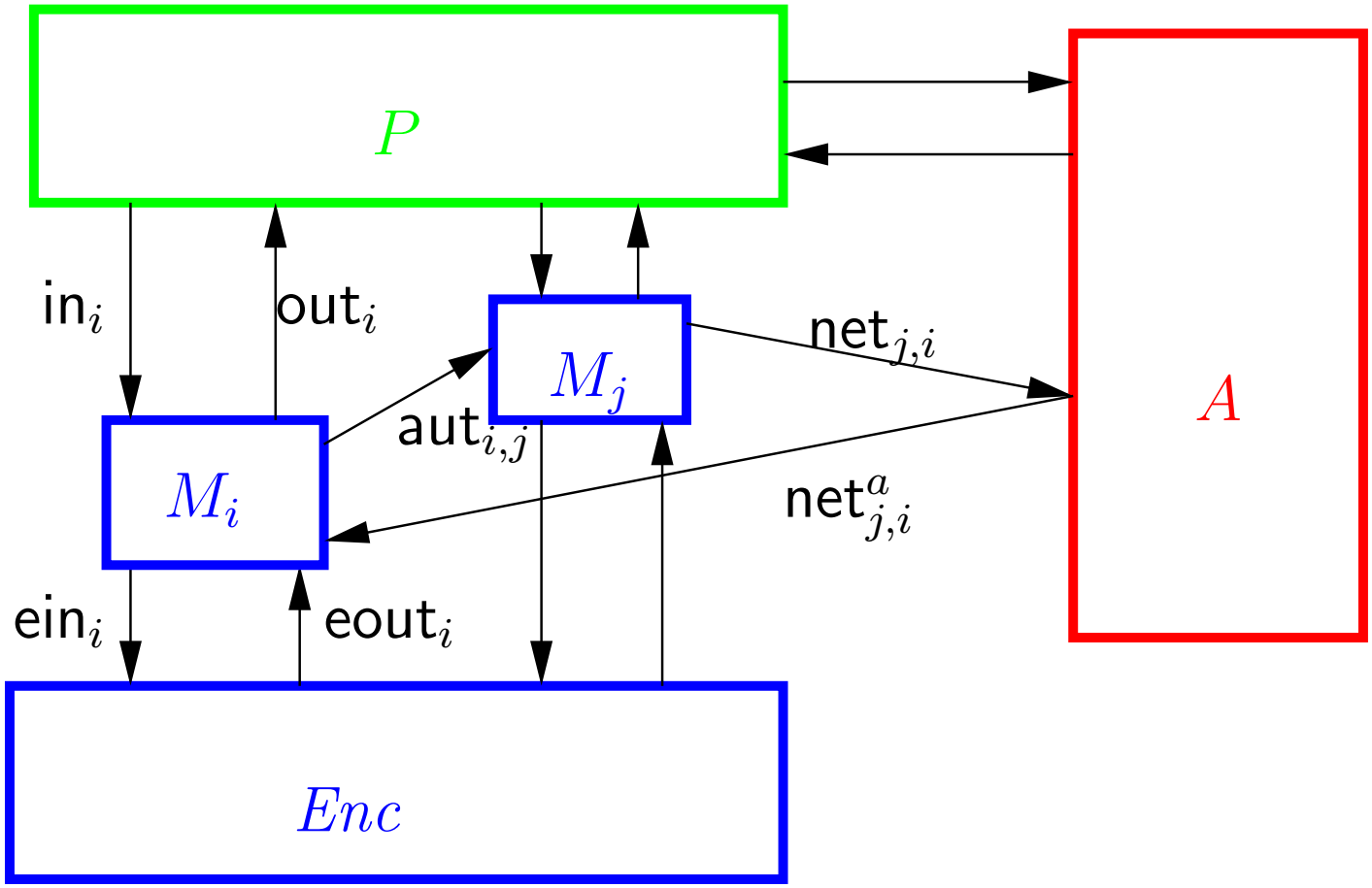
Masin Enc

- Sisemine olek:
  - juba loodud võtmepaaride arv v;
  - võtmepaarid (K_r^+, K_r^-) ise ($1 \leq r \leq v$);
  - iga r jaoks: port in_i, millelt tulnud päringu peale genereeriti (K_r^+, K_r^-).
- Käsud:
  - Kui in_i-st tuleb (gen), siis
    - suurenda v-d;
    - genereeri uus võtmepaar (K_v^+, K_v^-) ja jäta meelde, et päring tuli pordist in_i;
    - saada K_v^+ out_i-sse.

Masin Enc

- Veel käske:
  - Kui in_i-st tuleb (enc, K, m), ja $K = K_r^+$ mingi juba loodud võtmepaari (K_r^+, K_r^-) jaoks, siis saata $enc(K_r^+, m)$ out_i-sse.
  - Kui in_i-st tuleb (dec, K, m) ja $K = K_r^+$ mingi juba loodud võtmepaari (K_r^+, K_r^-) jaoks, mis genereeriti in_i-st tulnud päringu peale, siis saata $dec(K_r^-, m)$ out_i-sse.

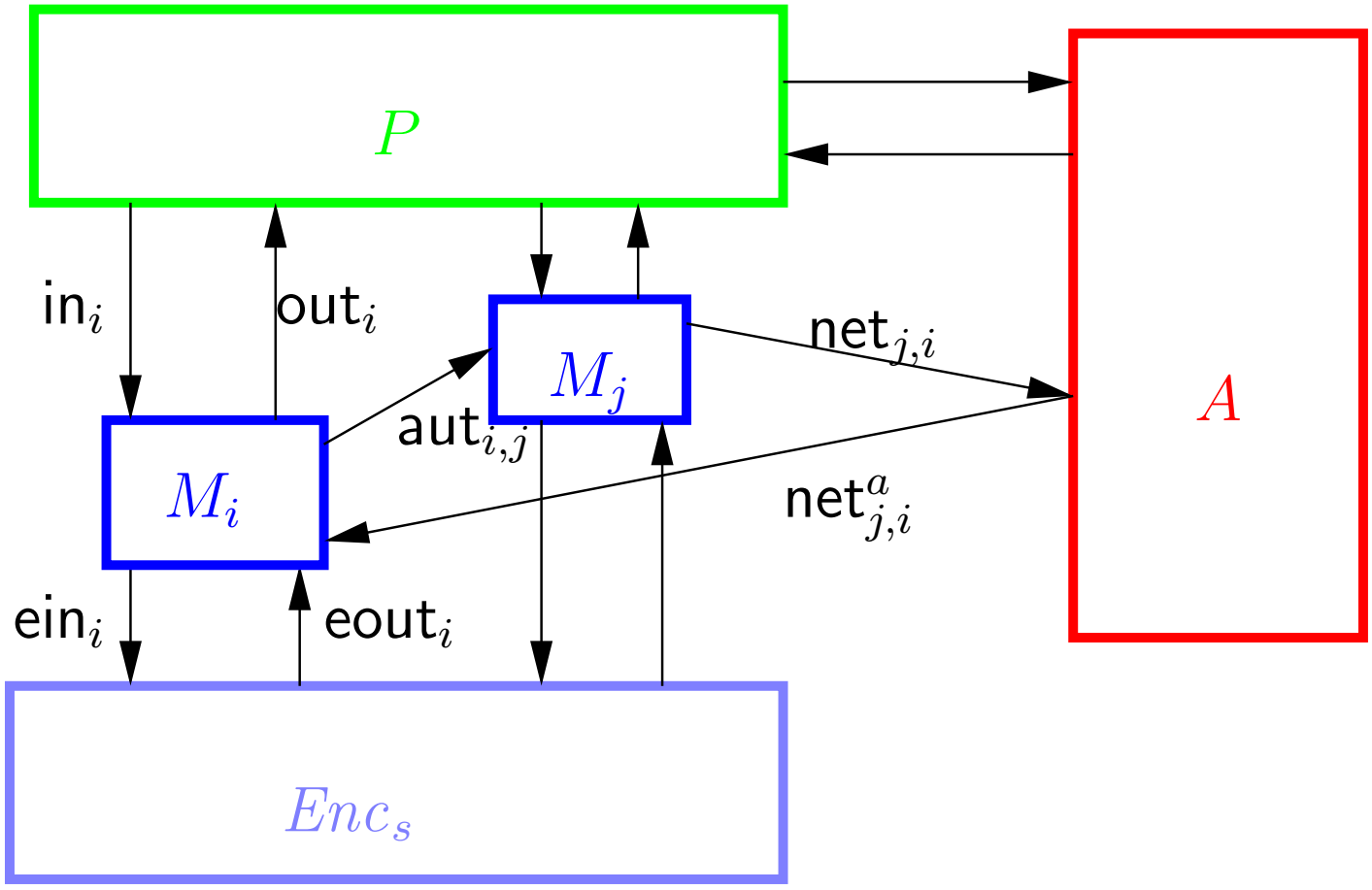
Modifitseeritud reaalne struktuur



Masin Enc_s

- Samad pordid kui Enc-l.
- Muudatused käskudes:
  - Kui in_i-st tuleb (enc, K, m), ja $K = K_r^+$ mingi juba loodud võtmepaari (K_r^+, K_r^-) jaoks, siis
    - olgu $c = enc(K_r^+, 0^{|m|})$;
    - salvesta (m, K, c) ja saada c out_i-sse.
  - Kui in_i-st tuleb (dec, K, m) ja täidetud on samad tingimused, mis masinal Enc, siis
    - kui leidub c, et (m, K, c) on salvestatud, siis saada c out_i-sse;
    - muidu saada $dec(K_r^-, m)$ out_i-sse.
- $Enc \geq Enc_s$
  - turvalisus adaptiivse, valitud krüptotekstiga ründe vastu

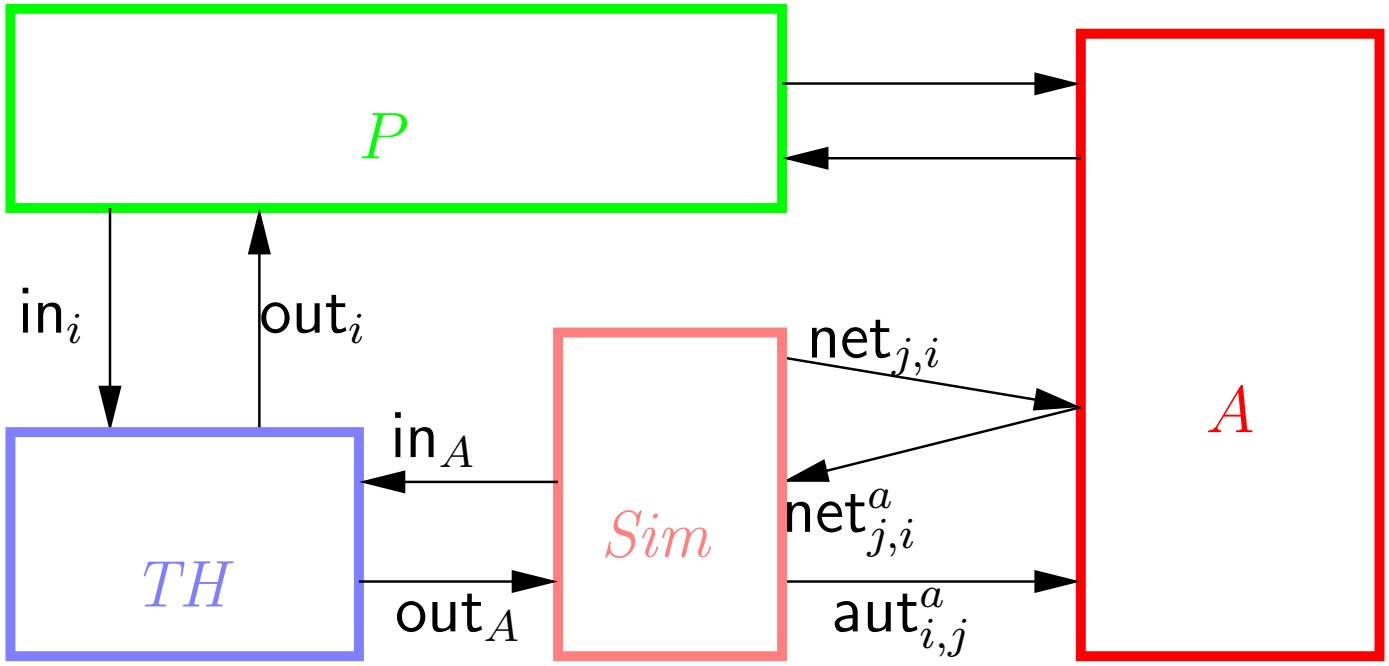
Kasutame kompositsiooniteoreemi



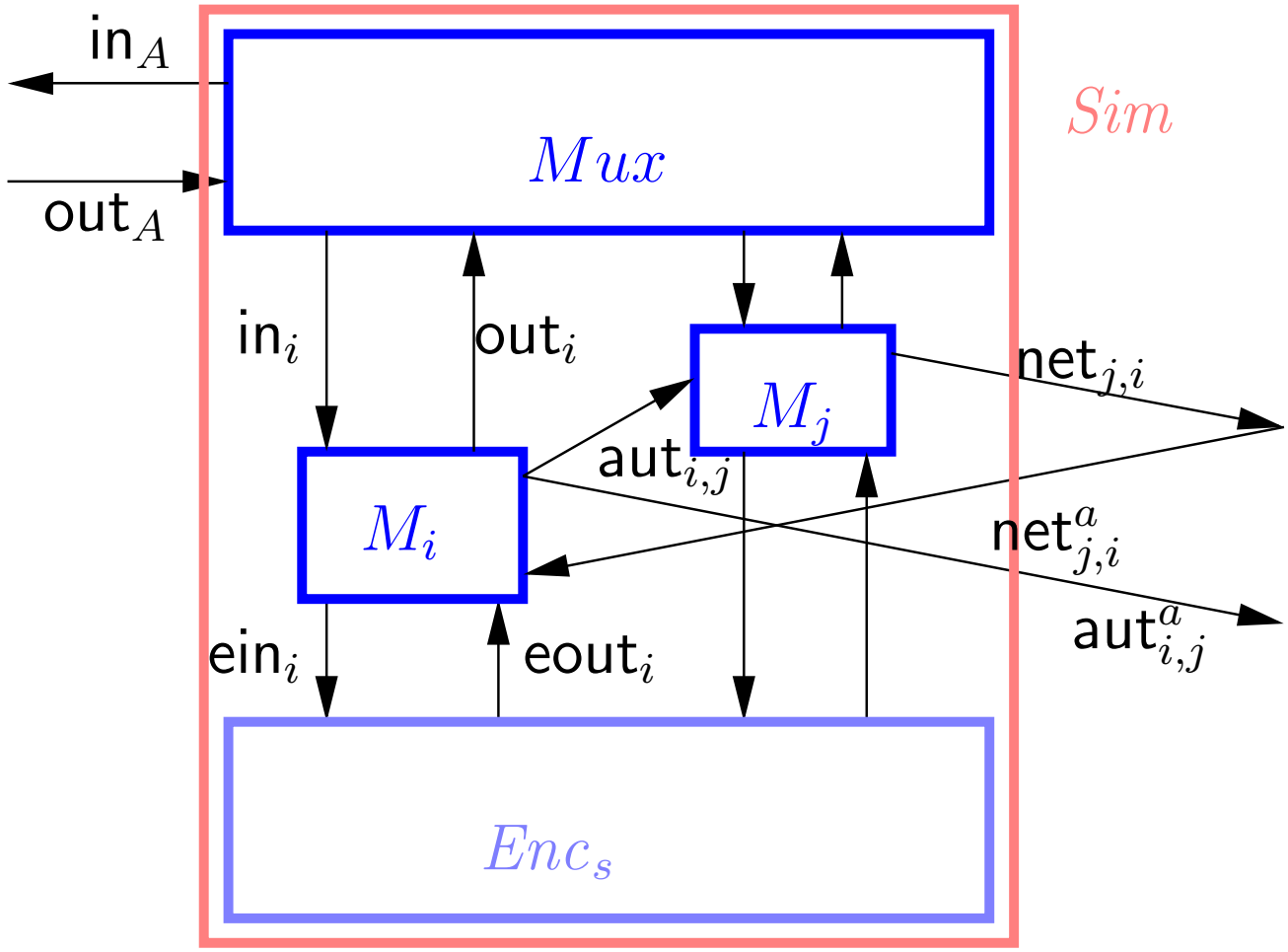
Reaalne struktuur on vähemalt sama turvaline kui see.

Simulaator

2. Konstrueeri simulaator.



Simulaatori ehitus



Mux mõtleb ise teateid välja.

Running in lock-step

3. Näita, et struktuurid $(M_1, \ldots, M_n, Enc_s)$ ja (TH, Sim) reageerivad ühesugustele sisenditele ühtemoodi.
   - Defineeri vastavus nende kahe struktuuri olekute vahel.
   - Näita, et see on bisimulatsioon.
     - ... peaaegu kõikjal.

Komponeeritavate süsteemide arv

- Olgu $Sys \in Sys_1 \times \cdots \times Sys_n$ ja $Sys' \in Sys'_1 \times \cdots \times Sys'_n$.
- Olgu $Sys_i \geq Sys'_i$ iga $i \in \{1, \ldots, n\}$ jaoks.
- Sisaldagu Sys' piisavalt palju struktuure.
- Siis $Sys \geq Sys'$.
  - Rakendame eelmist teoreemi n korda.
- Siin n ei sõltu turvalisusparameetrist k.

Parametriseeritud süsteemid

- Olgu $Sys = \dot{\bigcup}_{k \in \mathbb{N}} Sys_k$.
- Sys on *piiratud* funktsiooniga $p : \mathbb{N} \to \mathbb{N}$, kui iga $Str \in Sys_k$ suurus/keerukus on ülimalt $p(k)$.
- Süsteemi kasutaja $\{P_{Str}\}_{Str \in Sys}$ ja ründaja $\{A_{Str}\}_{Str \in Sys}$ on parametriseeritud üle struktuuride.
  - Nõuame, et nad oleksid uniformsed: leiduvad polünomiaalses ajas töötavad algoritmid P ja A nii, et $P(Str) = P_{Str}$ ja $A(Str) = A_{Str}$.
  - Nõuame ka, et nad töötaksid polünomiaalses ajas (üks polünoom terve P või A jaoks).

Parametriseeritud eristamatus

- Olgu
  - $I = \dot{\bigcup}_{k \in \mathbb{N}} I_k$;
  - iga $i \in I$ jaoks: $D_i, D_i' \in \mathcal{D}(\{0,1\}^*)$.
- Siis pered $D = \{D_i\}_{i \in I}$ ja $D' = \{D_i'\}_{i \in I}$ on *eristamatud*, kui
  - iga PPT algoritmi $\mathcal{A}$ jaoks
  - leidub kaduvväike funktsioon δ

  nii, et

$$\mathbf{P}[\mathcal{A}(1^k, i, x) \mid x \leftarrow D_i] - \mathbf{P}[\mathcal{A}(1^k, i, x) \mid x \leftarrow D_i'] \leq \delta(k)$$

  iga $i \in I_k$ jaoks.

Simuleeritavus

$Sys \geq Sys'$, kui

- leidub polünoom p nii, et
- leidub kujutus $f : Sys \longrightarrow Sys'$ nii, et
  - $f(Sys_k) \subseteq Sys'_k$.
- iga kasutaja $\{P_{Str}\}_{Str \in Sys}$ jaoks
- iga (uniformse ja pol. ajas) ründaja $\{A_{Str}\}_{Str \in Sys}$ jaoks
- leidub ründaja $\{S_{Str}\}_{Str \in Sys}$ nii, et
  - S_{Str} ühendub $f(Str) \in Sys'$ ja P_{Str} külge.
- $\{view_{(P_{Str}, Str, A_{Str})}(P_{Str})\}_{Str \in Sys} \approx$
  $\{view_{(P_{Str}, f(Str), S_{Str})}(P_{Str})\}_{Str \in Sys}$;
- $|S_{Str_k}| \leq p(k, |(P_{Str_k}, Str_k, A_{Str_k})|)$.

Universaalne simuleeritavus

$Sys \geq Sys'$, kui

- leidub polünoom p nii, et
- leidub kujutus $f : Sys \longrightarrow Sys'$ nii, et
- iga (uniformse ja pol. ajas) ründaja $\{A_{Str}\}_{Str \in Sys}$ jaoks
- leidub ründaja $\{S_{Str}\}_{Str \in Sys}$ nii, et
- iga kasutaja $\{P_{Str}\}_{Str \in Sys}$ jaoks

- $\{view_{(P_{Str}, Str, A_{Str})}(P_{Str})\}_{Str \in Sys} \approx$
  $\{view_{(P_{Str}, f(Str), S_{Str})}(P_{Str})\}_{Str \in Sys}$;
- $|S_{Str_k}| \leq p(k, |(P_{Str_k}, Str_k, A_{Str_k})|)$.

Musta kastina simuleeritavus

$Sys \geq Sys'$, kui

- leidub kujutus $f : Sys \longrightarrow Sys'$ nii, et
- leidub (unif. ja pol. ajas) simulaator $\{Sim_{Str,\mathcal{P}}\}_{Str \in Sys}$ nii, et
  - Siin $\mathcal{P}$ on ründaja portide hulk.
- iga (uniformse ja pol. ajas) ründaja $\{A_{Str}\}_{Str \in Sys}$ jaoks,
- iga kasutaja $\{P_{Str}\}_{Str \in Sys}$ jaoks
- $\{view_{(P_{Str}, Str, A_{Str})}(P_{Str})\}_{Str \in Sys} \approx$
  $\{view_{(P_{Str}, f(Str), Sim_{Str, \mathsf{ports}(A_{Str})}(A_{Str}))}(P_{Str})\}_{Str \in Sys}$.

Süsteemide kompositsioon

- Olgu $Sys^{(i)}$, kus $i \in \mathbb{N}$, parametriseeritud süsteemid.
  - S.t. $Sys^{(i)} = \dot{\bigcup}_{k \in \mathbb{N}} Sys_k^{(i)}$.
- Olgu $q : \mathbb{N} \to \mathbb{N}$.
- Parametriseeritud süsteem Sys^* on $\{Sys^{(i)}\}_{i \in \mathbb{N}}$ *kompositsioon suurusega* q, kui
  - iga $k \in \mathbb{N}$ jaoks
  - iga $Str_k \in Sys_k^*$ jaoks
  - Str_k esitub üheselt kui $Str_k^{(1)} \times \cdots \times Str_k^{(q(k))}$, kus $Str_k^{(i)} \in Sys_k^{(i)}$.
- Tähistame $Sys^* \in \prod^{(q)} \{Sys^{(i)}\}_{i \in \mathbb{N}}$.

Kompositsiooniteoreem

- Olgu $Sys^{(i)}$ ja $Sys'^{(i)}$, kus $i \in \mathbb{N}$, parametriseeritud süsteemid.
- Olgu q mingi polünoom. Olgu $Sys \in \prod^{(q)}\{Sys^{(i)}\}_{i\in\mathbb{N}}$ ja $Sys' \in \prod^{(q)}\{Sys'^{(i)}\}_{i\in\mathbb{N}}$, kus Sys' sisaldab piisavalt paju struktuure.
- ...

Vahepeal: teoreemi kasutamisest

- Olgu antud mingi krüptoprimitiivi reaalne implementatsioon Sys^R ja ideaalne funktsionaalsus Sys^I, nii et $Sys^R \geq^{\mathsf{bb}} Sys^I$.
- Olgu antud krüptoprotokoll Sys^P, mis seda primitiivi mitmes kohas (kuni $q(k)$-s kohas, kus k on turvaparameeter) kasutab.
- Defineerime $Sys^{(i)}$ ja $Sys'^{(i)}$ järgmiselt:
  - $Sys^{(1)} = Sys'^{(1)} = Sys^P$;
  - $Sys^{(i)} = Sys^R$, kui $i > 1$.
  - $Sys'^{(i)} = Sys^I$, kui $i > 1$.
- Siis $Sys \geq^{\mathsf{bb}} Sys'$.

Kompositsiooniteoreem (jätkub)

- Olgu $Sys^{(i)} \geq^{\mathsf{bb}} Sys'^{(i)}$, kusjuures
  - leidub polünoom p_1, mis piirab kõiki süsteeme $Sys^{(i)}$;
  - Simulaatorite $Sim^{(i)}$, mis leiduma peavad, pere keerukus on piiratud mingi polünoomiga p_2;
  - Leidub PPT algoritm Sim nii, et $Sim^{(i)} = Sim(i)$.
  - leidub kaduvväike funktsioon, mis sobib $\approx$-i definitsiooni kõigi PPT eristajate $\mathcal{A}$ ja kõigi indeksite i jaoks.
    - Jutt on kasutaja vaadete eristamisest reaalse ja ideaalse süsteemiga suhtlemisel.
    - δ võib sõltuda ainult $\mathcal{A}$ tööajast (sõltuvuses turvalisusparameetrist).
- Siis $Sys \geq^{\mathsf{bb}} Sys'$.

leidub δ...

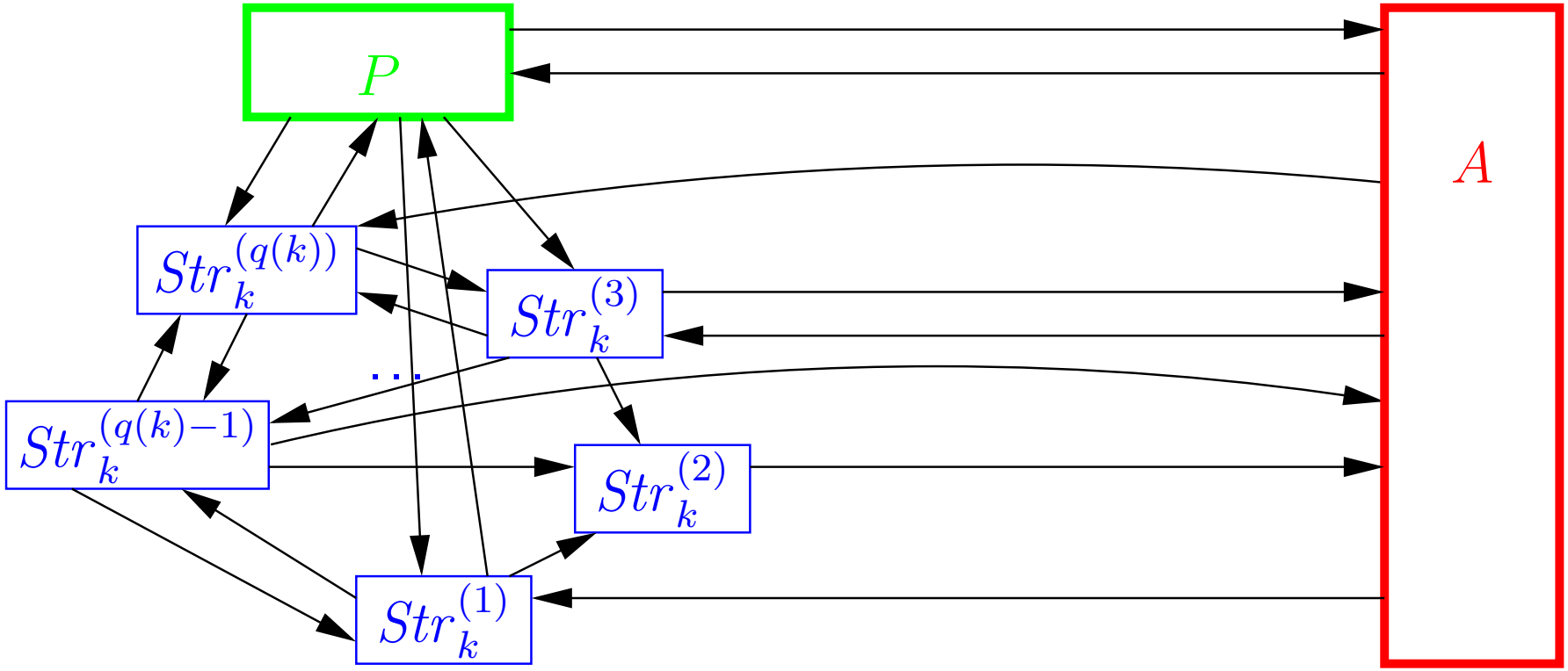
- Iga polünoomi t jaoks
- leidub kaduvväike funktsioon δ_t nii, et
- iga PPT algoritmi $\mathcal{A}$ jaoks, mille tööaega piirab t,
- iga $i \in \mathbb{N}$ ja $k \in \mathbb{N}$ jaoks,
- iga $Str_k^{(i)} \in Sys_k^{(i)}$, $P_{Str_k^{(i)}}$ ja $A_{Str_k^{(i)}}$ jaoks, mis pole suuremad kui $t(k)$:

$$\mathbf{P}[\mathcal{A}(1^k, i, r) | r \leftarrow view_{(P_{Str_k^{(i)}}, Str_k^{(i)}, A_{Str_k^{(i)}})}(P_{Str_k^{(i)}})] -$$

$$\mathbf{P}[\mathcal{A}(1^k, i, r) | r \leftarrow$$

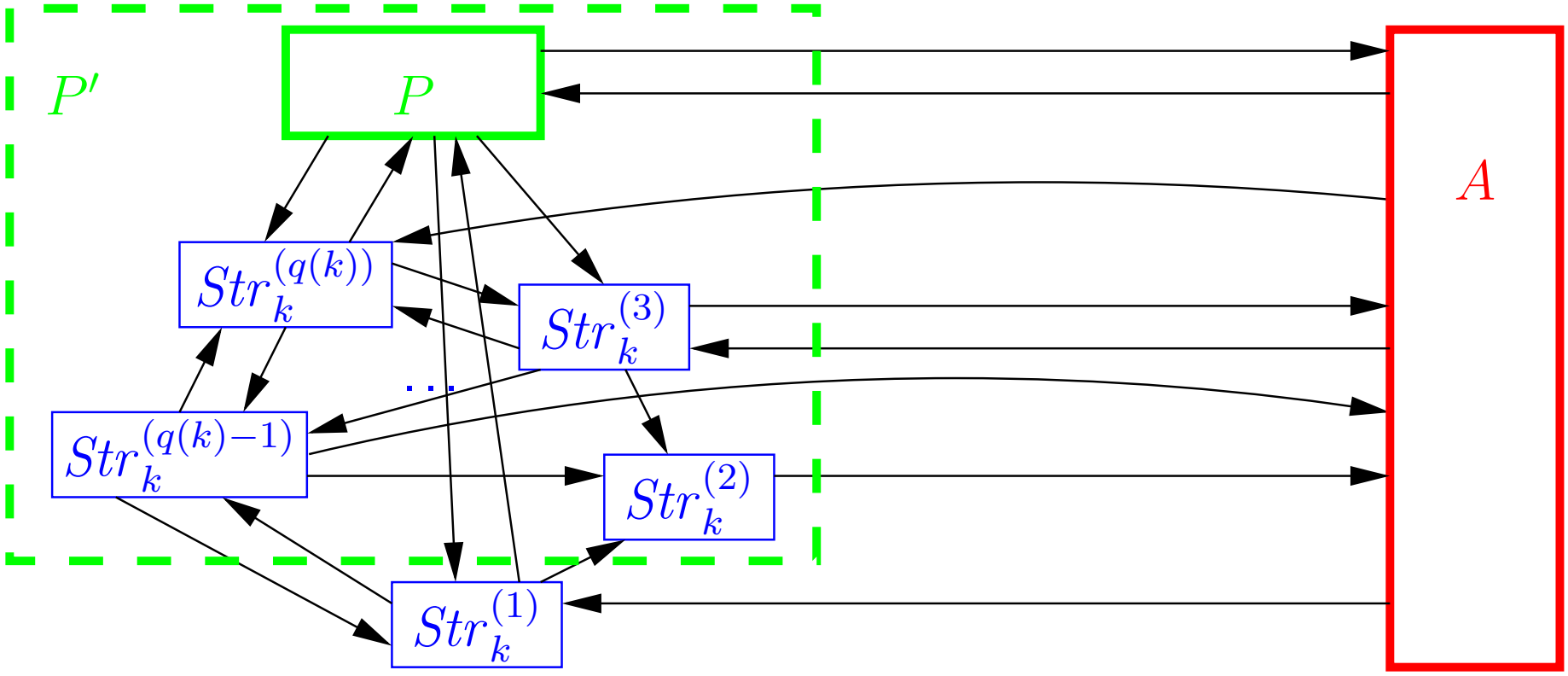
$$view_{(P_{Str_k^{(i)}}, f^{(i)}(Str_k^{(i)}), Sim^{(i)}_{Str_k^{(i)}, \mathsf{ports}(A_{Str_k^{(i)}})}(A_{Str_k^{(i)}}))}(P_{Str_k^{(i)}})] \leq \delta_t(k) \quad .$$

Tõestus



- Olgu $\{P_{Str}\}_{Str \in Sys}$ suurus piiratud t_P-ga.
- Olgu $\{A_{Str}\}_{Str \in Sys}$ suurus piiratud t_A-ga.
- Olgu $\mathcal{A}$ mingi P vaadete eristaja tööaja piiranguga $t_{\mathcal{A}}$.
- Olgu t polünoom, nii et $t \geq \max\{t_{\mathcal{A}}, t_P + p_1 q, t_A + p_2 q\}$.

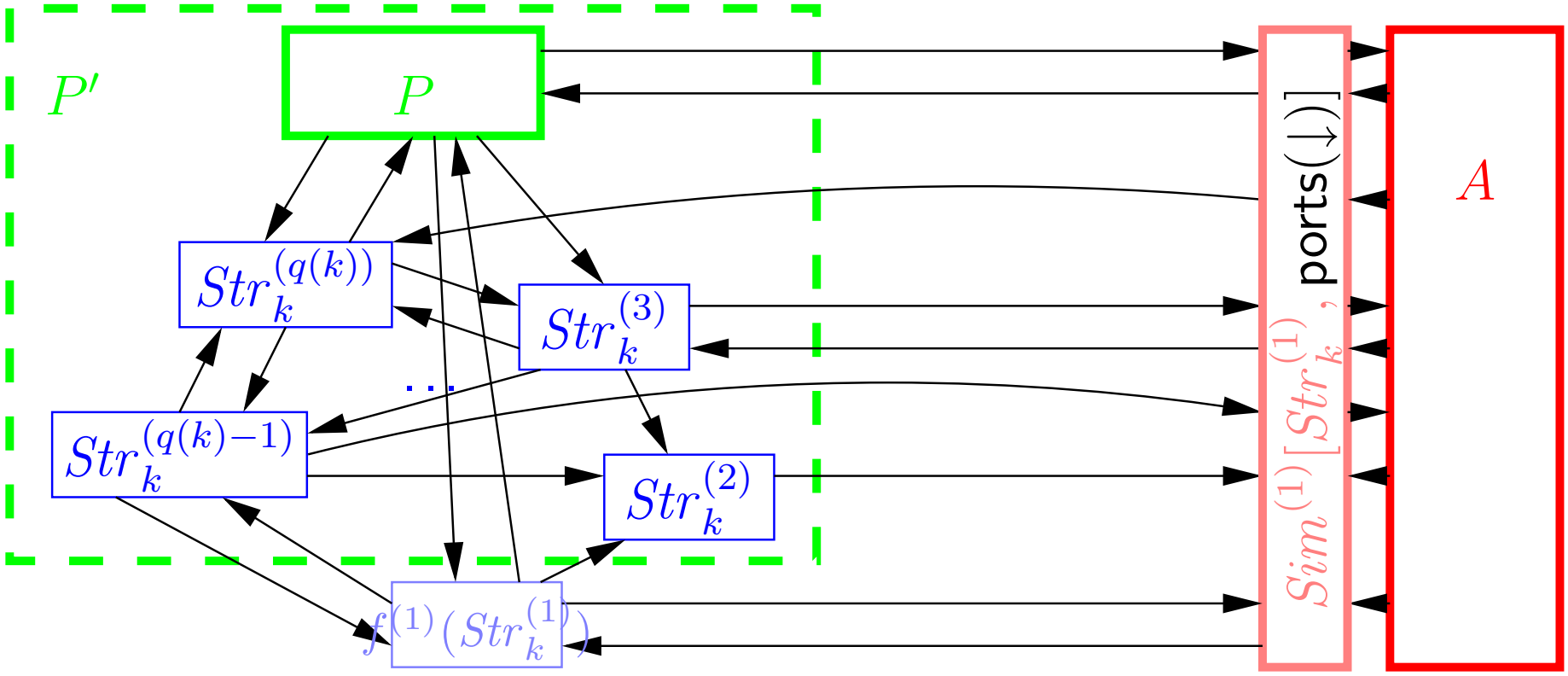
Tõestus



$\mathcal{A}$ eelis P vaate eristamisel tema esialgsest vaatest on ülimalt:

$$0$$

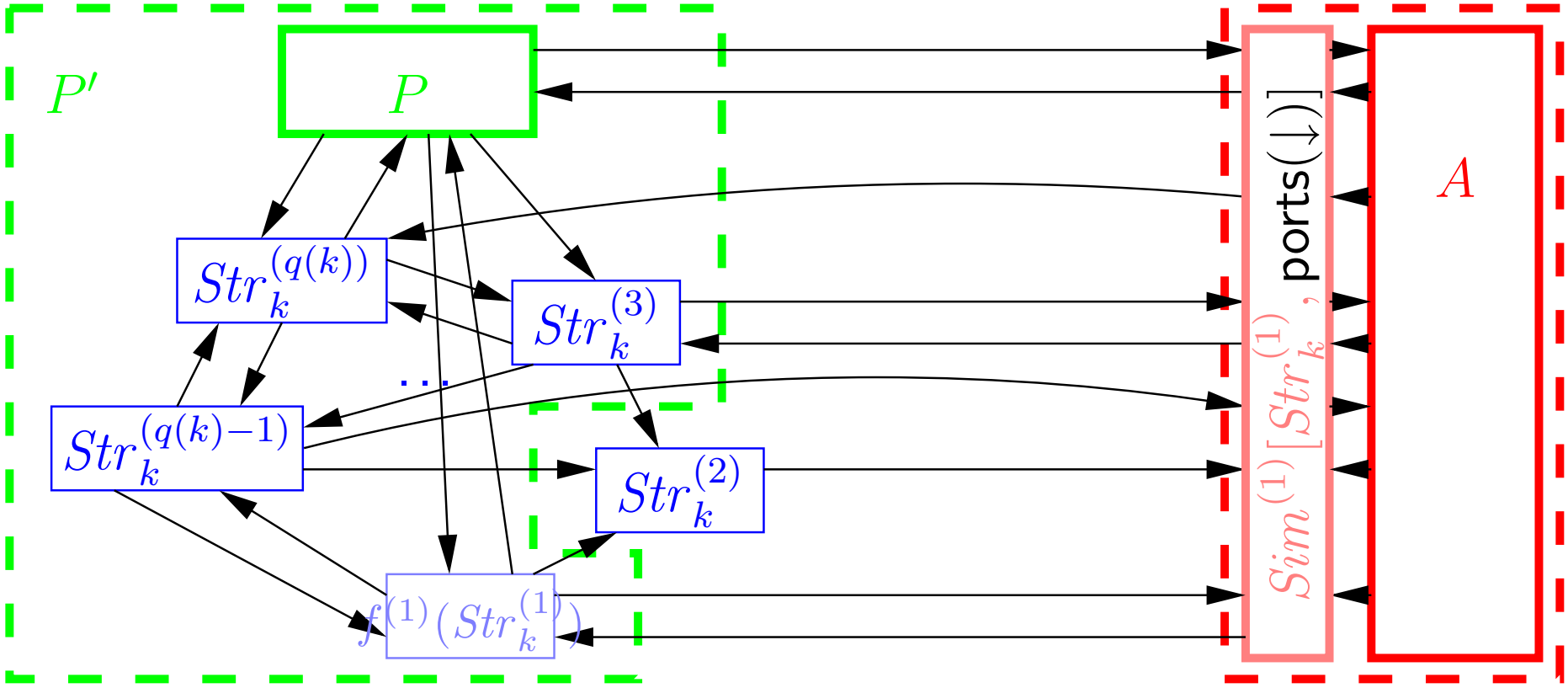
Tõestus



$\mathcal{A}$ eelis P vaate eristamisel tema esialgsest vaatest on ülimalt:

$$\delta_t(k)$$

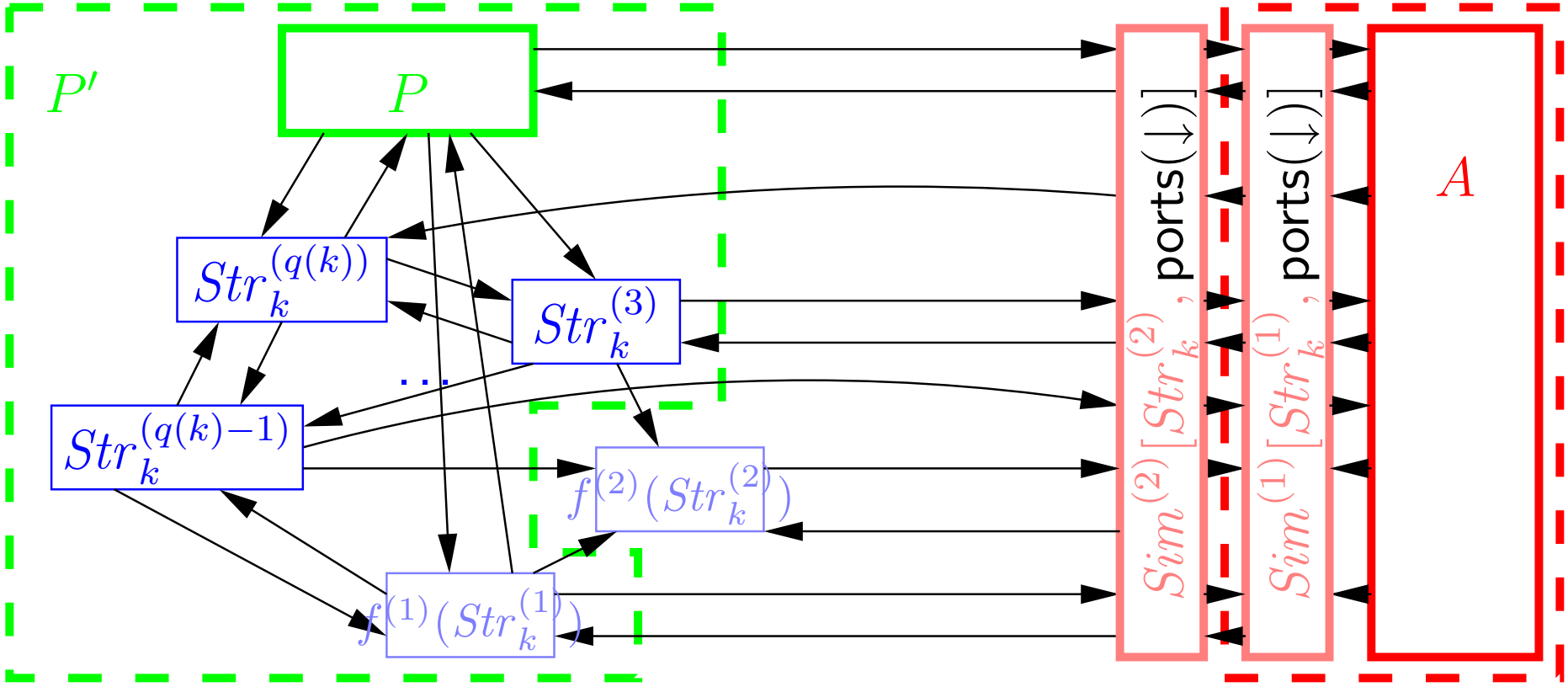
Tõestus



$\mathcal{A}$ eelis P vaate eristamisel tema esialgsest vaatest on ülimalt:

$$\delta_t(k)$$

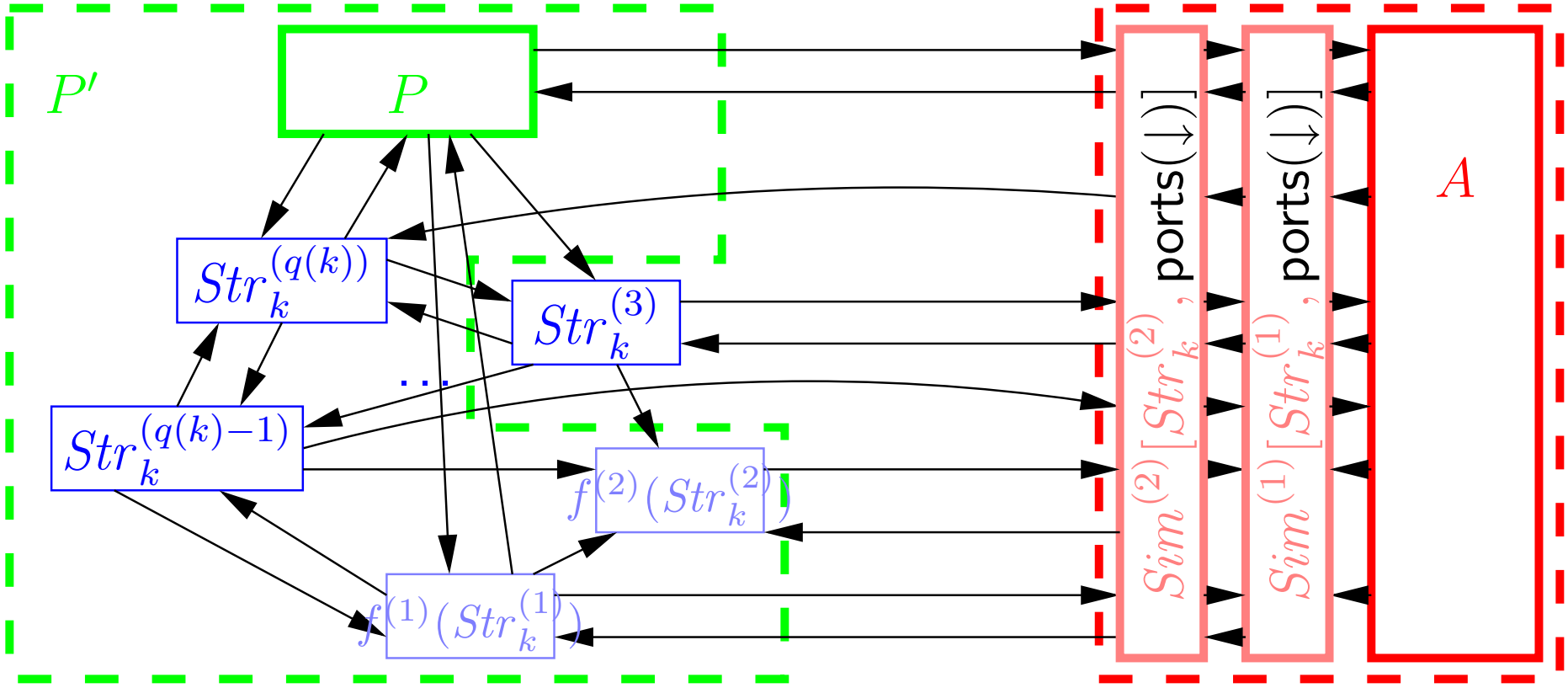
Tõestus



$\mathcal{A}$ eelis P vaate eristamisel tema esialgsest vaatest on ülimalt:

$$2\delta_t(k)$$

Tõestus



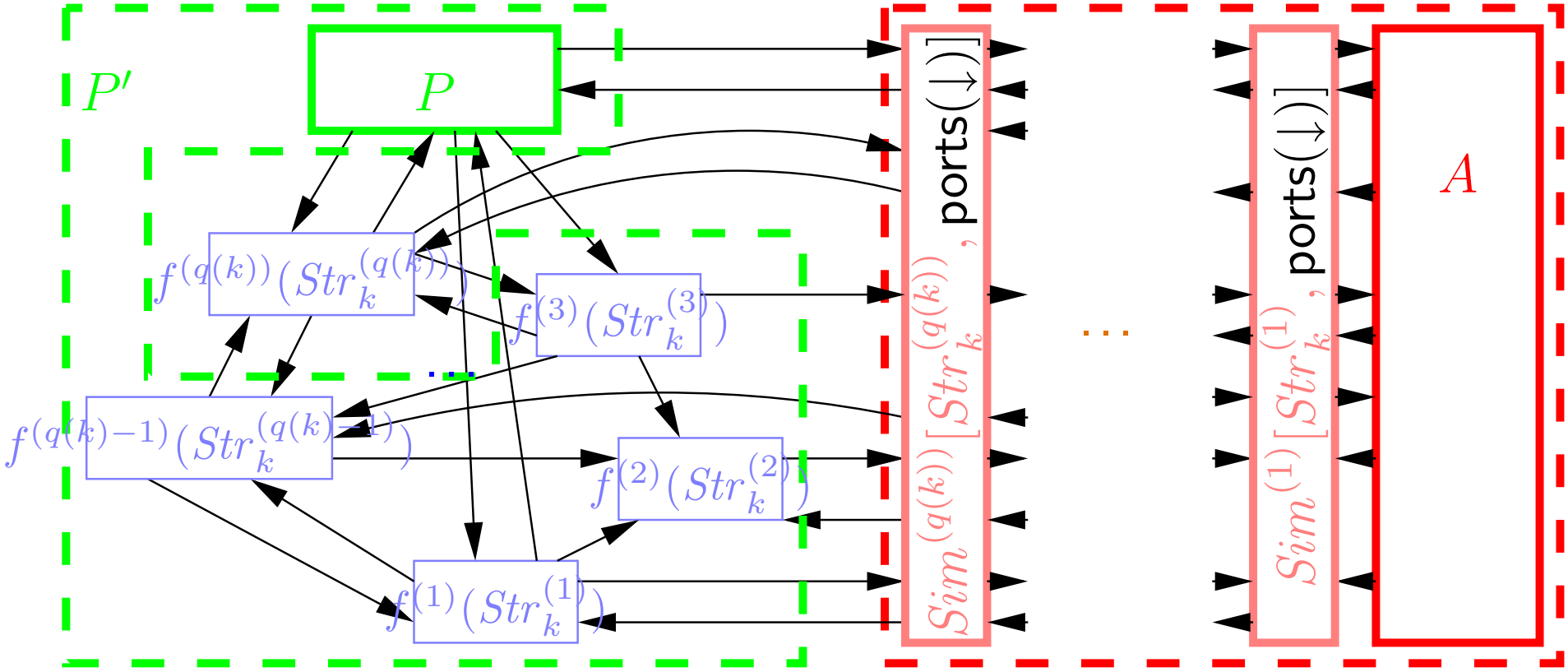
$\mathcal{A}$ eelis P vaate eristamisel tema esialgsest vaatest on ülimalt:

$$2\delta_t(k)$$

Tõestus

jne.

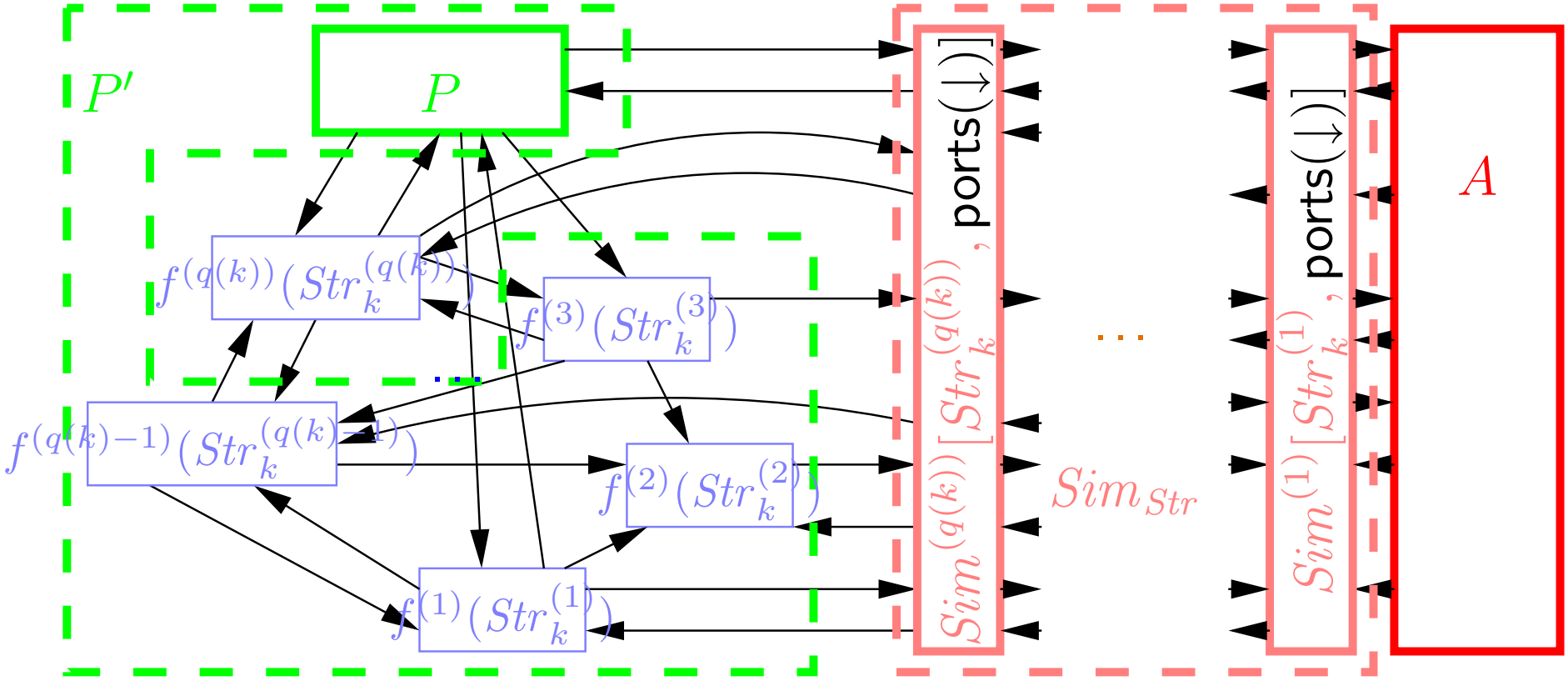
Tõestus



$\mathcal{A}$ eelis P vaate eristamisel tema esialgsest vaatest on ülimalt:

$$q(k)\delta_t(k)$$

Tõestus



$\mathcal{A}$ eelis P vaate eristamisel tema esialgsest vaatest on ülimalt:

$$q(k)\delta_t(k)$$

See on kaduvväike.