UNIVERZITET U NOVOM SADU
FAKULTET TEHNIČKIH NAUKA

Doriana Medić

FORMALNI MODEL ZA REVERZIBILNE KONKURENTNE KOMUNIKACIONE SISTEME

MASTER RAD

Mentor: Prof. dr Jovanka Pantović

Novi Sad, 2015.

Sadržaj

1 Uvod

Moderna vremena neminovno menjaju ljudske navike i ponašanja. Nekad je to posledica nastojanja da se kvalitet života poboljša uštedom vremena utrošenog na izvršavanje zadataka koje sa sobom nosi život u društvenoj zajednici, a nekad samo potreba za relaksacijom i ispunjavanjem priodnih čovekovih želja i potreba. Tako su konkurentni softverski sistemi postali sastavni i neizostavni deo života svakog čoveka. Od komunikacionih sistema, npr. bankomata, koji olakšavaju komunikaciju između klijenta i banke, preko mobilne telefonije, koja olakšava svakodnevnu komunikaciju, sve do društvenih mreža, u kojima pojedinci pre svega pronalaze trenutke relaksacije.

Sistemi kod kojih više procesa izvršava svoje akcije istovremeno i pri tom mogu, ali ne moraju, međusobno da komuniciraju nazivaju se *konkurentni sistemi*. Upravo ta mogućnost interakcije, razlikuje konkurentne od paralelnih sistema. Jedan od primera konkurentnih sistema jesu *baze podataka*. Struktura takvog sistema je da negde na serveru postoji organizovano skladište podataka kojima korisnici pristupaju pomoću *upita*. Konkurentnost ovom sistemu obezbeđuje da se on ne radi upit samo jednog korisnika, već da može da obrađuje više upita istovremeno i da pritom upiti mogu međusobno da komuniciraju. Ukoliko neki korisnik pravi izmene u svom upitu, ostalim korisnicima koji koriste konkurentne upite, prikazuje se stanje baze podataka pre promene, sve dok traje proces izmene. Nakon završetka procesa, korisnicima se prikazuje novo stanje podataka. Komunikacija između upita omogućava da se korisnicima u trenutku opisanog procesa ne prikazuju i stari i novi podaci.

Nastanak procesnog računa je zasnovan na težnji da se konstuiše formalni model koji bi opisao prirodu konkurentnih sistema i doprineo boljem razumevanju njihovog funkcionisanja. Sedamdesetih godina 20-og veka ovaj problem postaje predmet istraživanja i 1980. godine Robin Milner uvodi procesni račun CCS [9] (eng. Calculus of Concurrent Communicating Systems). Ovim modelom je omogućeno opisivanje komunikacije, interakcije i sinhronizacije između dva procesa. Kasnije je iz ovog modela razvijen formalni model mobilnih komunikacionih sistema, koji je nazvan π-račun [10, 12]. Kako su ovi modeli našli svoju primenu u analizi velikih softverskih komunikacionih sistema, prethodne decenije donele su niz računa dobijenih proširivanjem ovih inicijalnih modela, u skladu sa potrebama iz industrije. Možemo izdvojiti račun ambijenata [2], tipove sesija

[6], [7] itd.

Sa razvitkom računa CCS, ovaj formalni model, delimično ili u potpunosti, počinje da se upotrebljava i u drugim oblastima nauke. Tako, na primer, apstraktna hemijska mašina, koja je zasnovana na odredjenim osobinama molekula koji plutaju slobodno, opisana je jednim od podskupova računa CCS [8]. Proširivanjem računa CCS, a u želji da se njegove akcije kontrolišu, nastaje model kojim mogu da se opišu stvaranja i raskidanja molekularnih veza između proteina u ćeliji, pomoću kojih se prenosi signal sa površine ćelije do DNK u ćelijskom jedru [11].

Potreba da se proces iz jednog stanja vrati u prethodno stanje, dovodi do pojave reverzibilnosti. Ona predstavlja vraćanje procesa u tačno ono stanje u kojem se nalazio pre izvršavanja akcije koju želimo da poništimo. Kod sekvencijalnih procesa, jasno je definisano koji je poslednji korak izvrašavanja, zbog čega je reverzibilnost postignuto direktnim vraćanje u nazad poslednjeg izvršenog izračunavanja. Kod konkurentnih sistema, ne znamo koje je poslednje izvršeno izračunavanje. Ne možemo da utičemo na proces i jednostavno ga vratimo korak nazad, već moramo da obratimo pažnju i na sve njegove interakcije sa ostalim procesima prilikom izvršavanja date akcije. Ako bi zanemarili komunikaciju između procesa, on bi mogao da se vrati u stanje u kome nikad nije bio.
U svom radu iz 1961. godine, Rolf Landauer postavlja princip u kojem tvrdi da je fizička ireveraibilnost povezana sa fizičkom ireverzibilnošću, što dovodi do minimalnog utroška toplote i rasipanja energije. Ovaj princip je nedavno i eksperimentalo potvrđen [1]. Efikasnije korišćenje energije, kao i težnja za konstrukcijom što pouzdanijih hardverskih i softverskih sistema, svakako predstavlja jedan od osnovnih motiva za proučavanje reverzibilnih procesa. Drugi motiv svakako leži u činjenici da su mnogi prirodni sistemi, koje želimo formalnim metodama da proučavamo, sami po sebi reversibilnog karaktera.

Prvi formalni model za reverzibilne konkurentne komunikacione preces jeste račun RCCS [4] (eng. Reversible Concurrent Communicating Systerms). Račun RCCS je razvijen proširivanjem računa CCS, dodavanjem memorijske komponente. Kako struktura računa CCS dozvoljava svakom procesu da se razgrana ili sinhronizuje sa drugim procesom u bilo kom trenutku, zadatak memorije jeste da skladišti podatke svake od navedenih akcija i to tačno onim redom kojim su se izvršavale. Cilj ovog master rada na prvom mestu jeste da se opiše račun RCCS koji je uveden u radu [4]. Navedeni rad uvodi operacionu semantiku koristeći relaciju sa označenim prelascima. U ovom master radu, uvedena je i druga standardna operaciona semantika, pomoću relacije redukcije i dokazano je da su dve posmatrane semantike ekvivalentne.

Uvodno poglavlje daje kratak osvrt na motivaciju i trenutno stanje u oblasti, kao i sadržaj organizacije rada.

U drugom poglavlju dat je primer na kome su šematski i formalno prikazani osnovni pojmovi koje ćemo korisitit u radu: procesi, prefiksi, suma, paralelna kompozicija i restrikcija.

U prvom delu trećeg poglavlja, formalno uvodimo sintaksu računa CCS, definišemo slobodna i vezana imena, α-konverziju, procesni kontekst, standardnu formu i strukturnu kongruenciju za proste procese. Nakon toga pokazujemo da je svaki proces strukturno kongruentan sa nekim procesom u standardnoj formi. U drugom delu trećeg poglavlja uvodimo reverzibilnost, tako što CCS procesima pridružujemo memoriju. Zatim definišemo strukturnu kongruenciju na reverzibilnim procesima i osobine koje se prenose iz CCS-a i pokazujemo da je svaki reverzibilni proces kongruentan sa reverzibilnim procesom u standardnoj formi.

U četvrtom poglavlju, bavimo se operacionim semantikama za RCCS [5]. Uvodimo dve različite grupe pravila, LTS i Redukcionu semantiku. Kod LTS-a, pored standardnih pravila za CCS, uvodimo i tri dodatna koja će nam biti potrebna za dokaz glavnog tvrđenja. Zatim dokazujemo dve leme kojima želimo da nadoknadimo nedostatak strukturnog pravila u LTS-u. Nakon što uvedemo redukciona pravila, imamo sve što je potrebno da dokažemo glavno tvrđenje, tj. da ukoliko se proces redukuje na neki drugi proces, tada za isti taj proces postoji τ-prelaz u LTS-u, takav da je dobijeni proces kongruentan sa procesom nastalim redukcijom, i obratno.

U poslednjem poglavlju sumiramo prezentovani rad i dajemo zaključne komentare.

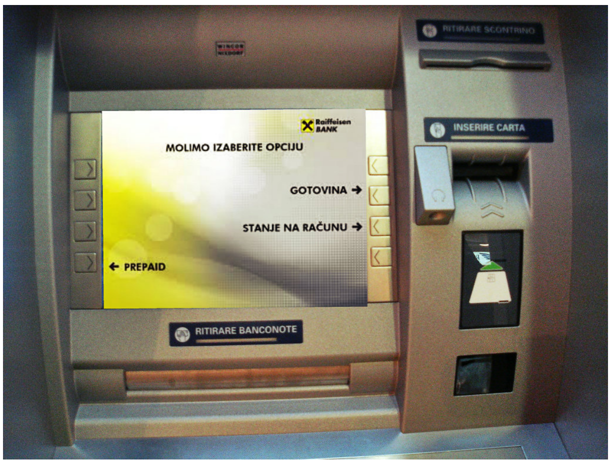
2 Primer

Procese u računu CCS možemo ilustrovati na primeru bankomata. U nastavku rada uvešćemo formalni model, u kojem u procesu $P = \alpha.Q$, α predstavlja prefiks, a Q potproces procesa P koji se izvšava sekvencijalno, nakon prefiksa. Tako proces P kojim klijent pristupa korišćenju bankomata, možemo da definišemo kao

$$P = kartica.lozinka.0$$

Vidimo da se nakon izvršavanja akcije $kartica$, kada ubacimo karticu u bankomat, aktivira proces $lozinka$, tj. bankomat nam daje povratnu informaciju u kojoj traži da unesemo lozinku.

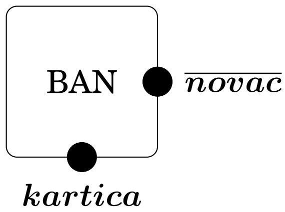
Primer procesa koji nam pruža mogućnost da izaberemo da se izvrši jedan od dva njegova potprocesa, možemo opisati na sledeći način: kada ubacimo karticu u bankomat, on nam kao povratnu informaciju daje mogućnost izbora, da li želimo da podignemo novac ili da proverimo stanje na računu. Korisnik bira jednu od te dve akcije. Podizanje novca označićemo akcijom $\overline{gotovina}$, proveru stanja akcijom $\overline{stanje}$, a proces, tj. bankomat koji nam to omogućava sa BANK. Na slici ispod je prikazan bankomat koji predstavlja dati proces.



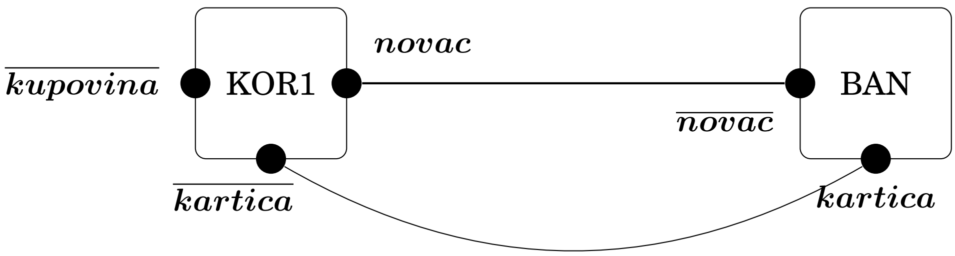
Formalno, proces koji predstavlja ovu konstrukciju zapisujemo na sledeći način

$$kartica.(\overline{gotovina}.BANK + \overline{stanje}.BANK)$$

Na sledećoj slici, prikazan je proces BAN, koji može da intereaguje sa sredinom preko dva kanala, čija su imena $kartica$ i $\overline{novac}$. Razlikujemo imena koja proces šelje i koja prima. Imena koja se šalju označavaćemo crtom iznad imena.

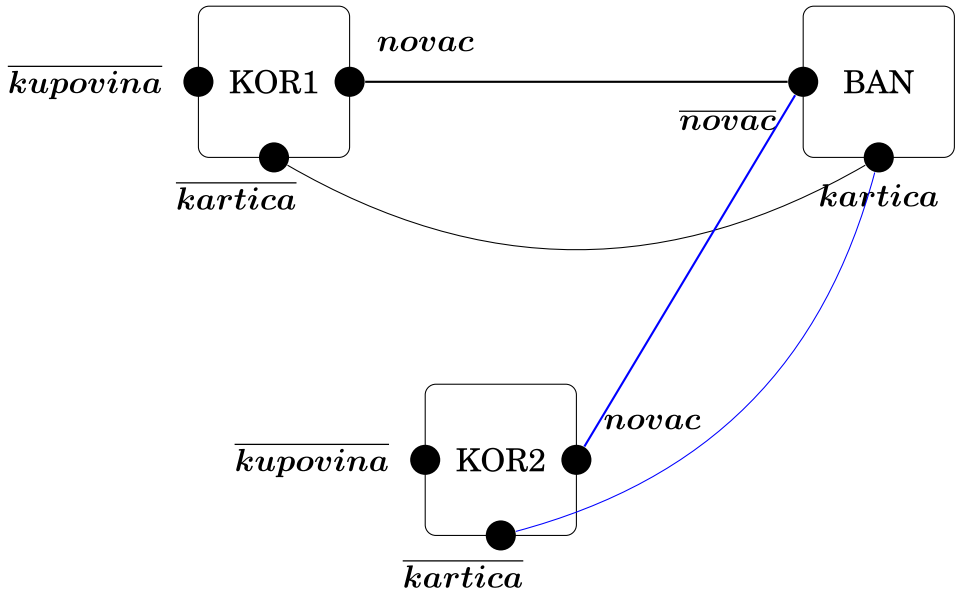


Kada korisnik upotrebljava bankomat, tada korisnika možemo definisati kao proces $KOR1$ sa kanalima $novac$, $\overline{kartica}$ i $\overline{kupovina}$. Primećujemo da će novi proces imati tri kanala pomoću kojih može da interreaguje sa sredinom. Ovaj sistem je tada konkurentni sistem komunikacije, jer se procesi izvršavaju paralelno i imaju mogućnost da komuniciraju jedan sa drugim preko kanala $novac$ i $kartica$. Na slici ispod dat je šematski prikaz opisanog sistema.



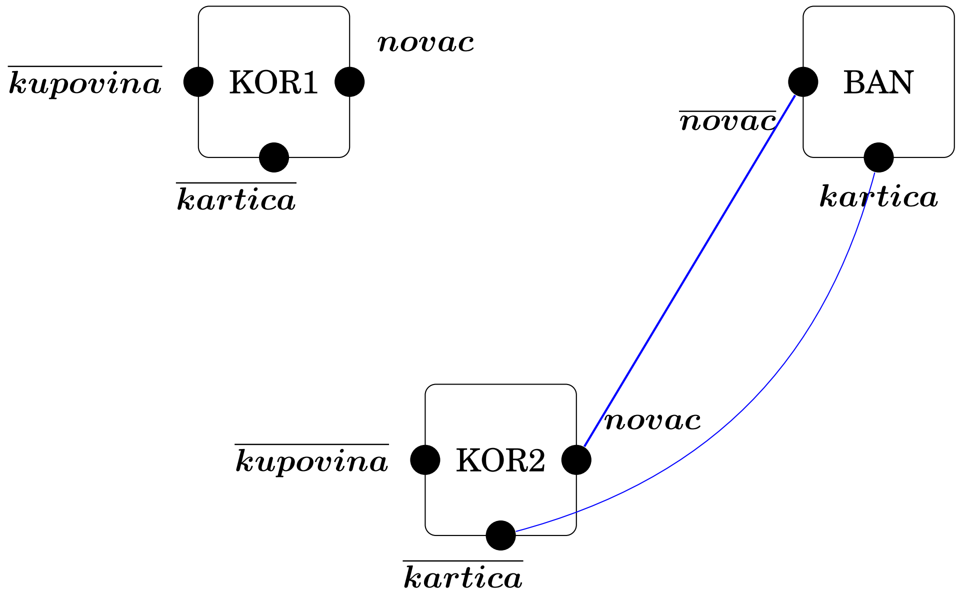
Formalni zapis ovog sistema je $KOR1 \mid BAN$.

Sistem može da se sastoji od dva korisnika i bankomata, pri čemu oba korisnika žele da podignu novac i odu u kupovinu. Tada uvodimo još jedan proces koji nazivamo $KOR2$ koji će sadržati iste kanale kao i proces $KOR1$. Sada procesi $KOR1$ i $KOR2$ mogu da komuniciraju sa procesom BAN preko kanala $novac$ i $kartica$, ali ne mogu da komuniciraju međusobno. Šematski prikaz dat je na sledećoj slici



Formalni zapis sistema bi bio $KOR1 \mid BAN \mid KOR2$.

Ako je potrebno da obezbedimo da samo drugi korisnik može da upotrebi bankomat, tada kanale $novac$ i $kartica$ treba da ograničimo samo na ova dva procesa. Ostali procesi ne mogu da komuniciraju putem ovih kanala. U tom slučaju, šematski prikaz je oblika



Ograničavanje kanala na date procese definišemo restrikcijom i dobijamo formalni zapis $(kartica)(novac)(KOR2 \mid BAN) \mid KOR1$.

3 Sintaksa

U ovom poglavlju definisaćemo jezik reverzibilnih procesa koji je uveden u [4], a na koji ćemo se oslanjati u nastavku rada. Kako je račun RCCS dobijen proširivanjem računa CCS, prvo uvodimo sintaksu računa CCS.

3.1 Račun CCS

Neka je $\mathcal{N}$ skup svih imena i $\overline{\mathcal{N}}$ skup svih komplementarnih imena kanala. Ova dva skupa su disjunktna, a njihovu uniju $\mathcal{N} \cup \overline{\mathcal{N}}$ označavamo sa $\mathcal{L}$. Definišemo da je $\overline{\overline{a}} = a$. Akcije označavamo sa α i definišemo ih nad skupom $\mathcal{L} \cup \tau$, gde je τ unutrašnja akcija koja se dešava unutar procesa, bez interakcije sa okolinom. Skup svih akcija na kanalima, označavaćemo sa $\mathcal{A}$.

Definicija 3.1 (prosti procesi)**.** *Prosti procesi u računu CCS, definisani su na sledeći način:*

Akcije	α	$::=$	$a, b, c, ..$	*(imena kanala)*
		$\mid$	$\bar{a}, \bar{b}, \bar{c}, ..$	*(komplementarna imena kanala)*
		$\mid$	τ	*(unutrašnja akcija)*
Procesi	P	$::=$	M	*(suma)*
		$\mid$	$(a)P$	*(restrikcija)*
		$\mid$	$P \mid P$	*(paralelna kompozicija)*
Suma	M	$::=$	0	*(neaktivan proces)*
		$\mid$	$\alpha.P$	*(prefiks)*
		$\mid$	$M + M$	*(binarna suma)*

Proste procese ili procese bez memorije, označavaćemo sa P,Q,$T, M, \ldots$

Prosti procesi obuhvataju sledeće procese:

(1) Proces 0 je neaktivan proces, tj proces koji ništa ne radi. Kako je neaktivan proces uvek na kraju sekvencijalne kompozicije, često ga izostavljamo iz zapisa i podrazumevamo ga. Na primer, proces $a.b.0$ možemo zapisati kao $a.b.$

(2) Proces $\alpha.P$ obuhvata tri procesa: $a.P, \overline{a}.P$ i $\tau.P$:

- *Pozitivan prefiks* $a.P$ i *negativan prefiks* $\overline{a}.Q$ u paralelnoj kompoziciji omogućuju međusobnu interakciju.
- *Unutrašnja akcija* τ izvršava „tihu" (neprimetnu) akciju i nastavlja da se ponaša kao P.

(3) Proces $(a)P$ (restrikcija) se ponaša kao P, ali je domen pojavljivanja imena a ograničen na dati proces P.

(4) Proces $P \mid Q$ je paralelna kompozicija procesa P i Q. Konkurentni procesi mogu da se izvršavaju istovremeno, nezavisno jedan od drugog, ali postoji i mogućnost njihove medjusobne interakcije.

(5) Proces $P + Q$ se ponaša kao P ili Q. Lako se zaključuje da je suma M oblika $\sum_{i \in I} \alpha_i.P_i$, za $I \neq \emptyset$, ili je 0.

Primer. Neka su dati izrazi

$$(i)\ \overline{a}.0 + a.0 \qquad (ii)\ a.\overline{b}.0 \mid b.\overline{a}.0 \qquad (iii)\ 0.a.0.$$

Prema Definiciji 3.1, (i) i (ii) su prosti procesi, dok (iii) ne može da se konstruiše na osnovu sintakse prostih procesa.

U procesu $(a)P$, kažemo da je pojavljivanje imena a vezano. Ukoliko ime nije vezano u nekom procesu, onda je ono slobodno. Slobodna i vezana pojavljivanja imena u procesima formalno definišemo kao funkcije koje procesima dodeljuju skupove imena.

Definicija 3.2 (slobodna imena)**.** *Neka je funkcija fn prostih procesa u skupove imena definisana sa:*

$$\begin{array}{lcl} fn(0) & = & \emptyset \\ fn(a.P) & = & \{a\} \cup fn(P) \\ fn(\overline{a}.P) & = & \{a\} \cup fn(P) \\ fn((a)P) & = & fn(P) \setminus \{a\} \\ fn(P_1 \mid P_2) & = & fn(P_1) \cup fn(P_2) \\ fn(P_1 + P_2) & = & fn(P_1) \cup fn(P_2) \end{array}$$

Kažemo da je $fn(P)$ skup svih slobodnih imena u prostom procesu P.

Definicija 3.3 (vezana imena). *Neka je funkcija bn prostih procesa u skupove imena definisana sa:*

$$\begin{array}{lcl} bn(0) & = & \emptyset \\ bn(\alpha.P) & = & bn(P) \\ bn((a)P) & = & bn(P) \cup \{a\} \\ bn(P_1 \mid P_2) & = & bn(P_1) \cup bn(P_2) \\ bn(P_1 + P_2) & = & bn(P_1) \cup bn(P_2) \end{array}$$

Kažemo da je $bn(P)$ skup svih vezanih imena u prostom procesu P.

Skup svih imena koja se pojavljuju u procesu P, u oznaci $n(P)$, jeste unija skupa slobodnih i skupa vezanih imena u procesu P.

Primer. Neka je dat proces

$$P = (a)(a.b.0 \mid \bar{a}.c.0)$$

Tada se skupovi slobodnih i vezanih imena dobijaju na sledeći način:

$$\begin{array}{rcl} fn(P) & = & fn(a.b.0 \mid \bar{a}.c.0)\backslash\{a\} \\ & = & \{fn(a.b.0) \cup fn(\bar{a}.c.0)\}\backslash\{a\} \\ & = & \{\{a\} \cup fn(b.0) \cup \{a\} \cup fn(c.0)\}\backslash\{a\} \\ & = & \{\{a\} \cup \{b\} \cup fn(0) \cup \{c\} \cup fn(0)\}\backslash\{a\} \\ & = & \{\{a\} \cup \{b\} \cup \emptyset \cup \{c\} \cup \emptyset\}\backslash\{a\} \\ & = & \{\{a\} \cup \{b\} \cup \{c\}\}\backslash\{a\} \\ & = & \{b, c\} \end{array}$$

Vidimo da je skup svih slobodnih imena u datom prostom procesu $fn(P) = \{b, c\}$, dok je skup svih vezanih imena $bn(P) = \{a\}$.

Supstitucija σ je unarna operacija nad skupom imena koja je identičko preslikavanje, osim na jednom njegovom konačnom podskupu.

Definicija 3.4 (supstitucija). *Neka je k prirodan broj. Za preslikavanje $\sigma : \mathcal{N} \to \mathcal{N}$ definisano sa*

$$\sigma(x) = \begin{cases} b_j & \textit{ako je } x = a_j \textit{ i } j \in \{1, \ldots, k\} \\ x & \textit{inače} \end{cases}$$

kažemo da je supstitucija (zamena) imena $a_1, \ldots, a_k$ redom imenima $b_1, \ldots, b_k$. Kraće, pišemo $\{b_1, \ldots, b_k/a_1, \ldots, a_k\}$. Kažemo da su imena supstitucije $n(\sigma) = \{b_1, \ldots, b_k, a_1, \ldots, a_k\}$.

Definicija 3.5 (α-konverzija)**.** *Promena vezanog imena u procesu P je zamena potprocesa $(a)T$ iz P sa $(c)T\{c/a\}$, gde se ime c ne pojavljuje u procesu T.*
Procesi P i Q su α-konvertibilni ($P \equiv_\alpha Q$) ako se Q može dobiti od P promenom konačnog broja vezanih imena novim imenima.

Primer. Primenom α-konverzije zaključujemo

$$(a)b.a.\bar{c}.0 \equiv_\alpha (d)b.d.\bar{c}.0 \qquad\qquad (a)b.a.\bar{c}.0 \not\equiv_\alpha (a)b.a.\bar{a}.0.$$

Po dogovoru, smatraćemo da su u jednom procesu sva vezana imena različita, da se razlikuju od slobodnih imena, kao i od konačnog skupa imena supstitucije. Znači, u procesu $(a)P$, za supstituciju σ, važi $a \notin fn(P) \cup n(\sigma)$.

Definicija 3.6. *Primena supstitucije σ na proces P, u oznaci $P\sigma$, definisana je na sledeći način:*

$$\begin{aligned}
0\sigma &= 0\\
(a.P)\sigma &= a\sigma.P\sigma\\
(\bar{a}.P)\sigma &= \overline{a\sigma}.P\sigma\\
(\tau.P)\sigma &= \tau.P\sigma\\
((a)P)\sigma &= (a)P\sigma\\
(P_1 \mid P_2)\sigma &= P_1\sigma \mid P_2\sigma\\
(P_1 + P_2)\sigma &= P_1\sigma + P_2\sigma
\end{aligned}$$

Na primer, ako u procesu P želimo da pojavljivanja slobodnog imena a zamenimo sa imenom b, to zapisujemo kao $P\{b/a\}$.

Primer. Ako je dat proces

$$P = a.b.0 \mid a.\bar{b}.0,$$

i želimo da ime a zamenimo sa imenom c, primenom supstitucije $\{c/a\}$ na proces P dobijamo proces

$$P\{c/a\} = c.b.0 \mid c.\bar{b}.0.$$

Ponašanje nekog procesa tokom izvršavanja zavisi od procesnog okruženja u kojem sa nalazi tj. od konteksta u koji ga stavimo.

Definicija 3.7 (procesni kontekst)**.** *Procesni kontekst $C[\,]$ je definisan na sledeći način:*

$$\begin{array}{rcl} C[\,] & ::= & [\,] \\ & | & \alpha.C[\,] + M \\ & | & (a)C[\,] \\ & | & C[\,] \mid P \\ & | & P|C[\,] \end{array}$$

Sa $C[Q]$ označavamo proces dobijen zamenom praznine $[\,]$ u kontekstu $C[\,]$ procesom Q.

Primer. Za procesni kontekst

$$C[\,] = (a)(a.\bar{b}.0 \mid [\,])$$

važi

$$C[b.\bar{a}.0] = (a)(a.\bar{b}.0 \mid b.\bar{a}.0).$$

Ako je kontekst dobijen u jednom koraku, primenom operacija procesna algebre na procese i prazninu, bez rekurzivnog pozivanja definicije, onda kažemo da je kontekst elementaran.

Definicija 3.8 (elementarni kontekst)**.** *Elementarni kontekst $E[\,]$ definišemo na sledeći način:*

$$\begin{array}{rcl} E[\,] & ::= & [\,] \\ & | & \alpha.[\,] + M \\ & | & (a)[\,] \\ & | & [\,] \mid P \\ & | & P \mid [\,] \end{array}$$

Iz prethodnih definicija direktno sledi da je elementarni kontekst procesni kontekst.

Primer. Neka je dat elementarni kontekst $E_1[\,] = a.\bar{b}.0 \mid [\,]$ i procesi

$$P_1 = \bar{a}.b.0 \qquad \text{i} \qquad P_2 = b.\bar{a}.0.$$

Primenom konteksta E_1 na procese P_1 i P_2, dobijamo nove procese

$$P_1' = E_1[\bar{a}.b.0] = a.\bar{b}.0 \mid \bar{a}.b.0 \text{ i } P_2' = E_1[b.\bar{a}.0] = a.\bar{b}.0 \mid b.\bar{a}.0.$$

Konstrukcija novonastalog procesa P_1' je takva da će njegovi potprocesi moći dalje da interreaguju, dok se konstrukcija procesa P_2' naziva deadlock (procesi čekaju jedan drugog da se izvrše).

Definicija 3.9 (relacija ekvivalencije)**.** *Relacija ekvivalencije $\cong$ nad skupom prostih procesa je binarna relacija koja zadovoljava sledeće osobine:*

(R) refleksivnost ($P \cong P$),

(S) *simetričnost (ako je $P \cong Q$ tada je i $Q \cong P$) i*

(T) *tranzitivnost (ako je $P \cong Q$ i $Q \cong T$, tada je i $P \cong T$).*

Definicija 3.10 (kongruencija)**.** *Za relaciju ekvivalencije kažemo da je kongruencija na prostim procesima ako je očuvana elementarnim kontekstima, tj. ako iz $P \cong Q$ sledi:*

$$\begin{gathered} \alpha.P + M \cong \alpha.Q + M \\ (a)P \cong (a)Q \\ P \mid T \cong Q \mid T \\ T \mid P \cong T \mid Q. \end{gathered}$$

Tvrđenje 3.1. *Relacija ekvivalencije $\cong$ je kongruencija na prostim procesima ako i samo ako važi da ako je $P \cong Q$, tada je i $C[P] \cong C[Q]$, za sve kontekste C.*

Dokaz. Tvrđenje dokazujemo u dva smera:

($\Rightarrow$) Pretpostavimo da je relacija ekvivalencije $\cong$ procesna kongruencija. Tada važe pravila data u Definiciji 3.10. Pokazaćemo da važi implikacija

$$P \cong Q \Rightarrow C[P] \cong C[Q].$$

Pretpostavimo da je $P \cong Q$. Dokazaćemo indukcijom po složenosti konteksta $C[\,]$ da je $C[P] \cong C[Q]$.

(i) Ako je $C[\,] = [\,]$ onda je $C[P] = P \cong Q = C[Q]$.

(ii) Pretpostavimo da je $C_1[P] \cong C_1[Q]$ za neki kontekst $C_1[\,]$.

(iii) Prema Definiciji 3.10 i induktivnoj pretpostavci važi:

(1) ako je $C[\,] = \alpha.C_1[\,] + M$, onda je $C[P] = \alpha.C_1[P] + M \cong \alpha.C_1[Q] + M = C[Q]$;

(2) ako je $C[\,] = (a)C_1[\,]$, onda je $C[P] = (a)C_1[P] \cong (a)C_1[Q] = C[Q]$;

(3) ako je $C[\,] = C_1[\,] \mid T$, onda je $C[P] = C_1[P] \mid T \cong C_1[Q] \mid T = C[Q]$;

(4) ako je $C[\,] = T \mid C_1[\,]$, onda je $C[P] = T \mid C_1[P] \mid T \cong C_1[Q] = C[Q]$.

($\Leftarrow$) Pretpostavimo da za svaki kontekst $C[\,]$ važi implikacija $P \cong Q \Rightarrow C[P] \cong C[Q]$. Treba da pokažemo da je tada $\cong$ procesna kongruencija, tj. da važi implikacija $P \cong Q \Rightarrow E[P] \cong E[Q]$. Posmatrana implikacija je tačna zato što elementarni kontekst $E[\,]$ jeste procesni kontekst. ∎

Definicija 3.11 (strukturna kongruencija)**.** *Strukturna kongruencija $\equiv$ je kongruencija na prostim procesima koja zadovoljava sledeće osobine:*

- *ako je* $P \equiv_\alpha Q$ *onda važi* $P \equiv Q$
- *skup svih procesa u odnosu na sabiranje je komutativni monoid:*

$$\begin{aligned} P+Q &\equiv Q+P \\ (P+Q)+T &\equiv P+(Q+T) \\ P+0 &\equiv P \end{aligned}$$

- *skup svih procesa u odnosu na paralelnu kompoziciju je komutativni monoid:*

$$\begin{aligned} P \mid Q &\equiv Q \mid P \\ (P \mid Q) \mid T &\equiv P \mid (Q \mid T) \\ P \mid 0 &\equiv P \end{aligned}$$

- *važe pravila za restrikciju:*

$$\begin{aligned} (a)(P \mid Q) &\equiv P \mid (a)Q, \textit{ ako } a \notin fn(P) \\ (a)0 &\equiv 0 \\ (a)(b)P &\equiv (b)(a)P. \end{aligned}$$

Definicija 3.12 (standardna forma)**.** *Za proces* 0 *i proces* $(\vec{a})(\alpha_1.P_1+M_1 \mid ... \mid \alpha_n.P_n+M_n)$ *kažemo da su u standardnoj formi, gde je* $\vec{a}$ *prazan niz ili* $(a_1)(a_2)\dots(a_k)$, *za neko* $k \geq 1$.

Svaki proces je strukturno kongruentan sa nekim procesom u standardnoj formi.

Tvrđenje 3.2. *Ako je* P *prost proces, onda postoji proces* Q *u standardoj formi sa osobinom* $P \equiv Q$.

Dokaz. Tvrđenje dokazujemo indukcijom po složenosti procesa P.

(i) Prvi korak indukcije, ili baza je kada je proces P oblika:
$P = 0$ tada iz definicije sledi da je on već u standardnoj formi.

(ii) (Induktivna pretpostavka) Pretpostavimo da tvrđenje važi za procese složenosti k.

(iii) Dokazujemo da tvrđenje važi za procese čija je složenost $k+1$. Proces P onda može da bude oblika:

(1) $P = \alpha_1.P_1' + M_1' \mid \dots \mid \alpha_n.P_n' + M_n'$, gde je proces u standardnoj formi u kojoj nema restrikcije na najvišem nivou.

(2) $P = (a)T$, gde za proces T važi induktivna pretpostavka i on je u standardnoj formi

$$T \equiv (\vec{a})(\alpha_1.T_1' + M_1' \mid ... \mid \alpha_n.T_n' + M_n').$$

Tada je

$$P = (a)(\vec{a})(\alpha_1.T_1' + M_1' \mid \dots \mid \alpha_n.T_n' + M_n')$$

u standardnoj formi direktno.

(3) $P = Q' \mid T$, gde za procese Q' i T važi induktivna pretpostavka i oni su u standardnoj formi:

$$\begin{aligned} Q' &\equiv (\vec{a'})(\alpha'_1.Q''_1 + M''_1 \mid ... \mid \alpha'_n.Q''_n + M''_n) \\ T &\equiv (\vec{a})(\alpha_1.T'_1 + M'_1 \mid ... \mid \alpha_n.T'_n + M'_n) \end{aligned}$$

Tada sledi:

$$P \equiv (\vec{a'})(\alpha'_1.Q''_1 + M''_1 \mid \ldots \mid \alpha'_n.Q''_n + M''_n) \mid (\vec{a})(\alpha_1.T'_1 + M'_1 \mid ... \mid \alpha_n.T'_n + M'_n).$$

Na osnovu pravila $(a)(P \mid Q) \equiv P \mid (a)Q$ i α-konverzije, dobijamo

$$P \equiv (\vec{a'})(\vec{a})(\alpha'_1.Q''_1 + M''_1 \mid ... \mid \alpha'_n.Q''_n + M''_n \mid \alpha_1.T'_1 + M'_1 \mid ... \mid \alpha_n.T'_n + M'_n).$$

■

3.2 Račun RCCS

Reverzibilni procesi se dobijaju od prostih procesa, dodeljivanjem svakom procesu njegove individualne memorije koja čuva podatke iz prošlosti. Memorija skladišti podatke o izvršenim akcijama i time omogućuje povratak u prethodna stanja procesa.

Definicija 3.13 (reverzibilni procesi). *Reverzibilni procesi u računu RCCS su definisani na sledeći način:*

Memorije	m	$::=$	$\langle\rangle$	*(prazna memorija)*
		$\mid$	$\langle 1\rangle \cdot m$	*(memorija leve strane)*
		$\mid$	$\langle 2\rangle \cdot m$	*(memorija desne strane)*
		$\mid$	$\langle *, \alpha, P\rangle \cdot m$	*(polusinhronizovana memorija)*
		$\mid$	$\langle m, \alpha, P\rangle \cdot m$	*(sinhronizovana memorija)*
(Reverzibilni) procesi	R	$::=$	$m \triangleright P$	*(procesi sa memorijom)*
		$\mid$	$R \mid R$	*(paralelna kompozicija)*
		$\mid$	$(a)R$	*(restrikcija)*

Reverzibilne procese ćemo označavati sa $R, S, \ldots$

Reverzibilni procesi obuhvataju sledeće procese:

(1) Proces $m \triangleright P$, u kojem m predstavlja memoriju u kojoj se skladište podaci koji su nam potrebni ukoliko želimo da vratimo izvršavanje unazad. Informacije se u memoriji slažu tako da se poslednja akcija koju smo izvršili nalazi na vrhu memorije.

(2) Proces $R_1 \mid R_2$ predstavlja paralelnu kompoziciju reverzibilnih procesa R_1 i R_2.

(3) Proces $(a)R$ ograničava domen pojavljivanja imena a na reverzibilni proces R.

Za reverzibilne procese sa praznom memorijom, u kojima su sva pojavljivanja imena u prostom procesu vezana, kažemo da su zatvoreni.
Ako je dat reverzibilni proces R i memorije m_1 i m_2, zamenjivanjem svih memorija u R oblika $\langle *, \alpha, Q\rangle \cdot m_1$ sa $\langle m_2, \alpha, Q\rangle \cdot m_1$, dobijamo novi reverzibilni proces koji ćemo označavati sa $R_{m_2 @ m_1}$.

Definicija 3.14. *Neka je R reverzibilni proces i m_1, m_2 memorije. Reverzibilni proces $R_{m_2 @ m_1}$ definisan je na sledeći način:*

$$R_{m_2@m_1} = \begin{cases} (a)R'_{m_2@m_1} & \textit{ako je } R = (a)R' \\ R_{m_2@m_1} \mid S_{m_2@m_1} & \textit{ako je } R = R \mid S \\ \langle\rangle \triangleright P & \textit{ako je } R = \langle\rangle \triangleright P \\ \langle i\rangle \cdot m \triangleright P & \textit{ako je } R = \langle i\rangle \cdot m \triangleright P \textit{ i } i \in \{1,2\} \\ \langle m', \alpha, Q\rangle \cdot m \triangleright P & \textit{ako je } R = \langle m', \alpha, Q\rangle \cdot m \triangleright P \\ \langle *, \alpha, Q\rangle \cdot m \triangleright P & \textit{ako je } R = \langle *, \alpha, Q\rangle \cdot m \triangleright P \textit{ i } m \neq m_1 \\ \langle m_2, \alpha, Q\rangle \cdot m_1 \triangleright P & \textit{ako je } R = \langle *, \alpha, Q\rangle \cdot m_1 \triangleright P. \end{cases}$$

Svakom reverzibilnom procesu pridružena je jedinstvena memorija, koja ujedno služi i za imenovanje procesa. Za adresu memorije uzimamo reč nad azbukom $\{1, 2, \varepsilon\}$.

Definicija 3.15 (adresa memorije). *Adresa memorije λ je funkcija koja svakoj memoriji m dodeljuje niz slova azbuke $\{1, 2, \varepsilon\}$ na sledeći način:*

$$\lambda(m) = \begin{cases} \varepsilon & \textit{ako je } m = \langle\rangle, \\ \lambda(m') & \textit{ako je } m = \langle *, \alpha, P\rangle \cdot m' \\ \lambda(m') & \textit{ako je } m = \langle m, \alpha, P\rangle \cdot m' \\ i \cdot \lambda(m') & \textit{ako je } m = \langle i\rangle \cdot m' \textit{ i } i \in \{1,2\} \end{cases}$$

Primer. $\lambda(\langle *, \alpha, P\rangle m \cdot \langle 1\rangle\langle 2\rangle\langle\rangle) = 12\varepsilon$.

Kako želimo da memorija bude jedinstvena, uvodimo i pojam koherencije. Da bismo mogli da definišemo koherentnost na procesima, prvo uvodimo pojam koherentne memorije.

Definicija 3.16 (koherentnost). *Koherentnost, u oznaci $\frown$, je najmanja simerična relacija na memorijama sa osobinom:*

(i) $\langle 1\rangle \cdot m \frown \langle 2\rangle \cdot m$ *i*

(ii) $\forall m_1, m_2 : m \frown m' \Rightarrow m_1 \cdot m \frown m_2 \cdot m'$.

Znači da su grane (pog granama ćemo podrazumevati levu i desnu stranu memorije), koje su direktni potomci neke memorije, koherentne. Ako su dve memorije koherentne, onda su svi njihovi potomci po parovima koherentni.

Primer.

(i) $\langle 1\rangle\langle\ \rangle \frown \langle 2\rangle\langle\ \rangle$;

(ii) $\langle 1\rangle\langle 2\rangle\langle 1\rangle\langle\ \rangle \frown \langle 2\rangle\langle 1\rangle\langle 1\rangle\langle\ \rangle$;

(iii) $\langle\ \rangle \not\frown \langle\ \rangle$;

(iv) $\langle *, \alpha, Q\rangle \cdot \langle\ \rangle \not\frown \langle\ \rangle$.

Reverzibilni proces R je koherentan ako su mu memorije po parovima koherentne (ako imamo paralelan proces, memorije od potprocesa moraju biti koherentne).
Osobina koherentnosti je očuvana prelazima i stukturnom kongruencijom.

Primer. Sledeći proces je koherentan

$$\langle 1\rangle\langle 1\rangle\langle 2\rangle\langle 1\rangle\langle\ \rangle \triangleright P \mid \langle 2\rangle\langle 2\rangle\langle 1\rangle\langle 1\rangle\langle\ \rangle \triangleright Q.$$

U ovom radu bavićemo se reverzibilnim procesima koji su koherentni.

Definicija 3.17. *Strukturna kongruencija $\equiv_R$ na reverzibilnim procese je najmanja relacija kongruencije koja zadovoljava naredna pravila:*

1. $R_1|R_2 \equiv_R R_2|R_2$

$R_1|(R_2|R_3) \equiv_R (R_1|R_2)|R_3$

$(a)(b)R \equiv_R (b)(a)R$

2. $m \triangleright P + Q \equiv_R m \triangleright Q + P$

$m \triangleright (P + Q) + T \equiv_R m \triangleright P + (Q + T)$

$m \triangleright P + 0 \equiv_R m \triangleright P$

3. $m \triangleright (P \mid Q) \equiv_R (\langle 1\rangle \cdot m \triangleright P \mid \langle 2\rangle \cdot m \triangleright Q)$

4. $m \triangleright (a)(P \mid Q) \equiv_R m \triangleright P \mid (a)Q$ ***ako*** $a \notin fn(P)$,

$m \triangleright (a)P \equiv_R (a)m \triangleright P$

U prvoj grupi pravila predstavljene su osobine koje važe za reverzibilne procese. U drugoj grupi imamo osobine koje važe za proste procese, unutar reverzibilnog procesa. Trećim pravilom pokazujemo da ukoliko reverzibilni proces koji sadrži paralelnu kompoziciju prostih procesa podelimo na dva procesa, svaki od njih nasleđuje osnovnu memoriju zajedno sa brojem koji označava sa koje strane paralelne kompozicije je dati proces. Četvrtom grupom pravila pokazujemo da se na reverzibilne procese prenosi osobina koja važi za proste procese, kada se restrikcija nalazi ispred paralelne kompozicije i pokazujemo da restrikcija ne utiče na memoriju.

Direktna posledica osobina strukturne kongruencije za proste i reverzibilne proces jeste da se preostali identiteti koji važe za restrikciju direktno prenose.

Lema 3.1. *Neka su data imena* $a\,b$*, memorija* m *i proces* P*. Tada važi:*

1. $m \triangleright (a)0 \equiv_R m \triangleright 0$ *i*

2. $m \triangleright (a)(b)P \equiv_R m \triangleright (b)(a)P.$

Zbog dodeljivanja adresa memorijama, pravila za paralelnu kompoziciju na prostim procesima se ne prenose na reverzibilne procese.

Definicija 3.18 (standardna forma za reverzibilni proces)**.** *Za proces oblika*

$$(\vec{a})(m_1 \triangleright M_1 \mid ... \mid m_n \triangleright M_n),$$

gde su $M_1, \dots, M_n$ *sume,* $m_1, \dots, m_n$ *proizvoljne memorije i* $\vec{a}$ *proizvoljna imena, kažemo da je u standardnoj formi.*

Lema 3.2. *Ako je* R *reverzibilni proces, onda postoji reverzibilni proces* R' *u standardnoj formi sa osobinom* $R \equiv_R R'$*.*

Dokaz. Lemu dokazujemo indukcijom po strukturi procesa R.

(i) Prvi korak indukcije je da dokažemo da lema važi za osnovni proces R koji je oblika $R = m \triangleright P$. Možemo pretpostaviti, bez gubitka opštosti, da je proces P dat u standardnoj formi:

$$P = (\vec{a})(T_1 \mid ... \mid T_n) \quad \text{gde je} \quad T_i = \alpha_i.P_i + M_i, i \in \{1, ..., n\}.$$

Tada je

$$R = m \triangleright (\vec{a})(T_1 \mid ... \mid T_n).$$

Ako na proces R primenimo pravilo za restrikciju $m \triangleright (a)P \equiv_R (a)m \triangleright P$, dobijamo:

$$m \triangleright (\vec{a})(T_1 \mid ... \mid T_n) \equiv_R (\vec{a})m \triangleright (T_1 \mid ... \mid T_n).$$

Zatim, na osnovu pravila za grananje memorije $m \triangleright (P \mid Q) \equiv_R (\langle 1 \rangle \cdot m \triangleright P \mid \langle 2 \rangle \cdot m \triangleright Q)$ dobijamo

$$\begin{aligned}
(\vec{a})m \triangleright (T_1 \mid ... \mid T_n) \quad &\equiv_R \quad (\vec{a})(\langle 1 \rangle \cdot m \triangleright (T_1 \mid ... \mid T_{n-1}) \mid \langle 2 \rangle \cdot m \triangleright T_n) \\
&\equiv_R \quad (\vec{a})(\langle 1 \rangle\langle 1 \rangle \cdot m \triangleright (T_1 \mid \dots \mid T_{n-2}) \mid \langle 1 \rangle\langle 2 \rangle \cdot m \triangleright T_{n-1} \mid \\
&\qquad \langle 2 \rangle \cdot m \triangleright T_n) \\
&\equiv_R \quad \dots \\
&\equiv_R \quad (\vec{a})(\langle 1 \rangle ... \langle 1 \rangle \cdot m \triangleright T_1 \mid \langle 1 \rangle \dots \langle 1 \rangle\langle 2 \rangle \cdot m \triangleright T_2 \mid \\
&\qquad \dots \mid \langle 1 \rangle\langle 2 \rangle \cdot m \triangleright T_{n-1} \mid \langle 2 \rangle \cdot m \triangleright T_n)
\end{aligned}$$

Memorije u dobijenom procesu su oblika $\langle 1\rangle...\langle 1\rangle\langle 2\rangle$, gde svaki naredni paralelni proces ima jedno grananje, pa time i jednu jedinicu u memoriji manje. Ovako dobijene memorije su koherentne, tako da ih možemo označiti sa m_i, za $i = 1, ...n$. Na ovaj način dobijamo traženu standardnu formu

$$(\vec{a})(m_1 \triangleright T_1 \mid m_2 \triangleright T_2 \mid ... \mid m_n \triangleright T_n).$$

(ii) (Indukcijska pretpostavka) pretpostavimo da lema važi za procese R_1 i R_2.

(iii) Treba pokazati da lema važi za sve procese oblika $(a)R_1$ i $R_1 \mid R_2$:

- $R = (a)R_1$: Prema indukcijskoj pretpostavci za R_1 važi:

  $$R = (a)(\vec{a})(m_1 \triangleright T_1 \mid ... \mid m_n \triangleright T_n).$$

  Vidimo da je R u standardnoj formi.

- $R = R_1 \mid R_2$: Prema indukcijskoj pretpostavci, procesi R_1 i R_2 su oblika:

  $$\begin{aligned} R_1 &= (\vec{a})(m_1 \triangleright T_1 \mid \ \dots \ \mid m_n \triangleright T_n) \\ R_2 &= (\vec{a'})(m'_1 \triangleright T'_1 \mid \ \dots \ \mid m'_n \triangleright T'_n). \end{aligned}$$

  Primenom α- konverzije, dobijamo:

  $$\begin{aligned} R &= R_1 \mid R_2 \\ &= (\vec{a})(m_1 \triangleright T_1 \mid \ \dots \ \mid m_n \triangleright T_n) \mid (\vec{a'})(m'_1 \triangleright T'_1 \mid \ \dots \ \mid m'_n \triangleright T'_n) \\ &\equiv_R (\vec{a})(\vec{a'})(m_1 \triangleright T_1 \mid ... \mid m_n \triangleright T_n \mid m'_1 \triangleright T'_1 \mid ... \mid m'_n \triangleright T'_n). \end{aligned}$$

■

4 Operacione semantike za RCCS

U ovom delu predstavljamo dve standardne operacione semantike koje se koriste u procesnim računima. Prvo predstavljamo semantiku koja se definiše pomoću relacija sa označenim prelascima. Ova semantika uvedena je u radu [4].

4.1 LTS semantika

Neka su oznake uređeni parovi (μ, ζ), čije su komponente definisane na sledeći način:

$$\begin{array}{rcll} \text{Direktne akcije } \zeta & ::= & \alpha & \text{(akcija unapred)} \\ & \mid & \alpha_* & \text{(akcija unazad)} \\ \text{Identifikatori } \mu & ::= & m & \text{(memorija)} \\ & \mid & m, m & \text{(memorijski par)} \end{array}$$

Definicija 4.1 (LTS)**.** *Sistem sa označenim prelascima, LTS, je uređena trojka* $(\mathcal{R}, \mathcal{O}, \{\xrightarrow{\mu:\zeta} : (\mu, \zeta) \in \mathcal{O}\})$ *sa osobinom:*

(i) skup stanja $\mathcal{R}$ je skup svih reverzibilnih procesa,

(ii) skup oznaka $\mathcal{O}$ je $\{(\mu, \zeta) : \mu$ je direktna akcija i ζ je identifikator $\}$ i

(iii) relacije sa označenim prelascima $\xrightarrow{\mu:\zeta} \in \mathcal{R} \times \mathcal{R}$, $(\mu, \zeta) \in \mathcal{O}$, su binarne relacije definisane u Tabeli 1.

Intuitivno, zapis $R \xrightarrow{\mu:\zeta} S$ znači da proces R, nakon izvođenja direktne akcije ζ prelazi u proces S. Identifikator μ pri tome sadrži memoriju ili par memorija. Često ćemo umesto memorije koristiti njenu adresu. Pravila LTS-a podelićemo u četiri grupe:

I Prva dva pravila predstavljaju **osnovna pravila** prelazaka unapred i unazad.

> [L-ACT] : U pravilu za prelazak unapred, u kojem proces izvršava akciju α, memorijska trojka $\langle *, \alpha, Q \rangle$, koja se sastoji od nepoznatog partnera „$*$", akcije α i izostavljenog procesa Q, stavlja se na vrh memorije. Prelazak koji je označen akcijom α, $\alpha \neq \tau$, može biti sinhronizovan samo sa drugim prelaskom koji je označen komplementarnom akcijom $\bar{\alpha}$.

[L-ACT*] : Prelazak unazad, u kojem proces izvršava akciju $\alpha *$, poništava poslednji izvršeni prelazak unapred. Pri tome, pravilo koristi informacije o poslednjoj izvršenoj akciji, koje su sačuvane na vrhu memorije.

II Naredna tri pravila predstavljaju **kontekstualna pravila** tj. pravila koja mogu biti izvedena unutar proizvoljnih konteksta.

[L-PAR1-L], [L-PAR1-D] : U pravilima za paralelne procese, prelazak se odvija na procesu sa leve ili desne strane paralelne kompozicije.

[L-RES1] : U pravilu za restrikciju, a ne sme da bude nijedna od direktnih akcija, tj. $a \notin \{\alpha, \alpha_*\}$.

III Pravila **sinhronizacije** obuhvataju pravila za prelaz unapred i prelaz unazad. Oba pravila pokazuju vezu između procesa R sa memorijama m_1 i m_2 i procesa koji nastaje od njega $R_{m_2 @ m_1}$. Ako je proces R koherentan, tada se ova dva procesa razlikuju tačno za drugi deo memorijskog para koji odgovara memoriji procesa R. Karakteristika pravila sinhronizacije unazad [L-SIN*] jeste da je njegova primena jedini način da se proces koji se sinhronizovao pravilom [L-SIN1] sa drugim procesom, vrati u prvobitno stanje.

[L-SIN1] : Procesi koji imaju prelaze takve da su im akcije komplementarne i identifikatori grane jedne memorije, m_1 i m_2, sinhronizacijom unapred (izvršavanjem akcije τ sa memorijama m_1 i m_2) prelaze u procese koji pamte memoriju i ime drugog procesa sa kojim su se sinhronizovali.

[L-SIN*] : Pravilo sinhonizacije možemo poništiti tako što ćemo na procese sa odgovarajućim prelazima koji su sinhonizovani, primeniti akciju τ_* sa memorijama m_1 i m_2 (ponovo ćemo ih sinhronizovati, samo unazad) i dobiti odgovarajuće nezavisne procese koje smo imali.

IV Četvrta grupa predstavlja četiri pravila kojima možemo da nadomestimo pravilo

$$\text{(STRUCT)}\quad \frac{R \equiv_R R' \quad R' \xrightarrow{(\mu,\zeta)} S' \quad S' \equiv_R S}{R \xrightarrow{(\mu,\zeta)} S}$$

Primer. Neka je dat proces

$$R = m \triangleright (a.\bar{b}.0 \mid \bar{a}.b.0).$$

Prema pravilu [L-ACT] zaključujemo

$$\langle 1 \rangle m \triangleright a.\bar{b}.0 \xrightarrow{\langle 1 \rangle m:a} \langle *, a, 0 \rangle \langle 1 \rangle m \triangleright \bar{b}.0$$

$$\langle 2 \rangle m \triangleright \bar{a}.b.0 \xrightarrow{\langle 2 \rangle m:\bar{a}} \langle *, \bar{a}, 0 \rangle \langle 2 \rangle m \triangleright b.0.$$

[L-ACT]
$$\frac{}{m \triangleright \alpha.P + Q \xrightarrow{m:\alpha} \langle *, \alpha, Q\rangle m \triangleright P}$$

[L-ACT*]
$$\frac{}{\langle *, \alpha, Q\rangle m \triangleright P \xrightarrow{m:\alpha_*} m \triangleright \alpha.P + Q}$$

[L-PAR1-L]
$$\frac{R \xrightarrow{\mu:\zeta} R'}{R \mid S \xrightarrow{\mu:\zeta} R' \mid S}$$

[L-PAR1-D]
$$\frac{R \xrightarrow{\mu:\zeta} R'}{S \mid R \xrightarrow{\mu:\zeta} S \mid R'}$$

[L-RES1]
$$\frac{R \xrightarrow{\mu:\zeta} R' \quad \zeta \notin \{a, \bar{a}, a_*, \bar{a}_*\}}{(a)R \xrightarrow{\mu:\zeta} (a)R'}$$

[L-SIN1]
$$\frac{R \xrightarrow{m_1:\alpha} R' \quad S \xrightarrow{m_2:\bar{\alpha}} S' \quad \alpha \neq \tau}{R \mid S \xrightarrow{m_1,m_2:\tau} R'_{m_2@m_1} \mid S'_{m_1@m_2}}$$

[L-SIN*]
$$\frac{R \xrightarrow{m_1:\alpha_*} R' \quad S \xrightarrow{m_2:\bar{\alpha}_*} S' \quad \alpha \neq \tau}{R_{m_2@m_1} \mid S_{m_1@m_2} \xrightarrow{m_1,m_2:\tau_*} R' \mid S'}$$

[L-RES2]
$$\frac{m \triangleright P \xrightarrow{\mu:\zeta} m' \triangleright P', \quad \zeta \notin \{a, \bar{a}, a_*, \bar{a}_*\}}{m \triangleright (a)P \xrightarrow{\mu:\zeta} m' \triangleright (a)P'}$$

[L-SIN2]
$$\frac{m_1 \triangleright P \xrightarrow{m_1:\alpha} R' \quad m_2 \triangleright Q \xrightarrow{m_2:\bar{\alpha}} S' \quad \alpha \neq \tau}{m \triangleright (P \mid Q) \xrightarrow{m_1,m_2:\tau} R'_{m_2@m_1} \mid S'_{m_1@m_2}}$$

[L-PAR2-L]
$$\frac{m_1 \triangleright P \xrightarrow{\mu:\zeta} R'}{m \triangleright (P \mid Q) \xrightarrow{\mu:\zeta} R' \mid m_2 \triangleright Q}$$

[L-PAR2-D]
$$\frac{m_2 \triangleright Q \xrightarrow{\mu:\zeta} R''}{m \triangleright (P \mid Q) \xrightarrow{\mu:\zeta} m_1 \triangleright P \mid R''}$$

Tabela 1: Relacija sa označenim prelascima (LTS).

Zatim primenjujući pravilo [L-SIN2] dobijamo

$$m \triangleright (a.\bar{b}.0 \mid \bar{a}.b.0) \xrightarrow{\langle 1\rangle\cdot m, \langle 2\rangle\cdot m:\tau} \langle\langle 2\rangle \cdot m, a, 0\rangle\langle 1\rangle \cdot m \triangleright \bar{b}.0 \mid \langle\langle 1\rangle \cdot m, \bar{a}, 0\rangle\langle 2\rangle \cdot m \triangleright b.0$$

Sa m_R i m_S označićemo memorije $\langle\langle 2\rangle \cdot m, a, 0\rangle\langle 1\rangle \cdot m$ i $\langle\langle 1\rangle \cdot m, \bar{a}, 0\rangle\langle 2\rangle \cdot m$.

Na osnovu pravila [L-ACT] zaključujemo

$$m_R \triangleright \bar{b}.0 \xrightarrow{m_R:\bar{b}} \langle *, \bar{b}, 0\rangle m_R \triangleright 0$$

$$m_S \triangleright b.0 \xrightarrow{m_S:b} \langle *, b, 0\rangle m_S \triangleright 0$$

Primenjujući pravilo [L-SIN1] dobijamo

$$m_R \triangleright \bar{b}.0 \mid m_S \triangleright b.0 \xrightarrow{m_R,m_S:\tau} \langle m_S, \bar{b}, 0\rangle m_R \triangleright 0 \mid \langle m_R, b, 0\rangle m_S \triangleright 0$$

Posmatraćemo sada proces koji je strukturno kongruentan procesu iz prethodnog primera.

Primer. Neka je dat proces

$$S = \langle 1\rangle \cdot m \triangleright a.\bar{b}.0 \mid \langle 2\rangle \cdot m \triangleright \bar{a}.b.0$$

Prema pravilu [L-ACT] zaključujemo

$$\langle 1\rangle \cdot m \triangleright a.\bar{b}.0 \xrightarrow{\langle 1\rangle \cdot m:a} \langle *, a, 0\rangle\langle 1\rangle \cdot m \triangleright \bar{b}.0$$

$$\langle 2\rangle \cdot m \triangleright \bar{a}.b.0 \xrightarrow{\langle 2\rangle \cdot m:\bar{a}} \langle *, \bar{a}, 0\rangle\langle 2\rangle \cdot m \triangleright b.0.$$

Tada primenjujući pravilo [L-SIN1], dobijamo

$$\langle 1\rangle \cdot m \triangleright a.\bar{b}.0 \mid \langle 2\rangle \cdot m \triangleright \bar{a}.b.0 \xrightarrow{\langle 1\rangle \cdot m, \langle 2\rangle \cdot m:\tau} \langle\langle 2\rangle \cdot m, a, 0\rangle\langle 1\rangle \cdot m \triangleright \bar{b}.0 \mid \langle\langle 1\rangle \cdot m, \bar{a}, 0\rangle\langle 2\rangle \cdot m \triangleright b.0$$

Dobijena dva procesa mogu opet da se sinhronizuju. Primenjujući pravilo [L-ACT] imamo

$$\langle\langle 2\rangle \cdot m, a, 0\rangle\langle 1\rangle \cdot m \triangleright \bar{b}.0 \xrightarrow{\langle\langle 2\rangle \cdot m,a,0\rangle\langle 1\rangle \cdot m:\bar{b}} \langle *, \bar{b}, 0\rangle\langle\langle 2\rangle \cdot m, a, 0\rangle\langle 1\rangle \cdot m \triangleright 0$$

$$\langle\langle 1\rangle \cdot m, \bar{a}, 0\rangle\langle 2\rangle \cdot m \triangleright b.0 \xrightarrow{\langle\langle 1\rangle \cdot m,\bar{a},0\rangle\langle 2\rangle \cdot m:b} \langle *, b, 0\rangle\langle\langle 1\rangle \cdot m, \bar{a}, 0\rangle\langle 2\rangle \cdot m \triangleright 0$$

Memorije $\langle\langle 2\rangle \cdot m, a, 0\rangle\langle 1\rangle \cdot m$ i $\langle\langle 1\rangle \cdot m, \bar{a}, 0\rangle\langle 2\rangle \cdot m$, označićemo sa m_R i m_S, respektivno. Kada primenimo pravilo [L-SIN1], dobijamo

$$m_R \triangleright \bar{b}.0 \mid m_S \triangleright b.0 \xrightarrow{m_R, m_S:\tau} \langle m_S, \bar{b}, 0\rangle m_R \triangleright 0 \mid \langle m_R, b, 0\rangle m_S \triangleright 0$$

Sledeći primer ilustruje kako se dobijeni proces sa memorijom može vratiti nazad u prvobitno stanje.

Primer. Dat je proces $R = \langle m_S, \bar{b}, 0\rangle m_R \triangleright 0 \mid \langle m_R, b, 0\rangle m_S \triangleright 0$. Koristimo oznake m_R i m_S, definisane kao u prethodnom primeru.

Primenom pravila [L-ACT*], dobijamo

$$\langle *, \bar{b}, 0\rangle m_R \triangleright 0 \xrightarrow{m_R:\bar{b}_*} m_R \triangleright \bar{b}.0$$

$$\langle *, b, 0\rangle m_S \triangleright 0 \xrightarrow{m_S:b_*} m_S \triangleright b.0$$

Tada primenjujući pravilo [L-SIN*], dobijamo

$$\langle m_S, \bar{b}, 0\rangle m_R \triangleright 0 \mid \langle m_R, b, 0\rangle m_S \triangleright 0 \xrightarrow{m_R, m_S:\tau_*} m_R \triangleright \bar{b}.0 \mid m_S \triangleright b.0$$

Kada zamenimo m_R i m_S, dobijeni proces je oblika

$$\langle\langle 2\rangle \cdot m, a, 0\rangle\langle 1\rangle \cdot m \triangleright \bar{b}.0 \mid \langle\langle 1\rangle \cdot m, \bar{a}, 0\rangle\langle 2\rangle \cdot m \triangleright b.0$$

Kada na dobijene potprocese ponovo primenimo pravilo [L-ACT*], imamo

$$\langle *, a, 0\rangle\langle 1\rangle \cdot m \triangleright \bar{b}.0 \xrightarrow{\langle 1\rangle \cdot m:a_*} \langle 1\rangle \cdot m \triangleright a.\bar{b}.0$$

$$\langle *, \bar{a}, 0\rangle\langle 2\rangle \cdot m \triangleright b.0 \xrightarrow{\langle 2\rangle\cdot m:\bar{a}_*} \langle 2\rangle \cdot m \triangleright \bar{a}.b.0$$

Primenom pravila [L-SIN*], dobijamo

$$\langle\langle 2\rangle \cdot m, a, 0\rangle\langle 1\rangle \cdot m \triangleright \bar{b}.0 \mid \langle\langle 1\rangle \cdot m, \bar{a}, 0\rangle\langle 2\rangle \cdot m \triangleright b.0 \xrightarrow{\langle 1\rangle\cdot m,\langle 2\rangle\cdot m:\tau_*} \langle 1\rangle \cdot m \triangleright a.\bar{b}.0 \mid \langle 2\rangle \cdot m \triangleright \bar{a}.b.0$$

Lema 4.1. *Neka su data dva strukturno kongruentna reverzibilna procesa* $R \equiv_R R_1$ *i neka važi* $R \xrightarrow{\mu:\zeta} R'$. *Tada postoji reverzibilni proces* R_1' *sa osobinom* $R_1 \xrightarrow{\mu:\zeta} R_1'$ *i* $R_1' \equiv_R R'$.

Dokaz. Lemu dokazujemo indukcijom po dužini izvodjenja zaključka $R \equiv_R R_1$. Pretpostavljamo da je zaključak $R \equiv_R R_1$ dobijen u jednom koraku. Posmatraćemo dva slučaja:

(i_1) $m \triangleright (a)P \equiv_R (a)m \triangleright P$:

- Ako je $m \triangleright (a)P \xrightarrow{\mu:\zeta} R'$, inverzijom zaključujemo da je

  $$R' = m' \triangleright (a)P' \quad \text{i} \quad m \triangleright P \xrightarrow{\mu:\zeta} m' \triangleright P' \quad \text{i} \quad a \notin \zeta.$$

  Kako prema pravilu [L-RES1] važi

  $$(a)m \triangleright P \xrightarrow{\mu:\zeta} (a)m' \triangleright P' \quad \text{i} \quad (a)m' \triangleright P' \equiv_R m' \triangleright (a)P',$$

  tvrđenje važi za $R_1' = (a)m' \triangleright P'$.
- Ako je $(a)m \triangleright P \xrightarrow{\mu:\zeta} R'$, inverzijom zaključujemo da je

  $$R' = (a)m' \triangleright P' \quad \text{i} \quad m \triangleright P \xrightarrow{\mu:\zeta} m' \triangleright P' \quad \text{i} \quad a \notin \zeta.$$

  Kako prema pravilu [L-RES2] važi

  $$m \triangleright (a)P \xrightarrow{\mu:\zeta} m' \triangleright (a)P' \quad \text{i} \quad m' \triangleright (a)P' \equiv_R (a)m' \triangleright P'$$

  tvrđenje važi za $R_1' = m' \triangleright (a)P'$.

(i_2) $m \triangleright (P \mid Q) \equiv_R (\langle 1\rangle \cdot m \triangleright P \mid \langle 2\rangle \cdot m \triangleright Q)$:
U daljem dokazivanju, koristićemo oznaku m_i umesto $\langle i\rangle \cdot m$, za $i = 1, 2$.

Ako je $m \triangleright (P \mid Q) \xrightarrow{\mu:\zeta} R'$, inverzijom (iz pravila [L-SIN2], [L-PAR2-L] i [L-PAR2-D]) zaključujemo da su moguća tri slučaja:

- $R' = R'_{m_2@m_1} \mid S'_{m_1@m_2}$ i $\mu : \zeta = m_1, m_2 : \tau$ i $m_1 \triangleright P \xrightarrow{m_1:\alpha} R'$ i $m_2 \triangleright Q \xrightarrow{m_2:\bar{\alpha}} S'$: Primenom pravila [L-SIN1] dobijamo

  $$m_1 \triangleright P \mid m_2 \triangleright Q \xrightarrow{m_1,m_2:\tau} R'_{m_2@m_1} \mid S'_{m_1@m_2},$$

  odakle sledi da tvrđenje važi za $R_1' = R'$.

- $R' = R' \mid m_2 \triangleright Q$ i $m_1 \triangleright P \xrightarrow{\mu:\zeta} R'$: Primenom pravila [L-PAR-L] dobijamo

$$m_1 \triangleright P \mid m_2 \triangleright Q \xrightarrow{\mu:\zeta} R' \mid m_2 \triangleright Q,$$

odakle sledi da tvrđenje važi za $R'_1 = R'$.

- $R' = m_1 \triangleright P \mid R''$ i $m_2 \triangleright Q \xrightarrow{\mu:\zeta} R''$: Primenom pravila [L-PAR-D] dobijamo

$$m_1 \triangleright P \mid m_2 \triangleright Q \xrightarrow{\mu:\zeta} m_1 \triangleright Q \mid R'',$$

odakle ponovo sledi da tvrđenje važi za $R'_1 = R'$.

Ako je $\langle 1\rangle \cdot m \triangleright P \mid \langle 2\rangle \cdot m \triangleright Q \xrightarrow{\mu:\zeta} R'$, inverzijom (iz pravila [L-PAR1-L], [L-PAR1-D], i [L-SIN1] zaključujemo da su mogući sledeći slučajevi:

- $R' = R'' \mid \langle 2\rangle \cdot m \triangleright Q$ i $\langle 1\rangle \cdot m \triangleright P \xrightarrow{\mu:\zeta} R''$: Primenom pravila [L-PAR2-L] dobijamo

$$m \triangleright (P \mid Q) \xrightarrow{\mu:\zeta} R'' \mid \langle 2\rangle \cdot m \triangleright Q,$$

odakle sledi da tvrđenje važi za $R'_1 = R'$.

- $R' = \langle 1\rangle \cdot m \triangleright P \mid R''$ i $\langle 2\rangle \cdot m \triangleright Q \xrightarrow{\mu:\zeta} R''$: Primenom pravila [L-PAR2-D] dobijamo

$$m \triangleright (P \mid Q) \xrightarrow{\mu:\zeta} \langle 1\rangle \cdot m \triangleright P \mid R'',$$

odakle sledi da tvrđenje važi za $R'_1 = R'$.

- $R' = R'_{m_2@m_1} \mid S'_{m_1@m_2}$ i $\langle 1\rangle \cdot m \triangleright P \xrightarrow{m_1:\alpha} R'$ i $\langle 2\rangle \cdot m \triangleright Q \xrightarrow{m_2:\bar{\alpha}} S'$ i $\alpha \neq \tau$: Primenom pravila [L-SIN2] dobijamo

$$m \triangleright (P \mid Q) \xrightarrow{\mu:\zeta} R'_{m_2@m_1} \mid S'_{m_1@m_2}$$

odakle sledi da tvrđenje važi za $R'_1 = R'$.

Svi ostali slučajevi, uključujući induktivni korak, slede direktno. ■

Lema 4.2. **(i)** *Ako* $R \xrightarrow{m:\alpha} R'$ *i* $\alpha \in \{a, \bar{a}\}$ *, tada važi*

$$\begin{aligned} R &\equiv_R (\vec{a})(m \triangleright \alpha.P + Q \mid R'') \quad i \\ R' &\equiv_R (\vec{a})(\langle *, \alpha, Q\rangle m \triangleright P \mid R'') \end{aligned}$$

za neki reverzibilni proces R''*, proste procese* P, Q*, akciju* α*, memoriju* m *i imena* $(\vec{a})$.

(ii) *Ako* $R \xrightarrow{m:\alpha_*} R'$ *i* $\alpha_* \in \{a_*, \bar{a}_*\}$ *i , tada važi:*

$$\begin{aligned} R &\equiv_R (\vec{a})(\langle *, \alpha, Q\rangle m \triangleright P \mid R'') \quad i \\ R' &\equiv_R (\vec{a})(m \triangleright \alpha.P + Q \mid R'') \end{aligned}$$

za neki reverzibilni proces R''*, proste procese* P, Q*, akciju* α*, memoriju* m *i imena* $(\vec{a})$.

Dokaz. Lemu dokazujemo indukcijom po dužini izvođenja prelaza $R \xrightarrow{m:\alpha} R'$.

(i) Baza indukcije pokazuje da lema važi ako je prethodni zaključak izveden u jednom koraku:

- Ako je $R \xrightarrow{m:\alpha} R'$ izvedeno na osnovu pravila [L-ACT] onda je

$$R = m \triangleright \alpha.P + Q \quad \text{i} \quad R' = \langle *, \alpha, Q\rangle m \triangleright P.$$

(ii) (Induktivna pretpostavka) Pretpostavimo da lema važi ako je $R \xrightarrow{m:\alpha} R'$ izvedeno u k koraka.

(iii) Dokazujemo da lema važi ako je prelaz $R \xrightarrow{m:\alpha} R'$ izveden u $k+1$ koraka. Poslednji korak izvođenja je onda dobijen primenom jednog od sledećih pravila:

[L-PAR1-L] Pretpostavimo da je zaključak $R \mid S \xrightarrow{m:\alpha} R' \mid S$ izveden iz zaključka $R \xrightarrow{m:\alpha} R'$ koji je dobijen u k koraka. Na osnovu induktivne pretpostavke sledi

$$\begin{aligned} R &\equiv_R (\vec{a})(m \triangleright \alpha.P + Q \mid R'') \quad \text{i} \\ R' &\equiv_R (\vec{a})(\langle *, \alpha, Q\rangle m \triangleright P \mid R''). \end{aligned}$$

Tada je

$$\begin{aligned} R \mid S &\equiv_R (\vec{a})(m \triangleright \alpha.P + Q \mid R'') \mid S \quad \text{i} \\ R' \mid S &\equiv_R (\vec{a})(\langle *, \alpha, Q\rangle m \triangleright P \mid R'') \mid S. \end{aligned}$$

Na osnovu Leme 3.2,

$$S \equiv_R (\vec{a'})(m'_1 \triangleright \alpha'_1.P'_1 + Q'_1 \mid ... \mid m'_l \triangleright \alpha_l.P'_l + Q'_l)$$

odakle primenom α-konverzije i osobine strukturne kongruencije za reverzibilne procese sledi tvrđenje.

[L-PAR1-D] Pretpostavimo da je zaključak $S \mid R \xrightarrow{m:\alpha} S \mid R'$ izveden iz zaključka $R \xrightarrow{m:\alpha} R'$ koji je dobijen u k koraka. Na osnovu induktivne pretpostavke sledi

$$\begin{aligned} R &\equiv_R (\vec{a})(m \triangleright \alpha.P + Q \mid R'') \quad \text{i} \\ R' &\equiv_R (\vec{a})(\langle *, \alpha, Q\rangle m \triangleright P \mid R''). \end{aligned}$$

Tada je

$$\begin{aligned} S \mid R &\equiv_R S \mid (\vec{a})(m \triangleright \alpha.P + Q \mid R'') \quad \text{i} \\ S \mid R' &\equiv_R S \mid (\vec{a})(\langle *, \alpha, Q\rangle m \triangleright P \mid R''). \end{aligned}$$

Na osnovu Leme 3.2,

$$S \equiv_R (\vec{a'})(m'_1 \triangleright \alpha'_1.P'_1 + Q'_1 \mid ... \mid m'_l \triangleright \alpha_l.P'_l + Q'_l)$$

tada primenom α-konverzije i osobine strukturne kongruencije za reverzibilne procese sledi tvrđenje.

[L-RES] Ako je zaključak $(a)R \xrightarrow{m:\alpha} (a)R'$ izveden iz zaključka $R \xrightarrow{m:\alpha} R'$ koji je dobijen u k koraka. Na osnovu induktivne pretpostavke sledi

$$\begin{array}{rcl} R & \equiv_R & (\vec{a})(m \triangleright \alpha.P + Q \mid R'') \quad \text{i} \\ R' & \equiv_R & (\vec{a})(\langle *, \alpha, Q\rangle m \triangleright P \mid R''). \end{array}$$

Tada je

$$\begin{array}{rcl} (a)R & \equiv_R & (a)(\vec{a})(m \triangleright \alpha.P + Q \mid R'') \quad \text{i} \\ (a)R' & \equiv_R & (a)(\vec{a})(\langle *, \alpha, Q\rangle m \triangleright P \mid R''). \end{array}$$

Odakle direktno sledi tvrđenje.

[L-PAR2-L] Pretpostavimo da je zaključak $S \xrightarrow{m:\alpha} R' \mid \langle 2\rangle \cdot m \triangleright T$ izveden iz zaključka $R \xrightarrow{m:\alpha} R'$ koji je dobijen u k koraka i gde su $S = m \triangleright (T_1 \mid T)$, $R = \langle 1\rangle \cdot m \triangleright T_1$ i pritom su T i T_1 prosti procesi. Tada na osnovu induktivne pretpostavke sledi

$$\begin{array}{rcl} R = \langle 1\rangle \cdot m \triangleright T_1 & \equiv_R & (\vec{a})(m \triangleright \alpha.P + Q \mid R'') \quad \text{i} \\ R' & \equiv_R & (\vec{a})(\langle *, \alpha, Q\rangle m \triangleright P \mid R''). \end{array}$$

Kako je S reverzibilni proces, na osnovu Leme 3.2, imamo

$$\begin{array}{rcl} S = m \triangleright (T_1 \mid T) & \equiv_R & (\vec{a})(m \triangleright \alpha.P + Q \mid R'' \mid \langle 2\rangle \cdot m \triangleright T) \quad \text{i} \\ R' \mid \langle 2\rangle \cdot m \triangleright T & \equiv_R & (\vec{a})(\langle *, \alpha, Q\rangle m \triangleright P \mid R'') \mid \langle 2\rangle \cdot m \triangleright T \end{array}$$

Primenom α-konverzije i osobine strukturne kongruencije za reverzibilne procese sledi tvrđenje.

[L-PAR2-D] Pretpostavimo da je zaključak $S \xrightarrow{m:\alpha} \langle 1\rangle \cdot m \triangleright T_1 \mid R'$ izveden iz zaključka $R \xrightarrow{m:\alpha} R'$ koji je dobijen u k koraka i gde su $S = m \triangleright (T_1 \mid T)$, $R = \langle 2\rangle \cdot m \triangleright T$ i pritom su T i T_1 prosti procesi. Tada na osnovu induktivne pretpostavke sledi

$$\begin{array}{rcl} R = \langle 2\rangle \cdot m \triangleright T & \equiv_R & (\vec{a})(m \triangleright \alpha.P + Q \mid R'') \quad \text{i} \\ R' & \equiv_R & (\vec{a})(\langle *, \alpha, Q\rangle m \triangleright P \mid R''). \end{array}$$

Kako je S reverzibilni proces, na osnovu Leme 3.2, imamo

$$\begin{array}{rcl} S = m \triangleright (T_1 \mid T) & \equiv_R & (\vec{a})(\langle 1\rangle \cdot m \triangleright T_1 \mid m \triangleright \alpha.P + Q \mid R'') \quad \text{i} \\ \langle 1\rangle \cdot m \triangleright T_1 \mid R' & \equiv_R & \langle 1\rangle \cdot m \triangleright T_1 \mid (\vec{a})(\langle *, \alpha, Q\rangle m \triangleright P \mid R'') \end{array}$$

Primenom α-konverzije i osobine strukturne kongruencije za reverzibilne procese sledi tvrđenje.

[L-RES2] Ako je zaključak $m \triangleright (a)T \xrightarrow{m:\alpha} m' \triangleright (a)T'$, gde su procesi R i R' dati sa $R = m \triangleright T$ i $R' = m' \triangleright T'$, izveden iz zaključka $R \xrightarrow{m:\alpha} R'$ koji je dobijen u k koraka. Na osnovu induktivne pretpostavke sledi

$$\begin{array}{rcl} R = & \equiv_R & (\vec{a})(m \triangleright \alpha.P + Q \mid R'') \quad \text{i} \\ R' = & \equiv_R & (\vec{a})(\langle *, \alpha, Q\rangle m \triangleright P \mid R''). \end{array}$$

Tada je na osnovu Leme 3.2 imamo

$$\begin{aligned} m \triangleright (a)T &\equiv_R (a)(\vec{a})(m \triangleright \alpha.P + Q \mid R'') \quad \text{i} \\ m' \triangleright (a)T' &\equiv_R (a)(\vec{a})(\langle *, \alpha, Q\rangle m \triangleright P \mid R''). \end{aligned}$$

Odakle direktno sledi tvrđenje. ■

Lema 4.3. *Za bilo koji prelaz unapred* $R \xrightarrow{\mu:\alpha} S$ *postoji prelaz unazad* $S \xrightarrow{\mu:\alpha_*} R$ *i obrnuto.*

Dokaz. Dokaz ove leme sledi direktno iz same konstrukcije LTS-a. Direktna akcija ζ može biti α i α_*, pa svako pravilo koje sadrži prelaze oblika $R \xrightarrow{\mu:\zeta} S$, obuhvata prelaze unapred i unazad.

4.2 Redukciona pravila

Redukciona semantika opisuje izračunavanja pojedinačnih procesa i definiše se kao redukciona relacija. Uveđnje redukcione semantike predstavlja originalni doprinos ovog master rada.

Definicija 4.2 (redukcija)**.** *Relacija redukcije, u oznaci $\rightarrow$, najmanja je binarna relacija na zatvorenim reverzibilnim procesima koja je zatvorena za pravila data u Tabeli 2. Refleksivno i tranzitivno zatvaranje relacije $\rightarrow$ označavamo sa $\rightarrow^*$.*

Ako je $P \rightarrow Q$, kažemo da je Q dobijeno od P jednim korakom izračunavanja, dok za $P \rightarrow^* Q$ kažemo da je Q dobijeno od P u više koraka.

Pravila u Tabeli 2 obuhvataju:

[R-TAU] : Ovo pravilo predstavlja unutrašnju τ akciju unapred pri kojoj se dati proces polusinhronizuje. U trojci $\langle *, \tau, Q\rangle$, prva komponenta „$*$" čuva mesto za memoriju procesa sa kojim može doći do potpune sinhronizacije. Treće mesto u trojci pripada procesu koji je bio u sumi sa procesom P pre nego što je izvršena τ akcija.

[R-TAU*] : Predstavlja unutrašnju τ_* akciju unazad koja vraća polusinhronizovani proces u stanje pre sinhronizacije.

[R-STRUCT] : Ovo pravilo dozvoljava upotrebu strukturne kongruencije u bilo kom trenutku izvođenja redukcije. Ako imamo dva para strukturno kongruentnih procesa $R \equiv_R R'$ i $S' \equiv_R S$ i proces R' se redukuje na proces S', tada možemo zaključiti da će se proces R redukovati na proces S.

[R-RES] : Prema pravilu za restrikciju, redukcija može biti izvedena unutar restrikcije.

[R-PAR] : U paralelnoj kompoziciji dva procesa, moguće je izvršiti redukciju jednog, dok drugi ostaje neizmenjen.

[R-REACT] : Pravilo reakcije omogućava sinhronizaciju dva procesa od kojih jedan sadrži akciju, a drugi komplement date akcije. Nakon redukcije, na vrhovima memorija sinhronizovanih procesa, čuvaju se informacije o prethodnim stanjima procesa.

[R-REACT*] : Pravilo reakcije unazad omogućava da se prethodno sinhronizovani procesi, redukcijom vrate u stanja u kojima su bili pre sinhronizacije.

Poslednja dva pravila, kao i prva dva, nemaju premisu (nemaju uslov iznad horizontalne linije u pravilu) pa tako predstavljaju aksiome.

Lema 4.4. *Ako postoji redukcija unapred $R \longrightarrow S$,tada za iste procese, postoji redukcija unazad $S \longrightarrow R$ i obrnuto.*

[R-TAU]
$$m \triangleright \tau.P + Q \longrightarrow \langle *, \tau, Q\rangle m \triangleright P$$

[R-TAU*]
$$\langle *, \tau, Q\rangle m \triangleright P \longrightarrow m \triangleright \tau.P + Q$$

[R-REACT]
$$m_1 \triangleright a.P + Q \mid m_2 \triangleright \bar{a}.P' + Q' \longrightarrow \langle m_2, a, Q\rangle m_1 \triangleright P \mid \langle m_1, \bar{a}, Q'\rangle m_2 \triangleright P'$$

[R-REACT*]
$$\langle m_2, a, Q\rangle m_1 \triangleright P \mid \langle m_1, \bar{a}, Q'\rangle m_2 \triangleright P' \longrightarrow m_1 \triangleright a.P + Q \mid m_2 \triangleright \bar{a}.P' + Q'$$

[R-STRUCT]
$$\frac{R \equiv_R R' \quad R' \longrightarrow S' \quad S' \equiv_R S}{R \longrightarrow S}$$

(R-RES)
$$\frac{R \longrightarrow R'}{(a)R \longrightarrow (a)R'}$$

[R-PAR]
$$\frac{R \longrightarrow R'}{R \mid S \longrightarrow R' \mid S}$$

Tabela 2: Redukciona pravila.

Dokaz. Dokaz leme sledi direktno iz same konstrukcije redukcionih pravila.

Primer. Dat je proces $\langle\rangle \triangleright (\bar{a}.\bar{b}.0 \;\mid\; a.(b.0 + c.0) \;\mid\; \bar{a}.\bar{c}.0)$.

Ovaj proces možemo redukovati na dva različita načina

(Prvi način) Posmatrajmo strukturnu kongruenciju

$$\langle\rangle \triangleright (\bar{a}.\bar{b}.0 \;\mid\; a.(b.0 + c.0) \;\mid\; \bar{b}.\bar{c}.0) \equiv_R \langle 1\rangle \triangleright \bar{a}.\bar{b}.0 \;\mid\; \langle 2\rangle \triangleright a.(b.0 + c.0) \;\mid\; \langle 3\rangle \triangleright \bar{a}.\bar{c}.0$$

Primenom pravila [R-REACT] na prvi i drugi potproces, dobijamo

$$\langle\langle 2\rangle, \bar{a}, 0\rangle\langle 1\rangle \triangleright \bar{b}.0 \;\mid\; \langle\langle 1\rangle, a, 0\rangle\langle 2\rangle \triangleright b.0 + c.0 \;\mid\; \langle 3\rangle \triangleright \bar{a}.\bar{c}.0$$

Ako memorije označimo memorije

$$m' = \langle\langle 2\rangle, \bar{a}, 0\rangle\langle 1\rangle$$

$$m'' = \langle\langle 1\rangle, a, 0\rangle\langle 2\rangle$$

Primenom pravila [R-REACT] na prvi i drugi potproces, imamo

$$\langle m'', \bar{b}, 0\rangle m' \triangleright 0 \;\mid\; \langle m', b, c.0\rangle m'' \triangleright 0 \;\mid\; \langle 3\rangle \triangleright \bar{a}.\bar{c}.0$$

(Drugi način) Počinjemo od iste strukturne kongruencije

$$\langle\rangle \triangleright (\bar{a}.\bar{b}.0 \;\mid\; a.(b.0 + c.0) \;\mid\; \bar{b}.\bar{c}.0) \equiv_R \langle 1\rangle \triangleright \bar{a}.\bar{b}.0 \;\mid\; \langle 2\rangle \triangleright a.(b.0 + c.0) \;\mid\; \langle 3\rangle \triangleright \bar{a}.\bar{c}.0$$

Primenom pravila [R-REACT] na drugi i treći potproces, imamo

$$\langle 1\rangle \triangleright \bar{a}.\bar{b}.0 \mid \langle\langle 3\rangle, a, 0\rangle\langle 2\rangle \triangleright (b.0 + c.0) \mid \langle\langle 2\rangle, \bar{a}, 0\rangle\langle 3\rangle \triangleright \bar{c}.0$$

Ako označimo memorije m_1' i m_1'' na sledeći način

$$m_1' = \langle\langle 3\rangle, a, 0\rangle\langle 2\rangle$$

$$m_1'' = \langle\langle 2\rangle, \bar{a}, 0\rangle\langle 3\rangle$$

Ponovnom primenom pravila [R-REACT] na drugi i treći potproces, dobijamo

$$\langle 1\rangle \triangleright \bar{a}.\bar{b}.0 \mid \langle m_1'', c, b.0\rangle m_1' \triangleright 0 \mid \langle m_1', \bar{c}, 0\rangle m_1'' \triangleright 0$$

U zavisnosti od toga koja dva potprocesa su reagovala, dobijamo da se početni proces može redukovati na dva različita procesa

$$\langle\langle\langle 1\rangle, a, 0\rangle\langle 2\rangle, \bar{b}, 0\rangle\langle\langle 2\rangle, \bar{a}, 0\rangle\langle 1\rangle \triangleright 0 \mid \langle\langle\langle 2\rangle, \bar{a}, 0\rangle\langle 1\rangle, b, c.0\rangle\langle\langle 1\rangle, a, 0\rangle\langle 2\rangle \triangleright 0 \mid \langle 3\rangle \triangleright \bar{a}.\bar{c}.0$$

$$\langle 1\rangle \triangleright \bar{a}.\bar{b}.0 \mid \langle\langle\langle 2\rangle, \bar{a}, 0\rangle\langle 3\rangle, c, b.0\rangle\langle\langle 3\rangle, a, 0\rangle\langle 2\rangle \triangleright 0 \mid \langle\langle\langle 3\rangle, a, 0\rangle\langle 2\rangle, \bar{c}, 0\rangle\langle\langle 2\rangle, \bar{a}, 0\rangle\langle 3\rangle \triangleright 0$$

Primer. U ovom primeru, prikazano je redukovanje dobijenih procesa iz prethodnog primera u početni proces. U ovom slučaju imaćemo dva različita procesa koja će se redukovati na isti proces.

Prvi proces: Posmatramo proces

$$\langle\langle\langle 1\rangle, a, 0\rangle\langle 2\rangle, \bar{b}, 0\rangle\langle\langle 2\rangle, \bar{a}, 0\rangle\langle 1\rangle \triangleright 0 \mid \langle\langle\langle 2\rangle, \bar{a}, 0\rangle\langle 1\rangle, b, c.0\rangle\langle\langle 1\rangle, a, 0\rangle\langle 2\rangle \triangleright 0 \mid \langle 3\rangle \triangleright \bar{a}.\bar{c}.0$$

Uvodeći oznake m' i m'' kao u prethodnom primeru, navedeni proces je oblika

$$\langle m'', \bar{b}, 0\rangle m' \triangleright 0 \mid \langle m', b, c.0\rangle m'' \triangleright 0 \mid \langle 3\rangle \triangleright \bar{a}.\bar{c}.0$$

Kada primenimo pravilo [R-REACT*] na prvi i drugi potproces, dobijamo

$$m' \triangleright \bar{b}.0 \mid m'' \triangleright b.0 + c.0 \mid \langle 3\rangle \triangleright \bar{a}.\bar{c}.0 \qquad \text{tj.}$$

$$\langle\langle 2\rangle, \bar{a}, 0\rangle\langle 1\rangle \triangleright \bar{b}.0 \mid \langle\langle 1\rangle, a, 0\rangle\langle 2\rangle \triangleright b.0 + c.0 \mid \langle 3\rangle \triangleright \bar{a}.\bar{c}.0$$

Ponovnom primenom pravila [R-REACT*] na prvi i drugi potproces dobijamo

$$\langle 1\rangle \triangleright \bar{a}.\bar{b}.0 \mid \langle 2\rangle \triangleright a.(b.0 + c.0) \mid \langle 3\rangle \triangleright \bar{a}.\bar{c}.0$$

Drugi proces: Posmatramo proces

$$\langle 1\rangle \triangleright \bar{a}.\bar{b}.0 \ \mid \ \langle\langle\langle 2\rangle, \bar{a}, 0\rangle\langle 3\rangle, c, b.0\rangle\langle\langle 3\rangle, a, 0\rangle\langle 2\rangle \triangleright 0 \ \mid \ \langle\langle\langle 3\rangle, a, 0\rangle\langle 2\rangle, \bar{c}, 0\rangle\langle\langle 2\rangle, \bar{a}, 0\rangle\langle 3\rangle \triangleright 0$$

Koristeći oznake m_1' i m_1'' kao u prethodnom primeru, navedeni proces je oblika

$$\langle 1\rangle \triangleright \bar{a}.\bar{b}.0 \ \mid \ \langle m_1'', c, b.0\rangle m_1' \triangleright 0 \ \mid \ \langle m_1', \bar{c}, 0\rangle m_1'' \triangleright 0$$

Primenom pravila [R-REACT*] na drugi i treći potproces dobijamo

$$\langle 1\rangle \triangleright \bar{a}.\bar{b}.0 \ \mid \ m_1' \triangleright c.0 + b.0 \ \mid \ m_1'' \triangleright \bar{c}.0$$

Kada zamenimo memorije m_1' i m_1'' imamo

$$\langle 1\rangle \triangleright \bar{a}.\bar{b}.0 \ \mid \ \langle\langle 3\rangle, a, 0\rangle\langle 2\rangle \triangleright c.0 + b.0 \ \mid \ \langle\langle 2\rangle, \bar{a}, 0\rangle\langle 3\rangle \triangleright \bar{c}.0$$

Ponovnom primenom pravila [R-REACT*] na drugi i treći potproces dobijamo

$$\langle 1\rangle \triangleright \bar{a}.\bar{b}.0 \ \mid \ \langle 2\rangle \triangleright a.(c.0 + b.0) \ \mid \ \langle 3\rangle \triangleright \bar{a}.\bar{c}.0$$

Konačno, možemo dokazati glavno tvrđenje, koje povezuje dve uvedene operacijske semantike.

Tvrđenje 4.1. *$R \to S$ ako i samo ako postoji proces S' i identifikator μ koji zadovoljava jedan od nevedenih uslova:*

(*i*) $R \xrightarrow{\mu:\tau} S'$ *i* $S \equiv_R S'$.

(*ii*) $R \xrightarrow{\mu:\tau_*} S'$ *i* $S \equiv_R S'$.

Dokaz. Tvrđenje dokazujemo u dva smera.

(⇒) Pretpostavimo da se proces R redukuje na proces S, pokazujemo da tada postoji proces S', i identifikator μ sa jednom od navedene dve osobine.
Dokazujemo indukcijom po dužini izvođenja redukcije $R \longrightarrow S$.

(i) Bazu indukcije predstavljaju redukcije dobijene u jednom koraku, a one se izvode pravilima:

[R-TAU] Kod redukcije ovog oblika procesi su $R = m \triangleright \tau.P + Q$ i $S = \langle *, \tau, Q\rangle m \triangleright P$. Primenjujući pravilo [L-ACT] na proces R za $\mu = m$ i $\alpha = \tau$ dobijamo

$$S' = \langle *, \tau, Q\rangle m \triangleright P \qquad \text{i} \qquad S \equiv_R S'$$

[R-TAU*] Kod redukcije ovog oblika procesi su $R = \langle *, \tau, Q\rangle m \triangleright P$ i $S = m \triangleright \tau.P + Q$. Primenjujući pravilo [L-ACT*] na proces R za $\mu = m$ i $\alpha_* = \tau_*$ dobijamo

$$S' = m \triangleright \tau.P + Q \qquad \text{i} \qquad S \equiv_R S'$$

[R-REACT] Primenom ovog pravila, procesi su
$R = m_1 \rhd a.P' + Q' \mid m_2 \rhd \bar{a}.P'' + Q''$ i
$S = \langle m_2, a, Q' \rangle m_1 \rhd P' \mid \langle m_1, \bar{a}, Q'' \rangle m_2 \rhd P''$
Primenom pravila [L-SIN1] na proces R sa odgovarajućim prelazima, imamo da je

$$S' = R'_{m_2 @ m_1} \mid R''_{m_1 @ m_2} \qquad \text{i} \qquad S \equiv_R S' \qquad \text{za procese}$$

$R' = \langle *, a, Q' \rangle m_1 \rhd P'$ i $R'' = \langle *, \bar{a}, Q'' \rangle m_2 \rhd P''$ i identifikator $\mu = m_1, m_2$

[R-REACT*] Primenom ovog pravila, procesi su oblika
$R = \langle m_2, a, Q' \rangle m_1 \rhd P' \mid \langle m_1, \bar{a}, Q'' \rangle m_2 \rhd P''$ i
$S = m_1 \rhd a.P' + Q' \mid m_2 \rhd \bar{a}.P'' + Q''$
Primenom pravila [L-SIN*] na proces R sa odgovarajućim prelazima, identifikatorom $\mu = m_1, m_2$ i akcijom τ_* dobijamo

$$S' = m_1 \rhd a.P' + Q' \mid m_2 \rhd \bar{a}.P'' + Q'' \quad \text{i} \quad S \equiv_R S'$$

(ii) (Induktivna pretpostavka) Pretpostavimo da Tvrđenje važi za sve redukcije sa k koraka.

(iii) Dokazujemo da važi za redukcije sa $k+1$ korakom. Poslednji korak redukcije, može biti izveden narednim pravilima:

[R-RES] Primenom ovog pravila, procesi su nam oblika $R = (a)R_1$ i $S = (a)R'_1$, pri čemu za redukciju $R_1 \longrightarrow R'_1$ važi induktivna pretpostavka.
Primenom pravila [L-RES1] na proces R, dobijamo

$$S' = (a)R'_1 \qquad \text{i} \qquad S \equiv_R S'$$

[R-PAR] Primenom ovog pravila, procesi su nam oblika $R = R_1 \mid R_2$ i $S = R'_1 \mid R_2$, pri čemu za redukciju $R_1 \longrightarrow R'_1$ važi induktivna pretpostavka.
Primenom pravila [L-PAR1-L] na proces R, dobijamo

$$S' = R'_1 \mid R_2 \qquad \text{i} \qquad S \equiv_R S'.$$

[R-STRUCT] Ako je poslednji korak redukcije dobijen ovim pravilom, tada na osnovu Leme 4.1, prelaz $R \xrightarrow{\mu:\tau} S'$ postoji i važi $S \equiv_R S'$.

($\Leftarrow$) Pretpostavićemo da postoje proces S' i identifikator μ sa jednom od navedenih osobina i pokazati da tada postoji redukcija $R \longrightarrow S$, pri čemu je $S \equiv_R S'$.
Dokazujemo indukcijom po dužini izvođenja zaključka $R \xrightarrow{\mu:\tau} S'$ ili $R \xrightarrow{\mu:\tau_*} S'$.

(i) Baza indukcije predstavlja prelaz $R \xrightarrow{\mu:\tau} S'$ ili $R \xrightarrow{\mu:\tau_*} S'$ koji je izveden u jednom koraku. Tada imamo dva slučaja:

* Kada je $R = m \triangleright \alpha.P + Q$. Za $\alpha = \tau$ i identifikator $\mu = m$, na osnovu pravila [L-ACT] imamo

$$m \triangleright \tau.P + Q \xrightarrow{m:\tau} \langle *, \tau, Q \rangle m \triangleright P \qquad \text{i} \qquad S' = \langle *, \tau, Q \rangle m \triangleright P$$

Na osnovu pravila [R-TAU] imamo

$$m \triangleright \tau.P + Q \longrightarrow \langle *, \tau, Q \rangle m \triangleright P \qquad \text{i} \qquad S = \langle *, \tau, Q \rangle m \triangleright P$$

$$\text{pri čemu je} \qquad S \equiv_R S'$$

* Kada je $R = \langle *, \tau, Q \rangle m \triangleright P$. Za $\alpha_* = \tau_*$ i identifikator $\mu = m$, na osnovu pravila [L-ACT*] imamo

$$\langle *, \tau, Q \rangle m \triangleright P \xrightarrow{m:\tau} m \triangleright \tau.P + Q \qquad \text{i} \qquad S' = m \triangleright \tau.P + Q$$

Na osnovu pravila [R-TAU*] dobijamo

$$\langle *, \tau, Q \rangle m \triangleright P \longrightarrow m \triangleright \tau.P + Q \qquad \text{i} \qquad S = m \triangleright \tau.P + Q$$

$$\text{pri čemu je} \qquad S \equiv_R S'$$

(ii) (Induktivna pretpostavka) Pretpostavimo da Tvrđenje važi za sva izvođenja sa k koraka.

(iii) Dokazujemo da Tvrđenje važi ako je prelaz $R \xrightarrow{\mu:\tau} S'$ ili $R \xrightarrow{\mu:\tau_*} S'$ izveden u $k+1$ koraku. Tada poslednji korak može biti izveden jednim od narednih pravila:

[L-PAR1-L] Tada je $R = R_1 \mid R_2$ i na osnovu induktivne pretpostavke, Tvrđenje važi za prelaz $R_1 \xrightarrow{m:\tau} R_1'$, gde je $\mu = m$. Primenom navedenog pravila imamo $S' = R_1' \mid R_2$
Na osnovu pravila [R-PAR] imamo

$$S = R_1' \mid R_2 \qquad \text{pri čemu je} \qquad S \equiv_R S'.$$

[L-PAR1-D] Tada na osnovu $R = R_1 \mid R_2$ i prelaza $R_2 \xrightarrow{m:\tau} R_2'$, za $\mu = m$, primenom navedenog pravila imamo $S' = R_1 \mid R_2'$.
Na osnovu pravila [R-PAR], [R-STRUCT] i osobine komutativnosti paralelnih procesa za reverzibilne procese, imamo

$$S = R_1 \mid R_2' \qquad \text{pri čemu je} \qquad S \equiv_R S'.$$

[L-RES1] Tada imamo da je $R = (a)R_1$ i prelaz $R_1 \xrightarrow{m:\tau} R_1'$, za $\mu = m$. Na osnovu navedenog pravila dobijamo da je $S' = (a)R_1'$.
Primenjujući pravilo [R-RES] na proces R, za prelaz $R_1 \longrightarrow R_1'$, dobijamo

$$S = (a)R_1' \qquad \text{pri čemu je} \qquad S \equiv_R S'.$$

[L-SIN1] Proces R je oblika $R = R_1 \mid R_2$ i prelazi su $R_1 \xrightarrow{m_1:\alpha} R'$ i $R_2 \xrightarrow{m_2:\bar{\alpha}} R''$, gde $\alpha \neq \tau$. Na osnovu Leme 4.2 imamo da za procese R_1, R_2, R' i R'' važi

$$\begin{aligned} R_1 &\equiv_R (\vec{a})(m_1 \triangleright \alpha.P_1 + Q_1 \mid T), \\ R_2 &\equiv_R (\vec{b})(m_2 \triangleright \bar{\alpha}.P_2 + Q_2 \mid T), \\ R' &\equiv_R (\vec{a})(\langle *, \alpha, Q_1\rangle m_1 \triangleright P_1 \mid T) \quad \text{i} \\ R'' &\equiv_R (\vec{b})(\langle *, \bar{\alpha}, Q_2\rangle m_2 \triangleright P_2 \mid T) \end{aligned}$$

za neki reverzibilni proces T.
Primenom navedenog pravila, dobijamo $S' = R'_{m_2@m_1} \mid R''_{m_1@m_2}$, gde je identifikator μ memorijski par m_1, m_2.
Primenom pravila [R-REACT] na proces R, dobijamo

$$S = (\vec{a})(\langle m_2, \alpha, Q_1\rangle m_1 \triangleright P_1 \mid T) \;\mid\; (\vec{b})(\langle m_1, \bar{\alpha}, Q_2\rangle m_2 \triangleright P_2 \mid T)$$

pri čemu je $S \equiv_R S'$, na osnovu pravila [R-STRUCT].

[L-SIN*] Proces R je oblika $R = R'_{m_2@m_1} \mid R''_{m_1@m_2}$ i prelazi su $R' \xrightarrow{m_1:\alpha_*} R_1$ i $R'' \xrightarrow{m_2:\bar{\alpha}_*} R_2$, gde $\alpha_* \neq \tau_*$.
Leme 4.2 imamo da za procese R_1, R_2, R' i R'' važi

$$\begin{aligned} R' &\equiv_R (\vec{a})(\langle *, \alpha, Q_1\rangle m_1 \triangleright P_1 \mid T), \\ R'' &\equiv_R (\vec{b})(\langle *, \bar{\alpha}, Q_2\rangle m_2 \triangleright P_2 \mid T), \\ R_1 &\equiv_R (\vec{a})(m_1 \triangleright \alpha.P_1 + Q_1 \mid T) \quad \text{i} \\ R_2 &\equiv_R (\vec{b})(m_2 \triangleright \bar{\alpha}.P_2 + Q_2 \mid T) \end{aligned}$$

za neki reverzibilni proces T.
Primenom navedenog pravila, dobijamo $S' = R_1 \mid R_2$, gde je identifikator μ memorijski par m_1, m_2.
Primenom pravila [R-REACT*] na proces R, dobijamo

$$S = (\vec{a})(m_1 \triangleright \alpha.P_1 + Q_1 \mid T) \;\mid\; (\vec{b})(m_2 \triangleright \bar{\alpha}.P_2 + Q_2 \mid T)$$

pri čemu je $S \equiv_R S'$, na osnovu pravila [R-STRUCT].

[L-RES2] Proces R je oblika $R = m \triangleright (a)P$. Za dati prelaz $m \triangleright P \xrightarrow{m:\tau} m' \triangleright P'$, primenom navedenog pravila imamo da je $S' = m' \triangleright (a)P'$.
Na osnovu pravila [R-RES], [R-STRUCT] i kongruencije $m \triangleright (P \mid Q) \equiv_R (\langle 1\rangle \cdot m \triangleright P \mid \langle 2\rangle \cdot m \triangleright Q)$, imamo

$$S = m' \triangleright (a)P' \qquad \text{pri čemu je} \qquad S \equiv_R S'.$$

[L-SIN2] Proces R je oblika $R = m \triangleright (P \mid T)$. Za prelaze $\langle 1\rangle \cdot m \triangleright P \xrightarrow{\langle 1\rangle \cdot m:\alpha} R'$ i $\langle 2\rangle \cdot m \triangleright T \xrightarrow{\langle 2\rangle \cdot m:\bar{\alpha}} R''$, gde $\alpha \neq \tau$

primenom navedenog pravila, imamo da je $S' = R'_{\langle 2\rangle m@\langle 1\rangle m} \mid R''_{\langle 1\rangle m@\rangle\langle 2\rangle m}$.
Na osnovu pravila [R-REACT], [R-STRUCT] i kongruencije
$m \triangleright (P \mid Q) \equiv_R (\langle 1\rangle \cdot m \triangleright P \mid \langle 2\rangle \cdot m \triangleright Q)$, imamo

$$S = \langle\langle 2\rangle{\cdot}m, \alpha, Q'\rangle\langle 1\rangle{\cdot}m{\triangleright}P' \mid \langle\langle 1\rangle{\cdot}m, \bar{\alpha}, Q''\rangle\langle 2\rangle{\cdot}m{\triangleright}P'' \qquad \text{pri čemu je} \qquad S \equiv_R S'$$

$$\text{za procese} \quad R' = \langle *, \alpha, Q'\rangle\langle 1\rangle \cdot m \triangleright P' \quad \text{i} \quad R'' = \langle *, \bar{\alpha}, Q''\rangle\langle 2\rangle \cdot m \triangleright P''.$$

[L-PAR2-L] Proces R je oblika $R = m{\triangleright}(P \mid T)$. Za prelaz $\langle 1\rangle{\cdot}m{\triangleright}P \xrightarrow{\langle 1\rangle\cdot m:\alpha} R'$, primenom navedenog pravila, imamo da je $S' = R' \mid \langle 2\rangle \cdot m \triangleright T$.
Na osnovu pravila [R-STRUCT], [R-PAR] i kongruencije
$m \triangleright (P \mid Q) \equiv_R (\langle 1\rangle \cdot m \triangleright P \mid \langle 2\rangle \cdot m \triangleright Q)$ imamo

$$S = R' \mid \langle 2\rangle \cdot m \triangleright T \qquad \text{pri čemu je} \qquad S \equiv_R S'.$$

[L-PAR2-D] Proces R je oblika $R = m{\triangleright}(P \mid T)$. Za prelaz $\langle 2\rangle{\cdot}m{\triangleright}T \xrightarrow{\langle 2\rangle\cdot m:\alpha} R''$, primenom navedenog pravila, imamo da je $S' = \langle 1\rangle \cdot m \triangleright P \mid R''$.
Na osnovu pravila [R-STRUCT], [R-PAR], osobina komutativnosti paralelne kompozicije i kongruencije
$m \triangleright (P \mid Q) \equiv_R (\langle 1\rangle \cdot m \triangleright P \mid \langle 2\rangle \cdot m \triangleright Q)$ imamo

$$S = \langle 1\rangle \cdot m \triangleright P \mid R'' \qquad \text{pri čemu je} \qquad S \equiv_R S'.$$

■

5 Zaključak

U ovom master radu, prikazan je detaljno račun RCCS. U originalnom računu, predstavljenom u radu [4], korišćema je LTS semantika. Ovaj rad uvodi i drugu, redukcionu semantiku, i sadrđi dokaz da su dve operacione semantike ekvivalentne. Bavili smo se samo procesima koje je moguće vratiti na prethodno stanje. Međutim postoje procesi koji sadrže akcije koje ne mogu da se vrate unazad. Takve procese nazivamo ireverzibilnim. Ovakve prelaze je moguće uvesti u RCCS, definisanjem njihovih osobina i proširivanjem semantike sistema odgovarajućim pravilima.

U ovom radu pretpostavili smo osnovnu ideju za uvođenje reverzibilnosti u konkurentne sisteme. Pokazalo se da je moguće na osnovu istih principa uvesti reverzibilnost u mobilne sisteme. Ovakvi sistemi imaju mnogo veću ekspresivnost od CCS-a, jer njihove komponente imaju mogućnost da putem kanala kojima komuniciraju, pored podataka šalju i ime nekog kanala. Sposobnost mobilnih procesa da pamte svoje akcije i komunikacije sa drugim procesima postiže se na način koji smo opisali ranije, uvođenjem individualne memorije u kojoj se slažu informacije određenim redom. Jedan od takvih sistema je π-račun. Želja da se konstrukcija reverzibilnosti klasičnog π-računa [3], prenese na sve njegove modifikacije kao i potreba za kontrolisanjem takvih sistema, predstavljaju jedan od bitnijih zadataka iz oblasti konkurentnih sistema.

6 Literatura

[1] Artyom Petrosyan Sergio Ciliberto Raoul Dillenschneider Eric Lutz Antoine Bérut, Artak Arakelyan. Experimental verification of landauer?s principle linking information and thermodynamics. *Nature*, 483(7388):187–190, 2012.

[2] Luca Cardelli and Andrew D. Gordon. Mobile ambients. *Theor. Comput. Sci.*, 240(1):177–213, 2000.

[3] Ioana Cristescu, Jean Krivine, and Daniele Varacca. A compositional semantics for the reversible p-calculus. In *28th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2013, New Orleans, LA, USA, June 25-28, 2013*, pages 388–397, 2013.

[4] Vincent Danos and Jean Krivine. Reversible communicating systems. In *CONCUR 2004 - Concurrency Theory, 15th International Conference, London, UK, August 31 - September 3, 2004, Proceedings*, pages 292–307, 2004.

[5] Vincent Danos and Jean Krivine. Transactions in RCCS. In *CONCUR 2005 - Concurrency Theory, 16th International Conference, CONCUR 2005, San Francisco, CA, USA, August 23-26, 2005, Proceedings*, pages 398–412, 2005.

[6] Kohei Honda, Vasco Thudichum Vasconcelos, and Makoto Kubo. Language primitives and type discipline for structured communication-based programming. In *Programming Languages and Systems - ESOP'98, 7th European Symposium on Programming, Held as Part of the European Joint Conferences on the Theory and Practice of Software, ETAPS'98, Lisbon, Portugal, March 28 - April 4, 1998, Proceedings*, pages 122–138, 1998.

[7] Kohei Honda, Nobuko Yoshida, and Marco Carbone. Multiparty asynchronous session types. In *Proceedings of the 35th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL 2008, San Francisco, California, USA, January 7-12, 2008*, pages 273–284, 2008.

[8] Anthony S. Kosky. Formal models for concurrent communicating systems, 1991.

[9] Robin Milner. *A Calculus of Communicating Systems*, volume 92 of *Lecture Notes in Computer Science*. Springer, 1980.

[10] Robin Milner. *Communicating and mobile systems - the Pi-calculus*. Cambridge University Press, 1999.

[11] Iain Phillips, Irek Ulidowski, and Shoji Yuen. A reversible process calculus and the modelling of the ERK signalling pathway. In *Reversible Computation, 4th International Workshop, RC 2012, Copenhagen, Denmark, July 2-3, 2012. Revised Papers*, pages 218–232, 2012.

[12] Davide Sangiorgi and David Walker. *The Pi-Calculus - a theory of mobile processes*. Cambridge University Press, 2001.

Biografija

Doriana Medić rođena je 8. aprila 1988. godine u Apatinu. Završila je gimnaziju „Nikola Tesla” 2007. godine. Diplomirala je 2013. godine na Prirodno-matematičkom fakultetu, smer Inženjer matematike u Novom Sadu, sa temom diplomskog rada „Matematičko modeliranje HIV infekcije”. U oktobru 2014. godine upisuje master studije primenjene matematike na Fakultetu tehničkih nauka u Novom Sadu. Položila je sve ispite predviđene nastavnim planom i programom čime je stekla uslov za odbranu master rada.