

Erős kavics kezelésű kavics makró fatranszformátorok

A *"Pebble Macro Tree Transducers with Strong Pebble Handling"* című doktori
disszertáció magyar nyelvű tézis füzet

Muzamel Loránd

Szegedi Tudományegyetem TTIK
Informatikai Tanszékcsoport
Számítástudomány Alapjai Tanszék
Árpád tér 2., H-6720 Szeged, Hungary
muzamel@inf.u-szeged.hu

Témavezető: **Dr. Fülöp Zoltán**

SZTE Informatika Doktori Iskola

2010. május 24.

1. Előkészületek és a számítási modell

A fatranszformációk fontos szerepet töltenek be többek között a programozási nyelvek szintaxis-vezérelt szemantikájának modellezésében [Iro61, Knu68, Knu71, WM95], a fa struktúrájú adatokon dolgozó funkcionális programozásban [Vog91], valamint a különböző XML transzformációk [MN01, BMN02, MBPS05] és XML lekérdező nyelvek [Via01] specifikációiban és implementációiban.

A fatranszformátorok olyan számítási eszközök, amelyek a gyakorlatban előforduló fatranszformációk absztrakt tulajdonságait modellezik. Például, a makró fatranszformátorok [Eng80, Eng81, CF82, EV85] a szintaxis-vezérelt fordítások alkalmas modelljei, az attribútumos fatranszformátorok [Fül81, FV98] az attribútum nyelvtanokkal [Knu68, Knu71] megvalósított fordítások modelljei, míg az n -kavics fatranszformátorokkal [MSV03] jól tudjuk modellezni az XML lekérdezéseket és transzformációkat. Az n -kavics makró fatranszformátor [EM03] a makró fatranszformátor és az n -kavics fatranszformátor kombinációja, tehát alkalmas arra, hogy segítségével az n -kavics fatranszformátorok és a makró fatranszformátorok közötti kapcsolatot tanulmányozzuk. Az említett fatranszformátorok mindegyikének szemantikája az általa kiszámított fatranszformáció, vagyis absztrakt fák (rangolt fák) feletti kétváltozós reláció.

A disszertáció témája kavics makró fatranszformátorok vizsgálata. Egy (*erős kavics kezelésű*) n -kavics makró fatranszformátor egy olyan M , véges állapotú eszköz, amelyben az input és output ábécé elemein kívül maguk az állapotok is rangolt szimbólumok. M szabályai $\langle q, \sigma, b, j \rangle (y_1, \dots, y_m) \rightarrow \zeta$ alakúak, ahol q egy állapot, σ egy input szimbólum, b egy legfeljebb n hosszúságú bit vektor, j egy nemnegatív egész, y_1, \dots, y_m paraméter változók, továbbá ζ egy olyan rangolt fa, melynek csúcsai output szimbólumok, $\langle p, \varphi \rangle$ alakú állapot-utasítás párok és (levélben) y_1, \dots, y_m . Megjegyezzük, hogy $\langle p, \varphi \rangle$ rangja megegyezik a p állapot rangjával, valamint a φ utasítás *stay*, *up* vagy *down_i* alakú lehet.

M rendelkezik egy mutatóval, amely egy s input fa tetszőleges csúcsára mutathat és annak éllein mozoghat, továbbá M veremszerűen elhelyezheti-felszedheti az $1, \dots, n$ számú kavicsokat s csúcsain. Az s -en elhelyezett kavicsok bizonyos információt nyújthatnak M -nek, amelyet az figyelembe vehet a számítás során.

M az s input fán vándorolva mondatformák egy sorozatát számítja ki. *Mondatforma* alatt egy olyan ξ fát értünk, amely output szimbólumokból és $\langle p, (u, [u_1; \dots; u_l]) \rangle$ alakú konfigurációkból áll valamely $0 \leq l \leq n$ -re, ahol p a konfiguráció állapota, u egy s -beli csúcs, amelyre M mutatója mutat, végül u_1, \dots, u_l szintén s -beli csúcsok, amelyek azon helyeket jelölik, ahová az $1, \dots, l$ számú kavicsok el vannak helyezve. Egy $\langle p, (u, [u_1; \dots; u_l]) \rangle$ konfiguráció $(u, [u_1; \dots; u_l])$ részét *kavics konfigurációnak* mondjuk. Megjegyezzük, hogy $\langle p, (u, [u_1; \dots; u_l]) \rangle$ rangja megegyezik p rangjával.

A számítás egy speciális mondatformával, a $\langle q_0, (\varepsilon, []) \rangle$ *kezdőkonfigurációval* indul, ahol q_0 jelenti M kezdőállapotát, ε a gyökrere mutató pointert, $[]$ pedig kavics pozícióknak az üres listáját.

Ezek után M a következőképpen működik. Tegyük fel, hogy a számítás egy ξ mondatformánál tart. Tekintsük ξ -nek egy olyan v csúcsát, amelynek a címkéje egy

$\langle p, (u, [u_1; \dots; u_l]) \rangle$ alakú konfiguráció, és a ξ gyökerétől v -ig tartó út minden v -től különböző csúcsának output szimbólum (tehát nem konfiguráció) a címkéje. Vegyünk továbbá egy olyan $r : \langle q, \sigma, b, j \rangle (y_1, \dots, y_m) \rightarrow \zeta$, M -beli szabályt, feltéve ha létezik, amelyre teljesül, hogy

- az s input fa u csúcsa σ -val címkézett,
- a b bit vektor l hosszú és pontosan azokon az i indexeken 1, ahol $u_i = u$, továbbá
- az s fa u csúcsa j -edik fia az apjának ($j = 0$ esetben $u = \varepsilon$ maga a gyökér).

Ekkor M az r szabályt az alábbiakban leírt módon *alkalmazza* a ξ mondatforma v csúcsában a következő mondatforma kiszámítására.

1) Minden ζ -ban előforduló φ utasítás végrehajtódik az $(u, [u_1; \dots; u_l])$, kavics konfiguráción, amelyből eredményül keletkezett kavics konfigurációt $\varphi((u, [u_1; \dots; u_l]))$ -l jelöljük. Ha $\varphi = \text{down}_i$ ($\varphi = \text{up}$), akkor $\varphi((u, [u_1; \dots; u_l])) = (u', [u_1; \dots; u_l])$, amely azt jelöli, hogy a mutató elmozdult az u csúcs i -edik fiára (apjára), vagyis u' -re. Továbbá, ha $\varphi = \text{drop}$, akkor $\varphi((u, [u_1; \dots; u_l])) = (u, [u_1; \dots; u_l; u])$, amely azt jelenti, hogy M az $l + 1$ -ik kavicsot az u csúcsra helyezte el (feltéve ha $l < n$). Végül, ha $\varphi = \text{lift}$, akkor $\varphi((u, [u_1; \dots; u_l])) = (u, [u_1; \dots; u_{l-1}])$, vagyis M az l -edik kavicsot felszedte az u_l csúcsról (feltéve ha $l > 0$).

2) Tetszőleges, ζ -ban előforduló φ utasításra a $\varphi((u, [u_1; \dots; u_l]))$ kavics konfiguráció behelyettesítődik a φ minden előfordulásának helyére. Jelöljük ζ' -vel az így keletkezett fát.

3) Végül ζ' (amelynek a levelei lehetnek y_1, \dots, y_m , paraméter változók) másodrendű módon behelyettesítődik a ξ mondatforma $\langle p, (u, [u_1; \dots; u_l]) \rangle$ konfigurációval címkézett v csúcsába. A helyettesítés eredményezi a számítás következő mondatformáját.

Ha a keletkezett mondatforma nem tartalmaz konfigurációt, akkor ez az M kavics fatranszformátor s inputjának egy *output fája*. A tézisben τ_M -mel jelöljük az M által kiszámított fatranszformációt, amely az összes ily módon keletkező input-output fából álló párok halmaza.

Az 1) pontban írtak szerint kavicsot csak arra a pozícióra helyezhetünk, ahová a mutató mutat, míg kavicsot felemelni tetszőleges pozícióról lehet. Ez utóbbit *erős kavics kezelésnek* hívjuk [EH07, MSS06], szemben a *gyenge kavics kezeléssel*, ami azt jelenti, hogy kavicsot csak akkor emelhetünk fel, ha annak pozíciója egyben a mutató pozíciója is. Más szóval, csak olyan $(u, [u_1; \dots; u_l])$ kavics konfiguráción hajthatjuk végre a *lift* utasítást, amelyre $u = u_l$. Az erős kavics kezelésű kavics makró fatranszformátorok fogalmát [FM09]-ben vezettük be, míg a gyenge kavics kezelésűt a [EM03] cikkben definiálták. Mivel a disszertációban csak erős kavics kezelésű kavics makró fatranszformátorokkal foglalkozunk, a továbbiakban az „erős kavics” jelzőt elhagyjuk és röviden csak kavics makró fatranszformátort mondunk. Az n -kavics makró fatranszformátorokat az angol nyelvű *n-pebble macro tree transducer* elnevezés alapján *n-pmtt*-vel rövidítjük. Továbbá, *kavics makró fatranszformátor* (rövidítve: *pmtt*) alatt egy n -pmtt-t értünk valamely $n \geq 0$ -ra.

Azt mondjuk, hogy az M , n -pmtt egy

- *n-kavics fatranszformátor* (angolul: *n-pebble tree transducer*, rövidítve: *n-ptt*), ha tetszőleges Q -beli állapot rangja nulla;
- *stay-makró fatranszformátor* (angolul: *stay-macro tree transducer*, rövidítve: *smtt*), ha $n = 0$ továbbá a szabályok jobb oldalában nem szerepelnek *up* utasítások;
- *makró fatranszformátor* (angolul: *macro tree transducer*, rövidítve: *mtt*) ha ez egy *smtt* és a szabályok jobb oldalában nem szerepelnek *stay* utasítások;
- *felszálló fatranszformátor* (angolul: *top-down tree transducer*), ha ez egy *0-ptt* és egy *smtt* is egyben.

Továbbá *kavics fatranszformátornak* (rövidítve: *ptt*-nek) mondunk tetszőleges *n-ptt*-t, valamely $n \geq 0$ -ra.

Az *n-pmtt*-ekkel, *n-ptt*-ekkel, *smtt*-ekkel, *mtt*-ekkel és felszálló fatranszformátorokkal kiszámított fatranszformációk osztályát rendre *n-PMTT*, *n-PTT*, *sMTT*, *MTT* és *T* jelöli. A fenti osztályok determinisztikus, totális, kontext lineáris részosztályait rendre a *d*, *t* és *cl* prefixek írásával jelöljük. Például *n-dPMTT* jelenti a determinisztikus *n-pmtt*-ekkel kiszámolt fatranszformációk osztályát.

2. A disszertáció eredményeinek összefoglalása

2.1. Cirkularitás

Előfordulhat, hogy az M , *n-pmtt*-nek egy adott s input fán futó számítása nem terminál. Ez úgy lehetséges, ha M -nek van egy olyan rész-számítása, amely egy ξ mondatforma $\langle q, h \rangle \in C_{M,s}$ konfigurációjának alkalmazásával kezdődik és egy olyan ξ' mondatformában ér véget, amelynek valamely csúcsában ugyancsak alkalmazható a $\langle q, h \rangle \in C_{M,s}$ konfiguráció. A számítást ezen konfiguráció újra és újra történő aktivizálásával folytatva végtelen ciklusba kerülhetünk. (Ha M determinisztikus, akkor ez mindeképpen be is következik.) Ezt a jelenséget röviden cirkularitásnak nevezzük.

Három cirkularitási fogalmat vezetünk be: a gyenge cirkularitást, a cirkularitást és az erős cirkularitást. Mint ahogy az elnevezések is mutatják, a cirkularitási tulajdonságok között a következő összefüggések állnak fenn.

4.3. Lemma ([FM08] 14. Lemma) Tetszőleges M *n-pmtt* esetén:

- Ha M erősen cirkuláris, akkor M cirkuláris.
- Ha M cirkuláris, akkor M gyengén cirkuláris. ◇

A fenti három cirkularitási fogalom közül a legtermészetesebb (és a számítástudományban a legkézenfekvőbb) az erős cirkularitás, mivel azt írja le, hogy M a kezdő konfigurációból kiindulva végtelen ciklusba eshet. A másik két (gyengébb) cirkularitási fogalmat technikai okok miatt vezettük be.

Az erősen cirkuláris, cirkuláris és gyengén cirkuláris tulajdonságok negáltjait rendre *nem erősen cirkuláris*, *nemcirkuláris* és *nem gyengén cirkuláris* tulajdonságoknak mondjuk.

2.2. Kompozíciós, dekompozíciós eredmények

A disszertáció általunk legjelentősebbnek ítélt eredményei az n -kavics makró fatranszformációk kompozícióira és dekompozícióira vonatkoznak. Általában a fatranszformációk kompozíció elméletében azt vizsgáljuk, hogy kettő (vagy több) fatranszformátor által kiszámított fatranszformáció kompozícióját ki lehet-e számolni egyetlen fatranszformátorral. A kompozíciók közvetlenül megjelennek az informatikai alkalmazások számos területén: a többmenetes fordítóknak, a funkcionális programozási nyelvek deforesztációjának [Küh98, Voi02] modelljeként és nem utolsósorban az XML adatbázisok iterált lekérdezéseinek megvalósításában.

Egy adott fatranszformátor által kiszámolt fatranszformáció dekompozícióján azt a vizsgálatot értjük, hogy lehet-e a fatranszformációt kettő vagy több egyszerűbb fatranszformátor által kiszámolt fatranszformáció kompozíciójaként megvalósítani.

Megadunk egy-egy példát mind a kompozícióra és a dekompozícióra: [EV85]-ben és [FV98]-ban megmutatták, hogy (1) a totális determinisztikus felszálló fatranszformációk [Eng75, Rou70] és a totális determinisztikus yield fatranszformációk kompozíciói kiszámolhatók totális és determinisztikus makró fatranszformátorokkal (kompozíciós eredmény), és fordítva, hogy (2) tetszőleges totális determinisztikus makró fatranszformáció egyenlő egy totális determinisztikus felszálló fatranszformáció és egy totális determinisztikus yield fatranszformáció kompozíciójával (dekompozíciós eredmény). Az (1)-(2) eredményeket összerakva kapjuk a $dtMAC = dtTOP \circ dtYIELD$ eredményt, ahol a jelölések egyértelműen kiderülnek a szövegkörnyezetből. Megjegyezzük, hogy a fenti karakterizációnak egy fontos következménye, hogy a totális determinisztikus makró fatranszformációk és az attribútumos fatranszformációk kompozícióra történő lezártjai megegyeznek (Lásd [FV98] 6. Fejezet).

Mivel a pmtt-eket ugyanúgy kapjuk ptt-kből, mint a makró fatranszformátorokat a felszálló fatranszformátorokból (ti. paraméterek alkalmazásával), joggal vetődik fel a kérdés, hogy vajon a makró fatranszformátorokra kapott „yield-szerű” kompozíciós és dekompozíciós eredmények átültethetők-e pmtt-kre. (Az a tény, hogy makró attribútumos fatranszformátorokra is léteznek ilyen „yield-szerű” kompozíciós és dekompozíciós eredmények, lásd [KV94]-t és [FV98] 7.29 Tételét, megerősíti bennünk a sejtést, hogy pmtt-kre is léteznek hasonló összefüggések.)

Először bebizonyítjuk, hogy tetszőleges M , n -ptt és $yield_g$, yield fatranszformáció esetén (ahol g egy levél szimbólumokból fa halmazokba történő leképezés, amely egyértelműen meghatározza a yield fatranszformációt) meg tudunk konstruálni egy olyan M' , n -pmtt-t, amelyre $\tau_{M'} = \tau_M \circ yield_g$ teljesül. Ha M és $yield_g$ determinisztikusak (totálisak), akkor ugyanúgy M' is determinisztikus (totális) lesz.

6.1. Lemma ([FM08] 41. Lemma) Tetszőleges $n \geq 0$ esetén $n\text{-}PTT \circ YIELD \subseteq n\text{-}PMTT$. \diamond

A fenti lemmához tartozó konstrukció a korábban elmített totális determinisztikus felszálló fatranszformációk és totális determinisztikus yield fatranszformációk kompozíciójára alkalmazott konstrukció egy általánosítása.

Utána kavics makró fatranszformációkat dekomponálunk. Nevezetesen, tetszőleges M , n -pmtt-hez effektíven megadunk olyan M' , n -ptt-t és egy $yield_g$, yield fatranszformációt, amelyre $\tau_M = \tau_{M'} \circ yield_g$ teljesül. Így kapjuk az alábbi eredményt.

7.9. Következmény ([FM09] 4.11. Következmény) Tetszőleges $n \geq 0$ -ra n -PMTT $\subseteq n$ -PTT \circ YIELD. \diamond

A fenti dekompozíció és a 6.1. Lemmában igazolt n -PTT \circ YIELD $\subseteq n$ -PMTT kompozíciót összekapcsolva az alábbi jellemzést kapjuk.

7.10. Tétel ([FM09] 4.12 Következmény) Tetszőleges $n \geq 0$ -ra n -PMTT = n -PTT \circ YIELD. \diamond

A 7.9. Lemma bizonyításában a determinisztikus totális makró fatranszformátorokra alkalmazott, [EV85, FV98]-ban ismertetett módszereket követjük. A konstrukciónk tetszőleges (nemdeterminisztikus, nem totális, sőt, nemcirkuláris) pmtt-kre működik. Gyengesége viszont, hogy a megkonstruált M' ptt erősen cirkuláris és nem determinisztikus (még akkor is, ha a kiindulási M determinisztikus és semelyik értelemben nem cirkuláris). Ezen problémát a disszertáció 7.3. Szekciójában részletesebben is tárgyaljuk.

Évégett megvizsgáljuk, vajon létezik-e olyan másik, M' -t és $yield_g$ -t előállító (és szemantikus is helyes) dekompozíciós konstrukció, amely megőrzi M bizonyos jó tulajdonságait (pl. a determinisztikusságot és a nemcirkularitást), legalábbis a pmtt-k valamely természetes részosztályára vonatkozóan. Kiderül, hogy találunk ilyen alternatív konstrukciót. Az első megszorítás, amit M -re teszünk, hogy az determinisztikus vagy ún. kontext-lineáris. Ezen megszorítások mellett a szemantikus helyesség biztosítása végett még annak is teljesülnie kell, hogy M' nemcirkuláris. (Az alábbi lemmákban M' és $yield_g$ az alternatív konstrukcióval M -ből kapott ptt-t és yield fatranszformációt jelentik.)

8.4. Lemma ([FM08] 30. Lemma) Ha M determinisztikus és M' nemcirkuláris, akkor $\tau_M = \tau_{M'} \circ yield_g$ teljesül. \diamond

8.6. Lemma ([FM08] 32. Lemma) Ha M kontext-lineáris és M' nemcirkuláris, akkor $\tau_M = \tau_{M'} \circ yield_g$ teljesül. \diamond

A következő, amit megvizsgálunk az, hogy M -re milyen alkalmas szintaktikus megszorításokat kell tenni ahhoz, hogy M' nemcirkuláris legyen. Egy triviális megszorítás az, ha M egy makró fatranszformátor. Ebben az esetben M' egy felszálló fatranszformátor (lásd [EV85]), amely nem lehet cirkuláris. Tehát a 8.4. és 8.6. Lemmák teljesülnek M -re. Ebben az esetben az [EV85]-beli $dtMTT \subseteq dtT \circ dtYIELD$ dekompozíciónak új változatait kapjuk meg, nevezetesen a $dMTT \subseteq dtT \circ dYIELD$ és $clMTT \subseteq tT \circ dYIELD$ tartalmazásokat.

Egy másik természetes megszorítás M -re nézve az lenne, hogy M nemcirkuláris. Sajnos ez a feltétel azonban nem elégséges ahhoz, hogy M' nemcirkuláris legyen, amit a disszertációban egy ellenpéldával is alátámasztottunk.

Megfelelő viszont az a megszorítás (ami miatt a gyengén cirkuláris fogalmat bevezettük), hogy M nem gyengén cirkuláris. Megmutatjuk, hogy ha M nem gyengén cirkuláris (nwc), akkor M' nemcirkuláris (nc). Tehát azt kapjuk, hogy tetszőleges nem gyengén cirkuláris és determinisztikus (vagy nem gyengén cirkuláris és kontext-lineáris) M esetén a kapott M' nem cirkuláris és determinisztikus (vagy nem cirkuláris és kontext-lineáris) és a $\tau_M = \tau_{M'} \circ \text{yield}_g$ dekompozíciós egyenlőség teljesül.

8.10. Következmény ([FM08] 37. Következmény)

$$n\text{-dPMTT}_{nwc} \subseteq n\text{-dtPTT}_{nc} \circ dYIELD \text{ és}$$

$$n\text{-clPMTT}_{nwc} \subseteq n\text{-tPTT}_{nc} \circ dYIELD. \quad \diamond$$

További eredmények következnek abból, hogy tetszőleges parciális yield fatranszformáció kiszámítható egy 0-ptt-vel (ha a yield fatranszformáció determinisztikus, akkor egy determinisztikus nemcirkuláris 0-ptt-vel). Tehát tetszőleges nem gyengén cirkuláris és determinisztikus (kontext-lineáris) n -kavics makró fatranszformáció előáll egy nemcirkuláris determinisztikus (kontext-lineáris) n -kavics fatranszformáció és egy nemcirkuláris determinisztikus 0-kavics fatranszformáció kompozíciójaként.

8.11. Tétel ([FM08] 38. Tétel)

$$n\text{-dPMTT}_{nwc} \subseteq n\text{-dtPTT}_{nc} \circ 0\text{-dPTT}_{nc} \text{ és}$$

$$n\text{-clPMTT}_{nwc} \subseteq n\text{-tPTT}_{nc} \circ 0\text{-dPTT}_{nc}. \quad \diamond$$

2.3. n -ptt-k szimulációja $(n - 1)$ -pmtt-kkel

A disszertáció következő témája egy, az [EM03] 8. Szekciójában felvetett nyitott kérdés megoldása. Nevezetesen, bebizonyítjuk, hogy ha $n \geq 1$, akkor tetszőleges M , (determinisztikus) n -ptt szimulálható egy M' , (determinisztikus) $(n - 1)$ -pmtt-vel.

9.6. Tétel ([FM09] 5.7. Tétel) Tetszőleges $n \geq 1$ -re, $n\text{-PTT} \subseteq (n - 1)\text{-PMTT}$ és $n\text{-dPTT} \subseteq (n - 1)\text{-dPMTT}$. \diamond

A fenti eredmény bizonyításában alkalmazott konstrukció alapötlete, hogy az utolsó kavics működését makró hívásokkal tudjuk helyettesíteni.

2.4. További eredmények

A 7.10. és 9.6. Tételeknek további következményeit az alábbi tételben foglaljuk össze.

10.3. Tétel ([FM09] 6.5. Tétel) Tetszőleges $n \geq 0$ esetén

- (1) $n\text{-PMTT} \subseteq 0\text{-PTT} \circ YIELD^{n+1}$,
- (2) $n\text{-PMTT} \subseteq 0\text{-PTT}^{n+2}$,
- (3) $n\text{-PMTT} \subseteq sMTT^{n+2}$.

Továbbá, tetszőleges $n \geq 1$ esetén

$$\begin{aligned} (4) \quad n\text{-}PTT &\subseteq 0\text{-}PTT \circ YIELD^n, \\ (5) \quad n\text{-}PTT &\subseteq 0\text{-}PTT^{n+1}, \\ (6) \quad n\text{-}PTT &\subseteq sMTT^{n+1}. \end{aligned}$$

◇

Megjegyezzük, hogy a fenti dekompozíciós eredmények gyenge kavics használat mellett már igazolás nyertek az [EM03] 10. és 35. Tétéleiben és a 8. Szekciójában. Ugyanakkor úgy tűnik, hogy az [EM03] dekompozíciós eredményeinek bizonyításához segítségül használt EncPeb fatranszformáció erősen kihasználja a gyenge kavics kezelést és nem használható a disszertációban vizsgált erős kavics kezelésű modelljeinkhez.

A fenti dekompozíciós eredményeinknek a további alkalmazásait nyertük.

[EM03] 27. Lemmájában bebizonyították, hogy $sMTT \subseteq MON \circ MTT$, ahol MON az ún. *monadikus beszúrásoknak* az osztálya. Továbbá, az is közismert, hogy mind a monadikus beszúrások, mind a makró fatranszformációk inverzei megőrzik a regularitást. Ebből egyenesen következik, hogy a stay-makró fatranszformációk (kompozícióinak) inverzei is megőrzik a fanyelvek regularitását. Tehát, a 10.3. Tétel $n\text{-}PMTT \subseteq sMTT^{n+2}$ dekompozíciós eredményéből következnek az alábbi eredmények.

10.4. Tétel ([FM09] 6.7. Tétel) A kavics makró fatranszformációk (kompozícióinak) inverzei effektíven megőrzik a regularitást. ◇

10.5. Következmény ([FM09] 6.8. Következmény) A kavics makró fatranszformációk (kompozícióinak az) értelmezési tartománya effektíven reguláris. ◇

A fenti eredményt jól alkalmazhatjuk a kavics makró fatranszformációkra vonatkozó típus ellenőrzési probléma eldöntésére. Röviden összefoglalva, az *XML transzformációk típus ellenőrzési problémáján* azt a kérdést értjük, hogy egy adott XML transzformáció input DTD-je kielégíti-e az output DTD-t.

Formálisan, a pmtt-kre vonatkozó típus ellenőrzési probléma [EM03, MBPS05] a következő eldönthetőségi kérdést jelenti. Adottak az L_{in} és L_{out} reguláris fanyelvek, valamint egy τ kavics makró fatranszformáció. Igaz-e, hogy tetszőleges $s \in L_{in}$ input fa esetén s -nek a τ melletti összes képe L_{out} -ban van, vagyis igaz-e $\tau(s) \subseteq L_{out}$?

Felhasználva a 10.3. Tételben kapott $n\text{-}PMTT \subseteq sMTT^{n+2}$ dekompozíciót és [EM03] 44. Következményét, amely kimondja, hogy a stay-makró fatranszformációk kompozícióinak a típus ellenőrzési problémája eldönthető, az alábbi eredményeket kapjuk.

10.6. Tétel ([FM09] 6.9. Tétel) A kavics makró fatranszformációkra vonatkozó típus ellenőrzési probléma eldönthető. ◇

További eldönthetőségi eredményeink a pmtt-k cirkularitásával kapcsolatosak. Kihasználtuk a 10.5. Következményt, amely kimondja, hogy a kavics makró fatranszformációk értelmezési tartományai effektíven regulárisak. Ebből (különböző algoritmusok megkonstruálásával) következnek az alábbi pozitív eldönthetőségi eredmények.

10.8. Tétel ([FM08] 20. Tétel) A pmtt-k erősen cirkularitási problémája eldönthető. ◇

10.9. Tétel ([FM08] 21. Tétel) A pmtt-k cirkularitási problémája eldönthető. \diamond

10.11. Következmény ([FM08] 36. Következmény) A pmtt-k gyengén cirkularitási problémája eldönthető \diamond

Végezetül megvizsgáljuk a kavics fatranszformációk értelmezési tartományait. Bevezetjük az n -kavics *alternáló fabejáró automata* (angolul: *n-pebble alternating tree-walking automaton*, rövidítve: *n-patwa*) fogalmát, amely az n -ptt-k működését modellezi az input fán. Leginkább az n -patwa-k determinisztikus és termináló speciális eseteit vizsgáljuk, amelyek által felismert fanyelvek osztályát n -dPATWA_{nl}-lél jelöljük. Megjegyezzük, hogy az nl alsóindex az angol „non-looping” kifejezés alapján a mindig termináló automaták által meghatározott részosztályt jelenti.

Nyilvánvaló, hogy a determinisztikus és termináló n -patwa-k által felismert fanyelvek osztálya megegyezik a determinisztikus és nem erősen cirkuláris n -kavics fatranszformációk értelmezési tartományainak az osztályával: n -dPATWA_{nl} = dom(n -dPTT_{nsc}).

A fő eredményünk az, hogy a determinisztikus és nem erősen cirkuláris n -kavics fatranszformációk értelmezési tartományai valódi tartalmazási hierarchiát képeznek n -re vonatkozóan. Ez egyenesen következik az alábbi tételből és az n -dPATWA_{nl} = dom(n -dPTT_{nsc}) összefüggésből.

10.25. Tétel ([Muz08] 5.7. Tétel) Tetszőleges $n \geq 0$ esetén, n -dPATWA_{nl} \subset $(n + 1)$ -dPATWA_{nl}. \diamond

Tehát a hierarchia tételünk a következő.

10.26. Tétel

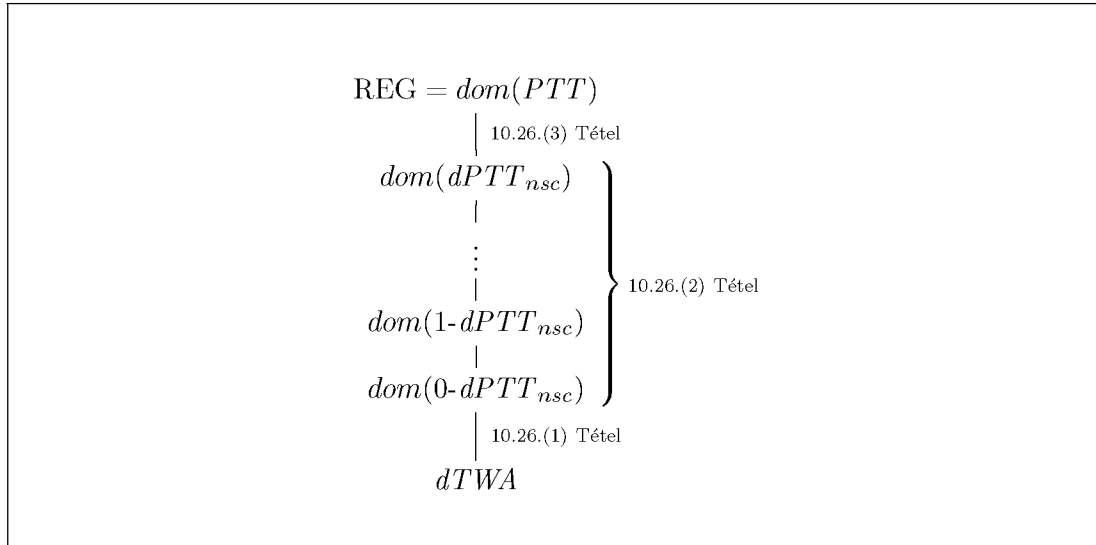
$$(1) \quad dTWA \subset \text{dom}(0\text{-dPTT}_{nsc}).$$

$$(2) \quad \text{Tetszőleges } n \geq 0 \text{ esetén } \text{dom}(n\text{-dPTT}_{nsc}) \subset \text{dom}((n + 1)\text{-dPTT}_{nsc}).$$

$$(3) \quad \text{dom}(dPTT_{nsc}) \subset \text{REG}. \quad \diamond$$

A fenti tételben $dTWA$ jelenti a determinisztikus fabejáró automatákkal felismert nyelvek osztályát, REG pedig a reguláris fanyelvek osztályát. Megjegyezzük, hogy a 10.26. Tétel [Muz08]-ban nem ebben a formában került publikálásra, mivel ott nem ptt-ket, hanem patwa-kat vizsgáltunk. Tehát az eredmény ebben a formában a disszertációban jelenik meg először.

Az 1. ábrán szemléltetjük a $(\text{dom}(n\text{-dPTT}_{nsc}) \mid n \geq 0)$ valódi hierarchiát és annak kapcsolatát REG-gel és $dTWA$ -val.



1. ábra. $dom(PTT)$ bizonyos részosztályainak a tartalmazási diagrammja.

3. A disszertációban alkalmazott bizonyítási módszerek

Tételeink bizonyítására a fatranszformációk elméletében használatos standard eszközöket használjuk. Egyik ilyen a kompozíciós technika, amely azt jelenti, hogy találunk egy olyan gépet (fatranszformátort), amely más gépek által kiszámolt fatranszformációk kompozícióját számítja ki. Ezzel a módszerrel be tudjuk látni fatranszformációk kompozícióra való zártságát. Másik módszerünk a dekompozíció, amely esetében azt vizsgáljuk, hogy egy adott fatranszformátor által kiszámított fatranszformációt meg lehet-e valósítani kettő (vagy több) egyszerűbb gép által kiszámított fatranszformáció kompozíciójaként. Mindkét módszer alapvető információt nyújthat a vizsgálandó fatranszformátorok működésével, tulajdonságaival kapcsolatban.

A cirkularitásra vonatkozó eldöntési problémáinkat a visszavezetés módszerével oldottuk meg: visszavezettük őket a reguláris fanyelvekkel kapcsolatos eldönthető problémákra. A disszertáció végén Hasse-diagram megadásával igazoltuk az adott kavics fatranszformációk értelmezési tartományainak a tartalmazási hierarchiáját.

A disszertáció fő eredményeit a makró fatranszformátorokra vonatkozó *szimultán indukciónak* (lásd [Eng75, FV98]) egy, az alábbi módon kiterjesztett változatával bizonyítottuk be:

Legyenek $K : (\mathbb{N} - \{0\}) \rightarrow \{true, false\}$ és $L : \mathbb{N} \rightarrow \{true, false\}$ predikátumok. Tetszőleges $l \geq 1$ -re azt mondjuk, hogy $K[l]$ teljesül, ha $K(1) = true, \dots, K(l) = true$ és azt mondjuk, hogy K teljesül, ha minden $l \geq 1$ -re, $K[l]$ teljesül. Ugyanazt a terminológiát használjuk minden $l \geq 0$ esetén, $L[l]$ -re, valamint L -re.

A szimultán indukció egy olyan bizonyítási módszer, amellyel bizonyos konkrét K és L esetén be tudjuk látni, hogy K és L teljesül.

Tekintsük ugyanis az alábbi három állítást:

IB: $L(0)$ teljesül.

IS1: Tetszőleges $l \geq 0$ -ra, ha $L[l]$ teljesül, akkor $K(l + 1)$ is teljesül.

IS2: Tetszőleges $l \geq 1$ -re, ha $K[l]$ teljesül, akkor $L(l)$ is teljesül.

A szimultán indukció elve azt mondja ki, hogy ha az IB, IS1 és IS2 állítások teljesülnek, akkor K és L is teljesül. A szimultán indukciónak ezen verziója alkalmasnak bizonyult a kavics makró fatranszformátorokkal kapcsolatos kompozíciós és dekompozíciós konstrukciók helyességének az igazolását, mivel K és L predikátumok előfordulásaiként meg tudtunk fogalmazni a helyesség igazolásához szükséges állításokat.

Gyakran alkalmaztuk még bizonyítási módszerként a fák struktúrájára vonatkozó indukciót.

4. A disszertációban hivatkozott saját publikációk

- [FM08] Z. Fülöp and L. Muzamel. Circularity and Decomposition Results for Pebble Macro Tree Transducers. *Journal of Automata, Languages and Combinatorics*, 13(1):3–44, 2008.
- [FM09] Z. Fülöp and L. Muzamel. Pebble Macro Tree Transducers with Strong Pebble Handling. *Fundam. Inf.*, 89(2-3):207–257, 2009.
- [Muz08] L. Muzamel. Pebble Alternating Tree-Walking Automata and Their Recognizing Power. *Acta Cybernetica*, 18(3):427–450, 2008.

Hivatkozások

- [BMN02] G. J. Bex, S. Maneth, and F. Neven. A formal model for an expressive fragment of XSLT. *Information Systems*, 27:21–39, 2002.
- [CF82] B. Courcelle and P. Franchi-Zannettacci. Attribute grammars and recursive program schemes I–II. *Theoret. Comput. Sci.*, 17:163–191, 235–257, 1982.
- [EH07] Joost Engelfriet and Hendrik Jan Hoogeboom. Automata with nested pebbles capture first-order logic with transitive closure. *Logical Methods in Computer Science*, 3(2), 2007.
- [EM03] J. Engelfriet and S. Maneth. A Comparison of Pebble Tree Transducers with Macro Tree Transducers. *Acta Informatica*, 39:613–698, 2003.
- [Eng75] J. Engelfriet. Bottom-up and top-down tree transformations — A comparison. *Math. Systems Theory*, 9:198–231, 1975.
- [Eng80] J. Engelfriet. Some open questions and recent results on tree transducers and tree languages. In R.V. Book, editor, *Formal language theory: perspectives and open problems*, pages 241–286. New York, Academic Press, 1980.
- [Eng81] J. Engelfriet. Tree transducers and syntax-directed semantics. Technical Report Memorandum 363, Technische Hogeschool Twente, March 1981. also in: Proceedings of the Colloquium on Trees in Algebra and Programming (CAAP 1992), Lille, France 1992.
- [EV85] J. Engelfriet and H. Vogler. Macro tree transducers. *J. Comput. System Sci.*, 31:71–146, 1985.
- [FM08] Z. Fülöp and L. Muzamel. Circularity and Decomposition Results for Pebble Macro Tree Transducers. *Journal of Automata, Languages and Combinatorics*, 13(1):3–44, 2008.
- [FM09] Z. Fülöp and L. Muzamel. Pebble Macro Tree Transducers with Strong Pebble Handling. *Fundam. Inf.*, 89(2-3):207–257, 2009.
- [Fül81] Z. Fülöp. On attributed tree transducers. *Acta Cybernet.*, 5:261–279, 1981.
- [FV98] Z. Fülöp and H. Vogler. *Syntax-Directed Semantics — Formal Models Based on Tree Transducers*. Monographs in Theoretical Computer Science, An EATCS Series. Springer-Verlag, 1998.
- [Iro61] E. T. Irons. A syntax directed compiler for ALGOL 60. *Comm. of the ACM*, 4:51–55, 1961.
- [Knu68] D. E. Knuth. Semantics of context-free languages. *Math. Systems Theory*, 2:127–145, 1968.
- [Knu71] D. E. Knuth. Semantics of context-free languages: Correction. *Math. Systems Theory*, 5(1):95–96, 1971. Errata of [Knu68].

- [Küh98] A. Kühnemann. Benefits of Tree Transducers for Optimizing Functional Programs. In V. Arvind and R. Ramanunjam, editors, *Foundations of Software Technology and Theoretical Computer Science*, volume 1530 of *LNCS*, pages 146–157. Springer-Verlag, 1998.
- [KV94] A. Kühnemann and H. Vogler. Synthesized and inherited functions — a new computational model for syntax-directed semantics. *Acta Inform.*, 31:431–477, 1994.
- [MBPS05] S. Maneth, A. Berlea, T. Perst, and H. Seidl. XML Type Checking with Macro Tree Transducers. In *Proceedings of the 24th ACM Symposium on Principles of Database Systems (PODS' 05)*, pages 283–294. ACM Press, 2005.
- [MN01] S. Maneth and F. Neven. Recursive structured document transformation. In R. Connor and R. Mendelzon, editors, *Research issues in structured and semistructured database programming - Revised papers DBLP 99*, volume 1949 of *Lect. Notes Comput. Sci.*, pages 80–98. Springer-Verlag, 2001.
- [MSS06] A. Muscholl, M. Samuelides, and L. Segoufin. Complementing deterministic tree-walking automata. *Information Processing Letters*, 99:33–39, 2006.
- [MSV03] T. Milo, D. Suciu, and V. Vianu. Typechecking for XML transformers. *J. of Comput. Syst. Sci.*, 66:66–97, 2003.
- [Muz08] L. Muzamel. Pebble Alternating Tree-Walking Automata and Their Recognizing Power. *Acta Cybernetica*, 18(3):427–450, 2008.
- [Rou70] W.C. Rounds. Mappings and grammars on trees. *Math. Systems Theory*, 4:257–287, 1970.
- [Via01] V. Vianu. A Web Odyssey: From Codd to XML. In *Proceedings of the 20th ACM Symposium on Principles of Database Systems (PODS' 01)*, pages 1–15. ACM Press, 2001.
- [Vog91] H. Vogler. Functional description of the contextual analysis in block-structured programming languages: a case study of tree transducers. *Science of Comput. Prog.*, 16:251–275, 1991.
- [Voi02] J. Voigtländer. Conditions for Efficiency Improvement by Tree Transducer Composition. In Sophie Tison, editor, *13th International Conference on Rewriting Techniques and Applications, Copenhagen, Denmark, Proceedings*, volume 2378 of *LNCS*, pages 222–236. Springer-Verlag, July 2002.
- [WM95] R. Wilhelm and D. Maurer. *Compiler Design*. Addison-Wesley, 1995.