The Composition Method

Wolfgang Thomas

RWTH AACHEN

Francqui Lecture, Mons, April 2013

Mastering compositions

Overview

1. Motivation
2. m-equivalence and the EF-game
3. Applications
4. m-types
5. Monadic types

Motivation

Composition and Decomposition

General problem:

How to know what is true in a composed system if one knows what is true in the components?

Essential question in verification.

Here we consider two kinds of compositions:

- **Ordered sums (e.g., concatentation of word models)**
- **Products**

Recall Automata

Given a DFA $\mathcal{A} = (Q, \Sigma, q_0, \delta, F)$

“If we know the behaviour of $\mathcal{A}$ on u and on v then we know the behaviour on uv.”

We capture “behaviour” by the state transformations over Q realized by words.

For $u \in \Sigma^*$ define $u^{\mathcal{A}} : Q \to Q$ by $u^{\mathcal{A}}(q) = \delta(q, u)$

The set $\{u^{\mathcal{A}} \mid u \in \Sigma^*\}$ of state transformations forms a finite monoid with composition and identity $\varepsilon^{\mathcal{A}}$.

$(uv)^{\mathcal{A}} = u^{\mathcal{A}} \circ v^{\mathcal{A}}$

Composition on Level of Automata

$\mathcal{A}$ accepts uv iff

$$\bigvee_{p \in Q} (u^{\mathcal{A}}(q_0) = p \ \wedge \ v^{\mathcal{A}}(p) \in F)$$

Nondeterministic version: Use relation

$\langle u \rangle^{\mathcal{A}} = \{(p,q) \mid \mathcal{A} : p \xrightarrow{u} q\}$

$\mathcal{A}$ accepts uv iff

$$\bigvee_{p \in Q, r \in F} \langle u \rangle^{\mathcal{A}}(q_0, p) \wedge \langle v \rangle^{\mathcal{A}}(p, r)$$

m-Equivalence and the EF-Game

Composition in Logic

How to obtain information whether $uv \models \varphi$ from knowledge about u and v?

Solution:

When dealing with a formula φ do not look into φ but consider all formulas of the same quantifier complexity.

More precisely:

The quantifier-depth qd(φ) of a formula φ is the maximal number of nested quantifiers in φ.

The quantifier alternation depth qad(φ) of φ is the number of blocks of existential resp. universal quantifiers in the prenex normal form of φ.

Format of Models

Fix a signature with unary relation symbols $Q_1, \ldots, Q_k$ and binary relation symbols $R_1, \ldots, R_\ell$.

Obtain relational structures $\mathcal{S} = (S, Q_1^S, \ldots, Q_k^S, R_1^S, \ldots, R_l^S)$

Satisfaction relation: $(\mathcal{S}, \overline{s}) \models \varphi(\overline{x})$

Special case: Word models ovder $\Sigma = \{a, b\}$:

$\underline{w} = (\mathbf{dom}(w), <, \mathbf{Suc}, Q_a, Q_b)$

Example $w = aaba$: $\mathbf{dom}(w) = \{1, 2, 3, 4\}$,
$Q_a = \{1, 2, 4\}, Q_b = \{3\}$.

Ordered Sums

For $i \in I$ we are given relational structures $\mathcal{M}_i = (M_i, R_1^i, \ldots)$ of the same signature

First focus on $I = \{1, 2\}$.

$\mathcal{M}_1 + \mathcal{M}_2$ is the structure $\mathcal{M} = (M_1 \cup M_2, R_1^1 \cup R_1^2, \ldots)$

If M_i is ordered by $<^i$, then the ordered sum has the following ordering $<$:

$a < b$ iff a, b belong to the same M_i and $a <^i b$, or
$a \in M_1, b \in M_2$

Similarly for arbitrary orderings $(I, <^I)$.

m-Equivalence

Two structures $(\mathcal{S}, \overline{s})$ and $(\mathcal{T}, \overline{t})$ are m-equivalent (short: $(\mathcal{S}, \overline{s}) \equiv_m (\mathcal{T}, \overline{t})$) if

$$(\mathcal{S}, \overline{s}) \models \varphi(\overline{x}) \iff (\mathcal{T}, \overline{t}) \models \varphi(\overline{x})$$

for all formulas $\varphi(\overline{x})$ of quantifier-depth $\leq m$.

The m-equivalence classes are also called m-types.

Plan:

1. **Find a way to show that two structures are m-equivalent.**
2. **Present some applications**
3. **Introduce descriptions of the m-types**

A. Ehrenfeucht

R. Fraïssé

Ehrenfeucht-Fraïssé game

allows to verify $(\mathcal{S}, \overline{s}) \equiv_m (\mathcal{T}, \overline{t})$.

A game position is a partial isomorphism:

a finite relation $\{(s_1, t_1), \ldots, (s_n, t_n)\} \subseteq S \times T$ denoted by $\overline{s} \mapsto \overline{t}$,

which is injective and preserves all relations Q^S, R^S under consideration:

$$s \in Q^S \iff p(s) \in Q^T$$

$$\text{and } (s, s') \in R^S \iff (p(s), p(s')) \in R^T$$

Game $G_m((\mathcal{S},\overline{s}),(\mathcal{T},\overline{t}))$

played between two players called Spoiler and Duplicator $(\mathcal{S},\overline{s})$ and $(\mathcal{T},\overline{t})$.

There are m rounds.

The initial configuration is $\overline{s} \mapsto \overline{t}$.

Given a configuration r, a round is composed of two moves:

first Spoiler picks an element s from S or t from T, and then Duplicator reacts by choosing an element in the other structure, i.e. by choosing some t from T, resp. some s from S.

The new configuration is $r \cup \{(s,t)\}$.

After m rounds, Duplicator has won if the final configuration is a partial isomorphism (otherwise Spoiler has won).

Example 1

Let $u = aabaacaa$ and $v = aacaabaa$

Consider $G_2(\underline{u}, \underline{v})$

(including the linear ordering $<$)

Duplicator looses:

Spoiler can pick the u-positions with the letters b and c,

whence Duplicator can only respond by picking the positions with b and c in v, in order to preserve the relations Q_b and Q_c;

but then the order between the positions $<$ is not preserved.

Consider $\exists x \exists y (x < y \wedge Q_b(x) \wedge Q_c(y))$

Example 2

as before, however with successor relation Suc only in word models, besides Q_a, Q_b, Q_c.

$u = aabaacaa$ **and** $v = aacaabaa$

Duplicator wins.

If Spoiler picks a position with b or c or a position adjacent to one of them, Duplicator reacts accordingly in the other word;

Otherwise Duplicator reacts by corresponding positions.

Consider, e.g., $\exists x \exists y(\mathbf{Suc}(x, y) \wedge Q_a(x) \wedge Q_b(y))$

Example 3

Word models with order but without successor,

singleton alphabet $\{a\}$.

Format: $(\mathbf{dom}(w), <, Q_a)$.

Duplicator wins $G_2(aaa, a^n)$ for any $n \geq 3$:

In the first round, Spoiler may pick a first position, a last position, or a non-border position in one of the two words, and Duplicator reacts accordingly.

This allows Duplicator also to respond correctly (i.e., order-preserving) in the second round.

$G_3(a^i, a^j)$

Here after the first round we get the situation

$a^i = a^{i_1} a a^{i_2}$ and $a^j = a^{j_1} a a^{j_2}$

Remembering the 2-rounds game, Duplicator will win if

i_1, j_1 are both ≥ 3 or else $i_1 = j_1$,

and similarly for i_2, j_2.

Duplicator can reach such a decomposition in the first round if i, j are both ≥ 7, or if $i = j$.

In general, . . .

With k rounds ahead,

Duplicator ensures that corresponding letter-blocks delimited by chosen positions are of length $\geq 2^k - 1$ or are of the same length.

Duplicator wins $G_m(a^i, a^j)$ for any $i, j \geq 2^m - 1$

Duplicator also wins $G_m(w^i, w^j)$ for any word w and $i, j \geq 2^m - 1$.

Keep in mind: Sentences of qd m can desribe repetitions up to threshold $2^m - 1$ but otherwise can just say "many".

Describing Winning Strategy

When does Duplicator win $G_m((\mathcal{S},\overline{s}),(\mathcal{T},\overline{t}))$?

Specify, for each $k = 0,\ldots,m$, a set I_k of partial isomorphisms (describing game positions) which would Duplicator allow to win with k rounds ahead.

There should be nonempty sets $I_m,\ldots,I_0$ of partial isomorphisms, each of them extending $\overline{s} \mapsto \overline{t}$, such that for all $k = m,\ldots,1$:

- ***(back property)***
  $\forall p \in I_k \; \forall t \in T \; \exists s \in S$ such that $p \cup \{(s,t)\} \in I_{k-1}$
- ***(forth property)***
  $\forall p \in I_k \; \forall s \in S \; \exists t \in T$ such that $p \cup \{(s,t)\} \in I_{k-1}$.

Write $(\mathcal{S},\overline{s}) \cong_m (\mathcal{T},\overline{t})$.

Ehrenfeucht-Fraïssé Theorem

For $m \geq 0$, the following are equivalent:

1. $(\mathcal{S}, \overline{s}) \equiv_m (\mathcal{T}, \overline{t})$
2. $(\mathcal{S}, \overline{s}) \cong_m (\mathcal{T}, \overline{t})$
3. Duplicator wins $G_m((\mathcal{S}, \overline{s}), (\mathcal{T}, \overline{t}))$.

Applications

Non-Definability

The language $\{a^n \mid n \text{ is even}\}$ is not first-order definable.

Suppose a defining first-order sentence φ exists, with $<$ only, say of quantifier-depth m.

We have $a^{2^m} \equiv_m a^{2^m+1}$

We have $a^{2^m} \models \varphi$.

So also $a^{2^m+1} \models \varphi$.

This model is of odd length, contradiction.

Finally Composition!

In order to know whether a formula φ of qd m holds in uv, it suffices to know the m-types of u and v.

Composition Lemma

If $\underline{u} \equiv_m \underline{u'}$ and $\underline{v} \equiv_m \underline{v'}$, then $\underline{u \cdot v} \equiv_m \underline{u' \cdot v'}$.

Use the Ehrenfeucht-Fraïssé Theorem.

Duplicator has winning strategies for the games $G_m(\underline{u}, \underline{u'})$ and $G_m(\underline{v}, \underline{v'})$.

The strategy "on the segments u and u' use the first strategy, on the segments v and v' use the second strategy"

guarantees Duplicator to win also the game $G_m(\underline{u \cdot v}, \underline{u' \cdot v'})$.

Products

The direct product of $\mathcal{S}_1, \mathcal{S}_2$ has $S_1 \times S_2$ as its universe, with relations

$R^{S_1 \times S_2}(a_1, b_1) \ldots (a_n, b_n)$

iff $R^{S_1} a_1 \ldots a_n$ and $R^{S_2} b_1 \ldots b_n$

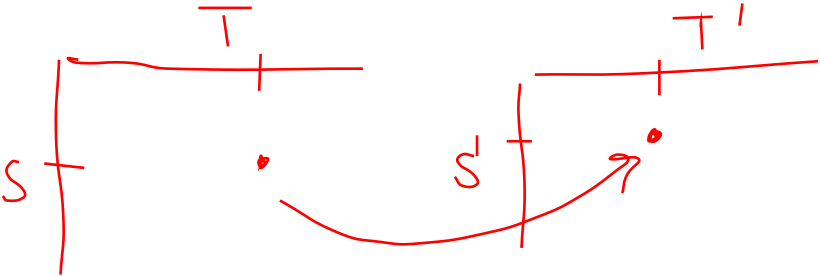
Special forms: Reduced product, synchronized product.

Landmark paper by Feferman and Vaught 1959

Composition for Products

In order to know whether a formula of qd m holds in $\mathcal{S} \times \mathcal{T}$, it suffices to know the m-types of $\mathcal{S}$ and $\mathcal{T}$.

Show: If $\mathcal{S} \equiv_m \mathcal{S}'$ and $\mathcal{T} \equiv_m \mathcal{T}'$, then $\mathcal{S} \times \mathcal{T} \equiv_m \mathcal{S}' \times \mathcal{T}'$.



m-Types

How to Describe Types?

Hintikka formulas:

$$\psi^{0}_{\mathcal{S},\bar{s}}(\bar{x}) := \bigwedge_{R,\ \bar{s}\in R^{S}} R\bar{x} \wedge \bigwedge_{R,\bar{s}\notin R^{S}} \neg R\bar{x}$$

$$\psi^{m+1}_{\mathcal{S},\bar{s}}(\bar{x}) := \bigwedge_{s\in S,\ \mathcal{S},\bar{s}\models\psi^{m}_{\mathcal{S},\bar{s},s}(\bar{x},x)} \exists x \psi^{m}_{\mathcal{S},\bar{s},s}(\bar{x},x)$$

$$\wedge\ \forall x \bigvee_{s\in S,\ \mathcal{S},\bar{s}\models\psi^{m}_{\mathcal{S},\bar{s},s}} \psi^{m}_{\mathcal{S},\bar{s},s}(\bar{x},x)$$

Shorter Notation

$$T_0(\mathcal{S}, \bar{s}) := \{\psi(\overline{x}) \mid \psi \textbf{ atomic}, (\mathcal{S}, \bar{s}) \models \psi(\overline{x})\}$$

$$T^{m+1}(\mathcal{S}, \bar{s}) := \{T^m(\mathcal{S}, \bar{s}, s) \mid s \in S\}$$

Elementary Facts

1. Each type is a finite object.
2. For each m and given tuple length n there are only finitely many m-types of structures $(\mathcal{S}, s_1, \ldots, s_n)$
3. $T^m(\mathcal{S}, s_1, \ldots, s_n)$ fixes for any formula $\varphi(\overline{x})$ whether $\mathcal{S}, \overline{s}) \models \varphi(\overline{x})$.
4. Each formula of quantifier depth m is effectively equivalent to a (finite) disjunction of m-Hintikka formulas
5. The first-order theory of $\mathcal{S}$ is decidable iff the function $m \mapsto T^m(\mathcal{S})$ is computable.

6. **Summarizing:** The m-types give a classification of reasons why a formula of quantifier depth m can be true.

Monadic Types

Saharon Shelah

An FO-Free MSO-Dialect

over structures $\mathcal{S} = (S, <, P_1, \ldots, P_k)$ with unary P_i

Atomic formulas are

$\mathbf{Nonempty}(X \cap Y)$,

$X \subseteq Y$,

$X < Y$ **for "some X-element is $<$ some Y-element"**

$X_1 \cup \ldots \cup X_n = \mathbf{All}$

$\overline{k}$-Types

For $\overline{k} = (k_1, \ldots, k_m)$

define the $\overline{k}$-n type $T_n^{\overline{k}}(\mathcal{S}, \overline{P})$ for a structure $\mathcal{S}(P_1, \ldots, P_n)$

$T_n^{\lambda}(\mathcal{S}, \overline{P}) =$ set of atomic formulas $\varphi(X_1, \ldots, X_n)$
which are true in $(\mathcal{S}, \overline{P})$

$T_n^{(\overline{k}, k_{m+1})}(\mathcal{S}, \overline{P}) =$ set of all types $T_{n+k_{m+1}}^{\overline{k}}(\mathcal{S}, \overline{P}, \overline{Q})$
with $\overline{Q} = (Q_1, \ldots, Q_{k_{m+1}}), Q_j \subseteq S$

Note that m measures quantifier alternation depth.

(2) We could have defined the sum more generally, by allowing the universe and the equality to be defined just as the other relations.

LEMMA 2.3. *For any $\sigma, n, m, \bar{k}$, if for $l = 1, 2$, $\bar{P}_i^l \in \underline{P}(M_i^l)^m$ and for every $i \in N$,*

$$Th_{\bar{k}}^n((M_i^1, \bar{P}_i^1), \Phi(\sigma)) = Th_{\bar{k}}^n((M_i^2, \bar{P}_i^2), \Phi(\sigma)) ,$$

then

$$Th_{\bar{k}}^n(\textstyle\sum_{i \in N}^{\sigma} (M_i^1, \bar{P}_i^1)) = Th_{\bar{k}}^n(\textstyle\sum_{i \in N}^{\sigma} (M_i^2, \bar{P}_i^2)) ,$$

THEOREM 2.4. *For any $\sigma, n, m, \bar{k}$ we can find an $\bar{r}$ such that: if $M = \sum_{i \in N} M_i$, $t_i = Th_{\bar{k}}^n((M_i, \bar{P}_i), \Phi(\sigma))$, and $Q_t = \{i \in N : t_i = t\}$, $l(\bar{P}_i) = m$, then from $Th_{\bar{r}}^n((N, \cdots, Q_t, \cdots), \Psi(\sigma))$ we can effectively compute $Th_{\bar{k}}^n(M, \bigcup_i \bar{P}_i)$ (which is uniquely determined).*

Definition 2.4. (1) For a class K of models

$$Th_{\bar{k}}^n(K, \Phi) = \{Th_{\bar{k}}^n(M, \Phi) : M \in K\} .$$

A Readable Version

Let $\overline{k} = (k_1, \dots, k_m)$.

To obtain the $\overline{k}$-n-type of

$(\mathcal{S}, P_1, \dots, P_n) = \sum_{i \in I} (\mathcal{S}_i, \overline{P}_i)$

consider $(I <^I, Q_1, \dots, Q_\ell)$

where Q_j collects those i where $(\mathcal{S}_i, \overline{P}_i)$ has the j-th $\overline{k}$-n-type;

indeed it suffices to know

$T_\ell^{(r_1,\dots,r_m)}(I <^I, Q_1, \dots, Q_\ell)$.

Essential: The quantifier alternation depth m is the same in $\overline{k}$ as in $\overline{r}$.

Büchi's Theorem via Shelah

Theorem: MTh$(\mathbb{N}, <)$ is decidable.

Show that for any $\overline{k}, n$ we can compute $T_n^{\overline{k}}(\mathbb{N}, <, P_1, \ldots, P_n)$.

Show this inductively over the length of $\overline{k}$, simultaneously for all n.

Case $\overline{k} = \lambda$ tedious but straightforward.

The set Fin(n) of types of finite structures $(M, <, P_1, \ldots, P_n)$ can be computed.

Induction Step

Assume we know the types $T_{n'}^{(k'_1,\ldots,k'_{m-1})}(\mathbb{N},<,\overline{P})$ for all $k'_1,\ldots,k'_{m-1}$, and $P_1,\ldots,P_{n'}$.

To compute a type $T_n^{(k_1,\ldots,k_m)}(\mathbb{N},<,\overline{P})$

we need the set of all $T_{n+k_m}^{(k_1,\ldots,k_{m-1})}(\mathbb{N},<,\overline{P},\overline{R})$

By Ramsey, any such type is presentable as a $(k_1,\ldots,k_{m-1})$-type $\tau+\sum_{i\in\mathbb{N}}\sigma$ with $\tau,\sigma\in\mathrm{Fin}(n+k_m)$

The finitely many possible types τ,σ are computable.

What remains is to compute the sum types $\sum_{i\in\mathbb{N}}\sigma$

By the Composition Theorem such a type can be obtained from a $(r_1,\ldots,r_{n-1})$-type of a structure $(\mathbb{N},<,\overline{Q})$

But these types are computable by induction hypothesis.

from Büchi's last paper

("State Strategies for Games in $F_{\sigma\delta} \cap G_{\delta\sigma}$]

show decidability at ω_1. Shelah [14] copied my AR (as Theorem 1.1), and my use of cofinal closed (as Conclusion 1.2), and now he can splice theories $MT[x, y)$ $x < y < \omega_1$, x and y from a cofinal closed set, where I splice runs $Z[x, y)$; he works on the ω_0-level of Fraïsse tree while I work on the 0-level. He says he does not use automata; I say he is joking, the automata are right there in the combinatorial lemma. He does show that PB (McNaughton's lemma) is not needed for ω_1.