# Определимост

Иво Стратев

22 октомври 2019 г.

## Съдържание

Нека първо кажем какво ще разбираме под определимост

# 1 Определимост

Ще дефинираме серия от изображения, които да ни помогнат формално да въведем понятието определимост.

## 1.1 Променливи участващи в терм

Дефинираме изображение $varsInTerm : Term_{\mathcal{L}} \rightarrow \mathcal{P}(Var_{\mathcal{L}})$, което на всеки терм съпоставя множеството от участващите в него променливи.
Действащо по следните правила:

- $(\forall \chi \in Var_{\mathcal{L}})[varsInTerm(\chi) = \{\chi\}]$ (единствената променлива участваща в терм, който е променлива е самата променлива);
- $(\forall c \in Const_{\mathcal{L}})[varsInTerm(c) = \emptyset]$ (не участват променливи в терм, който представлява константен символ);
- Ако $\delta \in Func_{\mathcal{L}}$ и $\tau_1$ е терм, ..., $\tau_{\#\delta}$ е терм, то
  $varsInTerm(\delta(\tau_1, \ldots, \tau_{\#\delta})) = \bigcup\limits_{i=1}^{\#\delta} varsInTerm(\tau_i)$.
  (Променливите участващи във "функционален" терм са променливите участващи в "аргументите").

### 1.1.1 Пример

$$\begin{aligned}
varsInTerm(f(x, g(y, x))) &= \\
varsInTerm(x) \cup varsInTerm(g(y, x)) &= \\
\{x\} \cup (varsInTerm(y) \cup varsInTerm(x)) &= \\
\{x\} \cup (\{y\} \cup \{x\}) &= \\
\{x\} \cup \{y, x\} &= \\
\{x, y, x\} &= \\
\{x, y\}
\end{aligned}$$

## 1.2 Свободни променливи на формула

Дефинираме изображение (индуктивно) $freeVars : \mathcal{F}_{\mathcal{L}} \rightarrow \mathcal{P}(Var_{\mathcal{L}})$, което на всяка формула съпоставя множеството от свободно участващите в нея променливи.
Действащо по следните правила:

- Ако $\rho \in Pred_{\mathcal{L}}$ и $\tau_1, \ldots, \tau_{\#\rho}$ са термове, то
$$freeVars(\rho < \tau_1, \ldots, \tau_{\#\rho} >) = \bigcup_{i=1}^{\#\rho} varsInTerm(\tau_i)$$
(В атомарна формула всяка участваща променлива е свободна, а участващите са променливите от термовете);

- Ако $\mathcal{L}$ е с формално равенство и $\alpha$ е терм и $\beta$ е терм, то $freeVars(< \alpha \doteq \beta >) = varsInTerm(\alpha) \cup varsInTerm(\beta)$;

- Ако $\varphi$ е формула, то $freeVars([\neg\varphi]) = freeVars(\varphi)$;

- Ако $\varphi$ е формула, $\psi$ е формула и $\sigma \in \{\vee, \&, \implies, \iff\}$, то $freeVars([\varphi\sigma\psi]) = freeVars(\varphi) \cup freeVars(\psi)$;

- Ако $\varphi$ е формула, $\chi \in Var_{\mathcal{L}}$ и $\Lambda \in \{\exists, \forall\}$, то $freeVars([\Lambda\chi\varphi]) = freeVars(\varphi) \setminus \{\chi\}$.
(Ако формулата е кванторна, то свободните променливи са свободните променливи на подформулата след променливата без самата променлива (тя участва свързано / под квантор) ).

### 1.2.1 Пример

$$\begin{aligned}
freeVars([p < x, x > \implies [\forall x\, p < x, y >]]) = \\
freeVars(p < x, x >) \cup freeVars([\forall x\, p < x, y >]) = \\
(varsInTerm(x) \cup varsInTerm(x)) \cup (freeVars(p < x, y >) \setminus \{x\}) = \\
\{x\} \cup ((varsInTerm(x) \cup varsInTerm(y)) \setminus \{x\}) = \\
\{x\} \cup (\{x, y\} \setminus \{x\}) = \\
\{x\} \cup \{y\} = \\
\{x, y\}
\end{aligned}$$

Коментар: Както се вижда от избрания пример променливата $x$ има и свободно и свързано участие (под универсален квантор) в подформули, но във формулата, която разглеждаме участието е свободно.

## 1.3 Естесвен начин за съпоставяне на функция на дадена формула

### 1.3.1 Едно важно твърдение (без доказателство)

Верността (сементиката) на една формула зависи само от структурата, в която се разглежда и оценките само на свободните променливи на формулата.

Нека $\mathcal{L}$ е език на предикатното смятане от първи ред. Нека $\varphi$ е произволна и фиксирана формула над езика $\mathcal{L}$. Нека $X = freeVars(\varphi)$. Нека $\mathcal{A}$ е фиксирана структура, за $\mathcal{L}$. Нека $\nu$ и $\kappa$ са оценки и нека $(\forall \chi \in X)[\nu(\chi) = \kappa(\chi)]$. Тогава $\| \varphi \|^{\mathcal{A}}_{\nu} = \| \varphi \|^{\mathcal{A}}_{\kappa}$.

Както се казва, удоволствието да се докаже горното твърдение оставяме на любознателния читател.

### 1.3.2 Естествена функция

Нека фиксираме произволна формула $\varphi$. Нека $X = freeVars(\varphi)$. Ако $X = \emptyset$, то случая не е интересен (за момента, понеже искаме да определяме разни неща), защото по горното твърдение верността на $\varphi$ ще зависи единствено от структурата. Тоест $\varphi$ ще е характеризация на света, а не на неговите елементи. За това нека $X \neq \emptyset$. Във всяка формула може да участват краен брой променливи, което влече и че $X$ е крайно множество. Нека тогава $n = |X|$. Всяко крайно множество може да бъде наредено, за това нека фиксираме една наредба на елементите на $X$: $x_1 < \cdots < x_n$. Естествена наредба на $X$ е според реда на срещане на променливите във $\varphi$. Нека фиксираме подходяща структура $\mathcal{A}$ и нека $\nu$ е произволна оценка. Така нека дефинираме функция: $f : |\mathcal{A}|^n \to \{\text{И}, \text{Л}\}$ действаща по правило:

$$f(a_1, \ldots, a_n) = \| \varphi \|^{\mathcal{A}}_{\nu {x_1 \, x_2 \, \ldots \, x_n \atop a_1 \, a_2 \, \ldots \, a_n}}$$

Тоест $f$ пресмята верността на $\varphi$ в $\mathcal{A}$ при оценка на свободните променливи на $\varphi$ - подадените аргументи.

Нека $charF_{\mathcal{L}} = \{\varphi \mid \varphi \in \mathcal{F}_{\mathcal{L}} \ \& \ freeVars(\varphi) = \emptyset\}$. И Нека $naturalFunc_{\mathcal{A}} : \mathcal{F}_{\mathcal{L}} \setminus charF_{\mathcal{L}} \to \bigcup\limits_{n \in \mathbb{N}^+} \{g \mid g : |\mathcal{A}|^n \to \{\text{И}, \text{Л}\}\}$. Която на формула имаща поне една свободна променлива съпоставя ествествената ѝ функция по горното обяснение. Като за оценка $\nu$ можем да си мислим, че взимаме оценката, която на всяка променлива съпоставя първия аргумент на естестевената функция и след това я модифицираме. По твърдението няма значние каква оценка модифицираме, защото фиксираме оценката на всяка свободна променлива.

## 1.4 Множеството, което една формула определя

Нека $\varphi$ е произволна формула с поне една свободна променлива (искаме да определяме множество все пак). Нека $\mathcal{A}$ е подходяща структура. Нека $n = |freeVars(\varphi)|$. Тогава множеството, което $\varphi$ определя ще бележим с $Set_{\mathcal{A}}(\varphi)$. И по дефиниция $Set_{\mathcal{A}}(\varphi) =$
$\{< a_1, \ldots, a_n > \in |\mathcal{A}|^n \mid naturalFunc_{\mathcal{A}}(\varphi)(a_1, \ldots, a_n) = \text{И}\}$. Тоест отделяме

от $|\mathcal{A}|^n$ онези елементи, върху които естествената функция, която $\varphi$ определя е истина.

### 1.5 Определимо множество

Нека $\mathcal{L}$ е език на предикатното смятане от първи ред. Нека $\mathcal{A}$ е структура за $\mathcal{L}$. Нека $B \subseteq |\mathcal{A}|^n$ за някое $n \in \mathbb{N}^+$. Казваме, че $B$ е определимо множество ако има формула $\varphi$ със свойството $B = Set_{\mathcal{A}}(\varphi)$. Тоест има формула, която определя множеството $B$.

### 1.6 Определимост на елемент

Нека $\mathcal{L}$ е език на предикатното смятане от първи ред. Нека $\mathcal{A}$ е структура за $\mathcal{L}$. Нека $a \in |\mathcal{A}|$. Казваме, че $a$ е определим, ако има формула $\varphi$ със свойството $\{a\} = Set_{\mathcal{A}}(\varphi)$. Тоест елемента $a$ е определим точно когато неговия синглетон е определимо множество.

### 1.7 Определимост на функция

Нека $\mathcal{L}$ е език на предикатното смятане от първи ред. Нека $\mathcal{A}$ е структура за $\mathcal{L}$. Нека $f : |\mathcal{A}|^n \to A$ за някое $n \in \mathbb{N}^+$. Казваме, че $f$ е определима функция, ако нейната графика е определимо множество. Тоест има формула $\varphi$ със свойството $\{< a_1, \ldots, a_n, b > \in |\mathcal{A}|^{n+1} \mid b = f(a_1, \ldots, a_n)\} = Set_{\mathcal{A}}(\varphi)$.

### 1.8 Определимост на предикат/свойство/релация

Нека $\mathcal{L}$ е език на предикатното смятане от първи ред. Нека $\mathcal{A}$ е структура за $\mathcal{L}$. Нека $R \subseteq |\mathcal{A}|^n$ за някое $n \in \mathbb{N}^+$. Казваме, че $R$ е определим предикат/определимо свойство/определима релация/определима връзка точно когато $R$ е определимо множество.

## 2 Няколко задачи върху определимост

Преминаваме към решаването на задачи.

### 2.1 Естесвени числа само с наредбата им

Нека $\mathcal{L}$ се състой от единствен двуместен предикатен символ $p$. Това ще записваме така $\mathcal{L} =< p >$.

Правим следната оговорка множеството на променливи се състой от латинските букви с и без индексни, за които не е изрично казано, че са предикатни символи, функционални или символи за константи.

Нека $\mathcal{A}$ е структура за $\mathcal{L}$. Носителя/Домейна/Универсума на $\mathcal{A}$ е множеството на естествените числа. Тоест $|\mathcal{A}| = \mathbb{N}$. И интерпретацията на $p$ е на релацията "по-малко". Тоест:

$$< x, y > \in p^{\mathcal{A}} \longleftrightarrow x < y$$

Горното условие ще записваме още така:
Нека $< \mathbb{N}\ ;\ p^{\mathcal{A}} >$ е структура за $\mathcal{L}$.

Да се докаже, че всеки елемент на $\mathbb{N}$ е определим.

Ще пишем формулите два пъти с различни цветове.
Със син цвят ще бъдат спрямо синтаксиса с много и различни скоби.
Със зелен цвят ще бъдат спрямо олекотен синтакс само с кръгли скоби и въведени приоритети за намаляване на използваните скоби.
Реда на оценяване е следния:

1. $\neg$
2. $\forall\chi,\ \exists\chi$
3. $\&$
4. $\vee$
5. $\implies$
6. $\iff$

Ще натоварим значението на червения цвят с още едно: синтактично заместване на формула, при което евентуално се преименуват променливи, така че смисъла да е този, който имаме пишейки формулата. По идея концепцията е същата като извикването на рекурсивна функция в езика C++, всяко извикване алокира памет на стека и по този начин всяка променлива е на различен адрес, евентуално с различни стойности.

Ще използваме $\leftrightharpoons$ със смисъл "синтактично дефинираме".

**Решение:**

Нека определим първо числото 0.

Идея: Няма по-малко число от него.

$\varphi_0 < x > \leftrightharpoons [\neg[\exists y\ p < y, x >]]$

$\varphi_0(x) \leftrightharpoons \neg\exists y\ p(y, x)$

Очевидно горните формули са истина само когато оценката на x е 0.

Както можем да се досетим можем да дефинираме числото 0 като най-малкият елемент. Нека преди това дефинираме формула със смисъл "по-малко или равно".

$\varphi_{\leq} < x, y > \leftrightharpoons [\neg p < y, x >]$

$\varphi_{\leq}(x, y) \leftrightharpoons \neg p(y, x)$

Очевидно горните формули са истина точно когато оценката на x е по-малка или равна от оценката на y.

$\varphi_0 < x > \leftrightharpoons [\forall y \; \varphi_{\leq} < x, y >]$

$\varphi_0(x) \leftrightharpoons \forall y \; \varphi_{\leq}(x, y)$

Очевидно горните формули са истина само когато оценката на x е 1.

Ще дадем пример за заместването, за което говорихме. Формулата:
$[\forall y \; \varphi_{\leq} < x, y >]$. По същество представлява: $[\forall y \; [\neg \, p < y, x >]]$.

След като имаме формула за 0, то по аналогичен начин можем да дефинираме 1. Като най-малкото естесвено число, което е по-голямо от 0. Но за да кажем по-голямо от 0, то ще трябва да въведем "временна променлива за която да кажем, че е 0 (да я инициализираме с 0). Това става по-следния начин:

$\varphi_1 < x > \leftrightharpoons [\exists y \; [\varphi_0 < y > \& \; [p < y, x > \& \; [\forall z \; [p < y, z > \Longrightarrow \; \varphi_{\leq} < x, z >]]]]]$

$\varphi_1(x) \; \leftrightharpoons \exists y \; (\varphi_0(y) \; \& \; p(y, x) \; \& \; \forall z \; (p(y, z) \; \Longrightarrow \; \varphi_{\leq}(x, z)))$

По-аналогичен начин можем да определим 2: като най-малкото число, което е по-голямо от 1.

Ясно се вижда, че с индукция по естествените числа можем да определим всяко. Нека го направим

База: базовия случай е да напишем формула, с която да определим 0, но ние вече имаме такава.

Индукционна хипотеза: За някое естествено число $n$ имаме формула $\varphi_n$.

Индукционна стъпка: ще напишем формула, която определя $n + 1$.

$\varphi_{n+1} < x > \leftrightharpoons$
$[\exists y \; [\varphi_n < y > \& \; [p < y, x > \& \; [\forall z \; [p < y, z > \Longrightarrow \; \varphi_{\leq} < x, z >]]]]]$

$\varphi_{n+1}(x) \; \leftrightharpoons \exists y \; (\varphi_n(y) \; \& \; p(y, x) \; \& \; \forall z \; (p(y, z) \; \Longrightarrow \; \varphi_{\leq}(x, z)))$

## 2.2 Естествени числа само с графика на умножение

Нека $\mathcal{L} =< p >$ е език състоящ се от единствен триместен предикатен символ.
Нека $S =< \mathbb{N} \; ; \; p^S >$ е структура за $\mathcal{L}$. Където:

$$< x, y, z > \in p^S \longleftrightarrow z = x.y$$