ГОДИШНИК

НА

СОФИЙСКИЯ УНИВЕРСИТЕТ „СВ. КЛИМЕНТ ОХРИДСКИ“

ФАКУЛТЕТ ПО МАТЕМАТИКА И ИНФОРМАТИКА

КНИГА 1 – МАТЕМАТИКА И МЕХАНИКА
КНИГА 2 – ПРИЛОЖНА МАТЕМАТИКА И ИНФОРМАТИКА

Том 92

ANNUAIRE

DE

L'UNIVERSITE DE SOFIA „ST. KLIMENT OHRIDSKI“

FACULTE DE MATHEMATIQUES ET INFORMATIQUE

LIVRE 1 – MATHEMATIQUES ET MECANIQUE
LIVRE 2 – MATHEMATIQUES APPLIQUEE ET INFORMATIQUE

Tome 92

СОФИЯ • 2000

УНИВЕРСИТЕТСКО ИЗДАТЕЛСТВО „СВ. КЛИМЕНТ ОХРИДСКИ“

SOFIA • 2000

ГОДИШНИК

НА

СОФИЙСКИЯ УНИВЕРСИТЕТ „СВ. КЛИМЕНТ ОХРИДСКИ“

ФАКУЛТЕТ ПО МАТЕМАТИКА И ИНФОРМАТИКА

Книга 1 — МАТЕМАТИКА И МЕХАНИКА
Книга 2 — ПРИЛОЖНА МАТЕМАТИКА И ИНФОРМАТИКА

Том 92
1998

ANNUAIRE

DE

L'UNIVERSITE DE SOFIA "ST. KLIMENT OHRIDSKI"

FACULTE DE MATHEMATIQUES ET INFORMATIQUE

Livre 1 — MATHEMATIQUES ET MECANIQUE
Livre 2 — MATHEMATIQUES APPLIQUEE ET INFORMATIQUE

Tome 92
1998

СОФИЯ • 2000 • SOFIA
УНИВЕРСИТЕТСКО ИЗДАТЕЛСТВО „СВ. КЛИМЕНТ ОХРИДСКИ“
PRESSES UNIVERSITAIRES "ST. KLIMENT OHRIDSKI"

Annuaire de l' Université de Sofia "St. Kliment Ohridski"
Faculté de Mathématiques et Informatique

Годишник на Софийския университет „Св. Климент Охридски"
Факултет по математика и информатика

Address for correspondence:

Faculty of Mathematics and Informatics
"St. Kliment Ohridski" University of Sofia
5 Blvd J. Bourchier, P.O. Box 48
BG-1164 Sofia, Bulgaria

Fax xx(359 2) 687 180
electronic mail:
annuaire@fmi.uni-sofia.bg

Aims and Scope. The *Annuaire* is the oldest Bulgarian journal, founded in 1904, devoted to pure and applied mathematics, mechanics and computer sciences. It is reviewed by *Zentralblatt für Mathematik*, *Mathematical Reviews* and the Russian *Referativnii Jurnal*. The *Annuaire* publishes significant and original research papers of authors both from Bulgaria and abroad in some selected areas that comply with the traditional scientific interests of the Faculty of Mathematics and Informatics at the "St. Kliment Ohridski" University of Sofia, i.e., algebra, geometry and topology, analysis, mathematical logic, theory of approximations, numerical methods, computer sciences, classical, fluid and solid mechanics, and their fundamental applications.

ISSN 0205-0808

CONTENTS

Book 1

MATHEMATICS AND MECHANICS

Book 2

APPLIED MATHEMATICS AND INFORMATICS

ГОДИШНИК

НА

СОФИЙСКИЯ УНИВЕРСИТЕТ „СВ. КЛИМЕНТ ОХРИДСКИ“

ФАКУЛТЕТ ПО МАТЕМАТИКА И ИНФОРМАТИКА

Книга 1 — МАТЕМАТИКА И МЕХАНИКА

Том 92
1998

ANNUAIRE

DE

L'UNIVERSITE DE SOFIA
"ST. KLIMENT OHRIDSKI"

FACULTE DE MATHEMATIQUES ET INFORMATIQUE

Livre 1 — MATHEMATIQUES ET MECANIQUE

Tome 92
1998

СОФИЯ • 2000 • SOFIA
УНИВЕРСИТЕТСКО ИЗДАТЕЛСТВО „СВ. КЛИМЕНТ ОХРИДСКИ“
PRESSES UNIVERSITAIRES "ST. KLIMENT OHRIDSKI"

ГОДИШНИК НА СОФИЙСКИЯ УНИВЕРСИТЕТ „СВ. КЛИМЕНТ ОХРИДСКИ“
ФАКУЛТЕТ ПО МАТЕМАТИКА И ИНФОРМАТИКА
Книга 1 — Математика и механика
Том 92, 1998

ANNUAIRE DE L'UNIVERSITE DE SOFIA „ST. KLIMENT OHRIDSKI“
FACULTE DE MATHEMATIQUES ET INFORMATIQUE
Livre 1 — Mathématiques et Mecanique
Tome 92, 1998

CHARACTERIZATION OF THE STRUCTURES WHICH ADMIT EFFECTIVE ENUMERATIONS*

ANGEL DITCHEV

In this paper a characterization of the partial structures with denumerable domains which admit an effective enumeration is given.

Keywords: effective enumeration, extended effective enumeration, recursive model

1991/95 Math. Subject Classification: 03D45, 03C57

0. INTRODUCTION

In the Recursive Model Theory there are a lot of attempts to characterize the structures which admit a recursive enumeration. There are some necessary conditions and some sufficient ones [1]. On the other hand, in many of them the considerations are restricted to a given class of structures, for example, Boolean algebras, partially ordered sets and so on [1]. Further, other definitions of recursive enumerations are given [1–3] which restrict or extend the class of structures satisfying these definitions, and attempts to characterize the corresponding classes are made. One of these definitions is the well-known strong constructivization (recursive presentation) [1]. In [2] Soskova and Soskov have defined another notion of effective enumeration (recursively enumerable (r.e.) enumeration) of a partial structure. Thus they have succeeded to characterize the structure satisfying their definition by means of REDS computability [2] with finitely many constants.

* Research partially supported by the Ministry of Education, Contract No И-604, 1997.

In connection with this and some other results [4–6] there have been stated many conjectures, but all of them have been rejected (cf. [7–9]).

In [7, 8], the structures with denumerable domains and unary functions and predicates which admit effective enumerations have been characterized. It is natural, using the result in [7, 8], to try to generalize it. One possible way to do this is the following: Let us consider the least set B^*, which contains the domain B of the structure and is closed under taking ordered pairs. Thus, we can consider all finite Cartesian products of B as subsets of B^* and we consider the basic functions and predicates as unary functions and predicates on B^*. In this case however, we need to generalize the notion of effective enumeration and introduce the so-called extended effective enumerations.

In Section 1 we give the necessary definitions.

In Section 2 we prove the following results: 1) Theorem 2.1 that a partial structure with a denumerable domain admits an effective enumeration iff the corresponding structure on B^* admits an extended effective enumeration; 2) Theorem 2.17 and Theorem 2.24 that a partial structure with a denumerable domain admits an effective enumeration iff the family of the types of all elements of the extended structure on B^* has an universal r.e. set, which satisfies certain natural conditions.

1. PRELIMINARIES

In what follows, by $\mathbb{N}$ we shall denote the set of all natural numbers. Let Π, L, R be defined as follows:

$$\Pi(i,j) = 2^{i+1}(2j+1), \quad L((\Pi(i,j)) = i, \quad R((\Pi(i,j)) = j,$$

$$L(i) = R(i) = i, \text{ for all even natural numbers.}$$

Let us note that for every natural number i exactly one of the following two conditions is valid:

a) i is odd;

b) i is even and $i = \Pi(i_1, i_2)$, for some unique i_1 and i_2.

Let U be a subset of $\mathbb{N}^{n+1}$ and $\mathcal{F}$ be a family of subsets of $\mathbb{N}^n$. The set U is said to be universal for the family $\mathcal{F}$ iff for any a the set $\{\overline{x} \mid (a, \overline{x}) \in U\}$ belongs to the family $\mathcal{F}$ and, conversely, for any A from $\mathcal{F}$ there exists such an a that $A = \{\overline{x} \mid (a, \overline{x}) \in U\}$. If U is an universal set, then by U_a we shall denote the set $\{\overline{x} \mid (a, \overline{x}) \in U\}$.

If f is a partial function, $\mathrm{Dom}(f)$ denotes the domain and $\mathrm{Ran}(f)$ denotes the range of values of the function f.

Let $\mathfrak{A} = (B; \theta_1, \ldots, \theta_k; F_1, \ldots, F_l)$ be a denumerable partial structure, i.e. B is an arbitrary denumerable set, $\theta_1, \ldots, \theta_k$ are partial functions of several arguments on B, and $F_1, \ldots, F_l$ are partial predicates of several arguments on B. We shall identify the predicates with the (partial) mappings which obtain values 0 or 1, taking 0 for true and 1 for false.

If every θ_i $(1 \leqq i \leqq k)$ and every F_j $(1 \leqq j \leqq l)$ are totally defined, then we say that the structure $\mathfrak{A}$ is a total one.

Effective enumeration of the structure $\mathfrak{A}$ is every ordered pair $\langle \alpha, \mathfrak{B} \rangle$, where $\mathfrak{B} = (\mathbb{N}; \varphi_1, \ldots, \varphi_k; \sigma_1, \ldots, \sigma_l)$ is a partial structure of the same relational type as $\mathfrak{A}$, and α is a partial surjective mapping of $\mathbb{N}$ onto B such that the following conditions hold:

(i) $\mathrm{Dom}(\alpha)$ is recursively enumerable and $\varphi_1, \ldots, \varphi_k$, $\sigma_1, \ldots, \sigma_l$ are partial recursive;

(ii) For all natural $x_1, \ldots, x_{a_i}$, $1 \leqq i \leqq k$,

$$\alpha(\varphi_i(x_1, \ldots, x_{a_i})) \cong \theta_i(\alpha(x_1), \ldots, \alpha(x_{a_i})).$$

(iii) For all natural $x_1, \ldots, x_{b_j}$, $1 \leqq j \leqq l$,

$$\sigma_j(x_1, \ldots, x_{b_j}) \cong F_j(\alpha(x_1), \ldots, \alpha(x_{b_j})).$$

The next proposition is obvious.

Proposition 1.1. *Let* $\mathfrak{A} = \langle B; \theta_1, \ldots, \theta_k; F_1, \ldots, F_l \rangle$, $\mathfrak{A}' = \langle B; \theta_1, \ldots, \theta_k; F_1', \ldots, F_l' \rangle$, $\mathfrak{A}'' = \langle B; \theta_1, \ldots, \theta_k; F_1'', \ldots, F_l'' \rangle$ *be partial structures such that*

$$F_j'(s_1, \ldots, s_{b_j}) \cong \begin{cases} 0, & \text{if } F_j(s_1, \ldots, s_{b_j}) \cong 0, \\ \text{not defined}, & \text{otherwise}, \end{cases}$$

$$F_j''(s_1, \ldots, s_{b_j}) \cong \begin{cases} 0, & \text{if } F_j(s_1, \ldots, s_{b_j}) \cong 1, \\ \text{not defined}, & \text{otherwise}, \end{cases}$$

$j = 1, \ldots, l$.

If $\mathfrak{A}$ *admits an effective enumeration, then* $\mathfrak{A}'$ *and* $\mathfrak{A}''$ *admit effective enumerations, as well.*

Let B be an arbitrary set, $0 \notin B$ and $B_0 = B \cup \{0\}$. Let in addition $(\cdot, \cdot)$ be a fixed operation ordered pair and assume the set B_0 does not contain ordered pairs. We define the set B^* as follows:

a) For any $a \in B_0$, $a \in B^*$;

b) If $a \in B^*$ and $b \in B^*$, then $\langle a, b \rangle \in B^*$.

Consequently, B^* is the least set which contains the set B_0 and is closed under the operation ordered pair $(\cdot, \cdot)$.

On the set of all partially defined functions on B^* we define two operations — composition and combination in the following way:

a) The composition of the functions φ_1 and φ_2 is denoted by $\varphi_1\varphi_2$ and

$$\varphi_1\varphi_2(s) \cong \varphi_1(\varphi_2(s));$$

b) The combination of the functions φ_1 and φ_2 is denoted by $\langle \varphi_1, \varphi_2 \rangle$ and

$$\langle \varphi_1, \varphi_2 \rangle(s) \cong \langle \varphi_1(s), \varphi_2(s) \rangle.$$

The functions π and δ are defined on B^* as follows:

$$\pi(\langle a, b \rangle) = a; \quad \delta(\langle a, b \rangle) = b, \quad \text{for any elements } a, b \text{ of } B^*;$$
$$\pi(a) = \delta(a) = \langle 0, 0 \rangle, \quad \text{if } a \in B;$$
$$\pi(0) = \delta(0) = 0.$$

For any natural positive number k and arbitrary elements $s_1, \ldots, s_k$ the ordered k-tuple $\langle s_1, \ldots, s_k \rangle$ is defined in the usual way:

$$\langle s_1 \rangle = s_1; \quad \langle s_1, \ldots, s_k, s_{k+1} \rangle = \langle \langle s_1, \ldots, s_k \rangle, s_{k+1} \rangle.$$

Let $B^k = \{\langle s_1, \ldots, s_k \rangle \mid s_1 \in B \,\&\, \ldots \,\&\, s_k \in B$; this way $B^k \subset B^*$. If φ is a k-ary partial function on B, then it is natural to think of φ as a partial function on B^k or even on B^*, and in addition if $s_1, \ldots, s_k$ are elements of B, then we shall write $\varphi(\langle s_1, \ldots, s_k \rangle)$ instead of $\varphi(s_1, \ldots, s_k)$ and conversely; thus in this case we can think of φ as a partial unary function on B^*.

Let $\mathfrak{L}$ be the first order language which consists of k unary functional symbols $f_1, \ldots, f_k$ and l unary predicate symbols $T_1, \ldots, T_l$. Let T_0 be a new unary predicate symbol which is intended to represent the unary total predicate $F_0 = \lambda s.0$ on B^*.

We shall define *functional terms* and *functional termal formulae* (in language $\mathfrak{L}$) as follows:

a) If f is a functional symbol in the language $\mathfrak{L}$, then f is a functional term;

b) If τ^1 and τ^2 are functional terms, then $\tau^1\tau^2$ and (τ^1, τ^2) are functional terms;

c) If τ is a functional term and T is a predicate symbol, then $T(\tau)$ and $\neg T(\tau)$ are functional termal formulae.

Let $\mathfrak{A} = (B; \theta_1, \ldots, \theta_k; F_1, \ldots, F_l)$ be a partial structure and $\mathfrak{A}^* = (B^*; \theta_1, \ldots, \theta_k; F_1, \ldots, F_l)$ be the corresponding partial structure on B^*. If τ is a functional term in the language $\mathfrak{L}$, we shall define *the value* $\tau_{\mathfrak{A}^*}$ *of the term* τ *in the structure* $\mathfrak{A}^*$, which will be a partial function on B^*:

a) If $f = f_i$, $1 \leqq i \leqq k$, is a functional symbol in the language $\mathfrak{L}$, then $f_{\mathfrak{A}^*}$ is the function θ_i;

b) If $\tau = \tau^1\tau^2$, then $\tau_{\mathfrak{A}^*}$ is the composition of the partial functions $\tau^1_{\mathfrak{A}^*}$ and $\tau^2_{\mathfrak{A}^*}$; If $\tau = \langle \tau^1, \tau^2 \rangle$, then $\tau_{\mathfrak{A}^*}$ is the combination of the functions $\tau^1_{\mathfrak{A}^*}$ and $\tau^2_{\mathfrak{A}^*}$.

Analogously, if Π is a functional termal formula in the language $\mathfrak{L}$, we define a *value* $\Pi_{\mathfrak{A}^*}$ *of the functional termal formula* Π *in the structure* $\mathfrak{A}^*$ and the value $\Pi_{\mathfrak{A}^*}$ in the structure $\mathfrak{A}^*$ will be a partially defined predicate on B^*:

a) If $\Pi = T_j(\tau)$, $1 \leqq j \leqq l$, then the partial predicate $\Pi_{\mathfrak{A}^*}$ is defined as follows:

$$\Pi_{\mathfrak{A}^*}(s) \cong F_j(\tau_{\mathfrak{A}^*}(s)) \quad \text{for any element } s \in B^*;$$

b) If $\Pi = \neg T(\tau)$, where T is a predicate symbol, then the partial predicate $\Pi_{\mathfrak{A}^*}$ is defined as follows:

$$\Pi_{\mathfrak{A}^*}(s) \cong \begin{cases} 1, & \text{if } T_{\mathfrak{A}^*}(s) \cong 0, \\ 0, & \text{if } T_{\mathfrak{A}^*}(s) \cong 1, \\ \text{not defined}, & \text{if } T_{\mathfrak{A}^*}(s) \text{ is not defined.} \end{cases}$$

We assume fixed an effective coding of the functional terms and the functional termal formulae of the language $\mathfrak{L}$. If v is a natural number, then we denote by τ^v (Π^v) the functional term (functional termal formula) with a code v.

If s is an element of B^*, then $\mathbf{T}_{\mathfrak{A}^*}[s]$ (the type of s) is the set of natural numbers

$$\{v \mid \Pi_v \text{ is a functional termal formula } \&\ \Pi^v_{\mathfrak{A}^*}(s) \cong 0\}.$$

In this section we shall extend the notion effective enumeration.

Suppose a partial structure $\mathfrak{A} = (B; \theta_1, \ldots, \theta_k; F_1, \ldots, F_l)$ is given, where θ_i is an a_i-ary partial function on B, $1 \leqq i \leqq k$, and F_j is a b_j-ary predicate on B, $1 \leqq j \leqq l$, and B is a denumerable set. We shall consider the structure $\mathfrak{A}^* = (B^*; \theta_1, \ldots, \theta_k; F_1, \ldots, F_l)$, where all the functions and predicates $\theta_1, \ldots, \theta_k$; $F_1, \ldots, F_l$ are unary on B^*.

Extended effective enumeration of the structure $\mathfrak{A}^*$ is every ordered pair $\langle \alpha^*, \mathfrak{B}^* \rangle$, where $\mathfrak{B}^* = (\mathbb{N}; \varphi_1^*, \ldots, \varphi_k^*; \sigma_1^*, \ldots, \sigma_l^*)$ is a partial structure with unary functions and predicates and α^* is a partial surjective mapping of $\mathbb{N}$ onto B^* such that the following conditions hold:

(i) $\mathrm{Dom}(\alpha^*)$ is recursively enumerable and $\varphi_1^*, \ldots, \varphi_k^*$, $\sigma_1^*, \ldots, \sigma_l^*$ are partially recursive;

(ii) $\alpha^*(\varphi_i^*(x)) \cong \theta_i(\alpha^*(x))$ for all natural x, $1 \leqq i \leqq k$;

(iii) $\sigma_j^*(x) \cong F_j(\alpha^*(x))$ for all natural x, $1 \leqq j \leqq l$;

(iv) $\alpha^{*-1}(B)$ and $\alpha^{*-1}(B^* \setminus B)$ are recursively separable and $\alpha^{*-1}(0) = \{0\}$;

(v) There exist total recursive functions Π', L', R' such that:

a) $\alpha^*(\Pi'(x, y)) \cong \langle \alpha^*(x), \alpha^*(y) \rangle$;

b) If $\alpha^*(x) \cong \langle a, b \rangle$, then $\alpha^*(L'(x)) \cong a$ and $\alpha^*(R'(x)) \cong b$.

We shall prove first the following theorem:

Theorem 2.1. *Given a partial structure $\mathfrak{A} = (B; \theta_1, \ldots, \theta_k; F_1, \ldots, F_l)$, where B is a denumerable set, $\mathfrak{A}$ admits an effective enumeration iff the corresponding structure $\mathfrak{A}^* = (B^*; \theta_1, \ldots, \theta_k; F_1, \ldots, F_l)$ admits an extended effective enumeration.*

Proof. First, let $\mathfrak{A} = (B; \theta_1, \ldots, \theta_k; F_1, \ldots, F_l)$ admit an effective enumeration $(\alpha, \mathfrak{B})$. We define the mapping $\alpha^* : \mathbb{N} \twoheadrightarrow B^*$ as follows:

a) $\alpha^*(2(i+1)) \cong \alpha(i)$, $\alpha(0) = 0$;

b) $\alpha^*(\Pi(i_1, i_2)) \cong \langle \alpha^*(i_1), \alpha^*(i_2) \rangle$.

The next lemmas follow from the definitions of α^* and Π.

Lemma 2.2. *For any natural x and y the following conditions hold:*

a) $\alpha^*(\Pi(x, y)) \cong \langle \alpha^*(x), \alpha^*(y) \rangle$;

b) *If* $\alpha^*(x) \cong \langle a, b \rangle$, *then* $\alpha^*(L(x)) \cong a$ *and* $\alpha^*(R(x)) \cong b$.

Lemma 2.3. *$\alpha^{*-1}(B)$ and $\alpha^{*-1}(B^* \setminus B)$ are recursively separable.*

The definition of α^* shows that $\mathrm{Dom}(\alpha^*)$ is defined by the next inductive way:

a) $0 \in \mathrm{Dom}(\alpha^*)$ and if $i \in \mathrm{Dom}(\alpha)$, then $2(i+1) \in \mathrm{Dom}(\alpha^*)$;

b) If $i_1 \in \mathrm{Dom}(\alpha^*)$ and $i_2 \in \mathrm{Dom}(\alpha^*)$, then $\Pi(i_1, i_2) \in \mathrm{Dom}(\alpha^*)$.

Therefore,

Lemma 2.4. $\mathrm{Dom}(\alpha^*)$ *is r.e.*

Further, let the sequence of functions $\{\Pi_k\}_{k\in\mathbb{N}\setminus\{0\}}$ be defined in the following manner:

a) $\Pi_1(i_1) = 2(i_1+1)$;

b) $\Pi_{k+1}(i_1,\ldots,i_k,i_{k+1}) = \Pi(\Pi_k(i_1,\ldots,i_k),i_{k+1})$.

The next lemmas are obvious.

Lemma 2.5. *Let $i_1,\ldots,i_k$ be natural numbers and $\alpha(i_1)\cong s_1,\ldots,\alpha(i_k)\cong s_k$. Then $\alpha^*(\Pi_k(i_1,\ldots,i_k))\cong\langle s_1,\ldots,s_k\rangle$.*

Lemma 2.6. $\mathrm{Ran}(\alpha^*) = B^*$.

Let the functions $\varphi_1^*,\ldots,\varphi_k^*;\sigma_1^*,\ldots,\sigma_l^*$ be defined by the next equivalences:

$$\varphi_i^*(x)\cong y \iff \exists x_1\ldots\exists x_{a_i}(y\cong\Pi_1(\varphi_i(x_1,\ldots,x_{a_i}))\ \&\ x=\Pi_{a_i}(x_1,\ldots,x_{a_i})),$$

$i=1,\ldots,k$;

$$\sigma_j^*(x)\cong y \iff \exists x_1\ldots\exists x_{b_j}(y\cong\sigma_j(x_1,\ldots,x_{b_j})\ \&\ x=\Pi_{b_j}(x_1,\ldots,x_{b_j})),$$

$j=1,\ldots,l$.

From these definitions the next lemma follows immediately.

Lemma 2.7. *$\varphi_1^*,\ldots,\varphi_k^*,\sigma_1^*,\ldots,\sigma_l^*$ are partial recursive functions.*

Let $\mathbb{N}_k = \{\Pi_k(i_1,\ldots,i_k)\mid i_1\in\mathbb{N}\ \&\ \ldots\ \&\ i_k\in\mathbb{N}\}$.

Lemma 2.8. *Let $i\in\mathrm{Dom}(\alpha^*)$. Then for all natural $k\geqq 1$ the following equivalence is true:*

$$i\in\mathbb{N}_k \iff \alpha^*(i)\in B^k. \qquad (*)$$

Proof. By induction on k.

If $i\in\mathbb{N}_1$, then $i=\Pi_1(i_1)=2(i_1+1)$ for some natural i_1 and $\alpha^*(i)=\alpha(i_1)\in B$.

If $\alpha^*(i)\in B$, then it is clear that $i=2(i_1+1)$ and $i\in\mathbb{N}_1$.

Let us assume that the equivalence $(*)$ is true for some natural $k\geqq 1$.

If $i\in\mathbb{N}_{k+1}$, then $i=\Pi_{k+1}(i_1,\ldots,i_k,i_{k+1})=\Pi(\Pi_k(i_1,\ldots,i_k),i_{k+1})$ and let fix $i'=\Pi_k(i_1,\ldots,i_k)$. According to the induction hypothesis, $\alpha^*(i')\in B^k$ and $\alpha^*(i_{k+1})\in B$. Then $\alpha^*(i)\cong\alpha^*(\Pi(\Pi_k(i_1,\ldots,i_k),i_{k+1})))\cong\langle\alpha^*(i'),\alpha^*(i_{k+1})\rangle\in B^{k+1}$.

If $\alpha^*(i)\in B^{k+1}$, then $\alpha^*(i)$ is defined by the second clause of the definition, i. e. $\alpha^*(i)\cong\langle\alpha^*(i'),\alpha^*(i'')\rangle$, where $\alpha^*(i')\in B^k$, $\alpha^*(i'')\in B$ and $i=\Pi(i',i'')$. According to the induction hypothesis, $i'\in\mathbb{N}_k$ and $i''\in\mathbb{N}_1$. Thus $i\in\mathbb{N}_{k+1}$.

Lemma 2.9. *For any $x\in\mathbb{N}$ the following conditional equalities hold:*

$$\alpha^*(\varphi_i^*(x))\cong\theta_i(\alpha^*(x)),\quad i=1,\ldots,k.$$

Proof. We shall consider two cases.

Case 1. $x\notin\mathbb{N}_{a_i}$. Then $x\notin\mathrm{Dom}(\varphi_i)$, i. e. $\theta_i(\alpha^*(x))$ is not defined.

If $x\in\mathrm{Dom}(\alpha^*)$, then $\alpha^*(x)\notin B^{a_i}$, i. e. $\theta_i(\alpha^*(x))$ is not defined. If $x\notin\mathrm{Dom}(\alpha^*)$, then obviously $\theta_i(\alpha^*(x))$ is not defined.

Case 2. $x \in \mathbb{N}_{a_i}$. Then $x = \Pi_{a_i}(i_1, \ldots, i_{a_i})$ for some natural $i_1, \ldots, i_{a_i}$ and

$$\alpha^*(\varphi^*(x)) \cong \alpha^*(\Pi_1(\varphi_i(i_1, \ldots, i_{a_i})) \cong \alpha(\varphi_i(i_1, \ldots, i_{a_i})) \cong \theta_i(\alpha(i_1), \ldots, \alpha(i_{a_i}))$$

$$\cong \theta_i(\langle \alpha(i_1), \ldots, \alpha(i_{a_i}) \rangle) \cong \theta_i(\alpha^*(\Pi_{a_i}(i_1, \ldots, i_{a_i}))) \cong \theta_i(\alpha^*(x)).$$

Lemma 2.10. *For any* $x \in \mathbb{N}$ *the following conditional equalities hold:*

$$\sigma_j^*(x) \cong F_j(\alpha^*(x)), \quad j = 1, \ldots, l.$$

Proof. Analogously to Lemma 2.9.

So, we have that if we fix $\Pi' = \Pi$, $L' = L$ and $R' = R$, then the conditions (i) – (v) of extended effective enumeration are fulfilled.

Conversly, let a partial structure $\mathfrak{A} = (B; \theta_1, \ldots, \theta_k; F_1, \ldots, F_l)$ be given and the structure $\mathfrak{A}^* = (B^*; \theta_1, \ldots, \theta_k; F_1, \ldots, F_l)$ admit an extended effective enumeration $\langle \alpha^*, \mathfrak{B}^* \rangle$, where $\mathfrak{B}^* = (\mathbb{N}; \varphi_1, \ldots, \varphi_k; \sigma_1, \ldots, \sigma_l)$ is a partial structure with unary functions and predicates and α is a partial surjective mapping of $\mathbb{N}$ onto B^* such that the conditions (i) – (v) hold and the recursive functions Π', L', R' which satisfy (v) are fixed.

We shall define an enumeration $\langle \alpha, \mathfrak{B} \rangle$ of $\mathfrak{A}$. For this purpose for every positive natural number k we define the sets $\mathbb{N}'_k$, $\mathbb{N}''_k$ as follows:

$$\mathbb{N}'_k = \{x \mid x \in \mathrm{Dom}(\alpha^*) \;\&\; \alpha^*(x) \in B^k\}, \quad \mathbb{N}''_k = \{x \mid x \in \mathrm{Dom}(\alpha^*) \;\&\; \alpha^*(x) \notin B^k\}.$$

Then

$$\alpha(x) \cong \begin{cases} \alpha^*(x), & \text{if } x \in \mathbb{N}'_1, \\ \text{not defined}, & \text{otherwise.} \end{cases}$$

Lemma 2.11. $\mathrm{Dom}(\alpha)$ *is r.e.*

In this case we define the sequence $\{\Pi'_k\}_{k \in \mathbb{N} \setminus \{0\}}$ by means of the following inductive definition:

a) $\Pi'_1(i_1) = i_1$;

b) $\Pi'_{k+1}(i_1, \ldots, i_k, i_{k+1}) = \Pi'(\Pi'_k(i_1, \ldots, i_k), i_{k+1})$.

Lemma 2.12. *For every positive natural number* k, *if* $i_1 \in \mathrm{Dom}(\alpha)$ &...& $i_k \in \mathrm{Dom}(\alpha)$, *then* $\Pi'_k(i_1, \ldots, i_k) \in \mathrm{Dom}(\alpha^*)$ & $\alpha^*(\Pi'_k(i_1, \ldots, i_k)) \in B^k$ *and*

$$\langle \alpha(i_1), \ldots, \alpha(i_{a_i}) \rangle \cong \alpha^*(\Pi'_k(i_1, \ldots, i_k)).$$

Proof. By standard unduction on k.

Lemma 2.13. *For every positive natural number* k *there exists a recursive set* M_k *such that* $\mathbb{N}'_k \subseteq M_k$ *and* $\mathbb{N}''_k \subseteq \mathbb{N} \setminus M_k$.

Proof. By induction. If $k = 1$, then let M_1 be a recursive set such that $\alpha^{*-1}(B) \subseteq M_1$ and $\alpha^{*-1}(B^* \setminus B) \subseteq \mathbb{N} \setminus M_1$. Then $\mathbb{N}'_1 \subseteq M_1$ and $\mathbb{N}''_1 \subseteq \mathbb{N} \setminus M_1$.

Let us assume that there exists a recursive set M_k such that $\mathbb{N}'_k \subseteq M_k$ and $\mathbb{N}''_k \subseteq \mathbb{N} \setminus M_k$. Set $M_{k+1} = \{x \mid L'(x) \in M_k \;\&\; R'(x) \in M_1 \;\&x \neq 0\; \&x \notin M_1\}$.

If $x \in \mathbb{N}'_{k+1}$, then $x \in \mathrm{Dom}(\alpha^*)$ and $\alpha^*(x) = \langle b_1, b_2 \rangle$, where $\alpha^*(L'(x)) = b_1 \in B^k$ and $\alpha^*(R'(x)) = b_2 \in B$. Therefore, $x \in M_{k+1}$.

Let $x \in \mathbb{N}''_{k+1}$. Then $x = 0$ or $x \in M_1$ or $x \notin M_1$.

If $x = 0$ or $x \in M_1$, then it is obvious that $x \notin M_{k+1}$.

If $x \notin M_1$, then $x \notin N_1$, since $\alpha^*(x) \cong \langle b_1, b_2 \rangle \cong \alpha^*(L'(x)), \alpha^*(R'(x))\rangle$. Therefore, $b_1 \notin B^k$ or $b_2 \notin B$, i.e. $L'(x) \notin M_k$ or $R'(x) \notin M_1$. Again $x \notin M_{k+1}$ and Lemma 2.13 is proved.

Let us define the functions $\varphi_1, \ldots, \varphi_k, \sigma_1, \ldots, \sigma_l$ in the following way:

$$\varphi_i(x_1, \ldots, x_{a_i}) \cong \varphi_i^*(\Pi'_{a_i}(x_1, \ldots, x_{a_i})), \quad i = 1, \ldots, k,$$

$$\sigma_j(x_1, \ldots, x_{b_j}) \cong \sigma_j^*(\Pi'_{b_j}(x_1, \ldots, x_{b_j})), \quad j = 1, \ldots, l.$$

Lemma 2.14. *$\varphi_1, \ldots, \varphi_k, \sigma_1, \ldots, \sigma_l$ are partial recursive functions.*

Lemma 2.15. *For all i, $1 \leqq i \leqq k$, and for any natural numbers $x_1, \ldots, x_{a_i}$ the following conditional equalities hold:*

$$\alpha(\varphi_i(x_1, \ldots, x_{a_i})) \cong \theta_i(\alpha(x_1), \ldots, \alpha(x_{a_i})), \quad i = 1, \ldots, k.$$

Proof. $\alpha(\varphi_i(x_1, \ldots, x_{a_i})) \cong \alpha(\varphi_i^*(\Pi'_{a_i}(x_1, \ldots, x_{a_i})))$

$$\cong \alpha^*(\varphi_i^*(\Pi_{a_i}(x_1, \ldots, x_{a_i}))) \cong \theta_i^*(\alpha^*(\Pi'_{a_i}(x_1, \ldots, x_{a_i}))) \cong \theta_i(\langle\alpha(x_1), \ldots, \alpha(x_{a_i})\rangle)$$

$$\cong \theta_i(\alpha(x_1), \ldots, \alpha(x_{a_i})), \quad i = 1, \ldots, k.$$

Lemma 2.16. *For all j, $1 \leqq j \leqq l$, and for any natural numbers $x_1, \ldots, x_{b_j}$ the following conditional equalities hold:*

$$\sigma_j(x_1, \ldots, x_{b_j}) \cong F_j(\alpha(x_1), \ldots, \alpha(x_{b_j})), \quad j = 1, \ldots, l.$$

Proof. Analogously to Lemma 2.15.

Theorem 2.1 is proved.

Theorem 2.17. *A partial structure $\mathfrak{A}$ with a denumerable domain admits an effective enumeration iff the family of the types of all elements of the structure $\mathfrak{A}^*$ has an universal r.e. set U which satisfies the next conditions:*

(i) *The type of the element 0 is recursive set;*

(ii) *If $L_1 = \cup\{\mathbf{T}_{\mathfrak{A}^*}[s] \mid s \in B\}$ and $L_2 = \cup\{\mathbf{T}_{\mathfrak{A}^*}[s] \mid s \in B^* \setminus B\}$, then L_1 and L_2 are recursively separable;*

(iii) *There exist such total recursive functions Π', L', R' that:*

a) *If $U_{x_1} = \mathbf{T}_{\mathfrak{A}^*}[s_1]$ and $U_{x_2} = \mathbf{T}_{\mathfrak{A}^*}[s_2]$, then $\mathbf{T}_{\mathfrak{A}^*}[\langle s_1, s_2\rangle] = U_{\Pi'(x_1,x_2)}$;*

b) *If $\mathbf{T}_{\mathfrak{A}^*}[\langle s_1, s_2\rangle] = U_x$, then $U_{L'(x)} = \mathbf{T}_{\mathfrak{A}^*}[s_1]$ and $U_{R'(x)} = \mathbf{T}_{\mathfrak{A}^*}[s_2]$.*

Proof. Analogously to [8] suppose that the partial structure $\mathfrak{A}$ admits an effective enumeration $\langle\alpha, \mathfrak{B}\rangle$. Then the partial structure $\mathfrak{A}^*$ admits an extended effective enumeration $\langle\alpha^*, \mathfrak{B}^*\rangle$, where $\mathfrak{B}^* = (\mathbb{N}; \varphi_1^*, \ldots, \varphi_k^*; \sigma_1^*, \ldots, \sigma_l^*)$. According to [8] we can consider that α^* is totally defined over $\mathbb{N}$. A simple construction shows that there exists a primitive recursive in $\{\varphi_1^*, \ldots, \varphi_k^*, \sigma_1^*, \ldots, \sigma_l^*\}$ function Ψ such that for each functional termal formula Π^v with code v

$$\Psi(v, x) \cong \Pi^v_{\mathfrak{A}^*}(\alpha^*(x))$$

for all x of $\mathbb{N}$. Consequently, Ψ is partially recursive. Then it is obvious that the set

$$U = \{(x, v) \mid \Psi(v, x) \cong 0 \ \& \ v \text{ is a code of a functional termal formula}\}$$

is r.e. and universal for the family of the types of all elements of the structure $\mathfrak{A}^*$ which satisfies the conditions (i) – (iii).

Suppose now that the types of all elements of the structure $\mathfrak{A}^*$ are r.e. and that the family of all these types has an universal r.e. set U^1 which satisfies the conditions (i) – (iii). Let $U = \{(a, x) \mid U^1_a$ is a type of some element of $B\}$. It is obvious that the set U is r.e. and satisfies the conditions (i) – (iii), as well. We may assume that for every x there exist infinitely many y such that $U_x = U_y$ [cf. 7, 8].

Set

$$\varphi^*_i = \lambda x.\Pi(i, x), \quad i = 1, \ldots, k;$$
$$\Pi_0(x, y) = \Pi(0, \Pi(x, y));$$
$$\mathbb{N}_0 = \mathbb{N} \setminus (\mathrm{Ran}(\varphi^*_1) \cup \ldots \cup \mathrm{Ran}(\varphi^*_n) \cup \Pi_0).$$

For any natural number x, let B_x be the set $\{s \mid s \in B \ \& \ \mathbf{T}_{\mathfrak{A}^*}[s] = U_x\}$ of all elements of B with type U_x and α^0 be an arbitrary surjective mapping of $\mathbb{N}_0$ onto B, satisfying the equalities $\alpha^0(\{y \mid U_x = U_y\}) = B_x$, $x \in \mathbb{N}$. Evidently, $\mathrm{Dom}(\alpha^0) = \mathbb{N}_0$ is r.e.

We define the partial mapping α^* of $\mathbb{N}$ onto B^* by the inductive clauses:
If $x \in \mathbb{N}_0$, then $\alpha^*(x) \cong \alpha^0(x)$;
If $x = \Pi(i, y)$, $1 \leqq i \leqq k$, $\alpha^*(y) \cong s$ and $\theta_i(s) \cong t$, then $\alpha^*(x) \cong t$;
If $z = \Pi(0, \Pi(x, y))$, $\alpha^*(x) \cong s_1$ and $\alpha(y)^* \cong s_2$, then $\alpha^*(z) \cong \langle s_1, s_2 \rangle$.
The proofs of the next simple lemmas are analogous of those in [7, 8].

Lemma 2.18. *For every $x \in \mathbb{N}$ and i, $1 \leqq i \leqq k$,*

$$\alpha(\varphi^*_i(x)) \cong \alpha^*(\langle i, x \rangle) \cong \theta_i(\alpha^*(x)).$$

Let us denote by $\overline{\mathfrak{B}}$ the partial structure $(\mathbb{N}; \varphi^*_1, \ldots, \varphi^*_k)$.

Corollary 2.19. *Let τ be a functional term and $y \in \mathbb{N}$. Then*

$$\alpha^*(\tau_{\overline{\mathfrak{B}}}(y)) \cong \tau_{\mathfrak{A}^*}(\alpha^*(y)).$$

Lemma 2.20. *There exists an effective way to define, for every x of $\mathbb{N}$, an element y of $\mathbb{N}_0$ and a functional term τ such that $x = \tau_{\overline{\mathfrak{B}}}(y)$.*

Lemma 2.21. *There exists an effective way to define, for every x of $\mathbb{N}$, an element y of $\mathbb{N}_0$ and a functional term τ such that $\alpha^*(x) \cong \tau_{\mathfrak{A}^*}(\alpha^*(y))$.*

Lemma 2.22. $\mathrm{Dom}(\alpha)^*$ *is recursively enumerable.*

Finally, let us define the partial predicates $\sigma^*_1, \ldots, \sigma^*_k$ on N using the conditional equalities

$$\sigma^*_j(x) \cong \begin{cases} 0, & \text{if } F_j(\alpha^*(x)) \cong 0, \\ 1, & \text{if } \neg F_j(\alpha^*(x)) \cong 0, \\ \text{undefined}, & \text{otherwise}, \end{cases}$$

$j = 1, \ldots, l$. Analogously, it follows:

Lemma 2.23. *The predicates $\sigma_1^*, \ldots, \sigma_l^*$ are partially recursive.*

Thus, it is proven that $\langle \alpha^*, (\mathbb{N}; \varphi_1^*, \ldots, \varphi_k^*; \sigma_1^*, \ldots, \sigma_l^*) \rangle$ is an extended effective enumeration of the structure $\mathfrak{A}^*$.

It is easy to see that the next theorem is also valid.

Theorem 2.24. *A partial structure $\mathfrak{A}$ with a denumerable domain admits an effective enumeration iff the family of the types of all elements of the structure $\mathfrak{A}$ has an universal r.e. set U such that there exist total recursive functions Π', L', R' satisfying the conditions:*

*) *If $W_x = \mathbf{T}_{\mathfrak{A}^*}[\langle s_1, s_2 \rangle]$, then $W_{L'(x)} = \mathbf{T}_{\mathfrak{A}^*}[s_1]$ and $W_{R'(x)} = \mathbf{T}_{\mathfrak{A}^*}[s_2]$;*

**) *If $W_{x_1} = \mathbf{T}_{\mathfrak{A}^*}[s_1]$ and $W_{x_2} = \mathbf{T}_{\mathfrak{A}^*}[s_2]$, then $\mathbf{T}_{\mathfrak{A}^*}[\langle s_1, s_2 \rangle] = W_{\Pi'(x_1, x_2)}$.*

Here we use W_e to denote the e-th recursively enumerable (r.e.) set.

REFERENCES

1. Ershov, Y. L. Decision problems and constuctivizable models. Nauka, Moscow, 1980 (in Russian).
2. Soskova, A. A., I. N. Soskov. Effective enumerations of abstract structures. In: *Heyting'88: Mathematical Logic* (ed. P. Petkov), Plenum Press, New York — London, 1989.
3. Jockusch, C. G., Jr. and A. Shlapentokh. Weak presentations of computable fields. *J. Symbolic Logic*, **60**, 1, 1995, 199–208.
4. Soskov, I. N. Definability via enumerations. *J. Symbolic Logic*, **54**, 2, 1989.
5. Soskov, I. N. Computability by means of effective definable schemes and definability via enumerations. *Arch. for Math. Log.*, **29**, 1990, 187–200.
6. Chisholm, J. The complexity of intrincically r.e. subset of existentially decidable models. *J. Symbolic Logic*, **55**, 3, 1990, 1213–1232.
7. Ditchev, A. V. On the effective enumerations of partial structures. *Ann. de l'Université de Sofia, Fac. de Math. et Inf.*, **83**, 1, 1989, 29–37.
8. Ditchev, A. V. On the effective enumerations of structures (submited for publication).
9. Ditchev, A. V. Examples of structures which do not admit recursive presentations. *Ann. de l'Université de Sofia, Fac. de Math. et Inf.*, **85**, 1, 1991, 3–11.

Received January 19, 1999

Faculty of Mathematics and Informatics
"St. Kliment Ohridski" University of Sofia
5 James Bourchier Blvd.
BG-1164 Sofia, Bulgaria
E-mail: ditchev@fmi.uni-sofia.bg

ГОДИШНИК НА СОФИЙСКИЯ УНИВЕРСИТЕТ „СВ. КЛИМЕНТ ОХРИДСКИ“
ФАКУЛТЕТ ПО МАТЕМАТИКА И ИНФОРМАТИКА
Книга 1 — Математика и механика
Том 92, 1998

ANNUAIRE DE L'UNIVERSITE DE SOFIA „ST. KLIMENT OHRIDSKI“
FACULTE DE MATHEMATIQUES ET INFORMATIQUE
Livre 1 — Mathématiques et Mecanique
Tome 92, 1998

VARIATIONS OF HODGE STRUCTURE, EXPRESSED BY MEROMORPHIC DIFFERENTIALS ON THE PROJECTIVE PLANE

AZNIV K. KASPARIAN

The tautological variations of Hodge structure over Siegel upper half space, the open quadric and the generalized ball are expressed explicitely by the variations of Hodge structure of Weil hypersurfaces in projective spaces. That realizes all the abelian-motivic variations of Hodge structure by families of Jacobians of plane curves, which are known to be described by meromorphic differentials on the projective plane. As a consequence, the geometric origin of a maximal dimensional variation of Hodge structure turns to be sufficient for expressing it by meromorphic differentials on the projective plane.

Keywords: tautological variations of Hodge structure and J-Hodge structure, abelian-motivic and hypersurface variations

1991/95 Math. Subject Classification: 14D07, 14K10

For a smooth projective manifold X, defined over a field $k \subset \mathbb{C}$ of finite type, Hodge has conjectured that $H^{2w}(X, \mathbb{Q}) \cap H^{w,w}$, $w \leq \dim_{\mathbb{C}} X$, consists of the $\mathbb{Q}$-linear combinations of the cohomology classes of the algebraic submanifolds of X. Let $\overline{k}$ be the algebraic closure of k and $H^{2w}_{sc}(X, \mathbb{Q}_l) \subset H^{2w}(X, \mathbb{Q}_l)$ be the subspace of l-adic cohomologies, over which the action of the Galois group $Gal(\overline{k}/k)$ reduces to multiplication by scalars. Tate has conjectured that $H^{2w}_{sc}(X, \mathbb{Q}_l)$ is generated by the cohomologies of the algebraic submanifolds of X. For abelian varieties X, Tate conjecture is known to imply Hodge conjecture (cf. Deligne [5]), but neither of them is proved. Let X be a separable scheme of finite type over $\mathbf{F}_q$ and $\overline{X}$ be the scheme obtained from X by extension of the scalars to $\overline{\mathbf{F}_q}$. For primes $l \neq q$, the l-adic cohomologies $H^w_c(\overline{X}, \mathbb{Q}_l)$ with compact support are acted by Frobenius

automorphism $\varphi \in Gal(\overline{\mathbf{F}_q}/\mathbf{F}_q)$, $\varphi(x) = x^q$. Weil conjecture asserts that the characteristic roots of φ^{-1} on $H_c^w(\overline{X}, \mathbb{Q}_l)$ are of absolute value $q^{-\frac{w}{2}}$. It is verified for the abelian varieties and therefore, for any X whose cohomologies are expressed through linear algebra constructions (by the cohomologies) of abelian varieties. Hodge, Tate and Weil conjectures motivate the interest in the abelian varieties A and their Hodge structures $H^w(A, \mathbb{Z}) = \wedge^w H^1(A, \mathbb{Z}) \simeq \wedge^w A$.

The present work concerns the variations of Hodge structure which are expressed by a special kind of abelian varieties, namely, by Jacobians of plane curves. The members of a family $\mathcal{J} \to S$ of Jacobians of plane curves, as well as the infinitesimal variations $T_s^{1,0}S$, $s \in S$, can be identified with subspaces of meromorphic differentials on $\mathbf{P}_2$. We exhibit explicit embeddings of the so-called tautological variations over Siegel upper half space $\mathcal{S}(p)$, the open quadric $\mathcal{Q}(p)$ and the generalized ball $\mathcal{B}(p,q)$, in the variations of Hodge structure of Weil hypersurfaces $X \subset \mathbf{P}_N$. Shermenev shows in [11] that the Hodge structures of X are expressed by meromorphic differentials on $\mathbf{P}_2$. So far, Kuga and Satake [8], Deligne [4], Carlson and Simpson [3] have established that the aforementioned tautological variations are expressed by abelian varieties. The abelian varieties are known to be from the tensor category of the Jacobians of all curves (cf. [12]). On the other hand, Rapoport [10] has classified the complete intersections Y, whose variations of Hodge structure are exactly the tautological variations over $\mathcal{S}(p)$ or $\mathcal{Q}(p)$. All such Y turn to be of Hodge level 1 or 2. Our X are of arbitrary Hodge level, equal to $\dim_{\mathbb{C}} X$, and we realize the tautological variations as proper subfamilies of the variations of X.

The provided construction reveals that all the abelian-motivic variations of Hodge structure are expressed by meromorphic differentials on $\mathbf{P}_2$. As another consequence, the geometrically arising variations of maximum dimension turn to be realized in the tensor category of Jacobians of plane curves.

1. PRELIMINARIES

1.1. TAUTOLOGICAL VARIATIONS OF HODGE STRUCTURE AND J-HODGE STRUCTURE

Hodge structure on a $\mathbb{C}$-vector space $V = V_{\mathbb{Q}} \otimes_{\mathbb{Q}} \mathbb{C}$ defined over $\mathbb{Q}$ consists of Hodge decomposition $V = \sum_{i=0}^{w} V^{w-i,i}$, compatible with the complex conjugation $\overline{V^{w-i,i}} = V^{i,w-i}$, and a non-degenerate bilinear polarization form $\Psi : V_{\mathbb{Q}} \otimes_{\mathbb{Q}} V_{\mathbb{Q}} \to \mathbb{Q}$, which is symmetric for an even weight w or skew-symmetric for an odd w. Hodge decomposition is orthogonal with respect to the Hermitian form $\Phi(a,b) := \Psi(a, \bar{b})$ for $a, b \in V$, and $\Phi|_{V^{w-2i,2i}} > 0$, $\Phi|_{V^{w-2i-1,2i+1}} < 0$. J-Hodge structure on $V = V_{\mathbb{Q}} \otimes_{\mathbb{Q}} \mathbb{C}$ is Hodge structure with an endomorphism $J : V_{\mathbb{Q}} \to V_{\mathbb{Q}}$, $J^2 = -Id$, such that J is orthogonal with respect to Ψ, unitary with respect to Φ and compatible with Hodge decomposition $J : V^{w-i,i} \to V^{w-i,i}$.

The classifying space of Hodge structures on V with fixed bilinear form Ψ, Hermitian form Φ, and Hodge numbers $h^i := \dim_{\mathbb{C}} V^{w-i,i} > 0$ is the homogeneous

space

$$D(V,\Psi,\Phi)=O(V,\Psi)\cap U(V,\Phi)/\prod_{i=0}^{[\frac{w}{2}]-1}U(h^i)\times(1-\varepsilon(w))O(h^{[\frac{w}{2}]},\mathbb{R}),$$

where $O(V,\Psi)$ is the orthogonal group of V with respect to Ψ, $O(h^{[\frac{w}{2}]},\mathbb{R})=O\left(V^{[\frac{w}{2}],[\frac{w}{2}]},\Psi|_{V^{[\frac{w}{2}],[\frac{w}{2}]}}\right)$, $U(V,\Phi)$ is the unitary group of V with respect to Φ, $U(h^i)=U\left(V^{w-i,i},\Phi|_{V^{w-i,i}}\right)$, and $\varepsilon(w):=w-2[\frac{w}{2}]$ stands for the parity of the weight w.

The semisimple linear automorphisms $J|_{V^{w-i,i}}$ split $V^{w-i,i}=V_+^{w-i,i}\oplus V_-^{w-i,i}$ in $\pm\sqrt{-1}$-eigenspaces with $\overline{V_+^{w-i,i}}=V_-^{i,w-i}$. As a result, Hodge decomposition of the Ψ-isotropic $V_+:=\sum_{i=0}^{w}V_+^{w-i,i}$ determines completely Hodge decomposition of $V=V_+\oplus\overline{V_+}$, and the classifying space of J-Hodge structures on V turns to be

$$D(V,\Psi,\Phi,J)\simeq U(h_+^0+h_+^2+\cdots,h_+^1+h_+^3+\cdots)/U(h_+^0)\times\cdots\times U(h_+^w),$$

where

$$h_+^i:=\dim_{\mathbb{C}}V_+^{w-i,i},$$
$$U(h_+^0+h_+^2+\cdots,h_+^1+h_+^3+\cdots)=U\left(V_+,\Phi|_{V_+}\right),$$
$$U(h_+^i)=U\left(V_+^{w-i,i},\Phi|_{V_+^{w-i,i}}\right).$$

The classifying spaces $D=D(V,\Psi,\Phi)$, respectively, $D=D(V,\Psi,\Phi,J)$ are open subsets of quotients $\check{D}=O(V,\Psi)/P(V)$, respectively, $\check{D}=GL(V_+,\mathbb{C})/P(V_+)$ of reductive complex algebraic groups $G^{\mathbb{C}}$ by parabolic subgroups P, stabilizing Hodge filtrations $F^i:=\sum_{j\geq i}V^{j,w-j}$, respectively, $F_+^i:=\sum_{j\geq i}V_+^{j,w-j}$. Hodge decompositions of V, V_+ induce weight zero Hodge decompositions $\mathfrak{g}^{\mathbb{C}}=LieG^{\mathbb{C}}=\sum_{i=-w}^{w}\mathfrak{g}^{i,-i}$ with $\mathfrak{g}^{i,-i}=\{\tau\in\mathfrak{g}^{\mathbb{C}}|\tau(V^{j,w-j})\subseteq V^{i+j,w-i-j}\text{ for all }0\leq j\leq w\}$. The parabolic subalgebras $LieP=\sum_{i\geq 0}\mathfrak{g}^{i,-i}$. The holomorphic tangent bundle $T^{1,0}D=T^{1,0}\check{D}|_D=[G^{\mathbb{C}}\times_P(LieG^{\mathbb{C}}/LieP)]_D$ contains an equivariant subbundle $T^hD:=\left[G^{\mathbb{C}}\times_P(\mathfrak{g}^{-1,1}+LieP)/LieP\right]_D$, associated with a non-integrable distribution and called horizontal. As far as an arbitrary family of Hodge structures with fixed Ψ, Φ, h^i is induced by the tautological family over D, there is no loss in regarding the base S of this family as a complex analytic subspace of $\Gamma\setminus D$ for some discrete subgroup Γ of the biholomorphism group G of $D=G/G\cap P$. Variation of Hodge structure is a family $\mathcal{V}\to S$, whose base S is locally tangent to the horizontal distribution T^hD. The complete tautological families of Hodge structures over $D(V,\Psi,\Phi)$ or, respectively, the complete tautological families of J-Hodge structures over $D(V,\Psi,\Phi,J)$, which are variations of Hodge structure are referred to as tautological variations.

Lemma 1. *All the tautological variations of Hodge structure are*

(i) $\mathcal{V}_{\mathcal{S}(p)}=\sum_{i=0}^{1}\mathcal{V}_{\mathcal{S}(p)}^{1-i,i}$ *of* $rank\mathcal{V}_{\mathcal{S}(p)}^{1-i,i}=p$ *over Siegel upper half spaces* $\mathcal{S}(p)=Sp(p,\mathbb{R})/U(p)$ *and*

(ii) $\mathcal{V}_{\mathcal{Q}(p)} = \sum_{i=0}^{2} \mathcal{V}_{\mathcal{Q}(p)}^{2-i,i}$ *of* $rank\mathcal{V}_{\mathcal{Q}(p)}^{2,0} = rank\mathcal{V}_{\mathcal{Q}(p)}^{0,2} = 1$, $rank\mathcal{V}_{\mathcal{Q}(p)}^{1,1} = p$ *over open quadrics* $\mathcal{Q}(p) = SO(2,p)/SO(2) \times SO(p)$.

All the tautological variations of J-Hodge structure are

(iii) $\mathcal{V}_{J,p,q}^{w} = \sum_{i=0}^{1} \left(\mathcal{V}_{+}^{w-i,i} + \overline{\mathcal{V}_{+}^{w-i,i}} \right)$ *of weight* $1 \leq w \leq 3$ *and* $rank\mathcal{V}_{+}^{w,0} = p$, $rank\mathcal{V}_{+}^{w-1,1} = q$ *over generalized balls* $\mathcal{B}(p,q) = U(p,q)/U_p \times U_q$.

The corresponding polarizations Ψ_D, $\Psi_{J,p,q}^{w}$ *are*

$$\begin{pmatrix} & I_p \\ -I_p & \end{pmatrix}, \quad \begin{pmatrix} & & 1 \\ & -I_p & \\ 1 & & \end{pmatrix}, \quad \begin{pmatrix} & & & I_p \\ & & (-1)^{w-1} I_q & \\ & -I_q & & \\ (-1)^{w} I_p & & & \end{pmatrix}.$$

Proof. The existence of tautological variations is equivalent to $T^{1,0}D = T^h D$ for the corresponding classifying space D. In the case of $w = 2k+1 \geq 3$ the symplectic Lie algebra $\mathfrak{g}^{\mathbb{C}} = o(V,\Psi)$ has $\mathfrak{g}^{-2,2} \neq 0$. For an even weight $w = 2k \geq 2$, the indefinite orthogonal Lie algebra $\mathfrak{g}^{\mathbb{C}} = o(V,\Psi) = \mathfrak{g}^{-1,1} + LieP$ if and only if $w = 2$ and $h^0 = 1$. That justifies the classification of the tautological variations of Hodge structure. In the case of J-Hodge structures one can assume that $h_{+}^{0} \neq 0$, after eventual shift $(w-i,i) \mapsto (i,w-i)$ of Hodge indices. Then $T^{1,0}D(V,\Psi,\Phi,J) = T^h D(V,\Psi,\Phi,J)$ holds only when $D(V,\Psi,\Phi,J) \simeq U(h_{+}^{0},h_{+}^{1})/U(h_{+}^{0}) \times U(h_{+}^{1})$. The weights $w \leq 3$, since otherwise for $2 \leq j \leq w-2$ there follow $h_{+}^{j} = 0$ and $h_{-}^{j} = 0$, contrary to the assumption $h^j \neq 0$. □

1.2. THE TAUTOLOGICAL VARIATIONS ARE ABELIAN-MOTIVIC

A variation is said to be abelian-motivic or expressed by abelian varieties if it is a direct summand of a tensor polynomial with $\mathbb{N}$-coefficients of variations of Hodge structure of abelian varieties. All the tautological variations are expressed by abelian varieties. More precisely:

Theorem 2. (i) (obvious) *The tautological variation* $\mathcal{V}_{\mathcal{S}(p)}$ *is the variation of Hodge structure of a polarized abelian variety* $A \simeq H^1(A,\mathbb{C})$ *of* $\dim_{\mathbb{C}} A = p$.

(ii) (Kuga and Satake [8], Deligne [4]) *Let* $C^+(V,\Psi_{\mathcal{Q}(p)})$ *be the even part of the Clifford algebra* $C(V,\Psi_{\mathcal{Q}(p)})$ *of the reference Hodge structure* $(V,\Psi_{\mathcal{Q}(p)}) \in \mathcal{V}_{\mathcal{Q}(p)}$. *Then there is a family* $\mathcal{A} \to \mathcal{Q}(p)$ *of* 2^{p+1}*-dimensional abelian varieties such that the variation of* $C^+(V,\Psi_{\mathcal{Q}(p)})$ *is*

$$C^+\left(\mathcal{V}_{\mathcal{Q}(p)},\Psi_{\mathcal{Q}(p)}\right) = End_{C^+(V,\Psi_{\mathcal{Q}(p)})}(\mathcal{A}).$$

(iii) (Carlson and Simpson [3]) *The tautological variation of J-Hodge structure* $\mathcal{V}_{J,p,q}^{1}$ *is the restriction of* $\mathcal{V}_{\mathcal{S}(p+q)}$ *to a holomorphically and equivariantly embedded* $\mathcal{B}(p,q) \hookrightarrow \mathcal{S}(p+q)$. *Let* E *be the elliptic curve* $\mathbb{C}/(\mathbb{Z}+\sqrt{-1}\mathbb{Z})$, J *be the endomorphism of* $H^1(E,\mathbb{C})$ *induced from the multiplication by* $\sqrt{-1}$ *on* E, $\mathbf{E} = \mathbf{E}^{1,0} + \mathbf{E}^{0,1}$ *be the constant family of the aforementioned J-Hodge structure and* $\mathbf{E}^{r,s}(m)$ *be the*

m-th tensor power of $\mathbf{E}^{r,s}$. *Then*

$$\mathcal{V}^w_{J,p,q} = \mathcal{V}^1_+ \otimes \mathbf{E}^{1,0}(w-1) \oplus \overline{\mathcal{V}^1_+} \otimes \mathbf{E}^{0,1}(w-1)$$

of weight $w = 2$ *or* 3 *are expressed by* $\mathcal{V}^1_{J,p,q} = \mathcal{V}^1_+ \oplus \overline{\mathcal{V}^1_+}$.

1.3. COMPLETE INTERSECTIONS WITH TAUTOLOGICAL VARIATIONS OF HODGE STRUCTURE

Let $X_n^{d_1,\dots,d_k} \subset \mathbf{P}_{n+k}$ be a complete intersection of hypersurfaces of degree $d_1, \dots, d_k$. The primitive cohomologies $H^*(X_n^{d_1,\dots,d_k}, \mathbb{C})_o$, i.e., the cohomologies which are not dual to intersections of $X_n^{d_1,\dots,d_k}$ with subspaces $\mathbf{P}_m \subset \mathbf{P}_{n+k}$, have only nonzero components $H^n(X_n^{d_1,\dots,d_k}, \mathbb{C})_o$. From now on, under a variation of Hodge structure of a complete intersection $X_n^{d_1,\dots,d_k}$ we mean the variation of Hodge structure on $H^n(X_n^{d_1,\dots,d_k}, \mathbb{C})_o$. If $h^j := \dim_{\mathbb{C}} H^{n-j,j}(X_n^{d_1,\dots,d_k})_o$ vanish for all $j < i$ and $j > n - i$, $h^i = h^{n-i} \neq 0$, then the integer $n - 2i$ (which is one less that the number of the non-trivial Hodge components of $H^n(X_n^{d_1,\dots,d_k}, \mathbb{C})_o$) is called level of $X_n^{d_1,\dots,d_k}$ or of its Hodge structure.

Theorem 3 (Rapoport [10]). (i) *All the complete families* $\mathcal{X}^{d_1,\dots,d_k}_{n,\mathcal{S}(p)}$ *of* $X_n^{d_1,\dots,d_k} \subset \mathbf{P}_{n+k}$, *whose associated variations of Hodge structure are discrete quotients of level one tautological variations over Siegel upper half spaces* $\mathcal{S}(p)$, *are* $\mathcal{X}^{2,2}_{2n-1,\mathcal{S}(n)}$, $\mathcal{X}^{2,2,2}_{2n-1,\mathcal{S}(2n^2+3n)}$, $\mathcal{X}^{3}_{3,\mathcal{S}(5)}$, $\mathcal{X}^{2,3}_{3,\mathcal{S}(20)}$, $\mathcal{X}^{3}_{5,\mathcal{S}(21)}$, $\mathcal{X}^{4}_{3,\mathcal{S}(30)}$.

(ii) *The complete families* $\mathcal{X}^{d_1,\dots,d_k}_{n,\mathcal{Q}(p)}$ *of* $X_n^{d_1,\dots,d_k} \subset \mathbf{P}_{n+k}$, *whose associated variations of Hodge structure are discrete quotients of level two tautological variations over open quadrics* $\mathcal{Q}(p)$, *are depleted by the families* $\mathcal{X}^{2,2,2}_{2,\mathcal{Q}(19)}$, $\mathcal{X}^{2,3}_{2,\mathcal{Q}(19)}$, $\mathcal{X}^{4}_{2,\mathcal{Q}(19)}$ *of* $K3$ *surfaces and the family* $\mathcal{X}^{3}_{4,\mathcal{Q}(20)}$ *of cubic fourfolds.*

2. EXPLICIT CONSTRUCTIONS

Let us fix some standard notations. The Hodge structure on the second cohomology group $H^2(\mathbf{P}_1, \mathbb{C}) = H^{1,1}(\mathbf{P}_1)$ of the projective line $\mathbf{P}_1$ or, equivalently, on the cup product $\wedge^2 H^1(E, \mathbb{C}) = H^{1,0}(E) \wedge H^{0,1}(E)$ of the first cohomology group of the elliptic curve $E = \mathbb{C}/(\mathbb{Z} + \sqrt{-1}\mathbb{Z})$ is called Tate Hodge structure. The constant family of Tate Hodge structures (over an arbitrary base) is denoted by $\mathbb{Q}(1)$. If $m \in \mathbb{N}$, then $\mathbb{Q}(m)$ and $\mathbb{Q}(-m)$ designate the m-th tensor powers of $\mathbb{Q}(1)$ and, respectively, $\mathbb{Q}(-1) = Hom(\mathbb{Q}(1), \mathbb{C})$. The polarization $\Psi^{\mathbf{E}} = \begin{pmatrix} & 1 \\ -1 & \end{pmatrix}$ of the constant family $\mathbf{E}$ of J-Hodge structures on E induces the polarization $\Psi^{\mathbb{Q}(1)} = \Psi^{\mathbf{E}} \otimes \Psi^{\mathbf{E}} = 1$. In other words, all $\Psi^{\mathbb{Q}(m)}$, $m \in \mathbb{Z}$, coincide with the multiplication by complex numbers.

Theorem 4. *For the Hermitian symmetric spaces*

$$D = \mathcal{S}(p) = \{z \in Mat_{p,p}(\mathbb{C}) \mid {}^t z = z, z^t\overline{z} < I_p\} \text{ or}$$
$$D = \mathcal{B}(p,q) = \{z \in Mat_{p,q}(\mathbb{C}) \mid z^t\overline{z} < I_p\}, \quad p \leq q,$$

let $\delta = p+1$ *or, respectively,* $\delta = q+1$ *and consider the hypersurfaces* $\mathcal{X}_D(z) \subset \mathbf{P}_{2p+2}$, $z \in D$, *determined by the homogeneous equations*

$$\sum_{i=-1}^{0} x_i^{4\delta} + \sum_{i=1}^{p} \left(x_{2i-1}^{4\delta} + \frac{1}{2\delta} \sum_{j=1}^{\delta-1} z_{ij} x_{2i-1}^{4\delta-4j} x_{2i}^{4j} + x_{2i}^{4\delta} \right) + x_{2p+1}^{4\delta} = 0$$

of degree $d = 4\delta$. *Over the open quadric*

$$D = \mathcal{Q}(p) = \{ z \in Mat_{p,1}(\mathbb{C}) \mid |{}^t zz| < 1, 2{}^t\overline{z}z < 1 + |{}^t zz|^2 \}$$

define the family $\mathcal{X}_{\mathcal{Q}(p)}$ *of hypersurfaces*

$$\mathcal{X}_{\mathcal{Q}(p)}(z) = \left\{ x \in \mathbf{P}_{2p-1} \mid \sum_{i=0}^{p-1} \left(x_{2i-1}^{4p} + \frac{1}{2p} z_i x_{2i-1}^{2p} x_{2i}^{2p} + x_{2i}^{4p} \right) = 0 \right\}$$

of degree $d = 4p$. *Let us denote by* $\mathcal{H}_D$ *the variations of Hodge structure of* $\mathcal{X}_D$ *with polarizations* $\Psi^{\mathcal{H}_D}$. *Put* $w_{\mathcal{S}(p)} := 1$, $w_{\mathcal{Q}(p)} := 2$ *for the weights* w_D *of the tautological variations of Hodge structure* $\mathcal{V}_D$, δ_j^i *for Kronecker's delta and introduce* $m_D := p - \delta_{w_D}^2 w_D$. *Then the components of the tautological variations of Hodge structure are the subbundles*

$$\mathcal{V}_D^{i,w_D-i} \subset \mathcal{H}_D^{m_D+i,m_D+w_D-i} \otimes \mathbb{Q}(-m_D)$$

of abelian-motivic variations, expressed by meromorphic differentials on $\mathbf{P}_2$, *and* $\Psi_D = \Psi^{\mathcal{H}_D} \otimes \Psi^{\mathbb{Q}(-m_D)}|_{\mathcal{V}_D}$. *The tautological variations of J-Hodge structure have*

$$\mathcal{V}_{J,p,q}^{i,w-i} \subset \sum_{j=0}^{1} \left(\sum_{k=0}^{1} \delta_{j(w-1)+k}^{i} \right) \mathcal{H}_{\mathcal{B}(p,q)}^{p+i-j(w-1),p+1-i+j(w-1)} \otimes \mathbf{E}^{j,1-j}(w-1) \otimes \mathbb{Q}(-p)$$

for $1 \le w \le 3$, $0 \le i \le w$, *expressed by meromorphic differentials on* $\mathbf{P}_2$, *and* $\Psi_{J,p,q}^{w} = \Psi^{\mathcal{H}_{\mathcal{B}(p,q)}} \otimes \Psi^{\mathbf{E}(w-1)} \otimes \Psi^{\mathbb{Q}(-p)}\Big|_{\mathcal{V}_{J,p,q}^{w}}$.

The proof is subdivided into several steps and presented by the rest of the section.

2.1. A SMOOTH FAMILY $\mathcal{X}$ OF HYPERSURFACES OVER A PRODUCT OF BALLS

Lemma 5. *All the hypersurfaces* $\mathcal{X}(z) = \{ x \in \mathbf{P}_{n+1} \mid f_z(x) = 0 \}$ *with homogeneous equations*

$$f_z(x) = \sum_{i=0}^{[\frac{n}{2}]} \left(x_{2i-1}^{d} + \frac{2}{d} \sum_{j=1}^{d-1} z_{ij} x_{2i-1}^{d-j} x_{2i}^{j} + x_{2i}^{d} \right) + \varepsilon(n) x_n^d = 0$$

of degree $d > n+2 \ge 3$, *parametrized by the product of balls*

$$\mathcal{B}(1,d-1)^{[\frac{n}{2}]+1}=\left\{z\in Mat_{[\frac{n}{2}]+1,d-1}(\mathbb{C})\ \Big|\ \sum_{j=1}^{d-1}|z_{ij}|^2<1,\ \ \forall i\ 0\le i\le\left[\frac{n}{2}\right]\right\},$$

are smooth.

Proof. One has to justify that the system of the polynomial equations $\frac{\partial f_z}{\partial x_{2i-1}}=\frac{\partial f_z}{\partial x_{2i}}=0$ with $0\le i\le\left[\frac{n}{2}\right]$ and $\frac{\partial f_z}{\partial x_n}=0$ for an odd n has only the trivial solution $x_{-1}=x_0=x_1=\cdots=x_n=0$. As far as the pairs $\frac{\partial f_z}{\partial x_{2i-1}}$, $\frac{\partial f_z}{\partial x_{2i}}$ depend only on x_{2i-1}, x_{2i}, this system splits in $\left[\frac{n}{2}\right]+1$ parts of two equations with two variables. For x_{2i-1}, x_{2i} with $|x_{2i-1}|\le|x_{2i}|\neq 0$ one puts $y:=\frac{x_{2i-1}}{x_{2i}}$ in order to express $\frac{\partial f_z}{\partial x_{2i}}$ as $\frac{2}{d}\sum_{j=1}^{d-1}jz_{ij}y^{d-j}+d=0$. According to Cauchy–Schwarz inequality,

$$\left(\frac{d^2}{2}\right)^2=\left|\sum_{j=1}^{d-1}z_{ij}(jy^{d-j})\right|^2\le\left(\sum_{j=1}^{d-1}|z_{ij}|^2\right)\left(\sum_{j=1}^{d-1}j^2|y|^{2(d-j)}\right).$$

Bearing in mind that $z\in\mathcal{B}(1,d-1)^{[\frac{n}{2}]+1}$ and $|y|\le 1$, one infers that

$$\left(\frac{d^2}{2}\right)^2<\sum_{j=1}^{d-1}j^2=\frac{(d-1)d(2d-1)}{6}<\frac{2d^3}{6},$$

which contradicts $d\ge 3$. Similarly, for x_{2i-1}, x_{2i} with $|x_{2i}|\le|x_{2i-1}|\neq 0$ one introduces $t:=\frac{x_{2i}}{x_{2i-1}}$. Then converting the equation $\frac{\partial f_z}{\partial x_{2i-1}}=0$ into the form $d+\frac{2}{d}\sum_{k=1}^{d-1}kz_{id-k}t^{d-k}=0$, one gets an absurd. □

2.2. EXPRESSING THE VARIATION OF HODGE STRUCTURE OF $\mathcal{X}$ BY MEROMORPHIC DIFFERENTIALS ON $\mathbf{P}_2$

Suppose that X_n of $\dim_{\mathbb{C}}X_n=n\ge 2$ is a hypersurface from the constructed smooth family, X_{n-1} is the intersection of X_n with the hyperplane $x_{-1}=0$, and Y_1 is the plane curve with homogeneous equation $y_1^d=y_{-1}^d+\frac{2}{d}\sum_{j=1}^{d-1}z_{0j}y_{-1}^{d-j}y_0^j+y_0^d$. The presence of a rational map $X_{n-1}\times Y_1\to X_n$,

$$((x_0:x_1:\ldots:x_n),(y_{-1}:y_0:y_1))\mapsto(x_0y_{-1}:x_0y_0:x_1y_1:\ldots x_iy_1:\ldots:x_ny_1)$$

of degree d with singular locus $(X_{n-1}\cap\{x_0=0\})\times(Y_1\cap\{y_1=0\})$ justifies the next

Lemma 6 (Shermenev [11], Shioda and Katsura [13]). *In the notations, introduced in Lemma 5, let us fix a hypersurface $X_n := \mathcal{X}(z)$ of dimension $n \geq 2$ and consider the complete intersections $X_{n-1} := X_n \cap \{x_{-1} = 0\}$, $X_{n-2} := X_{n-1} \cap \{x_0 = 0\}$, the plane curve*

$$Y_1 := \left\{ y \in \mathbf{P}_2 \;\middle|\; y_{-1}^d + \frac{2}{d} \sum_{j=1}^{d-1} z_{0j} y_{-1}^{d-j} y_0^j + y_0^d - y_1^d = 0 \right\},$$

and the points $\{p_1, \ldots, p_d\} := Y_1 \cap \{y_1 = 0\}$. Then X_n can be obtained by a blow up $\beta_1 : Z_1 \to X_{n-1} \times Y_1$ along $X_{n-2} \times \{p_1, \ldots, p_d\} \simeq dX_{n-2}$, a morphism $\zeta : Z_1 \to Z_2$ of degree d and a blow down $\beta_2 : Z_2 \to X_n$, contracting $\zeta(X_{n-1} \times p_i) = \mathbf{P}_{n-1} \times p_i$ to p_i and $\zeta(X_{n-2} \times Y_1) \simeq X_{n-2} \times \mathbf{P}_1$ to X_{n-2}.

As an immediate consequence, the variation of Hodge structure of $\mathcal{X}$ is expressed by the variations of plane curves.

Corollary 7. *Given a Fermat hypersurface*

$$Z = \left\{ z \in \mathbf{P}_{[\frac{n}{2}]+\varepsilon(n)} \;\middle|\; \sum_{i=0}^{[\frac{n}{2}]+\varepsilon(n)} z_i^d = 0 \right\}$$

of dimension $k := \left[\frac{n}{2}\right] + \varepsilon(n) - 1$, let us denote by $\mathcal{H}^m_{Fermat}$ the constant family of $H^m(Z, \mathbb{C}) = H^m(\mathbf{P}_{k+1}, \mathbb{C})$ for $0 \leq m \leq k-1$ and $H^k(Z, \mathbb{C}) = H^k(\mathbf{P}_{k+1}, \mathbb{C}) + H^k(Z, \mathbb{C})_o$. Then the variation of Hodge structure of the family of smooth hypersurfaces $\mathcal{X} \to \mathcal{B}_0 \times \mathcal{B}_1 \times \ldots \times \mathcal{B}_{[\frac{n}{2}]}$, defined in Lemma 5, is a direct summand of

$$\sum_{0 \leq i_1 < \ldots < i_s \leq [\frac{n}{2}]} \eta_i \mathcal{H}^1(\mathcal{B}_{i_1}) \otimes \ldots \otimes \mathcal{H}^1(\mathcal{B}_{i_s}) \otimes \mathcal{H}^{s+\varepsilon(n)-2}_{Fermat} \otimes \mathbb{Q}\left(\left[\frac{n}{2}\right] + 1 - s\right),$$

where $\mathcal{H}^1(\mathcal{B}_t)$ stands for the variation of the plane curves with homogeneous equations

$$y_{-1}^d + \frac{2}{d} \sum_{j=1}^{d-1} z_{tj} y_{-1}^{d-j} y_0^j + y_0^d \pm y_1^d = 0,$$

$(z_{t1}, \ldots, z_{td-1}) \in \mathcal{B}_t$, and η_i, $i = (i_1, \ldots, i_s)$, are natural numbers.

Proof. According to [5], if $N \to M$ is a finite map of equi-dimensional connected manifolds, then $H^*(M, \mathbb{C})$ is a direct summand of $H^*(N, \mathbb{C})$, and if M' is a blow up of M along a closed submanifold T of codimension c, then $H^*(M', \mathbb{C}) = H^*(M, \mathbb{C}) + \sum_{i=1}^{c-1} H^*(T, \mathbb{C}) \otimes \mathbb{Q}(i)$. By Künneth formula, the variation $\mathcal{H}^n(Z_2)$ of $Z_2 = \zeta\beta_1^{-1}(X_{n-1} \times Y_1)$ is a direct summand of $\mathcal{H}^{n-1}(\mathcal{B}_1 \times \ldots \times \mathcal{B}_{[\frac{n}{2}]}) \otimes \mathcal{H}^1(\mathcal{B}_0) + d\mathcal{H}^{n-2}(\mathcal{B}_1 \times \ldots \times \mathcal{B}_{[\frac{n}{2}]}) \otimes \mathbb{Q}(1)$. On the other hand, $Z_2 = \beta_2^{-1}(X_n)$ implies the equali-

ty $\mathcal{H}^n(Z_2) = \mathcal{H}^n(\mathcal{B}_0 \times \mathcal{B}_1 \times \ldots \times \mathcal{B}_{[\frac{n}{2}]}) + \mathcal{H}^{n-2}(\mathcal{B}_1 \times \ldots \times \mathcal{B}_{[\frac{n}{2}]}) \otimes \mathbb{Q}(1)$ for the variation $\mathcal{H}^n(\mathcal{B}_0 \times \mathcal{B}_1 \times \ldots \times \mathcal{B}_{[\frac{n}{2}]})$ of $\mathcal{X}$. Consequently, $\mathcal{H}^n(\mathcal{B}_0 \times \ldots \times \mathcal{B}_{[\frac{n}{2}]})$ turns to be a direct summand of $\mathcal{H}^1(\mathcal{B}_0) \otimes \mathcal{H}^{n-1}(\mathcal{B}_1 \times \ldots \times \mathcal{B}_{[\frac{n}{2}]}) + (d-1)\mathcal{H}^{n-2}(\mathcal{B}_1 \times \ldots \times \mathcal{B}_{[\frac{n}{2}]}) \otimes \mathbb{Q}(1)$.

The proof is proceeded by induction on $\left[\frac{n}{2}\right]$. □

2.3. EXPLICIT REALIZATIONS OF THE TAUTOLOGICAL VARIATIONS BY THE VARIATION OF $\mathcal{X}$

Let $\mathcal{X}$ be the family of smooth hypersurfaces, constructed in Lemma 5. Restricting to the bounded symmetric realizations of

$$D = \mathcal{S}(p) \subset \mathcal{B}(1,p)^p \subset \mathcal{B}(1,4p+3)^{p+1},$$

$$D = \mathcal{B}(p,q) \subset \mathcal{B}(1,q)^p \subset \mathcal{B}(1,4q+3)^{p+1},$$

$$D = \mathcal{Q}(p) \subset \mathcal{B}(p,1) \subset \mathcal{B}(1,1)^p \subset \mathcal{B}(1,4p-1)^p,$$

specified in Theorem 4, one obtains the families $\mathcal{X}_D = \cup_{z \in D}\{x \in \mathbf{P}_{n+1} \mid f_z(x) = 0\}$. Let $\mathbf{S}_D$ be the trivial family of polynomial rings $S = \mathbb{C}[x_{-1}, x_0, x_1, \ldots, x_n]$ over D, $\mathbf{J}_D$ be the family of Jacobian ideals $\mathbf{J}_z := \left\langle \frac{\partial f_z}{\partial x_i} \mid -1 \le i \le n \right\rangle \subset \mathbf{S}_z$, $z \in D$, and $\mathcal{R}_D := \mathbf{S}_D/\mathbf{J}_D$ be the family of Jacobian rings. Denote by $\mathbf{f}$ the sheaf of the equations f_z of $\mathcal{X}_D(z)$, $d := \deg f_z$, $\Delta(i) := -(n+2) + d(n+1-i)$, and $\Omega = \sum_{i=-1}^{n}(-1)^i x_i dx_{-1} \wedge dx_0 \wedge dx_1 \wedge \ldots \wedge \widehat{dx_i} \wedge \ldots \wedge dx_n$. Griffiths [6] shows that the residue map

$$Res_D : \mathcal{R}_D^{\Delta(i)} \frac{\Omega}{\mathbf{f}^{n+1-i}} \to \mathcal{H}_D^{i,n-i} \simeq \mathcal{F}_D^i/\mathcal{F}_D^{i+1}$$

is an isomorphism. Carlson and Griffiths [1] establish that the non-degenerate pairing $\Psi^{\mathcal{H}_D} : \mathcal{H}_D^{i,n-i} \times \mathcal{H}_D^{n-i,i} \to \mathcal{H}_D^{2n}$, i.e., the Serre duality map with values in the constant family of $H^{2n}(\mathcal{X}_D(z), \mathbb{C}) = \mathbb{C}$, can be naturally identified with the ring multiplication $\mathcal{R}_D^{\Delta(i)} \times \mathcal{R}_D^{\Delta(n-i)} \to \mathcal{R}_D^{(d-2)(n+2)}$.

Lemma 8. *In the notations from Theorem 4, let*

$$\mathcal{V}_{\mathcal{B}(p,q)} := \mathcal{V}^1_{J,p,q}, \quad w_{\mathcal{B}(p,q)} := 1,$$

$$k(i,j) := ij + (1-i)(4\delta - 2 - j), \quad l(i,j) := i(2\delta - 2 - j) + (1-i)(2\delta + j).$$

Consider the holomorphic subbundles

$$\mathcal{W}_D^{m_D+i, m_D+w_D-i} \subset \mathcal{H}_D^{m_D+i, m_D+w_D-i}, \quad 0 \le i \le w_D,$$

generated by the global sections

$$Res_D \left\{ \left[w_j^{D,i} + \mathbf{J}_D^{\Delta(m_D+i)} \right] \frac{\Omega}{\mathbf{f}^{m_D+w_D-i+1}} \right\},$$

where

$$w_j^{D,i} := x_{-1}^{k(i,j)} x_0^{l(i,j)} \left(\prod_{t=1}^{2p+1} x_t \right)^{2\delta-1} \qquad \text{for } 1 \le j \le p,\ D = \mathcal{S}(p) \text{ or } \mathcal{B}(p,q);$$

$$w_{j+p}^{\mathcal{B}(p,q),i} := x_{-1}^{k(i,j)} x_{2p+1}^{l(i,j)} \left(\prod_{t=0}^{2p} x_t \right)^{2\delta-1} \qquad \text{for } 1 \le j \le q;$$

$$w^{\mathcal{Q}(p),i} := \left(\prod_{t=-1}^{2p-2} x_t \right)^{2p+1-2i} \qquad \text{for } i = 0 \text{ or } 2;$$

$$w_j^{\mathcal{Q}(p),1} := \left(\frac{1}{2}\right)^{\frac{1}{2}} (x_{2j-1}^2 + x_{2j}^2) \left(\prod_{t \neq 2j-1,2j} x_t \right) \left(\prod_{t=-1}^{2p-2} x_t \right)^{2p-2} \qquad \text{for } 0 \le j \le p-1.$$

Then the bundles $\mathcal{W}_D = \sum_{i=0}^{w_D} \mathcal{W}_D^{m_D+i, m_D+w_D-i}$ *admit polarization preserving isomorphisms*

$$\varphi_D : \mathcal{W}_D \to \mathcal{V}_D \otimes \mathbb{Q}(m_D),$$
$$\Psi^{\mathcal{H}_D}(w_1, w_2) = \Psi_D \otimes \Psi^{\mathbb{Q}(m_D)}(\varphi_D(w_1), \varphi_D(w_2))$$

for sections w_1, w_2 *of* $\mathcal{W}_D$.

Proof. In the cases of $D = \mathcal{S}(p)$ or $\mathcal{B}(p,q)$ the claim is a straightforward consequence of $x_i^s \in \mathbf{J}_D(z)$ for $s \ge 4\delta - 1$, $i \in \{-1, 0, 2p+1\}$, $z \in D$, and the fact that the line bundles $\mathcal{R}_D^{(4\delta-2)(2p+3)}$ are associated with the sheaves of sections

$$\left(\prod_{j=-1}^{2p+1} x_j \right)^{4\delta-2} + \mathbf{J}_D^{(4\delta-2)(2p+3)}.$$

For $D = \mathcal{Q}(p)$, $r \in \{2j-1, 2j\}$ let us note that

$$\frac{\partial f_z}{\partial x_r} = 4p x_r^{4p-1} + z_j x_r^{2p-1} x_{4j-1-r}^{2p} \in \mathbf{J}_{\mathcal{Q}(p)}(z)$$

and the line bundle $\mathcal{R}_{\mathcal{Q}(p)}^{2p(4p-2)}$ is generated by

$$\sigma = \left(\prod_{t=-1}^{2p-2} x_t \right)^{4p-2} + \mathbf{J}_{\mathcal{Q}(p)}^{2p(4p-2)}.$$

Applying repeatedly the aforementioned relations of the Jacobian rings, one computes for $j \neq k$ that

$$w_j^{\mathcal{Q}(p),1} w_k^{\mathcal{Q}(p),1} = \frac{1}{2} \prod_{l=j,k} \left[\left(x_{2l-1}^2 + x_{2l}^2\right) \left(x_{2l-1} x_{2l}\right)^{4p-3} \right] \left(\prod_{t, [\frac{t+1}{2}] \neq j,k} x_t \right)^{4p-2}$$

$$= \frac{1}{2} \prod_{l=j,k} \left[\frac{-z_l}{4p} \left(x_{2l-1}^{2p-1} x_{2l}^{6p-3} + x_{2l-1}^{6p-3} x_{2l}^{2p-1} \right) \right] \left(\prod_{t,[\frac{t+1}{2}] \neq j,k} x_t \right)^{4p-2}$$

$$= \frac{z_j^2 z_k^2}{512p^4} \prod_{l=j,k} \left(x_{2l-1}^{4p-1} x_{2l}^{4p-3} + x_{2l-1}^{4p-3} x_{2l}^{4p-1} \right) \left(\prod_{t,[\frac{t+1}{2}] \neq j,k} x_t \right)^{4p-2},$$

i.e.,

$$\prod_{l=j,k} w_l^{\mathcal{Q}(p),1} = \frac{z_j^2 z_k^2}{256p^4} \left(\prod_{l=j,k} w_l^{\mathcal{Q}(p),1} \right).$$

However, $2^t \overline{z} z < 1 + |^t z z|^2 < 1 + 1$ for $z = {}^t(z_0, z_1, \ldots, z_{p-1}) \in \mathcal{Q}(p)$ reveals that $|z_j|^2 \leq \sum_{i=0}^{p-1} |z_i|^2 < 1$, whereas $\left| \frac{z_j^2 z_k^2}{256p^4} \right| < 1$. In the torsion free Jacobian rings $\mathcal{R}_{\mathcal{Q}(p)}(z)$ that siffices for the vanishing of $\prod_{l=j,k} w_l^{\mathcal{Q}(p),1}$, $j \neq k$. Similarly, the expression

$$\left(w_j^{\mathcal{Q}(p),1} \right)^2 = \frac{1}{2} \left(x_{2j-1}^2 + x_{2j}^2 \right)^2 (x_{2j-1} x_{2j})^{4p-4} \left(\prod_{t \neq 2j-1,2j} x_t \right)^{4p-2}$$

$$= \frac{-z_j}{8p} \left(x_{2j-1}^{2p} x_{2j}^{6p-4} + x_{2j-1}^{6p-4} x_{2j}^{2p} \right) \left(\prod_{t \neq 2j-1,2j} x_t \right)^{4p-2} + \left(\prod_{t=-1}^{2p-2} x_t \right)^{4p-2}$$

$$= \frac{z_j^2}{32p^2} \left(x_{2j-1}^{4p} x_{2j}^{4p-4} + x_{2j-1}^{4p-4} x_{2j}^{4p} \right) \left(\prod_{t \neq 2j-1,2j} x_t \right)^{4p-2} + \sigma$$

$$= \frac{z_j^2}{16p^2} \left[\left(w_j^{\mathcal{Q}(p),1} \right)^2 - \sigma \right] + \sigma$$

with $\left| \frac{z_j^2}{16p^2} \right| < 1$ forces $\left(w_j^{\mathcal{Q}(p),1} \right)^2 = \sigma$. □

That completes the proof of Theorem 4.

3. CONSEQUENCES

3.1. ABELIAN-MOTIVIC VARIATIONS

Corollary 9. *The following three tensor categories are equivalent:*

(I) *the category $\mathcal{A}$ of the abelian-motivic variations of Hodge structure,*

(II) *the category $\mathcal{AH}$ of the abelian-motivic hypersurface variations of Hodge structure, and*

(III) *the category $\mathcal{JPC}$ of the variations of Hodge structure, expressed by Jacobians of plane curves.*

Proof. The inclusions $\mathcal{JPC} \subseteq \mathcal{AH} \subseteq \mathcal{A}$ are obvious. As far as $\mathcal{A}$ is generated by the tautological variations of Hodge structure $\mathcal{V}_{\mathcal{S}(p)}$ over Siegel upper half spaces $\mathcal{S}(p)$, Theorem 4 implies that $\mathcal{A} \subseteq \mathcal{JPC}$. □

3.2. MAXIMAL DIMENSIONAL VARIATIONS

For Hodge structure H of weight $w = 2k+1 > 1$ let

$$\mu_1^{odd} := \sum_{i \geq 0} h^{k-2i} h^{k-1-2i}$$

and

$$\mu_2^{odd} := \frac{1}{2} h^k (h^k + 1) + \sum_{i \geq 0} h^{k-1-2i} h^{k-2-2i}.$$

In the case of $w = 2k$ let

$$\mu_1^{even} := \sum_{i \geq 0} h^{k-1-2i} h^{k-2-2i},$$

$$\mu_2^{even} := h^k + (h^{k-1} - 1) h^{k-2} + \sum_{i \geq 0} h^{k-3-2i} h^{k-4-2i}$$

for $w \geq 4$,

$$\mu_3^{even} := \overline{\mu}_3 + \sum_{i \geq 0} h^{k-2-2i} h^{k-3-2i}$$

with $\overline{\mu_3} := h^k$ for $h^{k-1} = 1$, $\overline{\mu_3} := \frac{1}{2} h^k h^{k-1}$ for an even h^k and $h^{k-1} > 1$, $\overline{\mu_3} := \frac{1}{2}(h^k - 1) h^{k-1} + 1$ for an odd h^k and $h^{k-1} > 1$. According to [9] or [2], the maximum dimension μ of a variation of Hodge structure is $\mu^{odd} = \max(\mu_1^{odd}, \mu_2^{odd})$ if $w = 2k+1$, or $\mu^{even} = \max(\mu_1^{even}, \mu_2^{even}, \mu_3^{even})$ if $w = 2k$.

The summands $h^j h^{j-1}$, $j < \left[\frac{w}{2}\right]$, of μ, including $(h^{k-1} - 1) h^{k-2}$ from μ_2^{even} and $\left(\frac{1}{2} h^k\right) h^{k-1}$ from $\overline{\mu_3}$ with an even h^k, $h^{k-1} > 1$, are realized by appropriate shifts of the tautological variations of J-Hodge structure over the generalized balls $\mathcal{B}(h^j, h^{j-1})$, respectively, $\mathcal{B}(h^{k-1} - 1, h^{k-2})$, $\mathcal{B}(\frac{1}{2} h^k, h^{k-1})$. The tautological variation of Hodge structure $\mathcal{V}_{\mathcal{S}(h^k)}$ provides a variation of dimension $\frac{1}{2} h^k (h^k + 1)$ in the case of μ_2^{odd}. The tautological variation of Hodge structure $\mathcal{V}_{\mathcal{Q}(h^k)}$ is an example of dimension h^k for μ_2^{even} or $\overline{\mu_3}$ with $h^{k-1} = 1$. The non-symmetric domain

$$\Omega(h^{k-1}, \tfrac{1}{2}(h^k - 1)) \subset \mathcal{B}(1,1) \times \mathcal{B}(h^{k-1} - 1, \tfrac{1}{2}(h^k - 1)) \times \mathcal{B}(1, \tfrac{1}{2}(h^k - 1)),$$

cut by the inequality ${}^t\overline{Y}Y < (1-|t|^2)^2(I_q - {}^t\overline{X}X)$ for $t \in \mathcal{B}(1,1)$, $X \in \mathcal{B}(h^{k-1}-1, \frac{1}{2}(h^k-1))$, $Y \in \mathcal{B}(1, \frac{1}{2}(h^k-1))$, is an instance of a variation of Hodge structure of dimension $\frac{1}{2}(h^k-1)h^{k-1}+1$ in the case of $\overline{\mu_3}$ with an odd h^k, $h^{k-1} > 1$.

If all Hodge numbers of H are greater than 1, $h^{[\frac{w-1}{2}]} > 2$ and $h^{[\frac{w}{2}]} \geq 4$, then the results of [9] and [7] imply that all the maximal dimensional simply connected variations of Hodge structure are isomorphic to products of the aforementioned bounded domains. The lack of quasiprojective discrete quotients of $\Omega(h^{k-1}, \frac{1}{2}(h^k-1))$ (cf. [7]) reveals that the maximal dimensional variations, covered by $\Omega(h^{k-1}, \frac{1}{2}(h^k-1)) \times \prod_{i\geq 0} \mathcal{B}(h^{k-2-2i}, h^{k-3-2i})$, do not arise from geometry. All the other maximal dimensional variations are direct sums of tautological ones, so that Theorem 4 implies

Corollary 10. *The geometrically arising maximal dimensional variations of Hodge structure with sufficiently large Hodge numbers are expressed by Jacobians of plane curves.*

Let us observe that our main result provides a "new" symplectic representation of $SO(2,p)$. Indeed, the inclusion $\mathcal{V}_{\mathcal{Q}(p)} \subset C(\mathcal{V}_{\mathcal{Q}(n)}, \Psi_{\mathcal{Q}(p)})$ from Theorem 2 (ii) induces a symplectic representation $SO(2,p) \hookrightarrow Sp(2^{p+1}, \mathbb{R})$. Since a plane curve Y of degree $4p$ has genus $\frac{1}{2}(4p-1)(4p-2)$, Theorem 4 interprets as a realization of $SO(2,p)$ in a product of Mumford–Tate groups $Sp((4p-1)(2p-1), \mathbb{R})$ of Hodge structures of such Y.

Acknowledgements. The author is extremely indebted to Tony Pantev for his comments and suggestions and for bringing to her attention the works [8], [4]. The research is partially supported by the Bulgarian Funding Organizations, sponsoring the Algebra Section, Institute of Mathematics and Informatics, Bulgarian Academy of Sciences.

REFERENCES

1. Carlson, J., P. Griffiths. Infinitesimal variations of Hodge structures and the global Torelli problem. *Journ. Geom. Alg. d'Angers*, 1980, 51–76.
2. Carlson, J., A. Kasparian, D. Toledo. Variations of Hodge structure of maximal dimension. *Duke Math. Journ.*, **58**, 1989, 669–694.
3. Carlson, J., C. Simpson. Shimura varieties of wieght two Hodge structures. In: *Hodge Theory, Lect. Notes in Math.*, **1246,** Springer, 1987, 1–15.
4. Deligne, P. La conjecture de Weil pour les surfaces $K3$. *Inv. Math.*, **15**, 1972, 206–226.
5. Deligne, P., J. Milne, A. Ogus, K.-Y. Shih. Hodge Cycles, Motives, and Shimura Varieties. *Lect. Notes in Math.*, **900**, Springer, 1982.
6. Griffiths, P. On the periods of certain rational integrals — I, II. *Ann. Math.*, **90**, 1969, 460–495, 496–541.

7. Kasparian, A. The variations of Hodge structure of maximal dimension with associated Hodge numbers $h^{2,0} > 2$ and $h^{1,1} = 2q + 1$ do not arise from geometry. *Trans. Amer. Math. Soc.*, **347**, 1995, 4985–5007.
8. Kuga, M., I. Satake. Abelian varieties attached to polarized $K3$-surfaces. *Math. Ann.*, **169**, 1967, 239–242.
9. Mayer, R. Coupled contact systems and rigidity of maximal dimensional variations of Hodge structure. Preprint.
10. Rapoport, M. Complement a l'article de Deligne "La conjecture de Weil pour les surfaces $K3$". *Inv. Math.*, **15**, 1972, 227–236.
11. Shermenev, A. On the motive of Weil hypersurfaces. *Mat. Zam.*, **12**, 1972, 555–560.
12. Shermenev, A. The motive of an abelian variety. *Functional Anal. and Appl.*, **8**, 1974, 55–61.
13. Shioda, T., T. Katsura. On Fermat varieties. *Tohoku Math. Journ.*, **31**, 1979, 97–115.

Received January 27, 1999

Faculty of Mathematics and Informatics
"St. Kliment Ohridski" University of Sofia
5 James Bourchier Blvd.
BG-1164 Sofia, Bulgaria
E-mail: kasparia@fmi.uni-sofia.bg

ГОДИШНИК НА СОФИЙСКИЯ УНИВЕРСИТЕТ „СВ. КЛИМЕНТ ОХРИДСКИ“

ФАКУЛТЕТ ПО МАТЕМАТИКА И ИНФОРМАТИКА
Книга 1 — Математика и механика
Том 92, 1998

ANNUAIRE DE L'UNIVERSITE DE SOFIA „ST. KLIMENT OHRIDSKI“

FACULTE DE MATHEMATIQUES ET INFORMATIQUE
Livre 1 — Mathématiques et Mecanique
Tome 92, 1998

A SEMANTICS OF LOGIC PROGRAMS WITHOUT SEARCHING*

VESSELA BALEVA

A generalized version of the declarative semantics of Horn clause programs on abstract structures is presented. The main feature of the semantics is that it does not admit searching in the domain of the structure.

Keywords: semantics, logic programming, abstract structures
1991/95 Math. Subject Classification: 03D75, 68Q05, 68Q55

1. INTRODUCTION

In this paper we represent and study a semantics of logic programs on abstract structures. A key feature for this semantics is that it does not admit searching in the domain of the structure. We consider partial abstract structures with enumerable domain. The main result is that the class of sets definable by logic programs (LP-definable sets) coincides with the class of domains of Fridman functions in the structure in some fixed point.

In order to prove this, we introduce several auxiliary terms. The names of these terms and the relations between them are given on Fig. 1. An arrow between two terms means that the first one implies the second one. Each arrow is labelled by the number of the proposition where the corresponding implication is proved.

* This work is partially supported under Grant I-604 by the Ministry of Science, Education and Technologies.

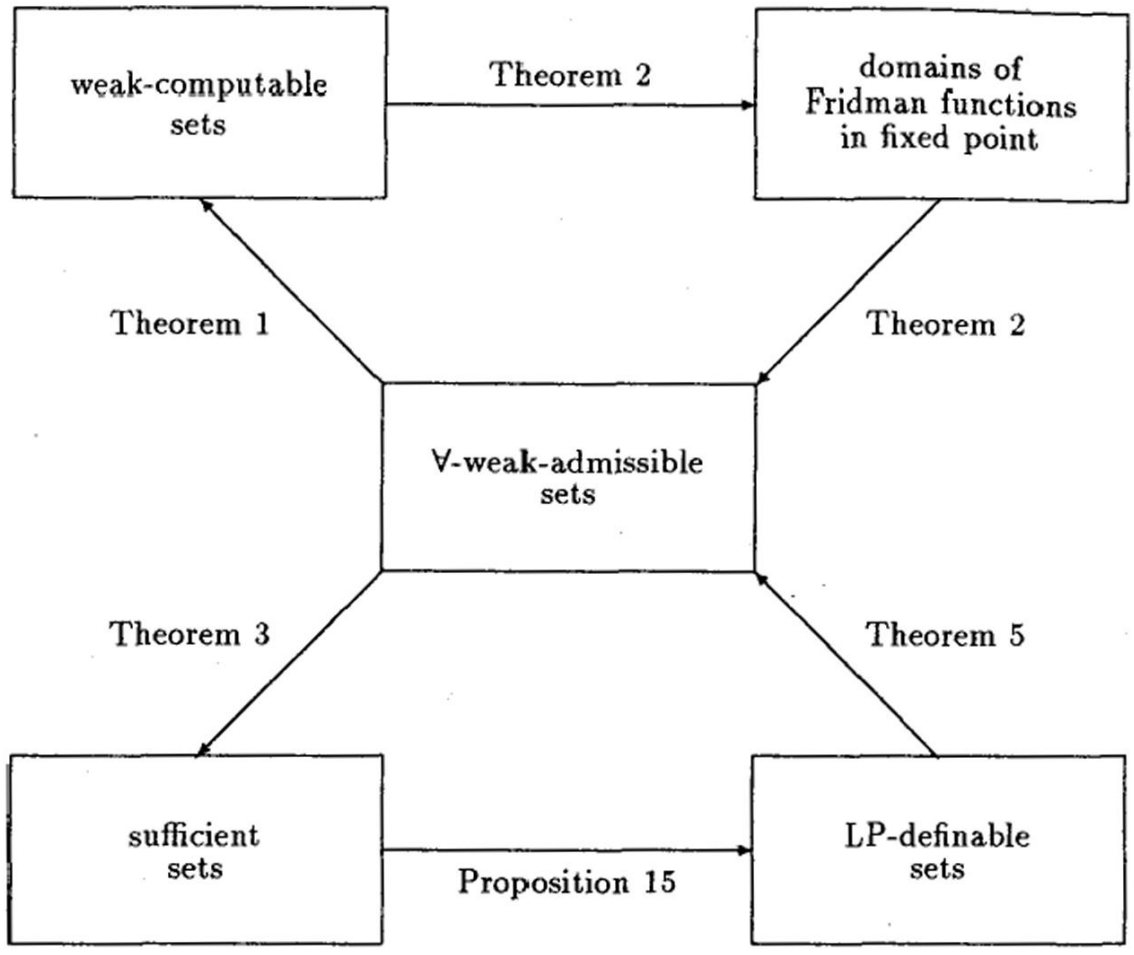

Fig. 1. Relations between the introduced terms

In Section 2 we introduce some basic notions needed in our considerations. In Section 3 we define standard enumerations, which are the main tool in proving all results in the paper. In Section 4 we prove the upper circle in Fig. 1 and in sections 5 and 6 the lower circle is proved.

For the sake of simplicity we consider only structures with unary functions, predicates and parameters. All definitions and results can be easily generalized for functions, predicates and parameters of arbitrary finite arity.

2. PRELIMINARIES

Let $\mathfrak{A} = (B; \theta_1, \ldots, \theta_n; \Sigma_0, \ldots, \Sigma_k)$ be a partial structure, where the domain of the structure B is a denumerable set, $\theta_1, \ldots, \theta_n$ are partial functions of one argument on B, $\Sigma_0, \ldots, \Sigma_k$ are partial predicates of one argument on B, $\Sigma_0 = \lambda s.true$ and $n, k \geq 0$. Let $\mathfrak{B} = (N; \varphi_1, \ldots, \varphi_n; \sigma_0, \ldots, \sigma_k)$ be a partial structure over the set N of the natural numbers. A subset W of N is said to be *recursively enumerable* (r. e.) in $\mathfrak{B}$ iff $W = \Gamma(\varphi_1, \ldots, \varphi_n; \sigma_0, \ldots, \sigma_k)$ for some enumeration operator Γ (see [1]).

An *enumeration* of the structure $\mathfrak{A}$ is any ordered pair $\langle \alpha, \mathfrak{B} \rangle$, where $\mathfrak{B} = (N; \varphi_1, \ldots, \varphi_n; \sigma_0, \ldots, \sigma_k)$ is a partial structure, $\sigma_0 = \lambda s.true$, and α is a partial surjective mapping of N onto B such that the following conditions hold:

(i) The domain of α ($Dom(\alpha)$) is closed with respect to the partial operations $\varphi_1, \ldots, \varphi_n$;

(ii) $\alpha(\varphi_i(x)) \simeq \theta_i(\alpha(x))$ for all x of $Dom(\alpha)$, $1 \leq i \leq n$;

(iii) $\sigma_j(x) \Leftrightarrow \Sigma_j(\alpha(x))$ for all x of $Dom(\alpha)$, $1 \leq j \leq k$.

We shall assume that an effective monotonic coding of finite sequences and sets of natural numbers is fixed. If $a_0, \ldots, a_m$ is a sequence of natural numbers, by $\langle a_0, \ldots, a_m \rangle$ we shall denote the code of the sequence $a_0, \ldots, a_m$ and by E_v — the finite set with code v. We shall use the following notations. The letters s, t, p will denote elements of B; x, y, z, u, v will be elements of N. We shall identify the predicates with partial mappings taking values 0 (for "true") and 1 (for "false").

Let $\langle \alpha, \mathfrak{B} \rangle$ be an enumeration of $\mathfrak{A}$. We shall call the set

$$D(\mathfrak{B}) = \{\langle i, x, y \rangle : 1 \leq i \leq n \ \& \ \varphi_i(x) \simeq y\}$$
$$\cup \{\langle j, x, \varepsilon \rangle : n+1 \leq j \leq n+k \ \ \& \ \sigma_{j-n}(x) \simeq \varepsilon \ \& \ \varepsilon \in \{0, 1\}\}$$

a *code* of the structure $\mathfrak{B}$. It is clear that for each $W \subseteq N$, W is r. e. in $\mathfrak{B}$ iff W is r. e. in $D(\mathfrak{B})$.

Let $A \subseteq B$. The set A is called *weak-admissible* in enumeration $\langle \alpha, \mathfrak{B} \rangle$ iff for some r. e. in $\mathfrak{B}$ subset W of N the following conditions hold:

(*) $W \subseteq Dom(\alpha)$;

(**) $\alpha(W) = A$.

A subset A of B is called $\forall$-*weak-admissible* in $\mathfrak{A}$ iff it is weak-admissible in every enumeration $\langle \alpha, \mathfrak{B} \rangle$ of $\mathfrak{A}$.

The equivalence between $\forall$-weak-admissible sets and the sets definable by logic programs will be considered. The $\forall$-weak-admissible sets have an explicit characterization which simplifies the considerations.

Let $\mathfrak{L} = (f_1, \ldots, f_n; T_0, \ldots, T_k)$ be the first-order language corresponding to the structure $\mathfrak{A}$, where $f_1, \ldots, f_n$ are functional symbols, $T_0, \ldots, T_k$ are symbols for predicates, T_0 represents the total predicate $\Sigma_0 = \lambda s.0$.

Let $\{Z_1, Z_2, \ldots\}$ be a denumerable set of variables. We shall use the capital letters X, Y, Z to denote the variables.

If τ is a term of the language $\mathfrak{L}$, then we shall write $\tau(\overline{Z})$ to denote that all of the variables in τ are among $\overline{Z} = (Z_1, \ldots, Z_a)$. If $\tau(\overline{Z})$ is a term and $\overline{t} = t_1, \ldots, t_a$ are arbitrary elements of B, then by $\tau_{\mathfrak{A}}(\overline{Z}/\overline{t})$ we shall denote the value, if it exists, of the term τ in the structure $\mathfrak{A}$ over the elements $t_1, \ldots, t_a$.

Termal predicates in the language $\mathfrak{L}$ are defined by the following inductive clauses:

(i) $T_j(\tau)$, $0 \leq j \leq k$, where τ is a term, are termal predicates;

(ii) If Π is a termal predicate, then $\neg\Pi$ is a termal predicate;

(iii) If Π^1 and Π^2 are termal predicates, then $\Pi^1 \& \Pi^2$ is a termal predicate.

Let $\Pi(\overline{Z})$ be a termal predicate and $t_1, \ldots, t_a$ be arbitrary elements of B. The value $\Pi_{\mathfrak{A}}(\overline{Z}/\overline{t})$ is defined as follows:

(i) If $\Pi = T_j(\tau)$, $0 \leq j \leq k$, then $\Pi_{\mathfrak{A}}(\overline{Z}/\overline{t}) \simeq \Sigma_j(\tau_{\mathfrak{A}}(\overline{Z}/\overline{t}))$;

(ii) If $\Pi = \Pi^1 \& \Pi^2$, where Π^1 and Π^2 are termal predicates, then

$$\Pi_{\mathfrak{A}}(\overline{Z}/\overline{t}) \simeq \begin{cases} \Pi^2_{\mathfrak{A}}(\overline{Z}/\overline{t}), & \text{if } \Pi^1_{\mathfrak{A}}(\overline{Z}/\overline{t}) \simeq 0, \\ 1, & \text{if } \Pi^1_{\mathfrak{A}}(\overline{Z}/\overline{t}) \simeq 1, \\ \text{undefined} & \text{otherwise.} \end{cases}$$

Let Π be a termal predicate and τ be a term. Then the term $(\Pi \supset \tau)$ is called *conditional term*. Let $Q = (\Pi \supset \tau)$ be a conditional term with variables among $X_1, \ldots, X_a$ and let $s_1, \ldots, s_a$ be arbitrary elements of B. A value $Q_{\mathfrak{A}}(\overline{X}/\overline{s})$ is defined as follows:

$$Q_{\mathfrak{A}}(\overline{X}/\overline{s}) \simeq t \Leftrightarrow (\Pi_{\mathfrak{A}}(\overline{X}/\overline{s}) \simeq 0 \ \& \ \tau_{\mathfrak{A}}(\overline{X}/\overline{s}) \simeq t).$$

Let fix an effective coding of expressions of $\mathfrak{L}$. The subset A of B is called *weak-computable* iff for some r. e. set V of codes of conditional terms $\{Q^v\}_{v \in V}$ with variables among $Z_1, \ldots, Z_r$ and for fixed elements $t_1, \ldots, t_r$ of B it is true that

$$s \in A \Leftrightarrow \exists v (v \in V \ \& \ Q^v_{\mathfrak{A}}(\overline{Z}/\overline{t}) \simeq s).$$

3. STANDARD ENUMERATIONS

In order to characterize the LP-definable sets in abstract structures, we shall examine their prototypes in the enumerations of the structures. For this purpose it is enough to restrict our considerations only to a special class of enumerations called standard ones (see [4]). In this section we briefly introduce some definitions and properties of standard enumerations.

Let φ^*_i, $1 \leq i \leq n$, be the unary recursive function $\lambda x.\langle i, x\rangle$, let $N^0 = N \setminus (Range(\varphi^*_1) \cup \ldots \cup Range(\varphi^*_n))$ and let α^0 be a partial mapping of N^0 onto B.

The partial mapping α of N onto B is defined by the following inductive clauses:

If $x \in N^0$, then $\alpha(x) \simeq \alpha^0(x)$;

If $x = \langle i, y\rangle$, $\alpha(y) \simeq s$ and $\theta_i(s) \simeq t$, then $\alpha(x) \simeq t$.

To the mapping α corresponds the set N_α of natural numbers defined by:

If $x \in Dom(\alpha^0)$, then $x \in N_\alpha$;

If $x = \langle i, y\rangle$ and $y \in N_\alpha$, then $x \in N_\alpha$.

Let $D_1, \ldots, D_n$ be unary partial predicates in N such that:

$$D_i(x) = \begin{cases} 0, & \text{if } x \notin N_\alpha, \\ 0, & \text{if } x \in N_\alpha \text{ and } \theta_i(\alpha(x)) \text{ is defined,} \\ \text{undefined} & \text{otherwise.} \end{cases}$$

The predicates $D_1, \ldots, D_n$ are used to describe the domains of the standard enumeration functions $\varphi_1, \ldots, \varphi_n$ defined as follows:

$$\varphi_i(x) = \begin{cases} \varphi^*_i(x), & \text{if } D_i(x) \simeq 0, \\ \text{undefined} & \text{otherwise.} \end{cases}$$

It is clear that each φ_i is r. e. in $\{D_i\}$ and each D_i is r. e. in $\{\varphi_i\}$, $1 \leq i \leq n$.
Let $\sigma_1, \ldots, \sigma_k$ be partial predicates in N satisfying the condition

$$x \in N_\alpha \Rightarrow \sigma_j(x) \simeq \Sigma_j(\alpha(x)),\ 1 \leq j \leq k.$$

Denote by $\mathfrak{B}$ the partial structure $(N; \varphi_1, \ldots, \varphi_n; \sigma_0, \ldots, \sigma_k)$. Each enumeration $\langle \alpha, \mathfrak{B} \rangle$ obtained by the method described above is called a *standard enumeration*. The mapping α^0 is called a *basis* of the enumeration $\langle \alpha, \mathfrak{B} \rangle$. It is clear that α^0 and the predicates $\sigma_1, \ldots, \sigma_k$ completely determine the enumeration $\langle \alpha, \mathfrak{B} \rangle$.
For each natural x we define $|x|$ as follows:

If $x \in N^0$, then $|x| = 0$;

If $x = \langle i, y \rangle$, $1 \leq i \leq n$, then $|x| = |y| + 1$.

The next properties are proved in detail in [4].

Proposition 1. *Let $\langle \alpha, \mathfrak{B} \rangle$ be a standard enumeration and $1 \leq i \leq n$. Then for each natural x, $\alpha(\langle i, x \rangle) \simeq \theta_i(\alpha(x))$.*

Proposition 2. *For each standard enumeration $\langle \alpha, \mathfrak{B} \rangle$, $Dom(\alpha) \subseteq N_\alpha$.*

Proposition 3. *Let $\langle \alpha, \mathfrak{B} \rangle$ be a standard enumeration and $1 \leq i \leq n$. Then for each natural x, $\alpha(\varphi_i(x)) \simeq \theta_i(\alpha(x))$.*

Proposition 4. *Each standard enumeration is an enumeration of the structure $\mathfrak{A}$.*

Define the unary recursive function g in the following way:

If $x \in N^0$, then $g(x) = x$;

If $x = \langle i, y \rangle$, then $g(x) = g(y)$.

Let $\mathfrak{B}^*$ denotes the structure $(N; \varphi_1^*, \ldots, \varphi_n^*)$.

Proposition 5. *There exits an effective way to define for each natural x and each variable Y a term $\tau(Y)$ such that $\tau_{\mathfrak{B}^*}(Y/g(x)) = x$.*

Proposition 6. *Let $\tau(Y)$ be a term and $y \in N$. Then for each standard enumeration $\langle \alpha, \mathfrak{B} \rangle$, $\alpha(\tau_{\mathfrak{B}^*}(Y/y)) \simeq \tau_{\mathfrak{A}}((Y/\alpha(y))$.*

Proposition 7. *There exits an effective way to define for each natural x and each variable Y a term $\tau(Y)$ such that for each standard enumeration $\langle \alpha, \mathfrak{B} \rangle$, $\alpha(x) \simeq \tau_{\mathfrak{A}}((Y/\alpha(g(x)))$.*

Let $\langle \alpha, \mathfrak{B} \rangle$ be a standard enumeration. Denote by $R_{\mathfrak{B}}$ the subset of N with the following definition:

$$\langle j, x, \varepsilon \rangle \in R_{\mathfrak{B}} \Leftrightarrow ((1 \leq j \leq k)\ \&\ \sigma_j(x) \simeq \varepsilon)$$

or

$$(k+1 \leq j \leq k+n\ \&\ D_{j-k}(x) \simeq \varepsilon)).$$

It is clear that the set W is r. e. in $\mathfrak{B}$ iff it is r. e. in $R_{\mathfrak{B}}$.

Proposition 8. *There exists an effective way to define for each triple $u = \langle j, x, \varepsilon \rangle$ and each variable Y an atomic predicate $\Pi(Y)$ such that for each standard enumeration $\langle \alpha, \mathfrak{B} \rangle$*

$$g(x) \in Dom(\alpha^0) \Rightarrow (u \in R_{\mathfrak{B}} \Leftrightarrow \Pi_{\mathfrak{A}}(Y/\alpha(g(x))) \simeq 0).$$

4. WEAK-COMPUTABILITY AND WEAK-ADMISSIBILITY

LP-definable sets are not convenient for direct examination. That is why we introduce and characterize $\forall$-weak-admissible sets which are later proved to coincide with the LP-definable sets. In this section we study the relation between $\forall$-weak-admissibility, weak-computability and Fridman computability.

Theorem 1. *If A is $\forall$-weak-admissible in $\mathfrak{A}$, then A is weak-computable.*

Proof. Assume A is not weak-computable. We shall construct a standard enumeration $\langle \alpha, \mathfrak{B} \rangle$ of $\mathfrak{A}$ such that A is not admissible in it.

To define the enumeration, we construct a partial surjective mapping α^0 of N^0 onto B. The mapping α^0 will be constructed by steps. On each step q we define a partial mapping α_q of N^0 onto B, a subset H_q of N^0 and partial predicates $\sigma_1^q, \ldots, \sigma_k^q$ such that:

(i) $Dom(\alpha_q)$ and H_q are finite and disjoint;

(ii) $\alpha_q \leq \alpha_{q+1}$ and $H_q \subseteq H_{q+1}$;

(iii) $\sigma_1^q, \ldots, \sigma_k^q$ are partial recursive and defined exactly for those natural y for which $g(y) \in H_q$;

(iv) $\sigma_j^q \subseteq \sigma_j^{q+1}$, $1 \leq j \leq k$.

We take $\alpha^0 = \bigcup_{q=0}^{\infty} \alpha_q$.

With the even steps we ensure that $Range(\alpha^0) = B$. With the odd steps $q = 2n + 1$ we ensure that if Γ_n is the n-th enumeration operator and

$$\langle \alpha, \mathfrak{B} = (N; \varphi_1, \ldots, \varphi_n; \sigma_1, \ldots, \sigma_k) \rangle$$

is a standard enumeration such that $\alpha_q \leq \alpha$, $H_q \cap Dom(\alpha) = \varnothing$, $\sigma_j^q \leq \sigma_j$, $1 \leq j \leq k$, then for $W = \Gamma(R_{\mathfrak{B}})$ at least one of the conditions $(*)$ and $(**)$ fails.

Let $s_0, s_1, \ldots$ be an arbitrary enumeration of B and $x_0, x_1, \ldots$ be an enumeration of N^0, $\alpha_0(x_0) = s_0$ and $\alpha_0(x)$ be undefined for $x \neq x_0$. Let $H_0 = \varnothing$ and each of $\sigma_1^0, \ldots, \sigma_k^0$ be the totally undefined predicate. Let $q > 0$ and α_r, H_r be $\sigma_1^r, \ldots, \sigma_k^r$ defined for $r < q$. We have to consider the following two cases:

I. $q = 2n$. Let z be the first element of the sequence $x_0, x_1, \ldots$ which does not belong to $Dom(\alpha_{q-1}) \cup H_{q-1}$, and s be the first element of the sequence $s_0, s_1, \ldots$ which does not belong to $Range(\alpha_{q-1})$. If such s does not exist, then let s be

an arbitrary element of B. Define $\alpha_q(z) \simeq s$ and $\alpha_q(x) \simeq \alpha_{q-1}(x)$ for $x \neq z$, $H_q = H_{q-1}$ and $\sigma_j^q = \sigma_j^{q-1}$, $1 \leq j \leq k$.

II. $q = 2n + 1$. Let E_v be the finite set of natural numbers with code v. The set E_v is called *q-consistent* iff:

(i) each element h of E_v is equal to $\langle j, x, \varepsilon \rangle$ for some $n + 1 \leq j \leq n + k$ and $\varepsilon \in \{0, 1\}$ or $1 \leq j \leq n$ and $\varepsilon = 0$;

(ii) if $\langle j, x, \varepsilon_1 \rangle$ and $\langle j, x, \varepsilon_2 \rangle$ belong to E_v, then $\varepsilon_1 = \varepsilon_2$;

(iii) if $\langle j, x, \varepsilon \rangle \in E_v$, $n + 1 \leq j \leq n + k$ and $g(x) \in H_{q-1}$, then $\sigma_j^{q-1}(x) \simeq \varepsilon$.

Let Γ_n be the n-th enumeration operator defined by W_n — the n-th r. e. set, that is for each set R of natural numbers

$$x \in \Gamma_n(R) \Leftrightarrow \exists v(\langle v, x \rangle \in W_n \,\&\, E_v \subseteq R).$$

Let $u = \langle v, x \rangle$ be element of W_n, $Dom(\alpha_{q-1}) = \{w_1, \ldots, w_m\}$, and $Z_1, \ldots, Z_m$ be distinct variables. Corresponding to u, we define predicates $\Pi^u(Z_1, \ldots, Z_m)$ and $P^u(Z_1, \ldots, Z_m)$ and a term $\tau^u(Z_1, \ldots, Z_m)$ as follows. If E_v is not q-consistent, then $\Pi^u = P^u = \neg T_0(Z_1)$.

Further we consider the case when E_v is q-consistent. We define Π^u in the following way. If E_v does not contain elements of the form $\langle j, x, \varepsilon \rangle$, such that $g(x) \in Dom(\alpha_{q-1})$, then $\Pi^u = T_0(Z_1)$.

Let $\langle j_1, x_1, \varepsilon_1 \rangle, \ldots, \langle j_p, x_p, \varepsilon_p \rangle$ be all elements of E_v such that $g(x_i) \in Dom(\alpha_{q-1})$, $1 \leq i \leq p$, and $\Pi^1(Y_1), \ldots, \Pi^p(Y_p)$ be atomic predicates such that:

If $g(x_i) = w_j$ for some j, $1 \leq j \leq m$, then $Y_i = Z_j$;

For each standard enumeration $\langle \alpha, \mathfrak{B} \rangle$, if $g(x_i) \in Dom(\alpha)$, then $\langle j_i, x_i, \varepsilon_i \rangle \in R_{\mathfrak{B}} \Leftrightarrow \Pi^i_{\mathfrak{A}}(Y_i/\alpha(g(x_i))) \simeq 0$.

Define Π^u to be the conjunction of $\Pi^1(Y_1), \ldots, \Pi^p(Y_p)$. Now we define P^u and τ^u to follow the behavior of x.

If $g(x) \notin Dom(\alpha_{q-1})$, then $P^u = \neg T_0(Z_1)$ and $\tau^u = Z_1$.

If $g(x) \in Dom(\alpha_{q-1})$ and $g(x) = w_j$ for some j, $1 \leq j \leq m$, then let $Y = Z_j$ and $\tau(Y)$ be a term such that for each standard enumeration $\langle \alpha, \mathfrak{B} \rangle$, $\alpha(x) \simeq \tau_{\mathfrak{A}}(Y/\alpha(g(x)))$ holds.

Define $P^u = \Pi^u \& T_0(\tau)$ and $\tau^u = \tau$.

We have described a way to construct the r. e. sets $\{\Pi^u\}_{u \in W_n}$, $\{P^u\}_{u \in W_n}$ and $\{\tau^u\}_{u \in W_n}$ for a given W_n.

Denote $\alpha(w_i)$ by t_i, $1 \leq i \leq m$, and let D be a subset of B such that $s \in D$ iff

$$\exists u(u \in W_n \,\&\, P^u_{\mathfrak{A}}(Z_1/t_1, \ldots, Z_m/t_m) \simeq 0 \,\&\, \tau^u_{\mathfrak{A}}(Z_1/t_1, \ldots, Z_m/t_m) \simeq s).$$

It is clear that D is weak-computable and hence $D \neq A$. There are two possible cases.

Case 1. There exists s, which is an element of B, such that $s \in A$ and $s \notin D$. In this case we have also two possibilities:

a. For some $u \in W_n$, $u = \langle v, x \rangle$, we have

$$\Pi^u_{\mathfrak{A}}(Z_1/t_1, \ldots, Z_m/t_m) \simeq 0 \tag{1}$$

and

$$P^u_{\mathfrak{A}}(Z_1/t_1, \ldots, Z_m/t_m) \not\simeq 0. \tag{2}$$

From (1) and (2) it follows that

$$g(x) \in Dom(\alpha_{q-1}) \;\Rightarrow\; \tau_{\mathfrak{A}}(Y/\alpha(g(x)) \text{ is undefined.} \tag{3}$$

Let

$$L = H_{q-1} \cup \{g(y) \mid \exists j \exists \varepsilon(\langle j, y, \varepsilon\rangle \in E_v \;\&\; g(y) \notin Dom(\alpha_{q-1}))\}.$$

We define $\alpha_q \equiv \alpha_{q-1}$. If $g(x) \in Dom(\alpha_q)$, then define $H_q \equiv L$, else $H_q \equiv H_{q-1} \cup \{g(x)\}$. And, finally, we define σ^q_j, $1 \le j \le k$, by the following clauses:

If $g(y) \notin H_q$, then $\sigma^q_j(y)$ is undefined;

If $g(y) \in H_q \setminus H_{q-1}$, then if $\langle n+j, y, \varepsilon\rangle \in E_v$ we have $\sigma^q_j(y) \simeq \varepsilon$, else $\sigma^q_j(y) \simeq 0$;

If $g(y) \in H_{q-1}$, then $\sigma^q_j(y) \simeq \sigma^{q-1}_j(y)$.

It follows from (1) that E_v is q-consistent and hence $\sigma^q_j(y)$ are correctly defined and $\sigma^{q-1}_j \le \sigma^q_j$.

Let $\langle \alpha, \mathfrak{B}\rangle$ be a standard enumeration such that $\alpha \ge \alpha_q$, $Dom(\alpha) \cap H_q = \varnothing$, $\sigma_j \ge \sigma^q_j$, $1 \le j \le k$, and let $W = \Gamma_n(R_{\mathfrak{B}})$. We shall prove that $E_v \subseteq R_{\mathfrak{B}}$.

Indeed, let $\langle j, y, \varepsilon\rangle \in E_v$. If $g(y) \in Dom(\alpha_q)$, then $g(y) \in Dom(\alpha)$ and (1) yields $\langle j, y, \varepsilon\rangle \in R_{\mathfrak{B}}$. If $g(y) \notin Dom(\alpha_q)$, then $g(y) \in H_q$ and we also have $\sigma_j \ge \sigma^q_j$ and $H_q \cap N_\alpha = \varnothing$, hence $\langle j, y, \varepsilon\rangle \in R_{\mathfrak{B}}$.

Suppose that $x \in Dom(\alpha)$. Then $g(x) \in Dom(\alpha_{q-1})$. From the definition of τ^u we obtain that $\tau^u_{\mathfrak{A}}(Y/\alpha(g(x))) \simeq \alpha(x)$ and $\tau^u_{\mathfrak{A}}(Y/\alpha(g(x)))$ is defined. This contradicts (3), hence $x \notin Dom(\alpha)$, which implies $W \not\subseteq Dom(\alpha)$.

b. For each $u \in W_n$ such that $u = \langle v, x\rangle$, $\Pi^u_{\mathfrak{A}}(Z_1/t_1, \ldots, Z_m/t_m) \simeq 0$ implies $P^u_{\mathfrak{A}}(Z_1/t_1, \ldots, Z_m/t_m) \simeq 0$. In this case let $\alpha_q \equiv \alpha_{q-1}$, $H_q \equiv H_{q-1}$ and $\sigma^q_j \equiv \sigma^{q-1}_j$, $1 \le j \le k$. Let $\langle \alpha, \mathfrak{B}\rangle$ be a standard enumeration such that $\alpha \ge \alpha_q$, $Dom(\alpha) \cap H_q = \varnothing$, $\sigma_j \ge \sigma^q_j$, $1 \le j \le k$, and $W = \Gamma_n(R_{\mathfrak{B}})$. Suppose that there exists $x \in W$ such that $\alpha(x) = s$. Then there exists $u \in W_n$ such that $u = \langle v, x\rangle$ and $E_v \subseteq R_{\mathfrak{B}}$. From the definitions it follows that $\Pi^u_{\mathfrak{A}}(Z_1/t_1, \ldots, Z_m/t_m) \simeq 0$ and hence $P^u_{\mathfrak{A}}(Z_1/t_1, \ldots, Z_m/t_m) \simeq 0$. We obtained $\alpha(x) \simeq \tau^u_{\mathfrak{A}}(Y/\alpha(g(x))) \simeq s$, which contradicts the assumption $s \notin D$. We conclude that $A \not\equiv \alpha(W)$.

Case 2. There exists $s \in B$ such that $s \notin A$ and $s \in D$. This implies the existence of $u \in W_n$ such that $u = \langle v, x\rangle$ and $P^u_{\mathfrak{A}}(Z_1/t_1, \ldots, Z_m/t_m) \simeq 0$ and $\tau^u_{\mathfrak{A}}(Z_1/t_1, \ldots, Z_m/t_m) \simeq s$. Then $\Pi^u_{\mathfrak{A}}(Z_1/t_1, \ldots, Z_{\tilde{m}}/t_m) \simeq 0$ and hence E_v is q-consistent. Let $\alpha_q \equiv \alpha_{q-1}$ and

$$H_q = H_{q-1} \cup \{g(y) \mid \exists j \exists \varepsilon(\langle j, y, \varepsilon\rangle \in E_v \;\&\; g(y) \notin Dom(\alpha_{q-1}))\}.$$

We define the predicates σ^q_j, $1 \le j \le k$, by the following clauses:

If $g(y) \notin H_q$, then $\sigma^q_j(y)$ is undefined;

If $g(y) \in H_q$, then

$$\sigma^q_j(y) \simeq \begin{cases} \varepsilon, & \text{if } \langle j, y, \varepsilon\rangle \in E_v, \\ 0, & \text{if } g(y) \in H_q/H_{q-1} \text{ and } \langle j, y, \varepsilon\rangle \notin E_v, \\ \sigma^{q-1}_j(y), & \text{if } q(y) \in H_{q-1}. \end{cases}$$

Let $\langle \alpha, \mathfrak{B} \rangle$ be a standard enumeration such that $\alpha \geq \alpha_q$, $Dom(\alpha) \cap H_q = \emptyset$, $\sigma_j \geq \sigma_j^q$, $1 \leq j \leq k$, and $W = \Gamma_n(R_{\mathfrak{B}})$. Analogously to *Case* **1a**, we can prove that $E_v \subseteq R_{\mathfrak{B}}$ and hence $x \in W$. From $P^u_{\mathfrak{A}}(Z_1/t_1, \ldots, Z_m/t_m) \simeq 0$ and from the definition of τ^u it follows that $x \in Dom(\alpha_q)$ and $\alpha(x) \simeq s$. And $s \notin A$ implies $A \not\equiv \alpha(W)$.

Now we are ready to complete the proof defining the required enumeration. Let $\alpha^0 = \bigcup_{q=0}^{\infty} \alpha_q$, $\sigma_j^* = \bigcup_{q=0}^{\infty} \sigma_j^q$, $1 \leq j \leq k$, and $H = \bigcup_{q=0}^{\infty} H_q$. Let $\langle \alpha, \mathfrak{B} = (N; \varphi_1, \ldots, \varphi_n; \sigma_0, \ldots, \sigma_m) \rangle$ be a standard enumeration with basis α^0 and

$$\sigma_j(x) \simeq \begin{cases} \sigma_j^*(x), & \text{if } x \notin N_\alpha, \\ \Sigma_j(\alpha(x)) & \text{otherwise,} \end{cases} \qquad 1 \leq j \leq k.$$

□

Let $F(\mathfrak{A})$ be the class of all Fridman computable functions in $\mathfrak{A}$.

Let $\mathfrak{A}_i = (B_i; \theta_1^i, \ldots, \theta_n^i; \Sigma_0^i, \ldots, \Sigma_k^i)$, $i = 1, 2$, be two partial structures, where the corresponding functions and predicates have the same arity. The mapping κ of B_1 onto B_2 is called a *strong homomorphism* iff:

(i) κ is a surjective mapping;

(ii) $\kappa(\theta_i^1(s_1, \ldots, s_{a_i})) \simeq \theta_i^2(\kappa(s_1), \ldots, \kappa(s_{a_i}))$ for each $(s_1, \ldots, s_{a_i}) \in B_1^{a_i}$, $1 \leq i \leq n$;

(iii) $\Sigma_j^1(s_1, \ldots, s_{b_j}) \Leftrightarrow \Sigma_j^2(\kappa(s_1), \ldots, \kappa(s_{b_j}))$ for each $(s_1, \ldots, s_{b_j}) \in B_1^{b_j}$, $1 \leq j \leq k$.

It is easy to show the following properties of the Fridman computability:

1. *Invariantness.* If κ is a strong homomorphism between $\mathfrak{A}_1$ and $\mathfrak{A}_2$ and $\theta_2 \in F(\mathfrak{A}_2)$, then there exists $\theta_1 \in F(\mathfrak{A}_1)$ (of the same arity) such that $\kappa(\theta_1(s_1, \ldots, s_a)) \simeq \theta_2(\kappa(s_1), \ldots, \kappa(s_a))$ for each $(s_1, \ldots, s_a) \in B_1^a$.

2. *Effectiveness.* If $\mathfrak{A} = (N; \varphi_1, \ldots, \varphi_n; \sigma_0, \ldots, \sigma_k)$ is a partial structure and $\varphi \in F(\mathfrak{A})$, then φ is r. e. in the functions and predicates of $\mathfrak{A}$.

3. *Substructure property.* Let $\mathfrak{A}_1$ and $\mathfrak{A}_2$ be partial structures such that $B_1 \subseteq B_2$. Let φ_j^1 and Σ_j^1 be the restrictions of φ_j^2 and Σ_j^2 on B_1, $1 \leq j \leq k$, $1 \leq i \leq n$, and $\theta_1 \in F(\mathfrak{A}_1)$. Then there exists $\theta_2 \in F(\mathfrak{A}_2)$ such that $\theta_1(s_1, \ldots, s_a) \simeq \theta_2(s_1, \ldots, s_a)$ for each $(s_1, \ldots, s_a) \in B_1^a$.

Using these properties we are ready to establish a relation between $\forall$-weak-admissibility and Fridman computability.

Theorem 2. *A is $\forall$-weak admissible iff there exists $\theta \in F(\mathfrak{A})$ such that $A \equiv \theta(t_1^0, \ldots, t_r^0)$ for some $(t_1^0, \ldots, t_r^0) \in B^r$.*

Proof. Let $\theta \in F(\mathfrak{A})$ and $A \equiv \theta(t_1^0, \ldots, t_r^0)$ for some $(t_1^0, \ldots, t_r^0) \in B^r$. Let $\langle \alpha, \mathfrak{B} \rangle$ be an arbitrary partial enumeration of $\mathfrak{A}$ and

$$\mathfrak{B}' = (Dom(\alpha); \varphi_1', \ldots, \varphi_n'; \sigma_0', \ldots, \sigma_k'),$$

where φ'_i and σ'_j are the restrictions of φ_i and σ_j on $Dom(\alpha)$, $1 \le i \le n$, $0 \le j \le k$. Then α is a strong homomorphism between $\mathfrak{B}'$ and $\mathfrak{A}$. Thus there exists $\varphi' \in F(\mathfrak{B}')$ such that φ' is an α-prototype of θ. From the substructure property of Fridman computability, there exists $\varphi \in F(\mathfrak{B})$ such that $\varphi(x_1, \ldots, x_r) \simeq \varphi'(x_1, \ldots, x_r)$ for each $(x_1, \ldots, x_r) \in (Dom(\alpha))^r$. Because of the effectiveness of Fridman computability, φ is r. e. in $\mathfrak{B}$. We obtained that $A \equiv \alpha(\varphi'(x_1, \ldots, x_r))$, $x_1, \ldots, x_r \in Dom(\alpha)$ and $\alpha(x_i) \simeq t_i$, $1 \le i \le r$. The set $W = \varphi(x_1, \ldots, x_r)$ is r. e. in $\mathfrak{B}$, because φ is such and $\varphi \ge \varphi'$. Finally, $W \subseteq Dom(\alpha)$ and $A = \alpha(W)$, i. e. A is $\forall$-weak-admissible.

Now let A be $\forall$-weak-admissible. Then from the previous theorem, A is weak-computable. Using the corresponding definitions, we can easily prove the "if" part of the theorem. □

5. SUFFICIENCY AND WEAK-ADMISSIBILITY

In this section we introduce the notion of sufficiency and establish the relation between sufficiency and weak-admissibility.

Further we assume that the structure $\mathfrak{A} = (B; \theta_1, \ldots, \theta_n; \Sigma_0, \ldots, \Sigma_k)$ is such that the predicates $\Sigma_1, \ldots, \Sigma_k$ take only value 0 (*true*) wherever defined. This assumption is not restrictive, because each predicate Σ can be represented by the following two predicates:

$$\Sigma^\delta(s) \simeq \begin{cases} 0, & \text{if } \Sigma(s) \simeq \delta, \\ \text{undefined} & \text{otherwise,} \end{cases} \qquad \delta = 0, 1.$$

The extra condition we impose is due to the syntax of Horn clause logic programs. The negative information of the structure cannot be used because a negation in clause tails is not allowed. Let fix the structure $\mathfrak{A}$ and modify some of the notions introduced according to the new limitation.

A standard enumeration $\langle \alpha, \mathfrak{B} \rangle$ is called *positive* iff $\sigma_1, \ldots, \sigma_k$ take only value 0 wherever defined. Further all the enumerations are assumed to be positive and thus we can simplify the code of the structure considering the set

$$\langle \mathfrak{B} \rangle = \{\langle j, x \rangle \mid n+1 \le j \le n+k \text{ and } \sigma_{j-n}(x) \text{ is defined}\} \cup$$
$$\{\langle i, x \rangle \mid 1 \le i \le n \text{ and } D_i(x) \text{ is defined}\}$$

instead of $R_{\mathfrak{B}}$. Note that $\langle \alpha, \mathfrak{B} \rangle$ is a positive standard enumeration and W is r. e. in $\mathfrak{B}$ iff $W = \Gamma(\langle \mathfrak{B} \rangle)$ for some enumeration operator Γ.

The pair $\langle \alpha', H' \rangle$ is called a *finite part* iff:

(i) α' is a finite mapping of N^0 onto B;

(ii) H' is a finite subset of N^0 and $Dom(\alpha') \cap H' = \varnothing$.

The positive standard enumeration $\langle \alpha, \mathfrak{B} \rangle$ *extends* the finite part $\langle \alpha', H' \rangle$ iff:

(i) $\alpha \ge \alpha'$;

(ii) $Dom(\alpha) \cap H' = \varnothing$;

(iii) $\sigma_j(x) \simeq 0$ for each $x \in N$ such that $g(x) \in H'$, $1 \le j \le k$.

Let fix an arbitrary finite part $\langle \alpha', H' \rangle$, $Dom(\alpha') = \{w_1, \ldots, w_r\}$ and $\alpha'(w_i) = s_i$, $1 \leq i \leq r$. Let $c_{s_1}, \ldots, c_{s_r}$ be new constants which we shall interpret as names of $s_1, \ldots, s_r$. Further we consider terms and predicates of the first order language $\mathfrak{L} = (c_{s_1}, \ldots, c_{s_r}, f_1, \ldots, f_n, T_0, \ldots, T_k)$.

The finite set with code v, E_v, is called *correct* iff it consists only of elements of the form $\langle j, x \rangle$ for some natural x and $1 \leq j \leq k + n$.

The next propositions are similar to the propositions for the standard enumerations and have straightforward proves.

Proposition 9. *Let $x \in N$ and $g(x) \in Dom(\alpha')$. Then there exists an effective way to define a term τ^x without variables such that each positive standard enumeration $\langle \alpha, \mathfrak{B} \rangle$ extending $\langle \alpha', H' \rangle$ satisfies $\alpha(x) \simeq \tau^x_{\mathfrak{A}}$.*

Proposition 10. *Let $x \in N$, $g(x) \in Dom(\alpha')$ and $1 \leq j \leq n + k$. Then there exists an effective way for $u = \langle j, x \rangle$ to define an atomic predicate Π^u without variables and negations such that for each standard enumeration $\langle \alpha, \mathfrak{B} \rangle$ we have*

$$\langle j, x \rangle \in \langle \mathfrak{B} \rangle \Leftrightarrow \Pi^u_{\mathfrak{A}} \simeq 0.$$

We shall identify each finite set of atoms without variables with their conjunctions. The empty set we shall identify with the logical constant *true*.

Let E be a correct finite set of naturals. By $\tilde{E}$ we denote the set

$$\{\Pi^u \mid u = \langle j, x \rangle \ \& \ g(x) \in Dom(\alpha') \ \& \ u \in E\}.$$

We shall call the set E *appropriate* for the finite part $\langle \alpha', H' \rangle$) iff it is correct and $\tilde{E}_{\mathfrak{A}} \simeq 0$.

Let W be a r. e. set and Γ be the enumeration operator defined by W. The notion of compatibility of a finite part and an enumeration operator introduced below reflects the fact that in logic programs, where a search in the domain of the structure is not allowed, only a finite information supplied by constants is available, while W contains much more information which is not accessible.

The finite part $\langle \alpha', H' \rangle$ and the enumeration operator Γ are *compatible* iff for each $u = \langle v, x \rangle \in W$, such that E_v is appropriate for $\langle \alpha', H' \rangle$, we have $g(x) \in Dom(\alpha')$.

Let Γ be an enumeration operator and let the finite part $\langle \alpha', H' \rangle$ be compatible to Γ. The subset A of B is called *sufficient* for $\langle \alpha', H' \rangle$ and Γ iff:

(i) For each positive standard enumeration $\langle \alpha, \mathfrak{B} \rangle$ extending $\langle \alpha', H' \rangle$, it is true that $\alpha(\Gamma(\langle \mathfrak{B} \rangle)) \subseteq A$;

(ii) For each $s \in A$ there exists a finite part $\langle \alpha'', H'' \rangle$ such that $\alpha'' \geq \alpha'$, $H'' \supseteq H'$ and for each positive standard enumeration $\langle \alpha, \mathfrak{B} \rangle$ extending $\langle \alpha'', H'' \rangle$ it is true that $s \in \alpha(\Gamma(\langle \mathfrak{B} \rangle))$.

It is easy to show the following

Proposition 11. *For each compatible finite part $\langle \alpha', H' \rangle$ and enumeration operator Γ there exists at most one subset of B which is sufficient for them.*

A class of sets $\mathfrak{P}$ is called *sufficient* iff for each compatible finite part $\langle \alpha', H' \rangle$ and enumeration operator Γ there exists a set $A \in \mathfrak{P}$ which is sufficient for Γ and $\langle \alpha', H' \rangle$.

Theorem 3. *Each sufficient class contains all $\forall$-weak-admissible sets.*

Proof. Let $\mathfrak{P}$ be a sufficient class and $A \notin \mathfrak{P}$. We will prove that A is not $\forall$-weak-admissible constructing by steps a positive standard enumeration in which A is not weak-admissible. For the purpose we shall construct a partial surjective mapping α^0 of N^0 onto B and a subset H of N^0 such that $Dom(\alpha^0) \cap H = \varnothing$. The set H defines the predicates $\sigma_1, \ldots, \sigma_k$ out of N_α. Even steps ensure that α^0 is a surjective mapping and the odd steps $q = 2n + 1$ ensure that A is not weak-admissible for the n-th enumeration operator Γ_n.

Let $s_0, s_1, \ldots$ be an arbitrary ordering of elements of B and $x_0, x_1, \ldots$ be an arbitrary ordering of elements of N^0. Let $\alpha_0(x_0) \simeq s_0$ and $\alpha_0(x)$ be undefined otherwise. Let $H_0 = \varnothing$. Now suppose we have defined $\langle \alpha_l, H_l \rangle$, $0 \leq l < q$. We define $\langle \alpha_q, H_q \rangle$ as follows:

I. $q = 2n$. Then let $H_q \equiv H_{q-1}$ and s be the first element of the sequence $s_0, s_1, \ldots$ which is not in $Range(\alpha_{q-1})$ (if there is no such element, let s be an arbitrary element of B). Let x be the first element of the sequence $x_0, x_1, \ldots$ which is not in $Dom(\alpha_{q-1}) \cup H_{q-1}$. Then let $\alpha_q(y) \simeq \alpha_{q-1}(y)$ for each $y \in N^0$, $y \neq x$ and $\alpha_q(x) \simeq s$.

II. $q = 2n+1$. Let Γ_n be the n-th enumeration operator. Consider the following cases:

Case 1. Γ_n and $\langle \alpha_{q-1}, H_{q-1} \rangle$ are incompatible. Let W_n be the r. e. set defining Γ_n. Then there exists $u \in W_n$ such that $u = \langle v, x \rangle$ and E_v is appropriate for $\langle \alpha_{q-1}, H_{q-1} \rangle$ and $g(x) \notin Dom(\alpha_q)$. Let $L = \{g(y) \mid \exists j(\langle j, x \rangle \in E_v \ \& \ g(y) \notin Dom(\alpha_{q-1}))\}$, $H_q = H_{q-1} \cup L \cup \{g(x)\}$ and $\alpha_q \equiv \alpha_{q-1}$. Let $\langle \alpha, \mathfrak{B} \rangle$ be a positive standard enumeration extending $\langle \alpha_q, H_q \rangle$ and $h = \langle j, y \rangle \in E_v$. If $g(y) \in Dom(\alpha_q)$, then $g(y) \in Dom(\alpha_{q-1})$, hence $\Pi^h_{\mathfrak{A}} \simeq 0$ and thus $h \in \langle \mathfrak{B} \rangle$. If $g(y) \notin Dom(\alpha_q)$, then $g(y) \in H_{q-1}$. If $k + 1 \leq j \leq k + n$, then we obtain $h \in \langle \mathfrak{B} \rangle$ from the definition of extension, and if $1 \leq j \leq n$, then we obtain this from the definition of standard enumeration and from $Dom(\alpha^0) \cap H_q = \varnothing$.

In this way we have proved that $E_v \subseteq \langle \mathfrak{B} \rangle$, which implies $x \in \Gamma_n(\langle \mathfrak{B} \rangle)$. From $g(x) \in H_q$ it follows that $x \notin Dom(\alpha)$, that is $\Gamma_n(\langle \mathfrak{B} \rangle) \not\subseteq Dom(\alpha)$ and A is not weak-admissible in $\langle \alpha, \mathfrak{B} \rangle$.

Case 2. Γ_n and $\langle \alpha_{q-1}, H_{q-1} \rangle$ are compatible. Let $D \in \mathfrak{P}$ be the set which is sufficient for Γ_n and $\langle \alpha_{q-1}, H_{q-1} \rangle$. Then $D \not\equiv A$. In this case there are two subcases possible:

a. There exists $s \in B$ such that $s \in A$ and $s \notin D$. Let $\alpha_q \equiv \alpha_{q-1}$ and $H_q \equiv H_{q-1}$. Sufficiency of D implies that for each standard enumeration $\langle \alpha, \mathfrak{B} \rangle$ extending $\langle \alpha_{q-1}, H_{q-1} \rangle$ we have $\alpha(\Gamma_n(\langle \mathfrak{B} \rangle)) \subseteq D$. Then $s \notin \alpha(\Gamma_n(\langle \mathfrak{B} \rangle))$, which means $A \not\equiv \alpha(\Gamma_n(\langle \mathfrak{B} \rangle))$.

b. There exists $s \in B$ such that $s \notin A$ and $s \in D$. In this case there exists a finite part $\langle \alpha', H' \rangle$ extending $\langle \alpha_{q-1}, H_{q-1} \rangle$, that is $\alpha' \geq \alpha_{q-1}$ and $H' \supseteq H_{q-1}$ and for each positive standard enumeration extending $\langle \alpha_{q-1}, H_{q-1} \rangle$ it is true that $s \in \alpha(\Gamma_n(\langle \mathfrak{B} \rangle))$. Let $\alpha_q \equiv \alpha'$ and $H_q \equiv H'$. Then for each standard enumeration extending $\langle \alpha_q, H_q \rangle$ it is true that $\alpha(\Gamma_n(\langle \mathfrak{B} \rangle)) \not\equiv A$.

Finally, let $\alpha^0 = \bigcup\limits_{q=1}^{\infty} \alpha_q$, $H^0 = \bigcup\limits_{q=1}^{\infty} H_q$ and $\sigma_j^* = \bigcup\limits_{q=1}^{\infty} \sigma_j^q$, where

$$\sigma_j^q(x) \simeq \begin{cases} 0, & \text{if } g(x) \in H_q, \\ \text{undefined} & \text{otherwise,} \end{cases} \qquad 1 \leq j \leq k.$$

We define the predicates $\sigma_1, \ldots, \sigma_k$ as follows:

$$\sigma_j(x) \simeq \begin{cases} \Sigma_j(\alpha(x)), & \text{if } x \in N_\alpha, \\ \sigma_j^*(x) & \text{otherwise.} \end{cases}$$

It is clear that the standard enumeration $\langle \alpha, \mathfrak{B} \rangle$, determined by α^0 and $\sigma_1, \ldots, \sigma_k$, is positive, correctly defined and it extends the finite parts $\langle \alpha_q, H_q \rangle$, $q = 0, 1, \ldots$, which means that A is not weak-admissible in $\langle \alpha, \mathfrak{B} \rangle$. □

6. LP-DEFINABILITY AND WEAK-ADMISSIBILITY

In this section we give a formal definition of LP-definable sets and for each compatible enumeration operator Γ and finite part $\langle \alpha', H' \rangle$ we construct a logic program $\langle P, F \rangle$ which defines a set sufficient for them. In this way we prove the equivalence between LP-definability and $\forall$-weak admissibility.

Let fix a structure $\mathfrak{A}$ which predicates are true wherever defined and let $\mathfrak{L} = (f_1, \ldots, f_n; T_0, \ldots, T_k)$ be a first order language corresponding to $\mathfrak{A}$. Let $\mathfrak{L}_C$ be the enrichment of $\mathfrak{L}$ with constants $c_1, \ldots, c_r$ and $\mathfrak{T}_C$ be the set of all terms without variables of $\mathfrak{L}_C$. We denote the set of all atoms of the form $T_j(\tau)$, where $0 \leq j \leq k$, $\tau \in \mathfrak{T}_C$ and $\Sigma_j(\tau_{\mathfrak{A}}) \simeq 0$ (the last means that $T_j(\tau)$ is true in $\mathfrak{A}$) by $\partial^C(\mathfrak{A})$.

Logic programs are called formulae of the form $F^1 \& \ldots \& F^l$, where F^i is an universal closure of Horn clause, i. e. F^i is of the form

$$\forall X_1 \ldots \forall X_r (\Pi \vee \neg \Pi_1 \vee \ldots \neg \Pi_n),$$

where $n \geq 0$ and $\Pi, \Pi_1, \ldots, \Pi_n$ are atomic predicates. We shall use the usual Prolog notation

$$\Pi :- \Pi_1, \ldots, \Pi_n.$$

for the Horn clauses. Π is called a *head* and $\Pi_1, \ldots, \Pi_n$ — a *tail* of the clause.

Let F be a new predicate symbol which is not among $T_0, \ldots, T_k$. For the sake of simplicity F is assumed to be an unary predicate symbol. All the definitions and proofs can be easily generalized for the case of a higher arity.

By $\mathfrak{L}_p$ we denote the language of the logic program P. The symbols from $\mathfrak{L}_C$ contained in $\mathfrak{L}_P$ are interpreted in the usual way, that is P does not redefine the predicates in $\mathfrak{L}_C$.

The subset A of B is *LP-definable* iff there exist a set of constants $C = \{c_1, \ldots, c_r\}$ and a pair $\langle P, F\rangle$, where P is a logic program and F is a new predicate symbol such that

$$s \in A \Leftrightarrow \exists \tau(\tau \in \mathfrak{T}_C \,\&\, \partial^C(\mathfrak{A}) \cup \{P\} \vdash F(\tau) \,\&\, \tau_{\mathfrak{A}} \simeq s),$$

where "$\vdash$" means deducibility in the sense of first order predicate calculus.

Let fix a finite part $\langle \alpha', H'\rangle$ such that $Dom(\alpha') = \{w_1, \ldots, w_r\}$ and $\alpha'(w_i) \simeq s_i$, $1 \leq i \leq r$. Let $c_{s_1}, \ldots, c_{s_r}$ be names for $s_1, \ldots, s_r$. We shall construct a logic program P such that the set LP-definable by $\langle P, F\rangle$ and $c_{s_1}, \ldots, c_{s_r}$ is sufficient for $\langle \alpha', H'\rangle$ and Γ. The program repeats the constructions in the proof of Theorem 1.

Let $\underline{0}$ and $\underline{nil}$ be new constant symbols, f_0 be a new unary functional symbol and h be a new binary functional symbol. Let $C = \{c_{s_1}, \ldots, c_{s_r}\}$, $\mathfrak{L}'_C = \{c_{s_1}, \ldots, c_{s_r}, \underline{0}, \underline{nil}, f_0, h, f_1, \ldots, f_n, T_0, \ldots, T_k\}$ and $\mathfrak{T}$ be the set of all terms without variables of $\mathfrak{L}'_C$. For each program P we consider Herbrand interpretations in $\mathfrak{L}_P$ with domain $\mathfrak{T}$. If Q is a predicate symbol of $\mathfrak{L}_P$ and I is a Herbrand interpretation of P, by $I(Q)$ we denote the corresponding predicate of $\mathfrak{T}$. An interpretation I of P is called a *model* for P iff all clauses of P are true in I.

For each natural n, by $\underline{n}$ we denote the term $f_0^n(\underline{0})$. Let $\underline{N}$ denote the set $\{\underline{n} \mid n \in N\}$.

It is well-known (see [3]) that:

Proposition 12. *For each r. e. subset W of N^k and for each k-ary predicate symbol Q there exists a logic program P with the following properties:*

(i) *If $(x_1, \ldots, x_k) \in W$, the $P \vdash Q(\underline{x}_1, \ldots, \underline{x}_k)$;*

(ii) *There exists a Herbrand interpretation I of P which is a model for P and*

$$I(Q)(a_1, \ldots, a_k) = 0$$
$$\Leftrightarrow \exists x_1 \ldots \exists x_k((x_1, \ldots, x_k) \in W \,\&\, a_1 = \underline{x}_1 \,\&\, \ldots a_k = \underline{x}_k).$$

Such an interpretation for P we shall call *standard*.

We define a *list* to be an element of $\mathfrak{T}$ such that: (i) $\underline{nil}$ is a list; (ii) if a is a term and b is a list, then $h(a, b)$ is a list.

We use the usual Prolog notation for lists.

Let $\underline{cod}$ be a new ternary predicate symbol and $\underline{nat}$ be a new unary predicate symbol. Let $P_{\underline{cod}}$ and $P_{\underline{nat}}$ be logic programs representing the r. e. sets $Cod = \{(\underline{x}, \underline{y}, \underline{z}) \mid x = \langle y, z\rangle\}$ and $Nat = \{\underline{x} \mid g(x) \notin Dom(\alpha')\}$ by $\underline{cod}$ and $\underline{nat}$, respectively, and

P_0

$\underline{tau}(\underline{w}_j, c_{s_j})$:- . $\quad j = 1, \ldots, r$

$\underline{tau}(X, f_i(V))$:- $\underline{cod}(X, \underline{i}, Y), \underline{tau}(Y, V)$. $\quad i = 1, \ldots n$

$P_{\underline{cod}}$.

The following proposition verifies the logic program P_0 using the method proposed in [3].

Proposition 13. *Let $x \in N$. Then $P_0 \vdash \underline{tau}(\underline{x}, \tau))$ iff $g(x) \in Dom(\alpha')$ and $\tau \equiv \tau^x$.*

Proof. The "if" part is easily proved by induction on $|x|$. To prove the "only if" part, let I be a standard Herbrand interpretation for $\underline{cod}$. We define I on τ as follows:

(i) $I(\underline{tau})(a, b) = 0$, if $a \notin \underline{N}$;

(ii) $I(\underline{tau})(a, b) = 0$, if $a \in \underline{N}$, $a = \underline{x}$, $g(x) \in Dom(\alpha')$ and $b \equiv \tau^x$;

(iii) $I(\underline{tau})(a, b) = 1$ otherwise.

It is easy to show that I is a model for P_0. □

Consider the following program:

P_1

$\underline{pi}([]) :- .$

$\underline{pi}([X|Y]) :- \underline{cod}(X, \underline{j}, Z), \underline{nat}(Z), \underline{pi}(Y). \quad j = 1, \ldots, n+k$

$\underline{pi}([X|Y]) :- \underline{cod}(X, \underline{j}, Z), \underline{tau}(Z, V), T_0(f_j(V)), \underline{pi}(Y). \quad j = 1, \ldots, n$

$\underline{pi}([X|Y]) :- \underline{cod}(X, \underline{j}, Z), \underline{tau}(Z, V), T_{j-n}(V), \underline{pi}(Y). \quad j = n+1, \ldots, n+k$

P_0.

$P_{\underline{nat}}$.

Proposition 14. *Let $E = \{v_1, \ldots, v_l\}$ be a correct finite set. Then for each finite set G' of atoms without variables in $\mathfrak{L}'_C$ it is true that $P_1 \vdash G' \Rightarrow \underline{pi}([\underline{v}_1, \ldots, \underline{v}_l])$ iff $G' \supseteq \tilde{E}$.*

Proof. The "if" part is easily proved by induction on l. To prove the "only if" part, we define a class $\mathfrak{K}$ of Herbrand interpretations of P_1. The Herbrand interpretation I belongs to class $\mathfrak{K}$ iff:

(i) I is standard for $P_{\underline{nat}}$ and the predicate symbols in P_0 are interpreted as in the prove of the previous proposition;

(ii) Let $a \in \mathfrak{T}$. If a is not of the form $[\underline{v}_1, \ldots, \underline{v}_l]$ for any correct set $\{v_1, \ldots, v_l\}$, then $I(\underline{pi}(a)) \simeq 0$. If a is of the form $[\underline{v}_1, \ldots, \underline{v}_l]$ for some correct set $E = \{v_1, \ldots, v_l\}$, then $I(\underline{pi}(a)) \simeq 0$ iff there exists a finite set of atoms without variables $G = \{\beta_1, \ldots, \beta_q\}$, $q \geq 0$ of $\mathfrak{L}'_C$, such that $\tilde{E} \subseteq G$ and $I(\beta_j) \simeq 0$, $1 \leq j \leq q$.

It is easy to show that each I of $\mathfrak{K}$ is a model of P_1. Let $G = \{\beta_1, \ldots, \beta_q\}$ be a finite set of atoms without variables of $\mathfrak{L}'_C$, $E = \{v_1, \ldots, v_l\}$ be a correct set and $P_1 \vdash G' \Rightarrow \underline{pi}([\underline{v}_1, \ldots, \underline{v}_l])$. Let $I \in \mathfrak{K}$ be such that if β is an atom without variables of $\mathfrak{L}'_C$, then $I(\beta) \simeq 0 \Leftrightarrow \beta \in G$. Since I is a model of P_1, we have $I(\underline{pi})([\underline{v}_1, \ldots, \underline{v}_l]) \simeq 0$, that is $\tilde{E} \subseteq G$. □

Proposition 15. *For each enumeration operator Γ compatible with $\langle \alpha', H' \rangle$ there exists a logic program P such that the set A, LP-definable by $\langle P, F \rangle$ and $C = \{c_{s_1}, \ldots, c_{s_r}\}$, is sufficient for $\langle \alpha', H' \rangle$ and Γ.*

Proof. Let Γ be an arbitrary enumeration operator and W be the r. e. set which determines Γ, i. e. if R is a set of natural numbers, then $x \in \Gamma(R) \Leftrightarrow \exists u(\langle u, x\rangle \in W \ \& \ E_n \subseteq R)$. Let $W_1 = \{\langle u, x\rangle \mid \langle u, x\rangle \in W$ and E_u is correct$\}$. It is clear that W_1 is r. e. set. Let Q be a new predicate symbol and P_2 be the logic program representing W_1 by Q. Let $\underline{list}$ be a new binary predicate symbol and P_3 be a logic program such that:

(i) If u is a code of the finite set $\{v_1, \ldots, v_l\}$, $l \geq 0$, then $P_3 \vdash \underline{list}(u, [\underline{v}_1, \ldots, \underline{v}_l])$;

(ii) There exists a Herbrand interpretation I of P_3, which is a model of P_3, and if $E_u = \{v_1, \ldots, v_l\}$ for some natural u, then $I(\underline{list})(\underline{u}, b) = 0 \Leftrightarrow b = [\underline{v}_1, \ldots, \underline{v}_l])$.

Consider the following logic program:

P

$F(Y) \text{ :- } Q(Z), \underline{cod}(Z, U, X), \underline{tau}(X, Y), T_0(Y), \underline{list}(U, V), \underline{pi}(V).$

$P_1.$

$P_2.$

$P_3.$

We shall use the next lemma, which proof is similar to the proof of the previous propositions.

Lemma 1. *Let G be a finite set of atoms without variables of $\mathfrak{L}'_C$. Then for each term τ without variables of $\mathfrak{L}'_C$, $P \vdash G \Rightarrow F(\tau)$ iff there exists $\langle u, x\rangle \in W_1$ such that:*

(i) $g(x) \in Dom(\alpha')$ *and* $\tau \equiv \tau^x$;

(ii) $\tilde{E}_u \cup \{T_0(\tau)\} \subseteq G$.

Let A be defined by $\langle P, F\rangle$ and $C = \{c_{s_1}, \ldots, c_{s_r}\}$. Let $\langle \alpha, \mathfrak{B}\rangle$ extend $\langle \alpha', H'\rangle$, $s \in \alpha(\Gamma(\langle \mathfrak{B}\rangle))$, x be such that $\alpha(x) \simeq s$ and let there exist $\langle u, x\rangle \in W_1$ such that $E_u \subseteq \mathfrak{B}$. This implies $(\tilde{E}_u)_{\mathfrak{A}} \simeq 0$. From the compatibility of Γ and $\langle \alpha', H'\rangle$ and the correctness of E_u it follows that $g(x) \in Dom(\alpha')$. And we also have $\tau^x_{\mathfrak{A}} \simeq \alpha(x) \simeq s$, hence $(T_0(\tau^x))_{\mathfrak{A}} \simeq true$. From the above arguments we obtain $P \vdash \tilde{E}_v \cup \{T_0(\tau^x)\} \Rightarrow F(\tau^x)$ and from the LP-definability of A it follows that $s \in A$, that is $\alpha(\Gamma(\langle \mathfrak{B}\rangle)) \subseteq A$.

Now let $s \in A$. Then there exists $\tau \in \mathfrak{T}_C$ such that $\tau_{\mathfrak{A}} \simeq s$ and $\partial(\mathfrak{A})^C \cup P \vdash H(\tau)$. From the reduction theorem it follows that there exists a finite set G of atoms without variables such that $G_{\mathfrak{A}} \simeq 0$ and $P \vdash G \Rightarrow F(\tau)$. Hence, there exists $\langle u, x\rangle \in W_1$ such that $g(x) \in Dom(\alpha')$, $\tau \equiv \tau^x$, $\tilde{E}_v \subseteq G$ and $\tau^x_{\mathfrak{A}} \simeq s$. Let

$$L = \{g(y) \mid \langle j, y\rangle \in E_v \ \& \ 1 \leq j \leq n + k \ \& \ g(y) \notin Dom(\alpha')\}.$$

Let $\alpha'' \equiv \alpha'$ and $H'' \simeq H' \cup L$. Let $\langle \alpha, \mathfrak{B}\rangle$ extend $\langle \alpha'', H''\rangle$. Then $\langle \alpha, \mathfrak{B}\rangle$ also extends $\langle \alpha', H'\rangle$. Consider the set E_v and let $t = \langle j, y\rangle \in E_v$. There are two possibilities:

1. $g(y) \in Dom(\alpha_q)$, hence $\Pi^t_{\mathfrak{A}} \simeq 0$, that is $\langle j, x\rangle \in \langle \mathfrak{B}\rangle$.

2. $g(y) \notin Dom(\alpha_q)$, hence $g(y) \in H''$. If $1 \leq j \leq n$, then $\langle j, x\rangle \in \langle\mathfrak{B}\rangle$ from the fact that $\langle\alpha, \mathfrak{B}\rangle$ is standard and $Dom(\alpha) \cap H'' = \varnothing$. In the case $n+1 \leq j \leq n+k$ we also obtain $\langle j, x\rangle \in \langle\mathfrak{B}\rangle$ from the definition of extension.

Finally, we have proved that $\tilde{E}_v \subseteq \langle\mathfrak{B}\rangle$, i. e. $x \in \Gamma(\langle\mathfrak{B}\rangle)$. On the other hand, $\alpha(x) \simeq \tau^x_{\mathfrak{A}} \simeq \tau_{\mathfrak{A}} \simeq s$ or $s \in \alpha(\Gamma(\langle\mathfrak{B}\rangle))$. Thus we obtained that A is sufficient for $\langle\alpha', H'\rangle$ and Γ. □

We already proved the next theorem.

Theorem 4. *Each weak-admissible set in $\mathfrak{A}$ is LP-definable.*

Theorem 5. *If the set A is LP-definable by $\langle P, H\rangle$ and $C = \{c_1, \ldots, c_r\}$, then A is $\forall$-weak-admissible.*

Proof. Let $\mathfrak{L}_C = (c_1, \ldots, c_r; f_1, \ldots, f_n; T_0, \ldots, T_k)$ be a first order language corresponding to $\mathfrak{A}$ and $\mathfrak{T}_C$ be the set of the terms without variables of $\mathfrak{L}_C$. Let

$$s \in A \Leftrightarrow \exists\tau(\tau \in \mathfrak{T}_C \,\&\, \partial^C(\mathfrak{A}) \cup P \vdash H(\tau) \,\&\, \tau_{\mathfrak{A}} \simeq s). \qquad (*)$$

Fix an arbitrary partial enumeration $\langle\alpha, \mathfrak{B}\rangle$ of $\mathfrak{A}$. We shall define a r. e. in $\mathfrak{B}$ set W of naturals such that $W \subseteq Dom(\alpha)$ and $\alpha(W) \equiv A$.

Let $\mathfrak{L} = (f_1, \ldots, f_n; T_0, \ldots, T_k)$ and let for each $(c_i)_{\mathfrak{A}} \in B$ choose $x_i \in N$ such that $\alpha(x_i) \simeq (c_i)_{\mathfrak{A}}$ (there exists such x_i since α is a surjective mapping). Let $K' = \{x_1, \ldots, x_r\}$ and $K = \{\tilde{x} \mid x \in K'\}$, where $\tilde{x}$ is a new constant for each $x \in K'$ and $K \cap \mathfrak{L}_C = \varnothing$. Since K is a finite set, K is r. e. in $\mathfrak{B}$. Let $L_K = L \cup K$ and let $\mathfrak{B}^*$ be the enrichment of $\mathfrak{B}$ in $\mathfrak{L}_K$, where $\tilde{x}$ is interpreted as x. Consider the set $\mathfrak{T}_K$ of terms without variables of $\mathfrak{L}_K$. For each term τ of $\mathfrak{T}_K$ we define a term $\lceil\tau\rceil$ of $\mathfrak{T}_C$ by the following inductive clauses:

(i) if $\tau = \tilde{x}$ for some $\tilde{x} \in K$, then $\lceil\tau\rceil = c_i \Leftrightarrow \alpha(x) = (c_i)_{\mathfrak{A}}$;

(ii) if $\tau = f_i(\tau^1)$, then $\lceil\tau\rceil = f_i(\lceil\tau^1\rceil)$.

It is easily seen that for each term $\tau \in \mathfrak{T}_K$, $\tau_{\mathfrak{B}^*}$ is defined iff $\lceil\tau\rceil_{\mathfrak{A}}$ is defined and also that $\alpha(\tau_{\mathfrak{B}^*}) \simeq \lceil\tau\rceil_{\mathfrak{A}}$.

Let $\partial^C_1(\mathfrak{B}) = \{T_j(\tau) \mid 0 \leq j \leq k \,\&\, \tau \in \mathfrak{T}_K \,\&\, T_j(\lceil\tau\rceil) \in \partial^C(\mathfrak{A})\}$. The set $\partial^C_1(\mathfrak{B})$ is r. e. in $\mathfrak{B}$, because for $1 \leq j \leq k$ and $\tau \in \mathfrak{T}_K$ the following equivalences hold:

$$T_j(\tau) \in \partial^C_1(\mathfrak{B}) \Leftrightarrow T_j(\lceil\tau\rceil) \in \partial^C(\mathfrak{A}) \Leftrightarrow \Sigma_j(\lceil\tau\rceil_{\mathfrak{A}}) \simeq 0$$
$$\Leftrightarrow \Sigma_j(\alpha(\tau_{\mathfrak{B}^*})) \simeq 0 \Leftrightarrow \sigma_j(\tau_{\mathfrak{B}^*}) \simeq 0.$$

By changing each appearance of c_i to $\tilde{x}_i$ in P and τ we obtain $\tilde{P}$ and $\tilde{\tau}$. Now we define the set W by

$$x \in W \Leftrightarrow \exists\tilde{\tau}(\tilde{\tau} \in \mathfrak{T}_K \,\&\, \tilde{\tau}_{\mathfrak{B}^*} \simeq x \,\&\, \partial^C_1(\mathfrak{B}) \cup \{\tilde{P}\} \vdash H(\tilde{\tau})).$$

From that definition it is clear that W is r. e. in $\mathfrak{B}$. Since $Dom(\alpha)$ is closed with respect to φ_i, $1 \leq i \leq n$, we have $W \subseteq Dom(\alpha)$. And finally, from the constant theorem and reduction theorem it follows that $\alpha(W) = A$, which proves the theorem. □

Now we are ready to state the main results of the paper as corollaries of the theorems already proved.

Corollary 1. *The subset A of B is LP-definable iff there exists $\theta \in F(\mathfrak{A})$ such that $A = \theta(t_{01}, \ldots, t_{0r})$ for some fixed $(t_{01}, \ldots, t_{0r}) \in B^r$.*

Corollary 2. *The subset A of B is LP-definable iff A is weak-computable.*

7. CONCLUSIONS AND RELATED WORK

The subject of this paper is a semantics of logic programs without searching in the structure domain. The paper is a part of a more general exploration being performed at the Department of Mathematical Logic of Sofia University. All these works use the enumerations approach which is extremely suitable for problems of finding normal form of objects obtained by certain kind of computations. In [3] is considered a semantic, for which searching in the domain of the structure is allowed. There are also results for more general parameterized semantics.

Acknowledgements. The author would like to thank Prof. Soskov for introducing her to the matter and for his inestimable help.

REFERENCES

1. Rogers, Jr. H. Theory of Recursive Functions and Effective Computability. McGraw–Hill, New York, 1967.
2. Soskov, I. Definability via Enumerations. *The Journal of Symbolic Logic*, **54**, 1989, 428–440.
3. Soskov, I. On the Computational Power of the Logic Programs. *In*: Heyting'88 Mathematical Logic Plenum Press, New York – London, 1989, 117–137.
4. Soskov, I. Computability by Means of Effectively Definable Schemes and Definability via Enumerations. *Archive for Mathematical Logic*, **29**, 1990, 187–200.

Received March 18, 1999

Faculty of Mathematics and Informatics
"St. Kliment Ohridski" University of Sofia
5 James Bourchier Blvd.
BG-1164 Sofia, Bulgaria
E-mail: vbaleva@@fmi.uni-sofia.bg

ГОДИШНИК НА СОФИЙСКИЯ УНИВЕРСИТЕТ „СВ. КЛИМЕНТ ОХРИДСКИ“
ФАКУЛТЕТ ПО МАТЕМАТИКА И ИНФОРМАТИКА
Книга 1 — Математика и механика
Том 92, 1998

ANNUAIRE DE L'UNIVERSITE DE SOFIA „ST. KLIMENT OHRIDSKI“
FACULTE DE MATHEMATIQUES ET INFORMATIQUE
Livre 1 — Mathématiques et Mecanique
Tome 92, 1998

DEFINABILITY VIA PARTIAL ENUMERATIONS WITH SEMICOMPUTABLE CODOMAINS*

STELA K. NIKOLOVA

Let $\mathfrak{A}$ be a total abstract structure. We prove that if a set $A \subseteq |\mathfrak{A}|^n$ is admissible in every partial enumeration of $\mathfrak{A}$ with semicomputable codomain, then A is semicomputable in $\mathfrak{A}$ in the sense of Friedman – Shepherdson.

Keywords: abstract computability, external definability, enumerations
1991/95 Math. Subject Classification: 03D70, 03D75

1. INTRODUCTION

There are two major ways to introduce a notion of computable function on an arbitrary abstract structure $\mathfrak{A}$. Using the first one, which we may call explicit, computable functions are defined by means of relativized programs, generalized algorithms, formulas, etc. The second approach, known as implicit, reduces the problem to computability on natural numbers making use of various types of enumerations of the structure.

It turns out that most of the well-known explicit notions of abstract computability can be characterized via enumerations. As a rule, when considering computability without "search" over the domain, we need a suitable notion of partial enumeration. A typical result of this type is the next theorem from [3], which

* This work was partially supported by the Ministry of Education, Science and Technologies, Contract I 604.

characterizes the semicomputable sets, i.e. the sets, whose semicharacteristic functions are computable in the sense of Friedman – Shepherdson ([1, 2]), using finitely many constants from $|\mathfrak{A}|$.

1.1. Theorem. *A set $A \subseteq |\mathfrak{A}|^n$ is semicomputable in $\mathfrak{A}$ if and only if A is admissible in every partial enumeration of $\mathfrak{A}$.*

Here an interesting question is whether there exists a subclass $\mathfrak{K}$ of the class of all partial enumerations such that admissibility in every enumeration in $\mathfrak{K}$ guarantees semicomputability, and further, whether there exists minimal such class. This question is answered partially in [4], where it is proved that admissibility in all partial enumerations with Σ_2^0 domains yields semicomputability. Here we show that the same is true if we confine ourselves to the class of all enumerations with Π_1^0 domains and this is the minimal class with this property.

2. PRELIMINARIES

Let an abstract structure $\mathfrak{A} = (B; \theta_1, \ldots, \theta_k, \Sigma_1, \ldots, \Sigma_m)$ be given, where the set B is finite or denumerable, θ_i is a total function of a_i arguments in B and Σ_j is a total predicate of b_j arguments in B. The equality relation is not supposed to be among the initial predicates of $\mathfrak{A}$. We shall write $\Sigma_j(\bar{s}) = 0$ (1) when $\Sigma_j(\bar{s})$ is *true* (resp. *false*).

2.1. Definition. *Partial enumeration* of $\mathfrak{A}$ is an ordered pair $(f, \mathfrak{B})$, where f is a partial function from N (the set of all natural numbers) onto B, $\mathfrak{B} = (N; \varphi_1, \ldots, \varphi_k, Q_1, \ldots, Q_m)$ is a total structure in the signature of $\mathfrak{A}$, and the following conditions hold for $1 \leq i \leq k$ and $1 \leq j \leq m$:

(1) if $x_1, \ldots, x_{a_i}$ are in $Dom(f)$, then $\varphi_i(x_1, \ldots, x_{a_i}) \in Dom(f)$;

(2) $f(\varphi_i(x_1, \ldots, x_{a_i})) = \theta_i(f(x_1), \ldots, f(x_{a_i}))$ for $x_1, \ldots, x_{a_i}$ in $Dom(f)$;

(3) $Q_j(x_1, \ldots, x_{b_j}) \iff \Sigma_j(f(x_1), \ldots, f(x_{b_j}))$ for $x_1, \ldots, x_{b_j}$ in $Dom(f)$.

In other words, the pair $(f, \mathfrak{B})$ is a partial enumeration of $\mathfrak{A}$ if the mapping $f \restriction Dom(f)$ is a strong homomorphism from $\mathfrak{B} \restriction Dom(f)$ onto $\mathfrak{A}$.

The set $Dom(f)$ is called *domain* of the enumeration $(f, \mathfrak{B})$.

A set $W \subseteq N^n$ is *semicomputable* in $\mathfrak{B}$ iff the semicharacteristic function of W is Turing computable relative to $\varphi_1, \ldots, \varphi_k, Q_1, \ldots, Q_m$.

2.2. Definition. A set $A \subseteq B^n$ is *admissible* in the enumeration $(f, \mathfrak{B})$ iff there exists a semicomputable in $\mathfrak{B}$ set $W \subseteq N^n$ such that for all $x_1, \ldots, x_n$ in $Dom(f)$

$$(x_1, \ldots, x_n) \in W \iff (f(x_1), \ldots, f(x_n)) \in A.$$

The set W is called an *associate* of A (in the enumeration $(f, \mathfrak{B})$).

Next we introduce the notion of semicomputable set in the sense of Friedman – Shepherdson [1, 2]. Say that the n-ary predicate Π in B is *elementary* iff it is

a finite conjunction of atomic predicates or their negations. Suppose that some effective coding of all elementary predicates (of arbitrary number of arguments) is fixed and denote by Π^v the predicate with code v.

A set $A \subseteq B^n$ is *semicomputable* in $\mathfrak{A}$ iff there exist constants $t_1, \ldots, t_l$ in B, $l \geq 0$, and an unary recursive function γ such that for every v, $\gamma(v)$ is a code of an elementary predicate with variables among $X_1, \ldots, X_n, Y_1, \ldots, Y_l$ and the equivalence

$$(s_1, \ldots, s_n) \in A \iff \exists v \left(\Pi^{\gamma(v)}(X_1/s_1, \ldots, X_n/s_n, Y_1/t_1, \ldots, Y_l/t_l) = 0 \right)$$

is satisfied for every $(s_1, \ldots, s_n) \in B^n$.

3. STANDARD ENUMERATIONS

In order to save space, from now on we shall suppose that the initial functions and predicates of the structure $\mathfrak{A} = (B; \theta_1, \ldots, \theta_k, \Sigma_1, \ldots, \Sigma_m)$ are unary. We shall consider also subsets of B (instead of B^n).

For our goal it is sufficient to confine ourselves to some special type of enumerations, called standard enumerations [3]. To introduce the precise version of this notion that we will need here, let us first fix some recursive coding $\langle\ ,\ \rangle$ of ordered pairs of natural numbers, chosen in such a way that the decoding functions L and R satisfy the condition

$$L(x) < x \ \&\ R(x) < x$$

for all $x \in N$. (Take, for example, $\langle x, y \rangle = 2^x(2y+1)$.) We shall write sometimes $(x)_0$ and $(x)_1$ instead of $L(x)$ and $R(x)$, respectively.

Set

$$N_0 = N \setminus \{\langle i, x \rangle \mid 1 \leq i \leq k,\ x \in N\}.$$

Let f_0 be an arbitrary partial mapping from N_0 onto B. Using a course-by-value-recursion, define f as:

$$f(x) \simeq \begin{cases} f_0(x), & x \in N_0, \\ \theta_i(f(x_0)), & x = \langle i, x_0 \rangle \text{ for some } 1 \leq i \leq k. \end{cases}$$

Now for $1 \leq i \leq k$ and $1 \leq j \leq m$ set

$$\varphi_i(x) = \langle i, x \rangle$$

and

$$Q_j(x) = \begin{cases} \Sigma_j(f(x)), & x \in Dom(f), \\ \text{arbitrary}, & \text{otherwise}. \end{cases}$$

It is an easy exercise to check that the pair $(f, \mathfrak{B} = (N; \varphi_1, \ldots, \varphi_k, Q_1, \ldots, Q_m))$ is an enumeration of $\mathfrak{A}$. Every enumeration, obtained in the way just described, we shall call a *standard* enumeration.

Let $W \subseteq N$ is semicomputable in $\mathfrak{B}$. Since $\mathfrak{B}$ is total, it is equivalent to the fact that $W = \Gamma_e(\langle \mathfrak{B} \rangle)$ for some enumeration operator with index e, more precisely,

$$W = \{x \mid \exists v(\langle v, x \rangle \in W_e \ \&\ D_v \subseteq \langle \mathfrak{B} \rangle)\},$$

where W_e is the e-th r. e. subset of N, $D_v \subseteq N$ is the finite set with canonical index v and $\langle \mathfrak{B} \rangle$ is "the code" of $\mathfrak{B}$, i.e. the set, which consists of the codes $\langle i, x, y \rangle$ of the triples (i, x, y), such that

$$(1 \leq i \leq k \ \& \ y = \langle i, x \rangle) \ \vee \ (k+1 \leq i \leq k+m \ \& \ Q_i(x) = y).$$

Let $root(x)$ be the recursive function such that

$$root(x) = \begin{cases} x, & \text{if } x \in N_0, \\ root(x_0), & \text{if } x = \langle i, x_0 \rangle \text{ for some } 1 \leq i \leq k. \end{cases}$$

Clearly, $root(x) \in N_0$ for every $x \in N$. Define $F : N \times B \longrightarrow B$ as

$$F(x, s) = \begin{cases} s, & \text{if } x \in N_0, \\ \theta_i(F(x_0, s)), & \text{if } x = \langle i, x_0 \rangle \text{ for some } 1 \leq i \leq k. \end{cases}$$

So, thinking of x as a code of an one-variable term in the signature of $\mathfrak{A}$, $F(x, s)$ is the value (in $\mathfrak{A}$) of this term, when its variable is evaluated to s.

Below we introduce appropriate notions of a finite part and a forcing relation. Let $z_0, z_1, \ldots$ be an enumeration of the elements of N_0 in an ascending order.

3.1. Definition. *Finite part* (of a standard enumeration) is an $(m+2)$-tuple

$$\tau = (f_\tau; H_\tau; q_1, \ldots, q_m),$$

where f_τ is a finite function from N into B, $H_\tau \subseteq N_0$ is a finite set, $Dom(f_\tau) \cap H_\tau = \varnothing$, $Dom(f_\tau) \cup H_\tau = \{z_0, \ldots, z_l\}$ for some $l \geq 0$, and $q_1, \ldots, q_m$ are unary finite predicates satisfying the additional condition

$$x \in Dom(q_i) \implies root(x) \in H_\tau.$$

The set $Dom(f_\tau) \cup H_\tau$ we shall call *domain* of τ (to be denoted by $Dom(\tau)$). If $Dom(\tau) = \{z_0, \ldots, z_l\}$, then l is the *length* of τ (in symbols $|\tau|$).

Whenever $\tau = (f_\tau; H_\tau; q_1, \ldots, q_m)$ is a finite part, $x \in N_0$ is the first not in $Dom(\tau)$ and $s \in B$, by $\tau * s$ we shall denote the tuple $(g; H_\tau; q_1, \ldots, q_m)$, where g is the function with graph $G_{f_\tau} \cup \{(x, s)\}$. Clearly, $\tau * s$ is a finite part.

Let $\tau = (f_\tau; H_\tau; q_1, \ldots, q_m)$ and $\delta = (f_\delta; H_\delta; r_1, \ldots, r_m)$ be arbitrary finite parts. We introduce three types of partial relations between finite parts:

$$\tau \subseteq \delta \iff f_\tau \subseteq f_\delta \ \& \ H_\tau \subseteq H_\delta \ \& \ q_1 \subseteq r_1 \ \& \ \ldots \ \& \ q_m \subseteq r_m;$$

$$\tau \leqslant \delta \iff \tau \subseteq \delta \ \& \ f_\tau = f_\delta;$$

$$\tau \preccurlyeq \delta \iff \tau \leqslant \delta \ \& \ H_\tau = H_\delta.$$

As usual, we will write $\delta \supseteq \tau$, $\delta \geqslant \tau$, ... for $\tau \subseteq \delta$, $\tau \leqslant \delta$, etc.

The enumeration $(f, \mathfrak{B} = (N; \varphi_1, \ldots, \varphi_k, Q_1, \ldots, Q_m))$ *extends* τ ($\tau \subseteq (f, \mathfrak{B})$) iff $f_\tau \subseteq f$, $H_\tau \subseteq N \setminus Dom(f)$ and $q_i \subseteq Q_i$ for $i = 1, \ldots, m$.

Now set

$$\begin{aligned} \tau \Vdash u \iff \exists x \exists y \exists i (u = \langle i, x, y \rangle \ \& \ 1 \leq i \leq k \ \& \ y = \langle i, x \rangle \ \vee \\ u = \langle k+i, x, y \rangle \ \& \ 1 \leq i \leq m \ \& \ (q_i(x) = y \ \vee \\ root(x) \in Dom(f_\tau) \ \& \ \Sigma_i(F(x, f_\tau(root(x)))) = y)); \end{aligned}$$

$$\tau \Vdash D_v \iff \forall u (u \in D_v \implies \tau \Vdash u);$$

$$\tau \Vdash R_e(x) \iff \exists v (\langle v, x \rangle \in W_e \ \& \ \tau \Vdash D_v).$$

The next simple observation will be of use in the sequel.

3.2. Lemma.

(1) $\tau \Vdash R_e(x)$ & $\tau \subseteq \rho \implies \rho \Vdash R_e(x)$ (*monotonisity*);

(2) $\tau \Vdash R_e(x)$ & $\tau \subseteq (f, \mathfrak{B}) \implies x \in \Gamma_e(\langle \mathfrak{B} \rangle)$;

(3) $x \in \Gamma_e(\langle \mathfrak{B} \rangle) \implies \exists \tau(\tau \subseteq (f, \mathfrak{B})$ & $\tau \Vdash R_e(x))$;

(4) $\tau \subseteq (f, \mathfrak{B})$ & $\forall \rho(\rho \geqslant \tau \implies \rho \nVdash R_e(x)) \implies x \notin \Gamma_e(\langle \mathfrak{B} \rangle)$.

Proof. The verification of (1) – (3) is straightforward.

(4) Towards contradiction assume that $x \in \Gamma_e(\langle \mathfrak{B} \rangle)$. Then by (3) there exists some $\delta = (f_\delta; H_\delta; r_1, \ldots, r_m)$ such that $\delta \subseteq (f, \mathfrak{B})$ and $\delta \Vdash R_e(x)$. Since $\tau = (f_\tau; H_\tau; q_1, \ldots, q_m) \subseteq (f, \mathfrak{B})$ as well, the sets $G = Dom(f_\tau) \cup Dom(f_\delta)$ and $H = H_\tau \cup H_\delta$ are disjoint and $G \cup H$ is an initial segment of N_0. It is clear also that the predicates $r'_i = q_i \cup r_i$, $1 \leq i \leq m$, are single-valued and the ordered tuple $\delta' = (f_\tau \cup f_\delta; H_\tau \cup H_\delta; r'_1, \ldots, r'_m)$ is a finite part. We have $\delta' \supseteq \delta$, hence $\delta' \Vdash R_e(x)$. It means that $\delta' \Vdash D_v$ for some v with $\langle v, x \rangle \in W_e$. Now consider the tuple $\rho = (f_\tau; H \cup (G \setminus Dom(\tau)); p_1, \ldots, p_m)$, where

$$p_i = r'_i \cup \{\langle x, y \rangle \mid x \in G \setminus Dom(\tau) \& \exists u(u \in D_v \ \& \ u = \langle k + i, x, y \rangle)\}.$$

Clearly, $\rho \Vdash D_v$ and $\rho \geqslant \tau$ — a contradiction. □

4. THE MAIN RESULT

In order to establish our result, we introduce a suitable notion of normal form of a subset of B.

4.1. Definition. A set $A \subseteq B$ has a *normal form*, if there exist a finite part δ and a natural number e such that if $x \in N_0$ is the first not in $Dom(\delta)$, then

$$s \in A \iff \exists \rho(\rho \geqslant \delta * s \ \& \ \rho \Vdash R_e(x))$$

for every $s \in B$.

Now the rest of the paper is devoted to the proof of the next theorem.

4.2. Theorem. *Let $A \subseteq B$. The following conditions are equivalent:*

(1) *A is semicomputable in $\mathfrak{A}$;*

(2) *A is admissible in every enumeration $(f, \mathfrak{B})$ such that $N \setminus Dom(f)$ is semicomputable in $\mathfrak{B}$;*

(3) *A has a normal form.*

Proof. The implication (1) $\implies$ (2) follows immediately from the definitions. To see that (3) $\implies$ (1) holds, take into account the next two observations:

(a) The set $R = \{(e, x) \mid \rho \Vdash R_e(x)\}$ is semicomputable in $\mathfrak{A}$.

(b) In order to find (if it exists) a finite part ρ such that $\rho \geqslant \tau$ (with τ fixed), it is sufficient to search over natural numbers (notice that for the more common inclusion "$\supseteq$" this is not true). More precisely:

$$\rho \geqslant \tau \iff \exists H \exists r_1, \ldots, r_m \big(H \supseteq H_\tau \;\&\; r_1 \supseteq q_1^\tau \;\&\; \ldots r_m \supseteq q_m^\tau \;\&\; (f_\tau; H; r_1, \ldots, r_m) \text{ is a finite part}\big).$$

The interesting part of the theorem is the direction (2) $\Longrightarrow$ (3), which now we prepare to prove using some auxiliary lemmas.

Indeed, assume that (2) holds, but the set A does not have a normal form. We are going to construct a standard enumeration $(f, \mathfrak{B})$ in which A is not admissible and such that $N \setminus Dom(f)$ is semicomputable in $\mathfrak{B}$.

Clearly, $A \neq \varnothing$ — check that the empty set has a normal form. Therefore, if $\Gamma_e(\langle \mathfrak{B} \rangle)$ is an associate of A, then $W_e \neq \varnothing$. So it will be sufficient to consider only the indexes of non-empty r. e. subsets of N. As it is well-known, every set of this type can be enumerated by some unary primitive recursive function. We will need in fact some uniform procedure that enumerates the elements of nonempty r. e. sets. So consider $U(n, x)$ — the universal for all unary primitive recursive functions. By the S_n^m-theorem, there exists recursive function σ:

$$W_{\sigma(e)} = Range(\lambda x.U(e.x)).$$

Our aim is to construct successively a sequence of finite parts $\tau^{(0)} \subseteq \tau^{(1)} \subseteq \ldots$ such that for every enumeration $(f, \mathfrak{B})$ of $\mathfrak{A}$ and every n it is true that

$$\text{if } \tau^{(2n+1)} \subseteq (f, \mathfrak{B}), \text{ then } \Gamma_{\sigma(n)}(\langle \mathfrak{B} \rangle) \text{ is not an associate of } A. \qquad (*)$$

It is clear from here that if $(f, \mathfrak{B})$ is an enumeration, such that $\tau^{(n)} \subseteq (f, \mathfrak{B})$ for every n, then A is not admissible in $\mathfrak{A}$. Indeed, assuming the contrary, we will have an index e such that $\Gamma_e(\langle \mathfrak{B} \rangle)$ is an associate of A. Since $W_e \neq \varnothing$, there exists n with $W_{\sigma(n)} = W_e$. Therefore $\Gamma_{\sigma(n)}(\langle \mathfrak{B} \rangle)$ is an associate of A, which contradicts $(*)$.

Let us fix some enumeration $s_0, s_1, \ldots$ of the elements of B. Now we are going to define the sequence $\{\tau^{(n)}\}_n$ satisfying $(*)$. The definition is by induction on n.

Set $\tau^{(0)} = \left(f^{(0)}; H^{(0)}; q_1^{(0)}, \ldots, q_m^{(0)}\right)$, where $f^{(0)}$ is the function (with graph) $\{(z_0, s_0)\}$, $H^{(0)} = \varnothing$ and all $q_1^{(0)}, \ldots, q_m^{(0)}$ are unary predicates with empty domains. Assuming that $\tau^{(2n)} = \left(f^{(2n)}; H^{(2n)}; q_1^{(2n)}, \ldots, q_m^{(2n)}\right)$, $n \geq 0$, is already determined, we define $\tau^{(2n+1)}$ and $\tau^{(2n+2)}$ as follows.

Let $x \in N_0$ be the first which does not belong to $Dom\left(\tau^{(2n)}\right)$. By assumption A does not have a normal form, so there exists $s \in B$ such that exactly one of the next two conditions holds:

- $s \in A \;\&\; \forall \rho(\rho \geqslant \tau^{(2n)} * s \implies \rho \nVdash R_{\sigma(n)}(x))$;
- $s \notin A \;\&\; \exists \rho(\rho \geqslant \tau^{(2n)} * s \;\&\; \rho \Vdash R_{\sigma(n)}(x))$.

In the first case put

$$\tau^{(2n+1)} = \tau^{(2n)} * s.$$

If the second case holds, take some finite part ρ such that

$$\rho \geqslant \tau^{(2n)} * s \ \& \ \rho \Vdash R_{\sigma(n)}(x)$$

and set $\tau^{(2n+1)} = \rho$.

Let us notice that this rather arbitrary choice of the above ρ is sufficient only to establish $(*)$. To claim that the codomain of $(f, \mathfrak{B})$ is semicomputable in $\mathfrak{B}$, we will further have to choose ρ more carefully.

Let s be the first in the list $s_0, s_1, \ldots$, which is not in $Range\left(\tau^{(2n+1)}\right)$. Set

$$\tau^{(2n+2)} = \tau^{(2n+1)} * s.$$

Now let $(f, \mathfrak{B})$ be an enumeration which extends $\tau^{(2n+1)}$. In order to establish $(*)$, assume that $\Gamma_{\sigma(n)}(\langle\mathfrak{B}\rangle)$ is an associate of A in $(f, \mathfrak{B})$. Then for every $x \in Dom(f)$

$$x \in \Gamma_{\sigma(n)}(\langle\mathfrak{B}\rangle) \iff f(x) \in A. \tag{4.1}$$

Now let $x \in N_0$ be the first not in $Dom\left(\tau^{(2n)}\right)$.

By definition $f^{(2n+1)}(x) = s$, hence $x \in Dom(f)$. According to the choice of s we have that either $\bullet$ or $\bullet\bullet$ is true. Suppose first that $s \in A \ \& \ \forall\rho(\rho \geqslant \tau^{(2n)} * s \implies \rho \nVdash R_{\sigma(n)}(x))$. By Lemma 3.2 (4) $x \notin \Gamma_{\sigma(n)}(\langle\mathfrak{B}\rangle)$, so using 4.1, we obtain $f(x) = s \notin A$ — a contradiction. Therefore it is the case $s \notin A \ \& \ \exists\rho(\rho \geqslant \tau^{(2n)} * s \ \& \ \rho \Vdash R_{\sigma(n)}(x))$. According to Lemma 3.2 (2) $x \in \Gamma_{\sigma(n)}(\langle\mathfrak{B}\rangle)$ and again by (4.1) $f(x) = s \in A$, which is also impossible.

Now set

$$f_0 = \bigcup_n f^{(n)}, \ H = \bigcup_n H^{(n)}, \ q_i = \bigcup_n q_i^{(n)} \quad \text{for } 1 \leq i \leq m.$$

Obviously, $Dom(f_0) \cup H = \varnothing$ (otherwise there will be some n such that $x \in Dom\left(f^{(n)}\right) \ \& \ x \in H^{(n)}$) and $Dom(f_0) \cup H = N_0$. Notice also that with the even steps of the definition of $\left\{\tau^{(n)}\right\}_n$ it is ensured that f_0 is a partial mapping onto B.

Define the predicates Q_j, $1 \leq j \leq m$, as

$$Q_j(x) = \begin{cases} \Sigma_j(f(x)), & \text{if } x \in Dom(f), \\ q_j(x), & \text{if } x \in Dom(q_j), \\ 0, & \text{otherwise.} \end{cases}$$

This definition is correct since for any $1 \leq j \leq m$:

$$x \in Dom(f) \iff root(x) \in Dom(f_0) \iff root(x) \notin \bigcup_n H^{(n)} \implies$$

$$\forall n \left(x \notin Dom\left(q_j^{(n)}\right)\right) \iff x \notin Dom(q_j).$$

Now putting

$$f(x) \simeq \begin{cases} f_0(x), & \text{if } x \in Dom(f_0), \\ \theta_i(f(x_0)), & \text{if } x = \langle i, x_0\rangle \text{ for some } 1 \leq i \leq k, \end{cases}$$

we obtain a standard enumeration $(f, \mathfrak{B} = (N; \lambda x.\langle 1, x\rangle, \ldots, \lambda x.\langle k, x\rangle, Q_1, \ldots, Q_m))$ of $\mathfrak{A}$. It is clear that for arbitrary n:

$$f^{(n)} \subseteq f_0 \subseteq f \;\&\; q_j^{(n)} \subseteq q_j \subseteq Q_j,\ 1 \leq j \leq m,\ \&\ H^{(n)} \subseteq H \subseteq N \setminus Dom(f).$$

Therefore $\tau^{(n)} \subseteq (f, \mathfrak{B})$ for every n, which immediately brings us to the conclusion that A is not admissible in $\mathfrak{B}$.

The work done so far repeats in essence the respective proof in [3]. Our aim is to show that if we choose more precisely the finite parts ρ (when it is the case $\bullet\bullet$), then we may claim that $N \setminus Dom(f)$ is semicomputable in $\mathfrak{B}$.

Set for brevity

$$\tau \Vdash P_e(x) \iff \tau \Vdash R_{\sigma(e)}(x).$$

We have by definition

$$\tau \Vdash P_e(x) \iff \exists v(\langle v, x\rangle \in W_{\sigma(e)} \;\&\; \tau \Vdash D_v) \iff$$

$$\exists t \exists v(U(e,t) = \langle v, x\rangle \;\&\; \tau \Vdash D_v)) \iff \exists t((U(e,t))_1 = x \;\&\; \tau \Vdash D_{(U(e,t))_0}).$$

Put

$$\tau \Vdash_t P_e(x) \iff \exists t_0 \left(t_0 \leq t \;\&\; (U(e,t_0))_1 = x \;\&\; \tau \Vdash D_{(U(e,t_0))_0}\right).$$

Obviously,

$$\tau \Vdash P_e(x) \iff \exists t(\tau \Vdash_t P_e(x)).$$

The first t with $\tau \Vdash_t P_e(x)$ may be thought of as the first step at which the validity of the fact that $\tau \Vdash P_e(x)$ is established.

For a finite part $\tau = (f_\tau; H_\tau; q_1, \ldots, q_m)$ with $|\tau| = w$ put

$$\tau_l = (f_\tau; H_\tau \cup \{z_{w+1}, \ldots, z_{w+l}\}; q_1, \ldots, q_m).$$

Clearly, for each $l \geq 1$, τ_l is a finite part, too.

The next lemma will be of use when constructing the modified sequence $\{\tau^{(n)}\}_n$.

4.3. Lemma. *Suppose that* $\exists \rho(\rho \geqslant \tau \;\&\; \rho \Vdash P_e(x))$. *Then there exist* $l \geq 1$ *and* $\rho^* \succcurlyeq \tau_l$ *such that* $\rho^* \Vdash_{l-1} P_e(x)$.

Proof. Let $\rho \Vdash_t P_e(x)$, where $\rho = (f_\rho; H_\rho; q_1, \ldots, q_m)$. Set $l = \max(l_0, t+1)$, where $l_0 = |\rho| - |\tau|$. We claim that the finite part

$$\rho^* = (f_\tau; H_\tau \cup \{z_{|\tau|+1}, \ldots, z_{|\tau|+l}\}; q_1, \ldots, q_m)$$

fulfills the requirements of the lemma.

Indeed, we have $f_{\rho^*} = f_{\tau_l}$, $H_{\rho^*} = H_{\tau_l}$ and q_i extends the i-th initial predicate of τ_l for $1 \leq i \leq m$, so $\rho^* \succcurlyeq \tau_l$. Besides, $\rho^* \supseteq \rho$ and $l - 1 \geq t$, hence $\rho^* \Vdash_{l-1} P_e(x)$. □

Now we make the following refinement in the definition of the odd members of $\{\tau^{(n)}\}_n$. Again assuming that $\tau^{(2n)}$ is already defined and denoting by x_n the first number in the list $z_0, z_1, \ldots$, which is not in $Dom\left(\tau^{(2n)}\right)$, we will have that there exists $p_n \in B$ such that

$$s \in A \;\&\; \forall \rho \left(\rho \geqslant \tau^{(2n)} * p_n \implies \rho \nVdash P_n(x_n)\right) \quad \text{or}$$

$$s \notin A \ \&\ \exists \rho \left(\rho \geqslant \tau^{(2n)} * p_n \ \&\ \rho \Vdash P_n(x_n) \right) .$$

Set for brevity $\tau = \tau^{(2n)} * p_n$. If the first of the above two cases holds, set $\tau^{(2n+1)} = \tau$, otherwise start to look for the least $l \geq 1$, for which there exists $\rho^* \geqslant \tau_l$ with $\rho^* \Vdash_{l-1} P_n(x)$ (the definition of τ_l is given immediately before Lemma 4.3). It follows from that lemma that such l exists. Now put $\tau^{(2n+1)} = \rho^*$, where $\rho^* \geqslant \tau_l$ and $\rho^* \Vdash_{l-1} P_e(x)$.

Now let $(f, \mathfrak{B} = (N; \varphi_1, \ldots, \varphi_k, Q_1, \ldots, Q_m))$ be a standard enumeration, obtained from the sequence $\{\tau^{(n)}\}_n$ in the way described before. We have that A is not admissible in $(f, \mathfrak{B})$, so to complete the proof of the theorem, it remains to see that $N \setminus Dom(f)$ is semicomputable in $\mathfrak{B}$.

Let us notice that

$$x \in N \setminus Dom(f) \iff x \notin Dom(f) \iff root(x) \notin Dom(f_0) \iff root(x) \in H.$$

Since $root(x)$ is a recursive function, it is sufficient to see that the set $H = \bigcup_n H^{(n)}$ is semicomputable in $\mathfrak{B}$.

For $t, l \in N$ set

$$\begin{aligned} \mathfrak{B}^{(t)} \vDash_l P_e(x) \iff \exists l_0 \exists v (l_0 \leq l \ \&\ U(e, l_0) = \langle v, x \rangle \ \&\ \forall u (u \in D_v \implies \\ \exists i \exists y (u = \langle i, x, y \rangle \ \&\ ((1 \leq i \leq k \ \&\ y = \langle i, x \rangle) \vee \\ (k+1 \leq i \leq k+m \ \&\ Q_i(x) = y \ \&\ root(x) \leq z_t))))). \end{aligned}$$

Below we describe a procedure P that generates effectively the elements of the set H, asking questions of the type "$\mathfrak{B}^{(t)} \vDash_l P_e(x)$?".

Obviously, the set $R = \{(t, l, e, x) \mid \mathfrak{B}^{(t)} \vDash_l P_e(x)\}$ is decidable in $\mathfrak{B}$, so the set generated by P is semicomputable in $\mathfrak{B}$.

Let $l_n = lh\left(\tau^{(2n)}\right) + 1$. Then $x_n = z_{l_n}$, in particular $x_0 = z_1$. Informally, the procedure for generating $N \setminus Dom(f)$ is the following.

We should begin with asking questions

$$\mathfrak{B}^{(l_0+1)} \vDash_0 P_0(x_0)?, \quad \mathfrak{B}^{(l_0+2)} \vDash_1 P_0(x_0)?, \ \ldots$$

in order to find (if it exists) the first t such that $\mathfrak{B}^{(l_0+t+1)} \vDash_t P_0(x_0)$. If such t does exist, then according to the construction of $\tau^{(1)}$ we put $H^{(1)} = \{z_{l_0+1}, \ldots, z_{l_0+t+1}\}$. Since z_{l_0+t+2} and z_{l_0+t+3} are added to $Dom\left(f^{(2)}\right)$ and $Dom\left(f^{(3)}\right)$, resp., they are not in H. So we should set $l_1 = l_0 + t + 3$, $x_1 = z_{l_1}$ and then start searching for t with $\mathfrak{B}^{(l_1+t+1)} \vDash_t P_1(x_1)$.

Here the problem is that we do not know in advance whether there exists t with $\mathfrak{B}^{(l_0+t+1)} \vDash_t P_0(x_0)$. So if two unsuccessful steps in this search are done (i.e. when $\mathfrak{B}^{(l_0+1)} \nvDash_0 P_0(x_0)$ and $\mathfrak{B}^{(l_0+2)} \nvDash_1 P_0(x_0)$), we decide temporarily that such t does not exist and start simultaneously a similar procedure for seeking the first $t \leq 1$: $\mathfrak{B}^{(l_1'+t+1)} \vDash_t P_1(x_1')$ for $l_1' = l_0 + 2$ and $x_1' = z_{l_1'}'$, i.e. $l_1' = 3$ and $x_1' = z_3$. If such $t \leq 1$ again does not exist, we repeat the same for $n = 2$ and so on. Meanwhile, if we have found (for example for $n = 0$) some t_0 such that $\mathfrak{B}^{(l_0+t_0+1)} \vDash_{t_0} P_0(x_0)$, we interrupt all started procedures for finding out t with $\mathfrak{B}^{(l_i'+t+1)} \vDash_t P_i(x_i')$ for

$i > 0$. Then we print numbers $z_{l_0+1}, \ldots, z_{l_0+t_0+1}$, set $l_1 = l_0 + t + 3$, $x_1 = z_{l_1}$, start searching for t: $\mathfrak{B}^{(l_1'+t+1)} \vDash_t P_1(x_1')$ etc.

However, this least t_0 with $\mathfrak{B}^{(l_0+t_0+1)} \vDash_t P_0(x_0)$ could be found after we have come across some (let say) t_1: $\mathfrak{B}^{(l_1'+t_1+1)} \vDash_{t_1} P_1(x_1')$, where l_1' and x_1' are calculated under the wrong supposition that $\forall t \mathfrak{B}^{(l_0+t+1)} \nvDash_t P_0(x_0)$. So, on the one hand, our algorithm requires the respective set $\{z_{l_1'+1}, \ldots, z_{l_1'+t_1+1}\}$ to be printed right after such a t_1 has been found (since it is supposed to be the set $H^{(3)}$). On the other hand, the "real" $H^{(3)}$ may be different (and in fact is different). However, thanks to the special choice of $\tau^{(1)}$ (to be long enough), it turns out that the printed numbers $z_{l_1'+1}, \ldots, z_{l_1'+t_1+1}$ actually belong to $H^{(1)}$ and hence to H.

Below we describe formally the procedure P that generates H. There the function $g(n,t)$ is intended to be such that $x_n = z_{g(t,n)}$ for sufficiently large t, namely $t \geq |\tau^{(2n-1)}|$. The function $G(t,n)$ from the program P is used to code the information about questions of the type "$\mathfrak{B}^{(t)} \vDash_v P_n(y)$?" for every $n \leq y$ (v, y depending on answers of similar questions for the numbers less than n).

Set

$$\langle y_1 \rangle = y_1; \quad \langle y_1, \ldots, y_{n+1} \rangle = \langle \langle y_1, \ldots, y_n \rangle, y_{n+1} \rangle \quad \text{for } n > 1.$$

Let $\lambda z.(z)_i$ be the recursive function such that if $z = \langle y_1, \ldots, y_n \rangle$ and $i \in \{1, \ldots, n\}$, then $(z)_i = y_i$, and $(z)_i = 1$ — otherwise. We shall obtain $G(t,n)$ in the format $\langle y_0, \ldots, y_t \rangle$, where each y_i will indicate (if t is large enough) whether $z_i \in H^{(n)}$ or not (writing $y_i = 0$ if $z_i \in H^{(n)}$, and $y_i = 1$ if $z_i \in Dom(f_0)$). The value $G(t+1,n)$ will depend on the last member y_n of $G(t,n)$, which is in fact $R(G(t,n))$. Since certainly $z_0 \in Dom(f)$ and $x_0 = z_1$, we put $G(t,0) = \langle 1 \rangle$ and $g(t,0) = 1$ for $t = 0$.

Here follows the exact description of the procedure P.

$t := 0$; $G(t,0) := \langle 1 \rangle$; $g(t,0) := 1$; 1 : $n := 0$;
2: **if** $R(G(t,n)) = 0$ **then**
if $g(t,n) = t+2$
then $G(t+1,n) := \langle G(t,n), 1 \rangle$; $g(t+1,n) := g(t,n)$; $t := t+1$; **go to** 1
else $G(t+1,n) := \langle G(t,n), 3 \rangle$; $g(t+1,n) := g(t,n)$; $n := n+1$; **go to** 2 **fi**
else
if $R(G(t,n)) = 1$ **then**
if $g(t,n) = t+2$
then $G(t+1,n) := \langle G(t,n), 1 \rangle$; $g(t+1,n) := g(t,n)$; $t := t+1$; **go to** 1
else if $g(t,n) = t+1$
then $G(t+1,n) := \langle G(t,n), 1 \rangle$; $G(t+1,n+1) := G(t+1,n)$;
$g(t+1,n) := g(t,n)$; $g(t+1,n+1) := t+3$; $t := t+1$; **go to** 1
else if $\mathfrak{B}^{(t+1)} \vDash_0 P_n(z_{g(t,n)})$
then $G(t+1,n) := \langle G(t,n), 0 \rangle$; **print**$(z_{t+1})$; $G(t+1,n+1) := G(t+1,n)$;
$g(t+1,n) := g(t,n)$; $g(t+1,n+1) := t+3$; $t := t+1$; **go to** 1

else $G(t+1,n) := \langle G(t,n), 2\rangle$; $g(t+1,n) := g(t,n)$; $n := n+1$; **go to** 2
fi fi fi
else
if $R(G(t,n)) = 2$
then $l := t - g(t,n)$; **if** $\mathfrak{B}^{(t+1)} \vDash_l P_n(z_{g(t,n)})$
then $G(t+1,n) := \langle L^l(G(t,n)), \underbrace{0,\ldots,0}_{l+1}\rangle$; **print** $(z_{g(t,n)+1}, z_{g(t,n)+2}, \ldots, z_{t+1})$;
$G(t+1,n+1) := G(t+1,n)$; $g(t+1,n) := g(t,n)$; $g(t+1,n+1) := t+3$;
$t := t+1$; **go to** 1
else $G(t+1,n) := \langle G(t,n), 2\rangle$; $g(t+1,n) := g(t,n)$; $n := n+1$; **go to** 2 **fi**
else $G(t+1,n) := \langle G(t,n), 3\rangle$; $g(t+1,n) := g(t,n)$; $n := n+1$; **go to** 2
fi fi fi.

Let us mention that some of the assignments in the above program are redundant. They are put there only to facilitate the verification of the algorithm.

Denote by $Output(P)$ the collection of all numbers, printed by P. We have to prove that

$$x \in H \iff x \in Output(P). \tag{4.2}$$

By an immediate inspection of P one can notice that for every t, n

$$g(t,n) \leq t+2 \;\&\; g(t,n) \leq g(t,n+1).$$

Set

$$S(n) = \begin{cases} 0, & \text{if } \exists\rho\left(\rho \geqslant \tau^{(2n)} * p_n \;\&\; \rho \Vdash P_n(x_n)\right), \\ 1, & \text{otherwise.} \end{cases}$$

To establish the first direction of (4.2), we will make use of the fact that $G\left(|\tau^{(2n+1)}|, n\right)$ is the code of the characteristic function C_H of H, restricted to the first $|\tau^{(2n+1)}|+1$ members of N_0. In order to prove this, the next more common observation will be needed.

4.4. Lemma.
(1) *For each* $t \geq |\tau_{(2n-1)}|$, $g(t,n) = |\tau^{(2n)}| + 1$ $(|\tau^{(-1)}| = 0)$;
(2) *For each* $t \geq |\tau^{(2n+1)}|$, $G(t,n) = \langle y_0, \ldots, y_t\rangle$, *where*

$$y_i = \begin{cases} 0, & \text{if } i \leq |\tau^{(2n+1)}| \;\&\; z_i \in H^{(2n+1)}, \\ 1, & \text{if } i \leq |\tau^{(2n+1)}| \;\&\; z_i \notin H^{(2n+1)}, \\ 2, & \text{if } i > |\tau^{(2n+1)}| \;\&\; S(n) = 1, \\ 3, & \text{if } i > |\tau^{(2n+1)}| \;\&\; S(n) = 0. \end{cases}$$

Proof. Induction on n. (1) The case $n = 0$ is obvious. In order to check (2) for $n = 0$, we shall separately consider the cases $S(0) = 0$ and $S(0) = 1$. If the latter is true, i.e.

$$\forall\rho\left(\rho \geqslant \tau^{(0)} * p_0 \implies \rho \nVdash P_0(z_0)\right),$$

then $\mathfrak{B}^{(u)} \not\vdash_v P_0(z_0)$ for no $u \geq 1$ and v. So for every t, $G(t,0) = \langle 1,1,\underbrace{2,\ldots,2}_{t-1}\rangle$.

Now suppose that $S(0) = 0$ and denote the finite part $\tau^{(0)} * p_0$ by τ. According to Lemma 4.3 there exists least $l \geq 1$ for which there are finite parts ρ^* such that $\rho^* \geqslant \tau_l$ and $\rho^* \Vdash_{l-1} P_0(x_0)$. Let us remind that by construction $\tau^{(1)} = \rho^*$, where ρ^* satisfies the last conditions. We claim that

$$G(t,0) = \langle 1,1,\underbrace{2,\ldots,2}_{t-1}\rangle \quad \text{for each } 1 \leq t < l+1 \tag{4.3}$$

and

$$G(l+1,0) = \langle 1,1,\underbrace{0,\ldots,0}_{l}\rangle. \tag{4.4}$$

Indeed, assuming that (4.3) does not hold and looking at the program P, we may claim that there is a step $t < l$ such that $1 \leq t < l$ with $\mathfrak{B}^{(t+1)} \models_{t-1} P_0(x_0)$. From here, there exists $\delta \subseteq (f,\mathfrak{B})$ with $|\delta| = t+1$ such that $\delta \Vdash_{t-1} P_0(x_0)$. Since $\tau \subseteq (f,\mathfrak{B})$, we may assume that $\delta \geqslant \tau$. We have $|\tau_t| = t+1$ and $\delta \geqslant \tau_t$. In addition, $\delta \Vdash_{t-1} P_0(x_0)$ and $t < l$ — a contradiction with the choice of l.

To see that (4.4) also holds, recall that $\tau^{(1)} \Vdash_{l-1} P_0(x_0)$, $|\tau^{(1)}| = l+1$ and $\tau^{(1)} \subseteq (f,\mathfrak{B})$. So $\mathfrak{B}^{(l+1)} \models_{l-1} P_0(x_0)$, and as we have just seen, l is the first one with this property. Hence at step $t = l$ we shall have for the first time that $\mathfrak{B}^{(t+1)} \models_{t-1} P_0(x_0)$, so $G(t+1,0) = \langle \underbrace{L(\ldots,L}_{t-1}(G(t,0))\ldots),\underbrace{0,\ldots,0}_{t}\rangle =$ $\langle 1,1,\underbrace{0,\ldots,0}_{t}\rangle$, or $G(l+1,0) = \langle 1,1,\underbrace{0,\ldots,0}_{l}\rangle$.

Clearly, $l \geq 1$, hence $R(G(l+1,0)) = 0$. Then for every $t > l$ we shall have $G(t+1,0) = \langle G(t,0),3\rangle$, in other words, $G(t+1,0) = \langle 1,1,\underbrace{0,\ldots,0}_{l}\underbrace{3,\ldots,3}_{t-l}\rangle$.

Now suppose that for each $j \leq n$ (1) and (2) are true. In particular, for $l = |\tau^{(2n+1)}|$ we have that $G(l,n) = \langle y_0,\ldots,y_l\rangle$, where $y_i = C_H(z_i)$, $0 \leq i \leq l$. Suppose first that $S(n) = 0$. According to the construction of $\tau^{(2n+1)}$ and the program P, at step t with $t+1 = |\tau^{(2n+1)}|$ we have $G(t+1,n+1) = G(t,n)$ and $g(t+1,n+1) = t+3$, in other words, $G(l,n+1) = \langle y_0,\ldots,y_l\rangle$ and $g(l,n+1) = |\tau^{(2n+1)}| + 2$. From the latter, $g(l,n+1) = |\tau^{(2n+2)}| + 1$, and since $g(t,n+1) = g(l,n+1)$ for every $t \geq l$, (1) is established for $n+1$. From here $x_{n+1} = z_{|\tau^{(2n+2)}|+1} = z_{g(t,n+1)}$ for $t \geq l$.

By the induction hypothesis, for every $t \geq l$ and $i < n$ we have that $R(G(t,i)) =$ 2 or 3 if $S(i)$ is 1, resp. 0. Further, $R(G(l,n)) = 0$ and hence for every $t > l$, $R(G(t,n)) = 3$. From here, for every $t \geq l$ there will not be situations that may cause changes in $G(t,n+1)$, due to the extension of an assignment of the type $G(t+1,i+1) := g(t+1,i)$ for some $i \leq n$. In other words, the value of $G(t+1,n+1)$ depends uniquely on the answers of the questions of the type

"$\mathfrak{B}^{(t+1)} \vdash_{t-(|\tau^{(2n+2)}|+1)} P_{n+1}(x_{n+1})$?" (recall that $g(t, n+1) = |\tau^{(2n+2)}| + 1$, as we have already noticed).

Now to complete the verification of (2) for $n+1$, we proceed in essence as in the case $n = 0$. The second case $S(n) = 1$ is treated similarly. □

Our last lemma, which asserts that the program P is correct, completes the proof of the theorem.

4.5. Lemma. $H = \mathit{Output}(P)$.

Proof. For the first inclusion, recall that $H = \bigcup_n H^{(n)}$, where $H^{(0)} = \varnothing$, $H^{(2n)} = H^{(2n-1)}$ and $H^{(0)} \subseteq H^{(1)} \subseteq \ldots$ So if $x \in H$, then there exists n such that $x \in H^{(2n+1)}$ and $x \notin H^{(2n)}$. Further, $x = z_j$ for some $j \leq |\tau^{(2n+1)}|$. According to Lemma 4.4, $G\left(|\tau^{(2n+1)}|, n\right) = \langle y_0, \ldots, y_{|\tau^{(2n+1)}|}\rangle$ with $y_j = 0$. Since $x \notin H^{(2n)}$, $j > |\tau^{(2n)}|$, then at step t with $t+1 = |\tau^{2n+1}|$ the number z_j will be among the numbers, printed by P.

Towards proving the inclusion $\mathit{Output}(P) \subseteq H$, let us notice that

$$z_j \in \mathit{Output}(P) \iff \exists n \exists t\, (\langle G(t+1, n)\rangle_j = 0 \;\&\; g(t, n) \leq j \leq t+1).$$

Then it is sufficient to show that

$$(\exists n \exists t\, \langle G(t, n)\rangle_j = 0) \implies z_j \in H. \tag{4.5}$$

Define the predicate T as follows:

$$T(n) \iff \forall j((\exists t \langle G(t, n)\rangle_j = 0) \implies z_j \in H).$$

We are going to establish $\forall t T(n)$ using induction on n. From here it follows (4.5) for arbitrary j.

To facilitate the inductive step, we suppose that when an assignment of the type $G(t+1, n+1) := G(t+1, n)$ is executed, the value $G(t+1, n)$ is assigned also to $G(t+1, k)$ for every $k > n+1$ for which there exists step $l \leq t$, at which $G(l, k)$ is determined. In other words, instead of single assignment $G(t+1, n+1) := G(t+1, n)$ we perform the finite list of assignments

$$\begin{aligned} &G(t+1, n+1) := G(t+1, n),\\ &G(t+1, n+2) := G(t+1, n),\\ &\ldots\\ &G(t+1, n') := G(t+1, n), \end{aligned}$$

where $n' > n$ can be found effectively.

Let $\mathit{In}(l, \langle y_0, \ldots, y_t\rangle) = \langle y_0, \ldots, y_l\rangle$ for $l \leq t$. The validity of $T(0)$ follows from the proof of Lemma 4.4. We obtained there that for $t < |\tau^{(1)}|$ we have $G(t, 0) = \langle 1, 1, \underbrace{2, \ldots, 2}_{t-1}\rangle$, so if for some j there exists t such that $\langle G(t, 0)\rangle_j = 0$, then $t \geq |\tau^{(1)}|$, hence $\langle G(t, 0)\rangle_j = C_H(z_j)$, i.e. $C_H(z_j) = 0$ and $z_j \in H$.

Now let for some $n > 0$ and some j there exists t_0 such that $\langle G(t_0, n)\rangle_j = 0$. Clearly $j \leq t_0$. We may assume that for every $k < n$ and every t, $\langle G(t, k)\rangle_j \neq 0$ since otherwise we can apply the induction hypothesis for that k.

If for every $t > t_0$ $In(t_0, G(t, n)) = G(t_0, n)$, then for $t > |\tau^{(2n+1)}|$ we will have according to Lemma 4.4 that

$$\langle G(t_0, n)\rangle_j = \langle In(t_0, G(t, n))\rangle_j = \langle G(t, n)\rangle_j = C_H(z_j) = 0,$$

hence $z_j \in H$.

Now assume that there exists $t' \geq t_0$ with $In(t_0, G(t'+1, n)) \neq G(t_0, n)$ and suppose that t' is the first one with that property. Clearly, $j > g(t_0, n)$ — otherwise we will have that $\langle G(g(t_0, n-1), n-1)\rangle_j = 0$, which contradicts the choice of n. So the fact that $G(t_0, n) \neq In(t_0, G(t'+1, n))$ is not due to an assignment of the type $G(t+1, n) := \langle L^l(G(t, n)), \underbrace{0, \ldots, 0}_{l+1}\rangle$ at step $t = t'$, since at the preceding step $t'-1$ we would have $\langle G(t', n)\rangle_j = 2$ (if $|\tau^{(2n+1)}| - |\tau^{(2n)}| > 2$) or $\langle G(t', n)\rangle_j = 1$ (if $|\tau^{(2n+1)}| - |\tau^{(2n)}| = 2$).

Therefore the change of $G(t'+1, n)$ is caused by an assignment of the type $G(t+1, n) := G(t+1, n_0)$ for some $n_0 < n$. It is easy to see that this is preceded by an operator of the type $G(t+1, n_0) := \langle L^l(G(t, n_0)), \underbrace{0, \ldots, 0}_{l+1}\rangle$ at the same step $t = t'$, where $l = t - g(t, n_0)$. In other words,

$$G(t'+1, n_0) = \langle L^l(G(t', n_0)), \underbrace{0, \ldots, 0}_{l+1}\rangle$$

for $l = t' - g(t', n_0)$. From here, for $g(t', n_0) < i \leq t'+1$ we have $\langle G(t'+1, n_0)\rangle_i = 0$. We may claim that $g(t', n_0) = g(t_0, n_0)$ — convince yourselves that any change of $g(t, n_0)$ for $t_0 \leq t < t'$ will produce changes in $In(t_0, G(t, n))$ and take into consideration that t' is the first one with that property.

Further, we have $g(t_0, n_0) < g(t_0, n)$, since $n_0 < n$. So

$$g(t', n_0) = g(t_0, n_0) < g(t_0, n) < j$$

and, obviously, $j \leq t_0 \leq t'$. Hence $\langle G(t'+1, n_0)\rangle_j = 0$ and using the induction hypothesis $T(n_0)$ we conclude that $z_j \in H$. □

Acknowledgements. The author thanks I. Soskov for his help in preparing the LaTeX-appearance of the present paper.

REFERENCES

1. Friedman, H. Algorithmic procedures, generalized turing algorithms and elementary recursion theory. In: *Logic Colloquium'69, Amsterdam*, eds. R. O. Gandy and C. E. M. Yates, North-Holland, 1971, 361–389.

2. Shepherdson, J. C. Computation over abstract structures. In: *Logic Colloquium'73, Amsterdam*, eds. H. E. Rose and J. C. Shepherdson, North-Holland, 1975, 445–513.
3. Soskov, I. N. Computability by means of effectively definable schemes and definability via enumerations. *Arch. Math. Logic*, **29**, 1990, 187–200.
4. Soskova, A. A. and I. N. Soskov. Admissibility in Σ^0_n-enumerations. *Ann. Sof. Univ.*, **90**, 1, 1996, 77–90.

Received March 22, 1999

Faculty of Mathematics and Informatics
"St. Kliment Ohridski" University of Sofia
5 James Bourchier Blvd.
BG-1164 Sofia, Bulgaria
E-mail: stenik@@fmi.uni-sofia.bg

ГОДИШНИК НА СОФИЙСКИЯ УНИВЕРСИТЕТ „СВ. КЛИМЕНТ ОХРИДСКИ“

ФАКУЛТЕТ ПО МАТЕМАТИКА И ИНФОРМАТИКА
Книга 1 — Математика и механика
Том 92, 1998

ANNUAIRE DE L'UNIVERSITE DE SOFIA „ST. KLIMENT OHRIDSKI“

FACULTE DE MATHEMATIQUES ET INFORMATIQUE
Livre 1 — Mathématiques et Mecanique
Tome 92, 1998

ON THE SUBPLANES OF THE HUGHES PLANES OF ODD SQUARE PRIME ORDER

ASSIA ROUSSEVA

In this paper we give a construction of a family of Baer subplanes of the Hughes plane **H** of odd square prime order q^2, $q \geq 5$, which are not isomorphic to its well-known desarguesian Baer subplane $\mathbf{H}_0$ [1, 5.4].

Keywords: finite geometries, Hughes planes, Baer subplanes

1991/95 Math. Subject Classification: 51E15

1. DEFINITION AND MAIN PROPERTIES OF THE HUGHES PLANES

We recall the well-known properties of the Hughes plane H over an arbitrary regular nearfield R of odd square prime power order q^2 [1]:

a. H is a projective plane of odd square prime power order q^2, and it is of Lenz–Barlotti type I.1 [1, 5.4].

b. Let $P := \{\mathrm{P} = (x_1, x_2, x_3)R,\ x_i \in Z(R),\ i = 1,2,3\}$; $T := \{$all the rest points$\}$; $L(1) := \{L_1^{\gamma^i},\ i = 0,1,2,\ldots,r-1\}$; $L(t) := \{L_t^{\gamma^i},\ t \in R \setminus Z(R),\ i = 0,1,2,\ldots,r-1\}$.

Then the points in the set P together with the lines joining them (that are the lines of the set $L(1)$) form a desarguesian Baer subplane H_0 of H.

c. The projective group of H_0 is faightfully induced by the collineation group $\Gamma = \mathrm{GL}_3(q)$ ($\mathrm{GL}_3(q)$ is the group of nonsingular (3,3)-matrices with elements in

$Z(R) = \mathrm{GF}(q)$). Every central collineation of H_0 extends to a central collineation of H in Γ.

d. The full collineation group G of H has two points — P and T, two lines — $L(1)$ and $L(t)$, and two flag orbits. Also, G is a semidirect product $\Gamma.\mathrm{Aut}\,R$; this product is direct iff q is prime [1, 5.4].

Further we consider the case when q is an odd prime ≥ 5 and we use the notation $\mathrm{F} = \mathrm{GF}(q)$. Let $f = z^2 - \alpha$ be an irreducible polynomial over F. Then one can describe the quadratic extension $\Phi = \mathrm{GF}(q^2)$ of F as follows: $\Phi = \{az+b,\ a, b \in \mathrm{F},\ z$ — a root of $f\}$.

Let θ be a primitive element of Φ. Then $\Phi^* = \Phi \setminus \{0\} = \Phi_S \cup \Phi_N$, where Φ_S (Φ_N) is the set of squares (nonsquares) in Φ, i.e. $\Phi_S = \{u : u \in \Phi^*,\ u = \theta^{2k},\ k \in \mathbb{N}\}$ and $\Phi_N = \{u : u \in \Phi^*,\ u = \theta^{2l+1},\ l \in \mathbb{N}\}$.

Let K be the regular nearfield of order q^2 with the same elements as Φ, in which the addition is the same as in Φ, while the multiplication, denoted by $\circ$, is defined as follows: $w \circ u = uw$ if $u \in \Phi_S$ and $w \circ u = uw^q$ if $u \in \Phi_N$ [2].

Since $\alpha^q = \alpha$ for each $\alpha \in F$, we have $\alpha \circ u = \alpha u$ when $\alpha \in F$ and $u \in \mathrm{K}$. It is known that the centre Z(K) is just the field F.

Let $\mathrm{H} = \mathrm{H(K)}$ be the Hughes plane over K. The points of H are all ordered triples $\mathrm{P} = (x_1, x_2, x_3) = (x_1, x_2, x_3) \circ k = (x_1 \circ k, x_2 \circ k, x_3 \circ k)$, $k \in \mathrm{K}^* = \mathrm{K} \setminus (0)$, $x_i \in \mathrm{K}$, $i = 1, 2, 3$, and $(x_1, x_2, x_3) \neq (0, 0, 0)$.

The theorem of Singer [3] gives us the existence of a transformation

$$(x_1, x_2, x_3) \mapsto \Big(\sum_{j=1}^{3} a_{1j}x_j, \sum_{j=1}^{3} a_{2j}x_j, \sum_{j=1}^{3} a_{3j}x_j\Big), \quad a_{ij} \in F,$$

such that the mapping $(x_1, x_2, x_3) = \mathrm{P} \mapsto \mathrm{AP} = (a_{11}x_1, \ldots, a_{33}x_3)$ is a collineation γ of order $r = q^2 + q + 1$ of the desarguesian plane $\pi(q)$ of order q over the field F.

The basic lines L of **H** are defined by the equations

$$L_t : x_1 + t \circ x_2 + x_3 = 0, \quad t \in \{\Phi \setminus F\} \cup \{1\}. \tag{1}$$

The point $\mathrm{P} = (x_1, x_2, x_3) \circ k$ is incident with the line L_t iff the triple (x_1, x_2, x_3) is a solution of (1). The remaining lines of **H** are $\mathrm{L}_t^{\gamma^i}$, $i = 0, 1, 2, \ldots, r-1$, and $\mathrm{A}^i\mathrm{P}$ is incident with $\mathrm{L}_t^{\gamma^i}$, $i = 0, 1, 2, \ldots, r-1$, iff P is incident with L_t [2].

2. AUTOMORPHISMS OF THE NEARFIELD $\mathrm{K} = \mathrm{K}(q^2)$

It is quite evident that the automorphism group Aut K is isomorphic to $\mathbb{Z}_2$. Indeed, on the one hand, $|\mathrm{Aut\,K}| \leq 2$ [1, 5.2.2]. On the other hand, it is easy to check that the mapping $\sigma : \mathrm{K} \to \mathrm{K}$, defined by the correspondance $az + b \mapsto (az+b)^\sigma = -az + b$, is a nontrivial automorphism of K, usually called conjugation in K.

3. THE MAIN RESULT

Let $\chi = \chi(S, L)$ be an arbitrary homology of order two of $\mathbf{H}$ with centre S and axis L (since $o(\chi) = 2$ and q is odd, S is not incident with L). Actually, $S \in P$ and $L \in L(1)$.

From **1.c**, **d** and **2** it follows that there exists a collineation $\psi = \sigma\chi$, where $\sigma \in \mathrm{Aut}\, K$, $\sigma \neq \mathrm{id}$. That gives us a reason to investigate the geometry of the ψ-invariant points and lines of $\mathbf{H}$. The main result in this paper is the following statement:

Theorem. *The ψ-invariant points and lines of $\mathbf{H}$ form a Baer subplane $\pi = \pi(S, L)$ of order q of $\mathbf{H}$, non-isomorphic to the subplane $\mathbf{H}_0$.*

Since the group Aut K is flag-transitive, it is sufficient to prove the theorem in the case when the homology χ has as an axis the line $L_1 : x_1 + x_2 + x_3 = 0$ and as a centre an arbitrary point $S \in P$ which is not incident with L_1.

Let χ^* be an arbitrary homology of $\mathbf{H}$ with centre $S = (a, b, c)$ $(a, b, c \in F)$ and axis L_1. Then the action of χ^* over the points of $\mathbf{H}$ can be presented as follows:

$$\chi^* : \bar{x}' \circ k^* = \mathbf{A}^* \bar{x}, \quad k^* \in K \setminus \{0\},$$

where the column vector $\bar{x} = (x_1, x_2, x_3)^t$ is an arbitrary point of $\mathbf{H}$, $\bar{x}' = (x_1', x_2', x_3')^t$ is its image under χ^*, and the matrix $\mathbf{A}^* \in \mathrm{GL}_3(q)$ has the form

$$\mathbf{A}^* = \begin{pmatrix} a+\rho & a & a \\ b & b+\rho & b \\ c & c & c+\rho \end{pmatrix},$$

where $\rho \in F$. Let us point out that $a + b + c \neq 0$ since S is not incident with L_1.

We denote by 2 the element $1 + 1$, where 1 is the unit element of the multiplication in F (and, respectively, in K). Since the characteristic of F is odd, $2 \neq 0$. Then the homology $\chi = (S, L_1)$ of order two with centre $S = (a, b, c)$ and axis the line L_1 is defined by the matrix

$$\mathbf{A} = \begin{pmatrix} 2^{q-2}(a-b-c) & a & a \\ b & 2^{q-2}(b-a-c) & b \\ c & c & 2^{q-2}(c-a-b) \end{pmatrix}.$$

If the collineation ψ fixes some quadrangle (four points, no three of which are incident with one and the same line) pointwisely, then ψ maps to itself some proper subplane of $\mathbf{H}$. We will show that there exist four points, no three of which are incident with one and the same line, and which are invariant with respect to ψ.

The ψ-invariant points and lines are exactly these, which the homology χ maps onto their conjugated ones, respectively. With the line L_1 are incident precisely $q+1$ different points of the orbit P. Obviously, these points together with the centre of ψ — the point $S = (a, b, c)$ — are invariant with respect to the collineation ψ. If the ψ-invariant points and lines of $\mathbf{H}$ form a Baer subplane, then every point of the supposed subplane will be incident with exactly $q + 1$ different ψ-invariant lines. Since all basic lines $L_t(1)$ are incident with the point $P_1 = (-1, 0, 1)$, one can expect that there exist $q + 1$ ψ-invariant ones among them. That is why we have

to find the number of lines $L_t(1)$, which are mapped under the homology of order two $\chi = \chi(S, L_1)$ in their conjugated lines $L_t\sigma = (L_t)^\sigma$, respectively.

Let $L_t \neq L_1$, $L_t : x_1 + t \circ x_2 + x_3 = 0$, $t \in K \setminus F$. Since the line L_t is different from the line $y = 0$, an arbitrary point T, not incident with L_t, $T \neq P_1$, has coordinates $(x_1, 1, -(x_1 + t))$, $x_1 \in K$. Then $\chi(T) = T'$, where $T' = (2^{q-2}(a - b - c)x_1 + a - a(x_1 + t), 2^{q-2}(b - a - c) - bt, cx_1 + c - 2^{q-2}(c - a - b)(x_1 + t))$. Therefore $\chi(L_t) = L'_t$, $L'_t = P_1T'$ and the element $t' \in K \setminus F$ is uniquely determined by the equation

$$t' \circ (m + bt) = 2m + b - mt, \tag{2}$$

where $m = 2^{q-2}(a - b + c)$. Hence $m \in F$ and m depends on the coordinates of the centre $S = (a, b, c)$ only.

Now we will find the number of distinct solutions of the equation (2) when $t' = t^\sigma$. It is conveniant to consider two cases with respect to b: $b = 0$ and $b \neq 0$.

Case **1**. Let $b = 0$, i.e. $S = (a, 0, c)$. In this case the equation (2) is of the form

$$mt' = m(2 - t). \tag{3}$$

Here $m = 2^{q-2}(a + c)$ and since S is not incident with L_1, $a + c \neq 0$, i.e. $m \neq 0$. Therefore $m^{-1} \in F^*$ and (3) yields that

$$t' = 2 - t. \tag{3'}$$

In general, each $t \in K \setminus F$ is of the form $t = dz + e$, $d \in F^*$, $e \in F$. As it was mentioned above, we are looking for the solutions of (3') when $t' = t^\sigma$, i.e. when $t' = -dz + e$. Then from (3') we obtain that

$$-dz + e = -dz + 2 - e. \tag{3''}$$

Hence d is an arbitrary element of F^* and $e = 1$. These solutions give us exactly $q - 1$ distinct L_t-lines ($t = dz + 1$, $d \in F^*$), which are mapped under the homology χ in their conjugated ones, respectively.

Case **2**. Let $b \neq 0$, i.e. $S = (a, 1, c)$. Now from (2) we have the following equation when $t^\sigma = t'$, namely

$$t^\sigma \circ (m + t) = 2m + 1 - mt. \tag{4}$$

Here $m = 2^{q-2}(a + c - 1)$ and since $a + 1 + c \neq 0$, $m \neq -1$. As t has the representation $t = dz + e$, so that $t^\sigma = -dz + e$, $d \in F^*$, $e \in F$, the relation (4) immediately gives

$$(-dz + e) \circ (dz + m + e) = -mdz + 2m - me + 1. \tag{4'}$$

For the result of multiplication in the left-hand side of (4') we have two possibilities, namely,

$$(-dz + e) \circ (dz + m + e) = \begin{cases} (-dz + e)(dz + m + e) & \text{if } dz + m + e \in \Phi_S, \\ (dz + e)(dz + m + e) & \text{if } dz + m + e \in \Phi_N. \end{cases}$$

If we assume that $dz + m + e \in \Phi_N$, then for d and e we have from (4') that $(m + 2e)dz + d^2z^2 + e(m + e) = -mdz + 2m + 1 - me$. Since $z^2 = \alpha$, we obtain

$$(m + 2e)dz + \alpha d^2 + e(m + e) = -mdz + 2m + 1 - me. \tag{4''}$$

Hence $e = -m$, i.e. $e \neq 1$, and $d^2\alpha = (e-1)^2$, i.e. $d \neq 0$. Therefore $\alpha = [d^{-1}(e-1)]^2$, i.e. α is a square in the field F, but this contradicts the choice of α. Therefore, if there exists a solution of the equation (4'), then $dz + m + e \in \Phi_S$.

Suppose that $dz + m + e \in \Phi_S$. Now (4') is reduced to

$$\alpha d^2 = e^2 + 2me - 2m - 1. \qquad (4''')$$

We transform the right-hand side of (4''') as follows: $e^2 + 2me - 2m - 1 = e^2 + 2me + m^2 - m^2 - 2m - 1 = (e+m)^2 - (m+1)^2$, and then (4''') becomes

$$(-\alpha)d^2 + (e+m)^2 = (m+1)^2, \qquad (5)$$

where $m \neq -1$ is a fixed element of F and $\alpha = z^2$.

Let η be a quadratic character of GF(q) (q — odd), i.e. $\eta(c) = 1$ if c is a square in GF(q) and $\eta(c) = -1$ if c is a nonsquare in GF(q). Define the function v on GF(q) by $v(b) = -1$ if $b \in \mathrm{GF}^*(q)$ and $v(0) = q - 1$. Let $N(\alpha_1 y_1^2 + \alpha_2 y_2^2 = b)$ ($b \in \mathrm{GF}(q)$, $\alpha_1, \alpha_2 \in \mathrm{GF}^*(q)$) be the number of the solutions of the equation $\alpha_1 y_1^2 + \alpha_2 y_2^2 = b$ in the field GF(q). Then [4, 6.24]

$$N(\alpha_1 y_1^2 + \alpha_2 y_2^2 = b) = q + v(b)\eta(-\alpha_1\alpha_2).$$

In the case of the equation (5) in the variables $y_1 = d$ and $y_2 = e + m$ we have $\alpha_1 = -\alpha$, $\alpha_2 = 1$ and $b = (m+1)^2$. Since $m \neq -1$ and α is a nonsquare in GF(q), $v((m+1)^2) = -1$ and $\eta(-\alpha_1\alpha_2) = \eta(\alpha) = -1$. Therefore $N((-\alpha)d^2 + (e+m)^2 = (m+1)^2) = q + 1$.

The solution $(0, 1+m)$ of the equation (5) gives the line L_1 and to the solution $(0, -2m-1) = (0, -(a+c))$ corresponds the line $\mathrm{SP}_1 : x_1 - (a+c)x_2 + x_3 = 0$, $\mathrm{SP}_1 \in L(1)$.

For each of the remaining $q-1$ solutions $(d, e+m)$ of the equation (5) we will prove that $dz + (m+e) \in \Phi_S$. Suppose that (d_1, e_1+m) is a solution of (5) such that $u = d_1 z + (m+e_1) \in \Phi_N$, i.e. $u = \theta^{2l+1}$. Then $u \circ u = uu^q = (-\alpha)d_1^2 + (e_1+m)^2 = (m+1)^2$, i.e. $\theta^{(q+1)(2l+1)} = (m+1)^2$, and therefore $\theta^{[(q^2-1)/2](2l+1)} = (m+1)^{q-1} = 1$. But, on the other hand, $\theta^{[(q^2-1)/2](2l+1)} = (-1)^{2l+1} = -1$. It turns out that $1 = -1$, which contradicts the oddness of the characteristic of the field $\mathrm{F} = \mathrm{GF}(q)$. Therefore every solution $(d, e+m)$ of the equation (5) with $d \neq 0$ gives an element $w = dz + (m+e) \in \Phi_S$. That means there exist exactly $q-1$ distinct basic lines L_t ($\neq \mathrm{L}_1$) invariant with respect to the collineation ψ.

Now it is easy to find ψ-invariant points, no three of which are incident with one and the same line. Let $\mathrm{P}_i \in P$, $i = 1, 2, 3$, be three different points which are incident with the axis L_1 of the homology of order two $\chi = \chi(\mathrm{S}, \mathrm{L}_1)$ and differ from the point $(-1, 0, 1)$. Then the points S, P_1, P_2, $\mathrm{T} = \mathrm{SP}_3 \cap \mathrm{L}_t$ have the desired property, L_t is an arbitrary invariant with respect to the ψ basic line different from L_1 and $\mathrm{S} = (a, b, c)$ is the centre of χ.

Hence for any point $\mathrm{S} \in P$ and any line $\mathrm{L} \in L(1)$, S non-incident with L, the points and lines of $\mathbf{H}$ invariant with respect to the collineation $\psi = \sigma\chi$ form a Baer subplane $\pi = \pi(\chi)$ of $\mathbf{H}$ of order q (χ is the homology of order two with centre S and axis L, $\sigma \in \mathrm{Aut\,K}$, $\sigma \neq \mathrm{id}$). It is clear that this subplane $\pi = \pi(\chi)$ is not isomorphic to the well-known Baer subplane $\mathbf{H}_0$ of $\mathbf{H}$ with respect to the group

Aut $\mathbf{H}$. Otherwise there exists an element $\phi \in \mathrm{Aut}\,\mathbf{H}$ such that $\phi(\mathrm{P}_0) = \mathrm{T}_0$, where P_0 and T_0 are arbitrary points from the orbits P and T, respectively. Hence Aut $\mathbf{H}$ acts transitively over the set of all points of $\mathbf{H}$, which is inadmissible in any Hughes plane.

It is naturaly to ask whether the subplanes of the kind $\pi = \pi(\chi)$ are isomorphic to each other with respect to the automorphism group of the plane $\mathbf{H}$. We claim that the answer of this question is positive.

Let S_1 and S_2 be arbitrary points of P and L^1, L^2 be lines of $L(1)$, S_i be non-incident with L^i $(i = 1, 2)$. Denote by χ_i the homology of order two with centre S_i and axis L^i $(i = 1, 2)$. Let $\pi_i = \pi_i(\chi_i)$ be the subplane generated by the collineation $\psi_i = \sigma\chi_i$ $(i = 1, 2)$. If there exists an element $\phi \in \mathrm{Aut}\,\mathbf{H}$ such that $\phi(\mathrm{S}_1) = \mathrm{S}_2$ and $\phi(\mathrm{L}^1) = L^2$, then $\phi(\pi_1) = \pi_2$.

Due to the fact that the group Aut $\mathbf{H}$ is flag-transitive, it is sufficient to consider only the case when the axis of χ_1 and χ_2 is the line $\mathrm{L}_1 : x_1 + x_2 + x_3 = 0$, the centre S_1 of χ_1 has coordinates $(1, 0, 0)$, and the centre S_2 of χ_2 is an arbitrary point from the orbit P non-incident with L_1. Then S_2 has coordinates (a, b, c), $a, b, c \in \mathrm{F}$ and $a + b + c \neq 0$.

In order to prove that the subplanes $\pi_1 = \pi(\chi_1)$ and $\pi_2 = \pi(\chi_2)$ are isomorphic, it is sufficient to show that there exists an automorphism $\phi \in \mathrm{Aut}\,\mathbf{H}$ which maps certain quadrangle of π_1 into a quadrangle of π_2.

In the case when $\mathrm{S}_2 = (a, 0, b)$, the points S_1, S_2 and $\mathrm{P}_1 = (-1, 0, 1)$ are incident with the line $x_2 = 0$. Then the isomorphism between the subplanes π_1 and π_2 is realized by the elation $\varepsilon_1 = \varepsilon_1(\mathrm{P}_1, \mathrm{L}_1)$ with centre $\mathrm{P}_1 = (-1, 0, 1)$, axis L_1 and $\varepsilon(\mathrm{S}_1) = \mathrm{S}_2$. It is obvious that in this case both subplanes π_1 and π_2 contain the lines of the form $\mathrm{L}_t : x_1 + t \circ x_2 + x_3 = 0$, where $t = dz + 1$, $d \in \mathrm{F}^*$.

Let the points $\mathrm{P}_2 \neq \mathrm{P}_3$ be in the orbit P, P_2 and P_3 be incident with L_1, and suppose that $\mathrm{P}_2 \neq \mathrm{P}_1$ and $\mathrm{P}_3 \neq \mathrm{P}_1$. The line $\mathrm{S}_1\mathrm{P}_2$ intersects an arbitrary line $\mathrm{L}_{t_1} \in \pi_1$, $t_1 = d_1 z + 1$, $d_1 \in \mathrm{F}^*$ at a point $\mathrm{T}_1 \in \pi_1$, and the line $\mathrm{S}_2\mathrm{P}_2$ intersects the same line $\mathrm{L}_{t_1} \in \pi_2$ at a point $\mathrm{T}_2 \in \pi_2$. Then the points S_1, P_2, P_3, T_1 form a quadrangle in π_1 and the points S_2, P_2, P_3, T_2 — quadrangle in π_2. Since the point P_1 is incident with all the lines L_t, $\varepsilon_1(\mathrm{L}_t) = \mathrm{L}_t$, hence $\varepsilon_1(\mathrm{S}_1, \mathrm{P}_2, \mathrm{P}_3, \mathrm{T}_1) = (\mathrm{S}_2, \mathrm{P}_2, \mathrm{P}_3, \mathrm{T}_2)$, which gives us that $\varepsilon_(\pi_1) = \pi_2$.

Let $\mathrm{S}_2 = (a, 1, c)$. Since S_2 is non-incident with L_1, $a + c + 1 \neq 0$ and the line $\mathrm{S}_1\mathrm{S}_2$ intersects the line L_1 at the point $\mathrm{P}_12 = (-1 - c, 1, c)$. Similarly, the isomorphism between the subplanes π_1 and π_2 is realized by the elation $\varepsilon_2 = \varepsilon_2(\mathrm{P}_12, \mathrm{L}_1)$ with centre P_12, axis L_1 and $\varepsilon_2(\mathrm{S}_1) = \mathrm{S}_2$. This elation is given by the matrix

$$\mathbf{B} = \begin{pmatrix} a & -1-c & -1-c \\ 1 & a+c+2 & 1 \\ c & c & a+2c+1 \end{pmatrix}.$$

Actually, we have that each point $\mathrm{G}_1 \in \pi_1$ which is incident with the line $\mathrm{S}_1\mathrm{P}_1$ is mapped under the elation ε_2 into a point $\mathrm{G}_2 \in \pi_2$ which is incident with the line $\mathrm{S}_2\mathrm{P}_1$.

The homology of order two $\chi_1 = \chi_1(S_1, L_1)$ with centre $S_1 = (1,0,0)$ is given by the matrix

$$\mathbf{C} = \begin{pmatrix} 2^{q-2} & 1 & 1 \\ 0 & -2^{q-2} & 0 \\ 0 & 0 & -2^{q-2} \end{pmatrix}.$$

The line $S_1P_1 \in L(1)$ has an equation $x_2 = 0$ and therefore it differs from the line $x_3 = 0$. Then each point $G \neq S_1$ which is incident with the line S_1P_2 has coordinates $(dz + e, 0, 1)$, $d, e \in F$. The point G belongs to the subplane π_1 iff $\chi_1(G) = \sigma(G)$. Hence the points G_1 of π_1, incident with the line S_1P_1, have coordinates $(dz - 1, 0, 1)$, $d \in F$ (when $d = 0$, $G_1 = P_1$).

The homology of order two $\chi_2 = \chi_2(S_2, L_1)$ with centre $S_2 = (a, 1, c)$ and axis L_1 is given by the matrix

$$\mathbf{D} = \begin{pmatrix} 2^{q-2}(a-c-1) & a & a \\ 1 & 2^{q-2}(-1-a-c) & 1 \\ c & c & 2^{q-2}(c-a-1) \end{pmatrix}.$$

The equation of the line $S_2P_2 \in L(1)$ is $x_1 - (a+c)x_2 - x_3 = 0$ and it is easy to see that the points $G_2 \in \pi_2$, which are incident with S_2P_2, have coordinates $(dz + a, 1, -dz + c)$, $d \in F$.

We have $\varepsilon(G_1) = G'$, where the coordinates of the point G' are $(adz - (a+c+1), dz, cdz + (a+c+1))$, $a + c + 1 \neq 0$. If $d \neq 0$, then the point G' has coordinates $(adz - (a+c+1), dz, cdz + (a+c+1)) \circ (dz)^{-1}$, that is $G' = (-(a+c+1)(dz)^{-1} + a, 1, (a+c+1)(dz)^{-1} + c)$. Since $(dz)^{-1} = \bar{d}z$, $G' = (d^* + a, 1, -d^* + c)$, where $d^* = -(a+c+1)\bar{d}$, $d^* \in F^*$.

If $d = 0$, then $G_1 = P_1 = G' = (-(a+c+1), 0, (a+c+1))$. Hence G' is incident with S_2P_1 and $G' \in \pi_2$.

Let P be an arbitrary point in the orbit P, let P be incident with the line L_1 and $P_1 \neq P \neq P_12$. That is why the point P belongs to the subplane π_1 as well as to the subplane π_2. Then the elation $\varepsilon_2 = \varepsilon_2(P_12, L_1)$ maps the quadrangle (P_12, P, S_1, G_1) of π_1 onto the quadrangle $(P_12, P, S_2, G_2 = G')$ of π_2 ($S_1 \neq G_1 \neq P_12$, $S_2 \neq G_2 \neq P_12$).

In this way we have proved that the subplanes π_1 and π_2 are isomorphic with respect to the automorphism group of **H**.

Acknowledgements. The author is thankful to Prof. Ch. Lozanov for his useful suggestions and attention shown to this paper.

REFERENCES

1. Dembowski, P. Finite Geometries. Springer-Verlag, 1968.
2. Hall, M. The Theory of Groups. The Macmillan Company, New York, 1959.
3. Singer, I. A Theorem in Finite Projective Geometry and Some Applications to Number Theory. *Trans. Amer. Math. Soc.*, **43**, 1938, 377–385.

4. Lidl, R., H. Niederreiter. Finite fields. Addison — Wesley Publishing Company, Advance Book Reprogram/World Science Division Reading, 1983.

Received March 26, 1999

Faculty of Mathematics and Informatics
"St. Kliment Ohridski" University of Sofia
5 James Bourchier Blvd.
BG-1164 Sofia, Bulgaria

ГОДИШНИК

НА

СОФИЙСКИЯ УНИВЕРСИТЕТ „СВ. КЛИМЕНТ ОХРИДСКИ“

ФАКУЛТЕТ ПО МАТЕМАТИКА И ИНФОРМАТИКА

Книга 2 — ПРИЛОЖНА МАТЕМАТИКА И ИНФОРМАТИКА

Том 92
1998

ANNUAIRE

DE

L'UNIVERSITE DE SOFIA "ST. KLIMENT OHRIDSKI"

FACULTE DE MATHEMATIQUES ET INFORMATIQUE

Livre 2 — MATHEMATIQUES APPLIQUEE ET INFORMATIQUE

Tome 92
1998

СОФИЯ • 2000 • SOFIA
УНИВЕРСИТЕТСКО ИЗДАТЕЛСТВО „СВ. КЛИМЕНТ ОХРИДСКИ“
PRESSES UNIVERSITAIRES "ST. KLIMENT OHRIDSKI"

ГОДИШНИК НА СОФИЙСКИЯ УНИВЕРСИТЕТ „СВ. КЛИМЕНТ ОХРИДСКИ“

ФАКУЛТЕТ ПО МАТЕМАТИКА И ИНФОРМАТИКА
Книга 2 — Приложна математика и информатика
Том 92, 1998

ANNUAIRE DE L'UNIVERSITE DE SOFIA „ST. KLIMENT OHRIDSKI“

FACULTE DE MATHEMATIQUES ET INFORMATIQUE
Livre 2 — Mathématiques Appliquée et Informatique
Tome 92, 1998

THE *SLRS* SYNCHRONOUS IMPERATIVE PROGRAMMING LANGUAGE

MAGDALINA V. TODOROVA

This paper describes the synchronous imperative programming language *SLRS*. After a brief overview of the language we define its behavioural semantics.

Keywords: synchronous language, reactive system, real time process

1991/95 Math. Subject Classification: main 68N15, secondary 68Q15

1. INTRODUCTION

Reactive systems are programs whose main role is to maintain an ongoing interaction with their environment, rather than to produce some final result on termination. Such systems should be specified and analysed in terms of their behaviour, i.e. the sequences of states or events they generate during their operation. A reactive program may be treated as a generator of computations which, for simplicity, we may assume to be infinite sequences of states or events [1]. Typical examples of reactive systems are real time process controllers, signal processing units, digital watches and video games. Operating system drivers and mouse interface drivers are examples of reactive programs too. Lustre [4], Esterel [2, 3], Signal [5] are programming languages devoted to program reactive systems.

Determinism is an important characteristic of reactive programs. A deterministic reactive program produces identical output sequences when fed with identical input.

In this paper a synchronous imperative programming language named *SLRS* (Synchronous Language to Reactive Systems) is considered. It is based on the synchrony hypothesis: each reaction is assumed to be instantaneous and therefore atomic in any possible sense. Control transmission, signal broadcasting, and elementary computations are supposed to take no time, making the outputs of a system perfectly synchronous with its inputs [2]. After a brief overview of the *Pure SLRS* we define its behavioural semantics.

2. THE *PURE SLRS* LANGUAGE

In this section we describe the *Pure SLRS* language intuitively and by examples.

A *SLRS* program:

program P;
    declaration part
    interface part
    body
end P.

has a *declaration part* that declares the external objects used by the program, an *interface part* that defines its input and output, and a *body* that is an executable statement.

Declaration part. Data declarations declare the constants, types, functions, and procedures that manipulate data. They are written in the host language (Pascal or C).

Interface part. The interface part

input I1 {, In};
output O1 {, On};
input relations;

defines program's input $I_1, \ldots, I_n$ and output $O_1, \ldots, O_n$ signals.

The basic object of the language is a *signal.* Signals are used for communication with the environment as well as for internal broadcast communication. There is a special signal called *tic.* It is assumed to be always present. In *Pure SLRS* there are only two kinds of interface signals: input and output signals.

Input signals come from the environment. *They cannot be produced internally.* They are declared in the form

input I1 {, In};

Output signals are directed towards the environment of the program by the *produce* statement. An output signal declaration has the form

output O1 {, On};

Input relations are assertions that can be used to restrict input events. That is very important for program specification and verification.

A *SLRS* program specifies a relation between input and output signals. It is activated by repeatedly giving it *input events*. These events consist of a possibly empty set of input signals assumed to be present. For each input event, the program reacts by executing its body and by outputting the produced output signals that form the *output event*. We assume that the reaction is perfectly synchronous and deterministic. A reaction is also called an *instant*.

The **kernel statements** in the language are:

- *Statement skip*:

  skip

It performs no action and terminates immediately.

- *Statement stop*:

  stop

It performs no action and never terminates.

- *Statement produce*:

  produce S

where S is a signal. It emits S and terminates immediately.

- *Statement sequence*:

  sequence stat_1, stat_2 end

where $stat_1$ and $stat_2$ are any statements. The statement $stat_2$ starts instantly when the statement $stat_1$ terminates. The sequencing operator takes no time by itself.

- *Statement parallel*:

  parallel stat_1, stat_2 end

where $stat_1$ and $stat_2$ are any statements. The statements $stat_1$ and $stat_2$ are started simultaneously when the parallel statement is started. The *parallel* statement terminates when its both branches are terminated.

- *Statement ifp-then-else-end*:

  ifp S then stat1 else stat2 end

where S is a signal, $stat_1$ and $stat_2$ are any statements. The *then* and *else* parts are optional. If some of them is omitted, it is supposed to be *skip* statement. The presence of S is tested and the *then* or *else* branch is immediately started accordingly.

- *Statement cycled-end*:

  cycled stat end

where *stat* is any statement. The body *stat* of a *cycled-end* starts immediately when the *cycled-end* statement starts and whenever *stat* terminates, it is instantly restarted. A *cycled-end* never terminates.

- *Statement watching-do*:

  watching S do stat end

where *stat* is any statement and S is a signal. S is called a *guard*. The statement *stat* is executed normally until *stat* terminates or until future occurrence of the signal S. If *stat* terminates just before S occurs or at the same time as S, so does the whole *watching-do* statement and the guard has no action. Otherwise, the occurrence of S provokes immediate preemption of the body *stat* and immediate termination of the whole *watching-do* statement.

Example. Let define

await S $=_{\text{def}}$ watching S do stop end.

When *await* S starts executing, it retains the control until the first future reaction where S is present. If such a reaction exists, the *await* statement terminates immediately. Otherwise it never terminates.

Example. Let us consider the statement

```
watching I1 do
   sequence
      watching I2 do
         sequence
            await I3,
            produce O1
         end
      end,
      produce O2
   end
end
```

If *I1* occurs before *I2* and *I3* or at the same time as them, then the external *watching-do* preempts its body and terminates instantly. In this case no signal is produced. If *I2* occurs before *I3* or at the same time as it, but before *I1*, then the internal *watching* preempts its body, *O1* is not produced even if *I3* is present, *O2* is produced and the external *watching* instantly terminates. If *I3* occurs just before *I1* and *I2*, then the *await* statement terminates, *O1* is produced, the internal *watching-do* terminates since its body terminates, *O2* is produced and the external *watching* also terminates.

- *Statement run-until*:

  run stat until X

where *stat* is any statement and X is a parameter. The body *stat* starts instantly and determines the behaviour of the *run-until* statement until it terminates or executes *exit* X. Then the execution of *stat* is preempted and the whole *run-until* constructor terminates. If body of a *run-until* statement contains parallel components, the *run-until* is exited when one of the components executes an *exit* X, the other component is preempted.

Example. Let consider the statement

```
run
   parallel
      sequence
         await I1,
         produce O
      end,
      sequence
         await I2,
         exit X
      end
   end
until X
```

If *I1* occurs before *I2*, then *O* is produced and *run* waits for *I2* to terminate. If *I2* occurs before *I1*, then the whole statement terminates instantly, the first branch is preempted and *O* will never be produced. If *I1* and *I2* occur simultaneously, then both branches do execute and *O* is produced.

Run-until statement provides a way for breaking loops:

```
run
   cycled ... exit X ... end
until X
```

Notice that the statement

```
run
   sequence
      run
         parallel
            exit X,
            exit Y
         end
      until Y,
      produce O
   end
until X
```

is ambiguous. We must define what it means to exit several *run-until* statements simultaneously.

Priorities between run-until statements — only the outermost *run-until* statement matters, the other ones are discarded.

In the above example the internal *run-until* is discarded and *O* is not produced.

- *Statement local:*

  local S {, S_i} in stat end

where *S* and S_i are signals and *stat* is any statement. It declares a lexically scoped signal *S* {, S_i} that can be used for internal broadcast communication within *stat*.

At each reaction, a signal has a single status — *present* or *absent*. The following law determines the status of local and output signals: *A local or output signal is present in a reaction if and only if it is produced by executing a produce statement in that reaction.* The default status of a signal is to be absent.

3. THE BEHAVIOURAL SEMANTICS OF THE *PURE SLRS*

This semantics defines program execution reaction by reaction using Structural Operational Semantics technique [6]. It defines transitions of the form

$$P \xrightarrow[I,\ O]{} P',$$

where P is a program, I is an input event, O is the corresponding output event, and P' is the new program, i.e. the new state of P after reaction to I. The sequence

$$P \xrightarrow[I_1,\ O_1]{} P_1 \xrightarrow[I_2,\ O_2]{} \cdots P_n \xrightarrow[I_{n+1},\ O_{n+1}]{} \cdots$$

defines the reaction $O_1, O_2, \ldots, O_n, \ldots$ to an input sequence $I_1, I_2, \ldots, I_n, \ldots$ The programs P_i are called *derivations of P*.

The transition

$$P \xrightarrow[I,\ O]{} P'$$

is defined using the following auxiliary relation:

$$stat \xrightarrow[E,\ E',\ t,\ S]{} stat',$$

where *stat* is the body of P, *stat'* is the body of P', E is the current event in which *stat* reacts, E' is the event composed of the signals produced by *stat*, t is an integer ($t \geq 0$) that codes the way in which *stat* terminates or exits, and S is a set of integers. S is called a *stopset* and t — a *termination level*. They are defined below. The current event E is composed of all signals that are present at a given reaction. By the law, which determines the state of local and output signals, E must contain the set E' of produced signals. The auxiliary relation is defined by structural induction on statements by means of inductive rules.

The connection between the transition and the auxiliary relation is as follows:

$$P \xrightarrow[I,\ O]{} P' \text{ if } stat \xrightarrow[I \cup O \cup \{tic\},\ O,\ t,\ S]{} stat'$$

for some t and S.

Termination level. To determine the termination level, it is useful to label the *exit X* part of a *run-until X* statement with the corresponding level $t + 2$, where t ($t \geq 0$) is an integer and is equal to the number of the *run-until* statements which one must traverse to reach the *run-until X* statement [2].

Example.

```
run
   parallel
      exit X : 2,
      run
```

```
            parallel
               exit X : 3,
               exit Y : 2
            end
         until Y
      end
   until X
```

The first *exit X* and the *exit Y* are labelled 2 since there is not intermediate *run-until* statement to traverse, while the second *exit X* is labelled 3 since one must traverse the *run-until Y* statement to reach the *run-until X* statement.

Definition. The *termination level t of a statement stat* is defined as *t(stat)*, where:

$t(\text{skip}) = 0,$
$t(\text{stop}) = 1,$
$t(\text{produce X}) = 0,$
$t(\text{sequence } \text{stat}_1, \text{stat}_2 \text{ end}) = \begin{cases} t(\text{stat}_1) & \text{if } t(\text{stat}_1) > 0, \\ t(\text{stat}_2) & \text{if } t(\text{stat}_1) = 0, \end{cases}$
$t(\text{parallel } \text{stat}_1, \text{stat}_2 \text{ end}) = \max\{t(\text{stat}_1), t(\text{stat}_2)\},$
$t(\text{cycled stat end}) = 1 \text{ if } t(\text{stat}) = 0,$
$t(\text{cycled stat end}) = t(\text{stat}) \text{ if } t(\text{stat}) > 0,$
$t(\text{watching X do stat end}) = t(\text{stat}),$
$t(\text{run stat until X}) = \begin{cases} 0 & \text{if } t(\text{stat}) = 0 \text{ or } t(\text{stat}) = 2, \\ 1 & \text{if } t(\text{stat}) = 1, \\ i - 1 & \text{if } t(\text{stat}) = i,\ i > 2, \end{cases}$
$t(\text{exit X : i}) = i,$
$t(\text{local X in stat end}) = t(\text{stat}).$

The termination level of the statement of the above example is 0.

Stopset. We number all occurrences of the *stop* statement in *stat* by different integers from 0 to n, $n > 0$. A *stopset* S is a subset of $[0..n]$ that satisfies the following condition: If $stat_1$ and $stat_2$ are the two statements of a *sequence* or two branches of an *ifp-then-else-end* statement, then S cannot contain an occurrence of *stop* in $stat_1$ together with an occurrence of *stop* in $stat_2$. Notice that $S = \emptyset$ when $t \neq 1$ and $S \neq \emptyset$ when $t = 1$.

Inductive Rules:

(IR1) skip $\xrightarrow[E,\ \emptyset,\ 0,\ \emptyset]{}$ skip;

(IR2) stop : i $\xrightarrow[E,\ \emptyset,\ 1,\ \{i\}]{}$ stop : i;

(IR3) produce X $\xrightarrow[E,\ \{X\},\ 0,\ \emptyset]{}$ skip;

$$\text{stat}_1 \xrightarrow[E,\ E'_1,\ 0,\ \varnothing]{} \text{stat}'_1$$

and

$$\text{(IR4)}\quad \frac{\text{stat}_2 \xrightarrow[E,\ E'_2,\ t_2,\ S_2]{} \text{stat}'_2}{\text{sequence stat}_1,\ \text{stat}_2\ \text{end} \xrightarrow[E,\ E'_1 \cup E'_2,\ t_2,\ S_2]{} \text{stat}'_2};$$

$$\text{(IR5)}\quad \frac{\text{stat}_1 \xrightarrow[E,\ E'_1,\ t_1,\ S_1]{} \text{stat}'_1,\quad t_1 > 0}{\text{sequence stat}_1,\ \text{stat}_2\ \text{end} \xrightarrow[E,\ E'_1,\ t_1,\ S_1]{} \text{sequence stat}'_1,\ \text{stat}_2\ \text{end}};$$

$$\text{stat}_1 \xrightarrow[E,\ E'_1,\ t_1,\ S_1]{} \text{stat}'_1$$

and

$$\text{(IR6)}\quad \frac{\text{stat}_2 \xrightarrow[E,\ E'_2,\ t_2,\ S_2]{} \text{stat}'_2}{\text{parallel stat}_1,\ \text{stat}_2\ \text{end} \xrightarrow[E,\ E'_1 \cup E'_2,\ \max(t_1,t_2),\ S]{} \text{parallel stat}''_1,\ \text{stat}''_2\ \text{end}},$$

where $S = \begin{cases} S_1 \cup S_2, & \text{if } \max(t_1, t_2) \le 1, \\ \varnothing, & \text{if } \max(t_1, t_2) > 1 \end{cases}$ and $\text{stat}''_i = \begin{cases} \text{stat}'_i, & \text{if } t_i \neq 0, \\ \text{skip}, & \text{if } t_i = 0; \end{cases}$

$$\text{(IR7)}\quad \frac{\text{stat} \xrightarrow[E,\ E',\ t,\ S]{} \text{stat}',\quad t > 0}{\text{cycled stat end} \xrightarrow[E,\ E',\ t,\ S]{} \text{sequence stat}',\ \text{cycled stat end end}};$$

$$\text{(IR8)}\quad \frac{X \in E \text{ and } \text{stat}_1 \xrightarrow[E,\ E'_1,\ t_1,\ S_1]{} \text{stat}'_1}{\text{ifp X then stat}_1\ \text{else stat}_2\ \text{end} \xrightarrow[E,\ E'_1,\ t_1,\ S_1]{} \text{stat}'_1};$$

$$\text{(IR9)}\quad \frac{X \notin E \text{ and } \text{stat}_2 \xrightarrow[E,\ E'_2,\ t_2,\ S_2]{} \text{stat}'_2}{\text{ifp X then stat}_1\ \text{else stat}_2\ \text{end} \xrightarrow[E,\ E'_2,\ t_2,\ S_2]{} \text{stat}'_2};$$

$$\text{(IR10)}\quad \frac{\text{stat} \xrightarrow[E,\ E',\ t,\ S]{} \text{stat}'}{\text{watching X do stat end} \xrightarrow[E,\ E',\ t,\ S]{} \text{ifp X else watching X do stat}' \text{ end end}};$$

$$\text{stat} \xrightarrow[E,\ E',\ t,\ \varnothing]{} \text{stat}'$$

and

$$\text{(IR11)}\quad \frac{t = 0 \text{ or } t = 2}{\text{run stat until X} \xrightarrow[E,\ E',\ 0,\ \varnothing]{} \text{skip}};$$

$$\text{stat} \xrightarrow[E,\ E',\ t,\ S]{} \text{stat}'$$

and

$$\text{(IR12)}\quad \frac{(t = 1 \text{ and } t' = 1) \text{ or } (t > 2 \text{ and } t' = t - 1)}{\text{run stat until X} \xrightarrow[E,\ E',\ t',\ S]{} \text{run stat}' \text{ until X}};$$

$$\text{(IR13)}\quad \text{exit X : i} \xrightarrow[E,\ \varnothing,\ i,\ \varnothing]{} \text{stop};$$

$$\text{(IR14)}\quad \frac{X \notin E' \text{ and } stat \xrightarrow[E,\ E' \cup \{X\},\ t,\ S]{} stat'}{\text{local X in stat end} \xrightarrow[E,\ E',\ t,\ S]{} \text{local X in } stat' \text{ end}};$$

$$\text{(IR15)}\quad \frac{X \notin E' \text{ and } stat \xrightarrow[E-\{X\},\ E',\ t,\ S]{} stat'}{\text{local X in stat end} \xrightarrow[E,\ E',\ t,\ S]{} \text{local X in } stat' \text{ end}}.$$

Definition. A program is *locally correct* if its body and its substatements are such that each local and output signal can have a single status for any input event that satisfies the input relations.

Definition. A program is *correct* if all its derivations are locally correct.

Correctness obviously implies determinism. In the sequel, we will consider a correct program P. For technical reasons (see Theorem 1 below), we assume also that the body of P never terminates, adding a trailing *stop* if it is necessary. This does not change the observable behaviours.

Let *stat* be a statement, S — a stopset, and *stat'* — a derivation of *stat*. We will define term *R(stat:S)* equal to *stat'*, i.e. by means of the operator R we recover the derivation *stat'* from *stat* and S. The argument of the operator R is a term labelled S. A labelled term *stat:S* is obtained by labelling the subterms of *stat* either $S+$, or $S-$. A subterm is labelled $S+$ if and only if it contains at least one occurrence of *stop* which number is in S, otherwise, the subterm is labelled $S-$. The labels are redundant, but they make the proofs simpler to write.

Definition. R(stat:S−) = stat
R(skip:S) = skip
R((stop:i):S) = stop:i
R((produce X):S) = skip
R(sequence $stat_1$:S+, $stat_2$:S− end) = sequence R($stat_1$:S+), $stat_2$ end
R(sequence $stat_1$:S−, $stat_2$:S+ end) = R($stat_2$:S+)
R(parallel $stat_1$:S+, $stat_2$:S+ end)= parallel R($stat_1$:S+), R($stat_2$:S+) end
R(parallel $stat_1$:S+, $stat_2$:S− end) = parallel R($stat_1$:S+), skip end
R(parallel $stat_1$:S−, $stat_2$:S+ end) = parallel skip, R($stat_2$:S+) end
R(ifp X then $stat_1$:S+ else $stat_2$:S− end) = R($stat_1$:S+)
R(ifp X then $stat_1$:S− else $stat_2$:S+ end) = R($stat_2$:S+)
R(cycled stat:S+ end) = sequence R(stat:S+), cycled stat end end
R(watching X do stat:S+ end) = ifp X else watching X do R(stat:S+) end end
R((run stat until X):S) = run R(stat:S) until X
R((local X in stat end):S) = local X in R(stat:S) end.

Theorem 1. Let *stat* be the body of a correct program and *stat* never terminate. Let S be a stopset in *stat*. Then for any transition of the form

$$\text{R(stat:S)} \xrightarrow[E,\ E',\ 1,\ S']{} stat'$$

the stopset S' contains only stops occurring in *stat'* and *stat'* = *R(stat:S')*.

Proof. Let E is a given current event. The proof is by structural induction on *stat*. All cases are similar, so we will consider the *sequence* and the *watching-do* statements as examples.

(i) Let *stat* = *sequence* $stat_1$, $stat_2$ *end*. There are two main subcases:

— If *stat:S* = *stat:S+* = *sequence* $stat_1$*:S−*, $stat_2$*:S+ end*, then *R(stat:S)* = *R(*$stat_2$*:S+)*. By correctness and by the hypothesis that *stat* stops, *R(*$stat_2$*:S+)* has a unique transition

$$\mathrm{R(stat_2{:}S+) = R(stat{:}S)} \xrightarrow[\mathrm{E,\ E',\ 1,\ S'}]{} \mathrm{stat'},$$

where S' is a non-empty stopset that contains only stops in $stat_2$. By induction,

$$\mathrm{stat'} = \mathrm{R(stat_2{:}S')} \tag{1}$$

and S' contains only stops in *stat'*. Since S' is non-empty and is a stopset in $stat_2$,

$$\mathrm{R(stat_2{:}S') = R(sequence\ stat_1{:}S'-,\ stat_2{:}S' + end) = R(stat{:}S').} \tag{2}$$

The result is achieved as a consequence of (1) and (2).

— If *stat:S* = *stat:S+* = *sequence* $stat_1$*:S+*, $stat_2$*:S− end*, then *R(stat:S)* = *sequence R(*$stat_1$*:S+)*, $stat_2$ *end*. By correctness and by the hypothesis that *stat* stops, *R(*$stat_1$*:S+)* has a unique transition

$$\mathrm{R(stat_1{:}S+)} \xrightarrow[\mathrm{E,\ E',\ 1,\ S'}]{} \mathrm{stat'_1},$$

where S' is a non-empty stopset that contains only stops in $stat_1$. By induction,

$$\mathrm{stat'_1} = \mathrm{R(stat_1{:}S')} \tag{3}$$

and S' contains only stops in $stat'_1$. By (IR5) we have

$$\mathrm{sequence\ R(stat_1{:}S+),\ stat_2\ end} \xrightarrow[\mathrm{E,\ E',\ 1,\ S'}]{} \mathrm{sequence\ stat'_1,\ stat_2\ end = stat'.} \tag{4}$$

From (3) and (4)

$$\begin{aligned}\mathrm{stat'} &= \mathrm{sequence\ stat'_1,\ stat_2\ end = sequence\ R(stat_1{:}S'),\ stat_2\ end}\\ &= \mathrm{R((sequence\ stat_1,\ stat_2\ end){:}S') = R(stat{:}S')}\end{aligned}$$

and the result is achieved.

(ii) Let *stat* = *watching X do* $stat_1$ *end*. There are also two main subcases:

— If *stat:S* = *stat:S−*, then *R(stat:S−)* = *stat*.

By correctness and by the hypothesis that *stat* stops, $stat_1$ has a unique transition

$$\mathrm{R(stat_1{:}S) = stat_1} \xrightarrow[\mathrm{E,\ E',\ 1,\ S'}]{} \mathrm{stat'_1},$$

where S' is a non-empty stopset that contains only stops in $stat_1$. By (IR10) we have

$$\mathrm{stat} \xrightarrow[\mathrm{E,\ E',\ 1,\ S'}]{} \mathrm{ifp\ X\ else\ watching\ X\ do\ stat'_1\ end\ end = stat'.}$$

By induction,

$$\mathrm{stat'_1} = \mathrm{R(stat_1{:}S')},$$

and by the fact that S' is a non-empty stopset that contains only stops in $stat_1$,

$$\begin{aligned} \text{stat}' &= \text{ifp X else watching X do } \text{stat}_1' \text{ end end} \\ &= \text{ifp X else watching X do R}(\text{stat}_1 : \text{S}'+) \text{ end end} \\ &= \text{R}(\text{stat} : \text{S}'). \end{aligned}$$

— If $stat{:}S = stat{:}S+$, then $R(stat{:}S+) = ifp\ X\ else\ watching\ X\ do\ R(stat_1{:}S+)\ end\ end$. By correctness and by the hypothesis that *stat* stops, $R(stat_1{:}S+)$ has a unique transition

$$\text{R}(\text{stat}_1{:}\text{S}+) \xrightarrow[\text{E, E', 1, S'}]{} \text{stat}_1',$$

where S' is a non-empty stopset that contains only stops in $stat_1$. By induction,

$$\text{stat}_1' = \text{R}(\text{stat}_1{:}\text{S}')$$

and S' contains stops in $stat_1'$. By (IR10) and (IR9) ($X \notin E$) we have

$$\begin{aligned} &\text{R}(\text{stat:S}+) = \text{ifp X else watching X do R}(\text{stat}_1{:}\text{S}+) \text{ end end} \\ &\xrightarrow[\text{E, E', 1, S'}]{} \text{ifp X else watching X do stat}_1' \text{ end end} = \text{stat}'. \end{aligned}$$

Then

$$\text{stat}' = \text{ifp X else watching X do R}(\text{stat}_1{:}\text{S}') \text{ end end} = \text{R}(\text{stat:S}').$$

Theorem 2. Let P be a correct program and *stat* be its body. Then any derivation *stat'* of *stat* is equal to $R(stat{:}S)$ for some stopset S and there are only finitely many derivations.

Proof. We shall use induction on the length of a transition sequence. Let the derivative *stat'* of *stat* be produced by means of the following sequence:

$$\text{stat} = \text{stat}_1 \xrightarrow[\cdots]{} \cdots \xrightarrow[\cdots]{} \text{stat}_n \xrightarrow[\text{E}_n, \text{E}_n', 1, \text{S}_n]{} \text{stat}'.$$

If $n = 0$, $stat' = stat = R(stat{:}\varnothing)$ and the result is achieved.

Let $stat_n = R(stat{:}S')$ for some stopset S'. Then

$$\text{R}(\text{stat:S}') \xrightarrow[\text{E}_n, \text{E}_n', 1, \text{S}_n]{} \text{stat}'.$$

By Theorem 1,

$$\text{stat}' = \text{R}(\text{stat:S}_n)$$

and the result is achieved.

The finiteness property is obvious since there are only finitely many possible stopsets in *stat*.

We can therefore completely replace a program P by its reaction graph considered as a finite state automaton with derivatives as states.

REFERENCES

1. Manna, Z., A. Pnueli. The Temporal Logic of Reactive and Concurrent Systems: Specification. Springer — Verlag, 1991.

2. Berry, G., G. Gonthier. The Esterel Synchronous Programming Language: Design, Semantics, Implementation.
`http://www.time-rover.com/reactive.html/islip_ps.gz`.
3. Berry, G., S. Ramesh, R. Shyamasunder. Communicating Reactive Processes.
`http://www.time-rover.com/reactive.html`.
4. Halbuachs, N., J-C. Fernansez, A. Bouajjani. An executable temporal logic to express safety properties and its connection with the language Lustre. Proc. in ISLIP'93, 1993.
5. Guernic, P., M. Borgne, T. Gauthier, C. Maire. Programming real time applications with Signal. Proc. of the IEEE, 1991.
6. Plotkin, G. A Structural Approach to Operational Semantics. Technical Report D AIMI FN-19, University of Aarhus, 1981.

Received November 9, 1998
Revised January 15, 1999

Faculty of Mathematics and Informatics
"St. Kliment Ohridski" University of Sofia
5 James Bourchier Blvd.
BG-1164 Sofia, Bulgaria
E-mail: `magda@fmi.uni-sofia.bg`

ГОДИШНИК НА СОФИЙСКИЯ УНИВЕРСИТЕТ „СВ. КЛИМЕНТ ОХРИДСКИ“

ФАКУЛТЕТ ПО МАТЕМАТИКА И ИНФОРМАТИКА
Книга 2 — Приложна математика и информатика
Том 92, 1998

ANNUAIRE DE L'UNIVERSITE DE SOFIA „ST. KLIMENT OHRIDSKI“

FACULTE DE MATHEMATIQUES ET INFORMATIQUE
Livre 2 — Mathématiques Appliquée et Informatique
Tome 92, 1998

DIRECT CONSTRUCTION OF MINIMAL ACYCLIC FINITE STATES AUTOMATA

STOYAN MIHOV

This paper presents automaton construction algorithms based on the method for direct building of minimal acyclic finite states automaton for a given list [2]. A detailed presentation of the base algorithm with correctness and complexity proofs is given. The memory complexity of the base algorithm is $O(m)$ and the worst-case time complexity is $O(n\log(m))$, where n is the total number of letters in the input list, m is the size of the resulting minimal automaton. Further we present algorithms for direct construction of minimal automaton presenting the union, intersection and difference of acyclic automata. In the cases of intersection and difference only the first input automaton has to be acyclic. The memory complexity of those construction algorithms is $O(m)$, and the time complexity is $O(n\log(m))$ for union and $O(n_1 + n\log(m))$ for intersection and difference, where n_1 is the total number of letters in the first automaton language, n is the number of all letters in the resulting automaton language and m is the number of states of the resulting minimal automaton. For construction of minimal automata for large scale languages, in the practice our algorithms deliver significantly better efficiency than the standard algorithms.

Keywords: minimal acyclic finite states automaton, construction of minimal automaton

1991/95 Math. Subject Classification: 68Q68, 68Q45

1. INTRODUCTION

The standard methods for constructing a minimal finite states (FS) automaton proceed in two stages. On the first stage a deterministic FS automaton is built and on the second stage this automaton is minimized. For an overview on the modern

automaton construction and minimization methods see [6, 7]. Those methods have the serious backdraw that the intermediate automaton is huge in respect to the corresponding minimal one.

For many practical applications the construction of large scale acyclic automata is an important task. Methods for construction and minimization of acyclic automata can be found in [3, 4]. Nevertheless, the Revuz' algorithm (which delivers the best efficiency) is also a two stage method and has the above mentioned backdraw. Therefore the use of Revuz' method for construction of very large automata is difficult.

We shall present bellow methods for direct construction of minimal automata where the whole construction is performed in one stage and no intermediate automata are built. Our methods are based on the method for direct construction of minimal automaton for a given lexicographically sorted list [2]. First we shall introduce the mathematical framework which is presented in more details in [2]. After that we present in details the corresponding algorithm and give correctness and complexity proofs. We proceed with a detailed presentation of the algorithms for direct construction of minimal automaton presenting the union, intersection and difference of acyclic automata. Those algorithms are direct descents of the base algorithm. At the end, we give some experimental comparisons of our algorithm with the corresponding Revuz' algorithm.

2. MATHEMATICAL CONCEPTS AND RESULTS

Definition 1. A deterministic FS automaton is a tuple $\mathcal{A} = \langle \Sigma, S, s, F, \mu \rangle$, where:

- Σ is a finite alphabet;
- S is a finite set of states;
- $s \in S$ is the starting state;
- $F \subseteq S$ is the set of final states;
- $\mu : S \times \Sigma \to S$ is a partial function called the transition function.

The function μ is extended naturally over $S \times \Sigma^*$ by induction:

$$\begin{cases} \mu^*(r, \varepsilon) = r, \\ \mu^*(r, \sigma a) = \begin{cases} \mu(\mu^*(r, \sigma), a), & \text{in case } \mu^*(r, \sigma) \text{ and } \mu(\mu^*(r, \sigma), a) \text{ are defined,} \\ \text{not defined} & \text{otherwise,} \end{cases} \end{cases}$$

where $r \in S$, $\sigma \in \Sigma^*$, $a \in \Sigma$.

We will work with a definition of FS automata with a partial transition function. The only difference from the definition with a total transition function is the absence of the necessity to introduce a dead state (a non-finite state r, for which $\forall a \in \Sigma\ (\mu(r, a) = r)$). Later, we will use $!\mu(r, \sigma)$ to denote that $\mu(r, \sigma)$ is defined, and when writing $\mu^*(r, \sigma) \cong x$, we will mean $!\mu(r, \sigma)\ \&\ \mu(r, \sigma) = x$.

Definition 2. Let $\mathcal{A} = \langle \Sigma, S, s, F, \mu \rangle$ be a deterministic FS automaton. Then the set $L(\mathcal{A}) \subseteq \Sigma^*$, defined as

$$L(\mathcal{A}) = \{\sigma \in \Sigma^* \mid !\mu^*(s, \sigma) \,\&\, \mu^*(s, \sigma) \in F\},$$

is called the language of the automaton $\mathcal{A}$ or the language recognized by $\mathcal{A}$.

Two automata $\mathcal{A}$ and $\mathcal{A}'$ are called equivalent when $L(\mathcal{A}) = L(\mathcal{A}')$. An automaton is called acyclic when $\forall r \in S \; \forall \sigma \in \Sigma^+ \; (\mu^*(r, \sigma) \not\cong r)$. The language of an acyclic FS automaton is finite.

Definition 3. Let $\mathcal{A} = \langle \Sigma, S, s, F, \mu \rangle$ be a deterministic FS automaton.

1. The state $r \in S$ is called reachable from $t \in S$ when $\exists \sigma \in \Sigma^* \; (\mu^*(t, \sigma) \cong r)$.
2. We define the subautomaton starting in $s' \in S$ as:

$$\mathcal{A}|_{s'} = \langle \Sigma, S', s', F \cap S', \mu|_{S' \times \Sigma} \rangle ,$$

where $S' = \{r \in S \mid r \text{ is reachable from } s'\}$.

3. Two states $s_1, s_2 \in S$ are called equivalent when $L(\mathcal{A}|_{s_1}) = L(\mathcal{A}|_{s_2})$.

Definition 4. The deterministic FS automaton $\mathcal{A} = \langle \Sigma, S, s, F, \mu \rangle$ with language $L(\mathcal{A})$ is called minimal (with language $L(\mathcal{A})$) when for every other deterministic FS automaton $\mathcal{A}' = \langle \Sigma, S', s', F', \mu' \rangle$ with language $L(\mathcal{A}') = L(\mathcal{A})$ it holds $|S| \leq |S'|$.

From the classical FS theory the following theorem is well-known:

Theorem 5. *A deterministic FS automaton with non-empty language is minimal if and only if every state is reachable from the starting state, from every state a final state is reachable and there are no different equivalent states. There exists an unique (up to isomorphism) minimal automaton for a given language of FS automaton.*

MINIMAL EXCEPT FOR A WORD AUTOMATA

Bellow we will assume that a finite alphabet Σ is given and there is a linear order in Σ. Later, writing lexicographical order of words in Σ^*, we will understand the lexicographical order induced by the linear order of Σ.

Definition 6. Let $\mathcal{A} = \langle \Sigma, S, s, F, \mu \rangle$ be an acyclic deterministic FS automaton with language $L(\mathcal{A})$. Then the automaton $\mathcal{A}$ is called *minimal except for the word* $\omega \in \Sigma^*$ when the following conditions hold:

1. Every state is reachable from the starting state and from every state a final state is reachable.

2. ω is a prefix of the last word in the lexicographical order of $L(\mathcal{A})$.

In that case we can introduce the following notations:

$$\omega = w_1^{\mathcal{A}} w_2^{\mathcal{A}} \dots w_k^{\mathcal{A}}, \text{ where } w_i^{\mathcal{A}} \in \Sigma \text{ for } i = 1, 2, \dots, k, \quad (1)$$

$$t_0^{\mathcal{A}} = s; \quad t_1^{\mathcal{A}} = \mu(t_0^{\mathcal{A}}, w_1^{\mathcal{A}}); \quad t_2^{\mathcal{A}} = \mu(t_1^{\mathcal{A}}, w_2^{\mathcal{A}}); \quad \dots; \quad t_k^{\mathcal{A}} = \mu(t_{k-1}^{\mathcal{A}}, w_k^{\mathcal{A}}), \quad (2)$$

$$T = \{t_0^{\mathcal{A}}, t_1^{\mathcal{A}}, \dots, t_k^{\mathcal{A}}\}. \quad (3)$$

3. In the set $S \setminus T$ there are no different equivalent states.

4. $\forall r \in S \forall i \in \{0, 1, \dots, k\} \forall a \in \Sigma (\mu(r, a) \cong t_i \leftrightarrow (i > 0 \,\&\, r = t_{i-1} \,\&\, a = w_i^{\mathcal{A}}))$.

Bellow, when working with minimal except for a given word automaton, we will use the notations (1)–(3) introduced in the former definition. In case the notation is not ambiguous, we will write t_i, w_i instead of $t_i^{\mathcal{A}}$, $w_i^{\mathcal{A}}$. Clearly, if an automaton is minimal except for two different words, one is a prefix of the other.

Proposition 7. *Let the automaton $\mathcal{A} = \langle \Sigma, S, s, F, \mu \rangle$ be minimal except for ω. Then:*

1. $\forall r \in S \setminus T \; \forall a \in \Sigma \; (!\mu(r, a) \rightarrow \mu(r, a) \in S \setminus T)$.
2. $\mu^*(s, \sigma) \cong t_i \leftrightarrow \sigma = w_1 w_2 \dots w_i$.

Proposition 8. *An automaton which is minimal except for the empty word ε is minimal.*

Lemma 9. *Let the automaton $\mathcal{A} = \langle \Sigma, S, s, F, \mu \rangle$ be minimal except for $\omega = w_1 w_2 \dots w_k$, $\omega \neq \varepsilon$. Let there be no state equivalent to t_k in the set $S \setminus T$. Then $\mathcal{A}$ is also minimal except for the word $w_1 w_2 \dots w_{k-1}$.*

Lemma 10. *Let the automaton $\mathcal{A} = \langle \Sigma, S, s, F, \mu \rangle$ be minimal except for $\omega = w_1 w_2 \dots w_k$, $\omega \neq \varepsilon$. Let the state $p \in S \setminus T$ be equivalent to the state t_k. Then the automaton $\mathcal{A}' = \langle \Sigma, S', s, F', \mu' \rangle$ defined as follows:*

$$S' = S \setminus \{t_k\},$$

$$F' = F \setminus \{t_k\},$$

$$\mu'(r, a) = \begin{cases} \mu(r, a), & \text{in case } r \neq t_{k-1} \vee a \neq w_k \text{ and } \mu(r, a) \text{ is defined,} \\ p, & \text{in case } r = t_{k-1},\ a = w_k, \\ \text{not defined} & \text{otherwise,} \end{cases}$$

is equivalent to the automaton $\mathcal{A}$ and is minimal except for the word $w_1 w_2 \dots w_{k-1}$.

Theorem 11. *Let the automaton $\mathcal{A} = \langle \Sigma, S, s, F, \mu \rangle$ be minimal except for $\omega' = w_1 w_2 \dots w_m$. Let $\psi \in L(\mathcal{A})$ be the last word in the lexicographical order of the language of the automaton. Let ω be a word which is greater in lexicographical order than ψ. Let ω' be the longest common prefix of ψ and ω. In that case we can denote $\omega = w_1 w_2 \dots w_m w_{m+1} \dots w_k; k > m$. Then the automaton $\mathcal{A}' = \langle \Sigma, S', s, F', \mu' \rangle$ defined as follows:*

$t_{m+1}, t_{m+2}, \dots, t_k$ *are new states such that* $S \cap \{t_{m+1}, t_{m+2}, \dots, t_k\} = \varnothing$,

$$S' = S \cup \{t_{m+1}, t_{m+2}, \dots, t_k\},$$

$$F' = F \cup \{t_k\},$$

$$\mu'(r,a) = \begin{cases} t_{i+1}, & \text{in case } r = t_i,\ m \le i \le k-1,\ a = w_{i+1}, \\ \mu(r,a), & \text{in case } r \in S \text{ and } \mu(r,a) \text{ is defined and} \\ & r \neq t_m \vee a \neq w_{m+1}, \\ \text{is not defined} & \text{otherwise,} \end{cases}$$

is minimal except for ω *and recognizes the language* $L(\mathcal{A}) \cup \{\omega\}$.

Lemma 12. *Let the automaton* $\mathcal{A} = \langle \Sigma, S, s, F, \mu \rangle$ *be minimal except for* $\omega = w_1 w_2 \ldots w_k$. *Then for* t_k *(refer to the notation introduced in Definition* 6*) the following statement holds:*

$$t_k \text{ is equivalent to } r \in S \setminus T \leftrightarrow \Big((t_k \in F \leftrightarrow r \in F)$$

$$\& \ \forall a \in \Sigma \left((\neg !\mu(t_k, a) \ \& \ \neg !\mu(r, a)) \vee (!\mu(t_k, a) \ \& \ !\mu(r, a) \ \& \ \mu(t_k, a) = \mu(r, a)) \right) \Big).$$

The proofs of the above results are presented in [2].

3. ON-LINE ALGORITHM FOR BUILDING A MINIMAL FS AUTOMATON FOR A GIVEN LIST

First we describe the method informally and give an example. After that we give the pseudo-code in a Pascal-like language (like the language used in [1]) with correctness proof.

Let a non-empty finite list of words L in lexicographical order be given. Let $\omega^{(i)}$ denote the i-th word of the list. We start with the minimal automaton which recognizes only the first word of the list. This automaton can be built trivially and is also minimal except for $\omega^{(1)}$. Using it as basis, we carry out an induction on the words of the list. Let us assume that the automaton $\mathcal{A}^{(n)} = \langle \Sigma, S, s, F, \mu \rangle$ with language $L^{(n)} = \{\omega^{(i)} \mid i = 1, 2, \ldots, n\}$ has been built and that $\mathcal{A}^{(n)}$ is minimal except for $\omega^{(n)}$. We have to build the automaton $\mathcal{A}^{(n+1)}$ with language $L^{(n+1)} = \{\omega^{(i)} \mid i = 1, 2, \ldots, n+1\}$ which is minimal except for $\omega^{(n+1)}$.

Let ω' be the longest common prefix of the words $\omega^{(n)}$ and $\omega^{(n+1)}$. Using several times Lemma 9 and Lemma 10 (corresponding to the actual case), we build the automaton $\mathcal{A}' = \langle \Sigma, S', s, F', \mu' \rangle$ which is equivalent to $\mathcal{A}^{(n)}$ and is minimal except for ω'. Now we can use Theorem 11 and build the automaton $\mathcal{A}^{(n+1)}$ with language $L^{(n+1)} = L^{(n)} \cup \{\omega^{(n+1)}\} = \{\omega^{(i)} \mid i = 1, 2, \ldots, n+1\}$ which is minimal except for $\omega^{(n+1)}$.

In this way by induction we build the minimal except for the last word of the list automaton with language the list L. At the end, using again Lemma 9 and Lemma 10, we build the automaton equivalent to the former one, which is minimal except for the empty word. From Proposition 8 we have that it is the minimal automaton for the list L. To distinguish efficiently between Lemma 9 and Lemma 10, we can use Lemma 12. □

Let us describe now the algorithm more formally. We will presume that there are given implementations for Abstract Data Types (ADT) representing the automaton state and the dictionary of automaton states. Later, we presume that NULL is the null constant for an arbitrary abstract data type.

On automaton state we shall need the following types and operations:

1. STATE is pointer to a structure representing an automaton state.

2. FIRST_CHAR, LAST_CHAR : char are the first and the last char in the automaton alphabet. We will assume that the chars are sequentially given in lexicographical order.

3. **function** NEW_STATE : STATE returns a new state.

4. **function** FINAL(STATE) : boolean returns true if the state is final and false otherwise.

5. **procedure** SET_FINAL(STATE,boolean) sets the finality of the state to the boolean parameter.

6. **function** TRANSITION(STATE,char): STATE returns the state to which the automaton transits from the parameter state with the parameter char.

7. **procedure** SET_TRANSITION(STATE,char,STATE) sets the transition from the first parameter state by the parameter char to the second parameter state.

8. **procedure** PRINT_AUTOMATON(file,STATE) prints the automaton starting from the parameter state to file.

Having defined the above operations, we make use of the following three functions and procedures:

```
function COPY_STATE (s : STATE) : STATE;
{ copies s to a new state}
var
   r : STATE;
   c : char;
begin
   r := NEW_STATE;
   SET_FINAL(r,FINAL(s));
   for c := FIRST_CHAR to LAST_CHAR do
      SET_TRANSITION(r,c,TRANSITION(s,c));
   return(r);
end; { COPY_STATE}

procedure CLEAR_STATE (s : STATE);
{ clears all transitions of s and sets it to non-final one }
var c : char;
begin
   SET_FINAL(s,false);
   for c := FIRST_CHAR to LAST_CHAR do
      SET_TRANSITION(s,c,NULL);
```

```
end; {CLEAR_STATE}

function COMPARE_STATES (s1,s2 : STATE) : integer;
{compares two states}
var c : char;
begin
   if FINAL(s1) < FINAL(s2) then return(-1)
   else if FINAL(s1) > FINAL(s2) then return(1);
   for c := FIRST_CHAR to LAST_CHAR do
      if TRANSITION(s1,c) < TRANSITION(s2,c) then return(-1)
      else if TRANSITION(s1,c) > TRANSITION(s2,c) then return(1);
      {here we compare only the pointers }
   return(0);
end; {COMPARE_STATES}
```

The ADT on Dictionary of automaton states uses the COMPARE_STATES function above to compare states. For the dictionary we need the following operations:

1. **function** NEW_DICTIONARY : DICTIONARY returns a new empty dictionary;

2. **function** MEMBER(DICTIONARY,STATE) : STATE returns state in the dictionary equivalent to the parameter state or NULL if not present;

3. **procedure** INSERT(DICTIONARY,STATE) inserts state to dictionary.

Implementations of the above ADTs can be found in [1]. Later we assume that the time complexity of PRINT_AUTOMATON is proportional to the size of the automaton and all other operations on Automaton states including COPY_STATE, CLEAR_STATE and COMPARE_STATES are performed in constant time. Depending on the concrete implementation of the dictionary, we could have different bounds for the time complexity of the operations. Using a typical implementation by, e.g., AVL balanced trees, we will have a logarithmic time complexity for the MEMBER and INSERT operations and the size of the dictionary will be proportional to the number of its elements.

Now we are ready to present the pseudo-code of our algorithm.

Algorithm 1. For on-line construction of minimal automaton presenting the input list of words given in lexicographical order.

```
1   program Create_Minimal_FS_Automaton_for_Given_List (input, output);
2   var
3      MinimalAutomatonStatesDictionary : DICTIONARY;
4      TempStates : array [0..MAX_WORD_SIZE] of STATE;
5      InitialState : STATE;
6      PreviousWord, CurrentWord : string;
7      i, PrefixLengthPlus1 : integer;
```

```
8       function FindMinimized ( s : STATE) : STATE;
9       {returns an equivalent state from the dictionary; if not present —
          inserts a copy of the parameter state to the dictionary and returns it}
10      var r : STATE:
11      begin
12          r := MEMBER(MinimalAutomatonStatesDictionary,s);
13          if r = NULL then begin
14              r := COPY_STATE(s);
15              INSERT(r);
16          end;
17          return(r);
18      end; { FindMinimized}

19  begin
20      MinimalAutomatonStatesDictionary := NEW_DICTIONARY;
21      for i := 0 to MAX_WORD_SIZE do
22          TempState[i] := NEW_STATE;
23      PreviousWord := '';
24      CLEAR_STATE(TempState[0]);

25      while not eof(input) do begin
26      {loop for the words in the input list}
27          readln(input,CurrentWord);

28          {the following loop calculates the length of the longest common}
             prefix of CurrentWord and PreviousWord}
29          i := 1;
30          while (i<length(CurrentWord)) and (i<length(PreviousWord)) and
                (PreviousWord[i] = CurrentWord[i]) do
31              i := i+1;
32          PrefixLengthPlus1 := i;

33          {here we are minimizing the states of the last word }
34          for i := length(PreviousWord) downto PrefixLengthPlus1 do
35              SET_TRANSITION(TempStates[i-1],PreviousWord[i],
                                    FindMinimized(TempStates[i]));
36          {this loop initializes the tail states for the current word}
37          for i := PrefixLengthPlus1 to length(CurrentWord) do begin
38              CLEAR_STATE(TempStates[i]);
39              SET_TRANSITION(TempStates[i-1],CurrentWord[i],
                                    TempStates[i]);
40          end;
41          SET_FINAL(TempStates[length(CurrentWord)], true);

42          PreviousWord := CurrentWord;
43      end; {while}
```

```
44      { here we are minimizing the states of the last word}
45      for i := length(CurrentWord) downto 1 do
46          SET_TRANSITION(TempStates[i-1],PreviousWord[i],
                                FindMinimized(TempStates[i]));
47      InitialState := FindMinimized(TempStates[0]);
48      PRINT_AUTOMATON(output,InitialState);
49  end.
```

Now we will prove the correctness and calculate the time and space complexity of the algorithm.

Theorem 13. *Given a lexicographically sorted list of words in the input file, Algorithm* 1 *builds the minimal FS automaton for the list and prints it out on the output file.*

Proof. To prove the theorem, we carry out an induction on the words of the input list.

In lines 20–24 the algorithm initializes the Dictionary of states of the minimal automaton to the empty dictionary and the temporary states. In line 24 *Tempstate*[0] is initialized to a non-final state with no transition. This corresponds to the automaton for the empty language. Line 27 reads the first word from the input. Because of the initialization in line 23, we have that at that moment *PreviousWord* is the empty string. Hence *PrefixLengthPlus1* will be set to 1 in lines 29–32. The loop in lines 33–35 will not be triggered and the loop 36–40 will construct a chain of states for recognizing the first word. In this way the algorithm constructs the minimal automaton for the language consisting of the first word in the input list. Clearly, this automaton is also minimal except for the first word $\omega^{(1)}$. In that moment the automaton is minimal except for $\omega^{(1)}$ and the states $t_0, t_1, \ldots, t_k$ are presented in *Tempstate*. In line 42 the first word is assigned to the string *PreviousWord*.

Now we will show that the loop in lines 25–43 adds the next word from the list to the automaton and produces a minimal except for this word automaton.

Let us assume that on stage j the algorithm has built the automaton recognizing $\{\omega^{(i)} \mid i = 1, 2, \ldots, j-1\}$, which is minimal except for $\omega^{(j-1)}$. The states $t_0, t_1, \ldots, t_k$ are presented in the array *TempStates*, all other states of the automaton are in *MinimalAutomatonStatesDictionary* and *PreviousWord* is $\omega^{(j-1)}$.

Being in line 27, the word $\omega^{(j)}$ is read from the input file into *CurrentWord*. The loop 28–32 calculates the longest common prefix of *PreviousWord* and *CurrentWord* with values at that moment $\omega^{(j-1)}$ and $\omega^{(j)}$. We will show that the loop 33–35 builds the equivalent automaton minimal except for the longest common prefix of $\omega^{(j-1)}$ and $\omega^{(j)}$. In downward order the transition to the state t_i is replaced by a transition to the state which returns the function *FindMinimized*, where i varies from the length of *PreviosWord* to *PrefixLengthPlus1* in reverse order. The function *FindMinimized* searches a state equivalent to the argument in *MinimalAutomatonStatesDictionary*. Here the conditions of Lemma 12 are fulfilled, therefore

the use of COMPARE_STATE in the function MEMBER will identify the equivalent to t_i state. If such a state exists, it is returned as result. This corresponds to the condition of Lemma 10. In the other case, the function copies the state and inserts it into *MinimalAutomatonStatesDictionary*. The copy of the state is returned as result. This corresponds to Lemma 9. According to those lemmata, in both cases the new automaton will be equivalent to the former and minimal except for the shorter prefix. After finishing the loop, we have an automaton recognizing $\{\omega^{(i)} \mid i = 1, 2, \ldots, j-1\}$, which is minimal except for the longest common prefix of $\omega^{(j-1)}$ and $\omega^{(j)}$.

The loop 36–40 simply constructs a tail of states in the array *TempStates* in order to recognize *CurrentWord*. In line 41 the last state is marked as final. This corresponds exactly to the conditions of Theorem 11. Therefore we have built the automaton for the language $\{\omega^{(i)} \mid i = 1, 2, \ldots, j\}$ minimal except for the word $\omega^{(j)}$. After assigning $\omega^{(j)}$ to *PreviousWord* in line 42, we are closing the main loop.

From the induction we have that after finishing the loop 25–43 the algorithm will build the automaton for the input list which is minimal except for the last word. The lines 44–47 in the same way as the loop 33–35 build the equivalent automaton which is minimal except for ε. From Proposition 8 we have that this is the minimal automaton for the list. Line 48 prints the automaton on the output file. □

Theorem 14. *Algorithm* 1 *builds the minimal automaton for a given alphabetically sorted list of words in* $O(n \log(m))$ *time, where* n *is the total number of letters in the input list and* m *is the size (number of states) of the resulting minimal automaton. The space complexity of Algorithm* 1 *is* $O(m)$.

Proof. For each letter from the input list the algorithm passes either through line 31 or through lines 38–39. Each of the statements of those lines are performed in constant time. In case we have passed through the lines 38–39, we later have to pass through line 35 or 46. The time complexity of the lines 35, 46 depends on the time complexity of *FindMinimized*. By using balanced tree implementation of the dictionary we have that the complexity of *FindMinimized* is logarithm of the size of the dictionary. The dictionary has at most m elements, where m is the number of the states of the minimal automaton for the list. Hence the time complexity of the whole algorithm is $O(n \log(m))$.

Clearly, the space needed by the algorithm is equal to the size of the dictionary of states of the minimal automaton plus the size of the *TempStates* array plus the constant size of the other fixed-size variables. *TempStates* is proportional to the size of the longest word in the list and in the case of using balanced tree implementation, the size of the dictionary of states of minimal automaton is proportional to the number of states of the minimal automaton. Clearly, the size of the longest word in the list is lower than the size of the minimal automaton representing this list. Therefore the space complexity is $O(m)$. □

The main advantage of our method is the excellent space to time proportion.

4. ALGORITHMS FOR DIRECT CONSTRUCTION OF MINIMAL AUTOMATON PRESENTING UNION, INTERSECTION AND DIFFERENCE OF ACYCLIC AUTOMATA

The standard methods for construction of automaton presenting union, intersection and difference are building first a temporary automaton which states are Cartesian product of states of the input automata. This temporary automaton in general is huge with respect to the resulting minimal automaton. Here we will present a new method for direct constructing the minimal automaton which drastically improves the efficiency.

By traversing an acyclic deterministic FS automaton in depth first by choosing the transitions in lexicographical order we can produce the automaton language in lexicographical order. Using this property, we can produce the lexicographical ordered list which is union, intersection or difference of the languages of the input automata. Using this list as input for Algorithm 1, we can construct directly the minimal automaton for the union, intersection and difference. Moreover, we do not have to build explicitely the whole lists in the memory. We can proceed word by word using only the top words of the lists. Bellow we give the formal description of our algorithm.

We will need the following declaration:

type States_Stack = **array** [1..MAX_WORD_SIZE+1] of STATE — type array of automaton states.

We will use array of states for representing automaton path. If we have a word w: string, we will have $S[i+1] = \text{TRANSITION}(S[i], w[i])$, $i = 1, 2, \ldots, \text{lenght}(w)$, where $S[0]$ is the initial automaton state.

For producing the language of an automaton word by word, we will use a function which for a given word and corresponding path returns the next word in lexicographical order in automaton language.

Algorithm 2. Given a word and a corresponding automaton, path returns the next word in lexicographical order in the automaton language.

We will assume that from any automaton state a final state is reachable.

```
function NEXT_AUTOM_WORD(S : States_Stack; var w : string) : boolean;
var
   c : char;
   sp : integer;

   function FIND_FORWARD_WORD : boolean;
   begin
      c := FIRST_CHAR;
      while (c<=LAST_CHAR) and (TRANSITION(S[sp], c) = NULL)
         do c := succ(c);
      if c>LAST_CHAR then return(false);
      S[sp + 1] := TRANSITION(S[sp], c);
```

```
        sp := sp + 1;
        w := concat(w, c);
        while not FINAL(S[sp]) do begin
            c := FIRST_CHAR;
            while TRANSITION(S[sp], c) = NULL do c :=succ(c);
            S[sp + 1] := TRANSITION(S[sp], c);
            sp := sp + 1;
            w := concat(w, c);
        end;
        return(true);
    end;

begin
    sp := length(w);
    if FIND_FORWARD_WORD then return(true);
    repeat
        if sp = 1 then return(false);
        sp := sp - 1;
        c := w[length(w)];
        delete(w, length(w), 1);
        while (c <= LAST_CHAR) and (TRANSITION(S[sp], c) = NULL)
            do c := succ(c);
    until c <= LAST_CHAR;
    S[sp + 1] := TRANSITION(S[sp], c);
    sp := sp + 1;
    w := concat(w, c);
    if not FINAL(S[sp])
        then return(FIND_FORWARD_WORD) else return(true);
end;
```

Theorem 15. *For a given word and a corresponding automaton, path Algorithm 2 finds the next word in lexicographical order in the automaton language and returns true or returns false in case there are no more words.*

We can proof the above theorem by induction on the words in the automaton language.

We will use the above function for producing the lexicographically sorted list representing the union, intersection and difference of automaton languages.

Algorithm 3. For producing the next word in lexicographical order of the union of two acyclic automaton languages.

We shall need the following global variables:

```
var
    p1, p2, f1, f2 : boolean;
    s1, s2 : States_Stack;
    w1, w2 : string;
```

We shall assume that they are initialized by the following procedure:

```
procedure INIT_NEXT_WORD;
begin
    s1[1] := init_state1;
    s2[1] := init_state2;
    w1 := '';
    w2 := '';
    p1 := true;
    p2 := true;
    f1 := true;
    f2 := true;
end;
```

init_state1, *init_state2* are the initial states of the two automata. In that case the function NEXT_WORD produces in the variable w the next word in lexicographical order of the union list or returns false in case there are no more words.

```
function NEXT_WORD(var w : string) : boolean;
begin
    if not f1 and not f2 then return(false);
    if p1 then f1 := NEXT_AUTOM_WORD(s1,w1);
    if p2 then f2 := NEXT_AUTOM_WORD(s2,w2);
    if not f1 and not f2 then
        return(false)
    else if (f1 and f2) and (w1 = w2) then begin
        w := w1;
        p1 := true;
        p2 := true;
    end else if not f1 or ((f1 and f2) and (w1 > w2)) then begin
        w := w2;
        p1 := false;
        p2 := true;
    end else if not f2 or ((f1 and f2) and (w1 < w2)) then begin
        w := w1;
        p1 := true;
        p2 := false;
    end;
    return(true);
end;
```

The function NEXT_WORD proceeds as follows: in the variables *p1*, *p2* we mark the necessity for reading the next word from the corresponding automaton. In the variables *f1*, *f2* we mark the ending of the corresponding automaton. In case both automata are traversed, the function returns false. In the other case the two current words from the automata lists are compared. In case the words are equal, we return one of them and mark in *p1*, *p2* that on the next call of the function the words from both lists have to be read. In case one of the words preceeds the other, we return that word and mark the corresponding automaton in order to read the next word from it. In case one of the automaton languages has finished, the other is listed until both are finished.

We have to note that the function NEXT_WORD returns word by word the list of the words in the union of the two input automata without using any extra memory for generating the lists. The time for listing the words in the union is proportional to the sum of the lengths (in letters) of the languages of the input automata. This follows from the next facts. To list the words, in the union the paths are traversing from the initial to the final states in the input automata. The sum of all those paths in an automaton is equal to the number of all letters in the automaton language. □

For producing the list of the intersection or difference of two automata, we proceed similar to the method above. But we shall present a more efficient method which is applicable also in case the second automaton is not acyclic.

First we present an additional function which returns true in case the word is recognized by the automaton and false otherwise.

```
function RECOGNIZE_WORD (w: string; s: STATE) : boolean;
var i : integer;
begin
   i := 1;
   while i <= length(w) do begin
      if TRANSITION(s, w[i]) = NULL then return(false);
      s := TRANSITION(s, w[i]);
      i := i + 1;
   end;
   return(FINAL(s));
end;
```

The functionality of the above function is clear. We have only to note that the recognition time for a word is proportional to the length of the word.

Algorithm 4. For producing the next word in lexicographical order of the intersection of an acyclic deterministic FS automaton with a deterministic FS automaton language.

We shall need the following global variables:

```
var
   f1 : boolean;
```

```
    s1 : States_Stack;
    w1 : string;
```

We shall assume that they are initialized by the following procedure:

```
procedure INIT_NEXT_WORD;
begin
    s1[1] := init_state1;
    w1 := '';
    f1 := true;
end;
```

init_state1, *init_state2* are the initial states of the two automata. In that case the function NEXT_WORD produces in the variable w the next word in lexicographical order of the intersection list or returns false in case there are no more words.

```
function NEXT_WORD(var w: string) : boolean;
begin
    if not f1 then return(false);
    repeat
        f1 := NEXT_AUTOM_WORD(s1, w1);
    until not f1 or RECOGNIZE_WORD(w1, init_state2);
    if not f1 then return(false);
    w := w1;
    return(true);
end;
```

In that case the function NEXT_WORD proceeds as follows: word by word the first automaton language is listed in lexicographical order. In case the current word is recognized by the second automaton, this word is returned as the next word in the intersection.

Here we have to note that the function NEXT_WORD produces the intersection list word by word without using extra memory for generating the whole lists. The time for producing the intersection list is obviously proportional to the number of all letters in the first automaton. □

For deriving an algorithm producing the next word from the difference of an acyclic deterministic FS automaton with a deterministic FS automaton, we have to make in the above algorithm the following change:

```
until not f1 or RECOGNIZE_WORD(w1, init_state2);
```

have to be exchanged with

```
until not f1 or not RECOGNIZE_WORD(w1, init_state2);
```

There will be almost no changes in the functionality of the algorithm and the time complexity for producing the difference list will be again proportional to the number of all letters in the first automaton language.

Let us present now the algorithm for direct construction of minimal automaton.

Algorithm 5. For direct construction of minimal automaton presenting the union, intersection or difference of acyclic automaton languages. (In case of intersection and difference, only the first automaton has to be acyclic.)

We shall use as base Algorithm 1. We assume that the global variables *init_state1*, *init_state2* are given, which represent the initial states of the first and second automaton. We shall assume also that the global variables of Algorithm 3 or Algorithm 4 and the corresponding functions NEXT_AUTOM_WORD, NEXT_WORD, INIT_NEXT_WORD, RECOGNIZE_WORD are defined. We need further the following changes of Algorithm 1:

1. Between lines 24 and 25 we have to call the initialization procedure

   INIT_NEXT_WORD;

2. Line 25 has to be changed to

   while NEXT_WORD(*CurrentWord*) **do begin**

3. Line 27 has to be deleted.

The only difference between the above algorithm and Algorithm 1 is the use of an input list which presents the union, intersection or difference of the input automata. We derive the following complexity results. For the union the time complexity of Algorithm 5 is $O((n_1+n_2)+n\log(m))$, where n_1, n_2 are the number of letters of the two input automata languages, n is the number of letters in the union language and m is the size (number of states) of the resulting minimal automaton. We obviously have that $n_1+n_2 < 2n$, hence the time complexity is $O(n\log(m))$. For the intersection and difference we have that the time complexity is $O(n_1+n\log(m))$. The memory complexity in all cases of Algorithm 5 is $O(m)$. □

5. IMPLEMENTATION RESULTS AND COMPARISONS

We have implemented various tools for constructing, updating and processing of lexicons presented as minimal automaton. They are programmed in GNU-C and JAVA. For a more efficient implementation we have used an open hash structure for the lexicon of automaton states presentation. This provides a nearly linear time complexity for practical applications.

We have experimented with grammatical lexicons for Bulgarian and Russian[1] languages. The middle-sized lexicon for Bulgarian common lexica has about 500000 wordforms and the Russian one — about 1500000 wordforms. They are encoded according to the DELAF format [5]. To provide additional grammatical information to the words, we have used a FS automaton with labels on the final states. A trivial change is needed to modify Algorithm 1 to build minimal automata with such labels. In the INTEX system [5] there is a similar tool for building the same kind of FS

[1] The DELAF format of the Russian lexicon is build in cooperation with the Computer Fond of the Russian Language in Moscow.

Table 1. Comparison of our and the INTEX tool for building minimal automata

		INTEX tool	our tool
Bulgarian lexicon	Size (Wordforms)	524473	
	Memory used	33450 KB	1660 KB
	Time needed	2:07 min	0:29 min
Russian lexicon	Size (Wordforms)	1486552	
	Memory used	129000 KB	4400 KB
	Time needed	17 min	2:04 min

automata. This tool is a highly efficient implementation of the Revuz' algorithm. Table 1 shows a comparison between our tool and the corresponding INTEX tool.

All time and memory parameters given in the paper are measured on a 32MB RAM Pentium 180 machine running under NEXTSTEP. The large time requirements of the INTEX tool for the Russian lexicon are explained with the heavy usage of virtual memory. On a small lexicon (when the trie structure for the INTEX tool fits into the operating memory) our tool is only slightly faster than the INTEX one.

6. CONCLUSION

The presented methods and algorithms are successfully used for construction and operations on large scale dictionaries. They are distinguished with significantly better memory efficiency than the others.

An open question is the existence of a method for direct construction of minimal automaton presenting the concatenation of acyclic automaton languages. There seems to be a problem producing the concatenation list lexicographically sorted.

Acknowledgements. The author would like to thank Professor Klaus Schulz, Professor Dimiter Skordev and Anton Zinoviev for the very valuable remarks on the presentation.

REFERENCES

1. Aho, A., J. Hopcroft, J. Ullman. Data Structures and Algorithms. Addison-Wesley, Reading, Massachutes, 1983.
2. Mihov, S. Direct Building of Minimal Automaton for Given List. *Ann. Sof. Univ.*, Fac. Math. et Inf., **91**, 1, 1999, 33–40.
3. Revuz, D. Dictionaires et lexiques, méthodes et algorithmes. Ph.D. dissertation, Université Paris 7, Paris, 1991.
4. Revuz, D. Minimization of acyclic deterministic automata in linear time. *Theoretical Computer Science*, **92**, 1, 1992.
5. Silberztein, M. Dictionnaires électroniques et analyse automatique de textes: le système INTEX. Masson, Paris, 1993.

6. Watson, B. W. A Taxonomy of Finite Automata Construction Algorithms. Computing Science Report 93/43, Fac. of Math. and Comp. Science, Eindhoven University of Technology, Eindhoven, The Netherlands, 1993.
7. Watson, B. W. A Taxonomy of Deterministic Finite Automata Minimization Algorithms. Computing Science Report 93/44, Fac. of Math. and Comp. Science, Eindhoven University of Technology, Eindhoven, The Netherlands, 1993.

Received November 13, 1998

Linguistic Modelling Laboratory
LPDP — Bulgarian Academy of Sciences
E-mail: stoyan@lml.acad.bg

ГОДИШНИК НА СОФИЙСКИЯ УНИВЕРСИТЕТ „СВ. КЛИМЕНТ ОХРИДСКИ“

ФАКУЛТЕТ ПО МАТЕМАТИКА И ИНФОРМАТИКА
Книга 2 — Приложна математика и информатика
Том 92, 1998

ANNUAIRE DE L'UNIVERSITE DE SOFIA „ST. KLIMENT OHRIDSKI“

FACULTE DE MATHEMATIQUES ET INFORMATIQUE
Livre 2 — Mathématiques Appliquée et Informatique
Tome 92, 1998

A METHOD FOR SOLVING THE SPECTRAL PROBLEM OF HAMILTONIAN MATRICES WITH APPLICATION TO THE ALGEBRAIC RICCATI EQUATION

IVAN GANTCHEV IVANOV

In this paper an effective iterative method for computing the eigenvalues and eigenvectors of a real Hamiltonian matrix is described and its applicability discussed. The method is an adaptation for Hamiltonian matrices of the methods for computing eigenvalues of real matrices due to Veselić and Voevodin. It uses symplectic similarity transformations and preserves the Hamiltonian structure of the matrix. Our method can be used for solving algebraic Riccati equation. The method is tested numerically and a comparison with the performance of other numerical algorithms is presented.

Keywords: Hamiltonian matrix, Jacobi-like methods, algebraic Riccati equation

1995 Math. Subject Classification: 65F10, 65F15

1. INTRODUCTION

Many applications lead to solving the real spectral problem

$$Hx = \lambda x,$$

where

$$H = H(A, B, D) = \begin{pmatrix} A & B \\ D & -A^T \end{pmatrix},$$

$A \in \mathbb{R}^{n\times n}$, $B = B^T \in \mathbb{R}^{n\times n}$, $D = D^T \in \mathbb{R}^{n\times n}$.

Recall that the real matrix H is called Hamiltonian or J-skewsymmetric if $J^T H J = -H^T$, where $J = \begin{pmatrix} 0_n & I_n \\ -I_n & 0_n \end{pmatrix}$, I_n is the $n \times n$ identity matrix and 0_n is the $n \times n$ zero matrix [7, 11, 12]. A matrix $U \in \mathbb{R}^{n \times n}$ is called a symplectic or J-orthogonal if $U^T J U = J$. It is well-known that if H is a Hamiltonian matrix and U is a symplectic matrix, then the matrix $U^{-1} H U$ is a Hamiltonian matrix.

Generalizations of the Jacobi process for arbitrary matrices, based on the fact that there exists a matrix P such that $\tilde{A} = P^{-1} A P$ is arbitrarily close to being normal, have been proposed [3, 15, 16]. In other words, the absolute value of every element of $\tilde{A}\tilde{A}^* - \tilde{A}^*\tilde{A}$ is arbitrarily small. Byers [2] has proposed a symplectic Jacobi-like algorithm for the computation of the Hamiltonian-Schur decomposition of Hamiltonian matrices. Byers' method is an adaptation of the non-symmetric Jacobi method proposed by Stewart [14].

In this paper an iterative method for solving the spectral problem for a Hamiltonian matrix is developed. It is a modification for Hamiltonian matrices of Veselić's and Voevodin's methods for computing eigenvalues of real matrices [15, 16]. The method uses similarity transformations with symplectic matrices. These transformations keep the block structure of a Hamiltonian matrix. This method can be used for solution of the algebraic Riccati equation.

The algebraic Riccati equation is of great practical importance due to its key role in control theory. There exist different procedures for solving this equation: a method solving a suitable matrix equation [6], a method solving a spectral problem for the Hamiltonian matrix [13]. Other methods are discussed in [8, 10, 17].

2. DESCRIPTION OF THE ALGORITHM

Now we describe the algorithm of our method. In this algorithm we construct the following sequence of Hamiltonian similar matrices:

$$\begin{aligned} H_1(A_1, B_1, D_1) &= H(A, B, D), \\ H_{k+1} = H(A_{k+1}, B_{k+1}, D_{k+1}) &= U_k^{-1} H_k U_k = (h_{rs}^{(k+1)}), \ldots, \end{aligned} \tag{2.1}$$

$k = 1, 2, 3$, where $U_k = U_{p_k q_k}(\varphi_k)$ is a suitable symplectic matrix. The matrix U_k depends on three parameters p_k, q_k and φ_k for each k. At each iteration step the parameters of U_k are chosen either to minimize

$$\|U_k^{-1} H_k U_k\|, \quad \text{where} \quad \|H_k\| = \sum_{rs} \left(h_{rs}^{(k)}\right)^2,$$

or to annihilate the off-diagonal elements of the symmetric matrix $H_{k+1} + H_{k+1}^T$.

To give an idea for the iterative process (2.1), we shall explain only the k-th iteration step of the algorithm. We introduce the notation

$$H_k = H(A_k, B_k, D_k) = (h_{rs}^{(k)}),$$
$$A_k = (a_{\beta\gamma}^{(k)}), \; B = (b_{\beta\gamma}^{(k)}), \; D_k = (d_{\beta\gamma}^{(k)}).$$

For the matrices $H_k + H_k^T$ and C_k we have

$$\begin{aligned} H_k + H_k^T &= H(A_k + A_k^T, B_k + D_k, D_k + B_k), \\ C_k = C(H_k) &= (c_{rs}^{(k)}) = H_k H_k^T - H_k^T H_k = H(F_k, E_k, E_k), \end{aligned}$$

where

$$\begin{aligned} F_k &= F_k^T = A_k A_k^T + B_k B_k - A_k^T A_k - D_k D_k = (f_{\beta\gamma}^{(k)}), \\ E_k &= E_k^T = A_k D_k - B_k A_k - A_k^T B_k + D_k A_k^T = (e_{\beta\gamma}^{(k)}). \end{aligned}$$

The strategy, determining U_k from (2.1) and parameters p_k, q_k, φ_k, is the following. At the k-th iteration step we find the numbers

$$c^{(k)} = \max_{r \neq s} |c_{rs}^{(k)}|^{\frac{1}{2}} \quad \text{and} \quad h^{(k)} = \max_{r \neq s} |h_{rs}^{(k)} + h_{sr}^{(k)}|$$

for the matrices C_k and H_k.

Then there are six possible cases to be considered successively:

A.1. $|f_{pq}^{(k)}|^{1/2} = c^{(k)} \geq h^{(k)}$, $1 \leq p = p_k < q = q_k \leq n$, $\varphi = \varphi_k$.

In this case we choose the matrix $U_k = U_{pq}(\varphi)$ of the form

$$U = U_{pq}(\varphi) = \begin{pmatrix} S_{pq}(\varphi) & 0 \\ 0 & S_{pq}^{-T}(\varphi) \end{pmatrix}, \tag{2.2}$$

where $S_{pq}(\varphi) \in \mathbb{R}^{n \times n}$ is the matrix

$$S_{pq}(\varphi) = (s_{ij}) = \begin{cases} s_{qp} = \varphi, \\ s_{ij} = \delta_{ij}, \quad (i,j) \notin \{(q,p)\}. \end{cases}$$

Note that $S_{pq}^{-1}(\varphi) = S_{pq}(-\varphi)$.

The parameter φ is computed by the formula

$$\varphi = \frac{2 f_{pq}^{(k)}}{\max\left(2|f_{pq}^{(k)}|, M_{qp}^{(k)}\right)}, \tag{2.3}$$

where

$$\begin{aligned} M_{qp}^{(k)} &= 2 \sum_{j \neq q} \left(\left(a_{jq}^{(k)}\right)^2 + \left(b_{pj}^{(k)}\right) \right)^2 + 2 \sum_{j \neq p} \left(\left(a_{pj}^{(k)}\right)^2 + \left(d_{qj}^{(k)}\right)^2 \right) \\ &+ 2 \left(a_{pp}^{(k)} - a_{qq}^{(k)}\right)^2 + 4 \left(b_{pq}^{(k)}\right)^2 + 4 \left(d_{pq}^{(k)}\right)^2 \\ &+ \tau^2 \left(\left(b_{pp}^{(k)}\right)^2 + \left(b_{qq}^{(k)}\right)^2 + \left(d_{pp}^{(k)}\right)^2 + \left(d_{qq}^{(k)}\right)^2 + 2 \left(a_{pq}^{(k)}\right)^2 + \left(a_{qp}^{(k)}\right)^2 \right). \end{aligned}$$

A.2. $|e_{pq}^{(k)}|^{1/2} = c^{(k)} \geq h^{(k)}$, $1 \leq p = p_k < q = q_k \leq n$, $\varphi_k = \varphi$.

Then the matrix $U = U_{pq}(\varphi)$ is of the form

$$U = \begin{pmatrix} I_n & S_{pq}(\varphi) \\ 0 & I_n \end{pmatrix}, \tag{2.4}$$

where $S_{pq}(\varphi) \in \mathbb{R}^{n\times n}$ is the matrix

$$S_{pq}(\varphi) = (s_{ij}) = \begin{cases} s_{pq} = \varphi, \\ s_{ij} = 0, \quad (i,j) \notin \{(p,q)\}. \end{cases}$$

In this case the parameter φ is computed by the formula

$$\varphi = \frac{2\, e_{pq}^{(k)}}{\max\left(2|e_{pq}^{(k)}|, M_{qp}^{(k)}\right)}, \tag{2.5}$$

where

$$\begin{aligned} M_{qp}^{(k)} &= 2\sum_j \left(\left(d_{qj}^{(k)}\right)^2 + \left(d_{pj}^{(k)}\right)\right)^2 + 2\sum_{j\neq q}\left(a_{jq}^{(k)}\right)^2 2\sum_{j\neq p}\left(a_{jp}^{(k)}\right)^2 \\ &+ 2\left(a_{pp}^{(k)} + a_{qq}^{(k)}\right)^2 + 4\left(a_{pq}^{(k)}\right)^2 + 4\left(a_{qp}^{(k)}\right)^2 \\ &+ 2\,\tau^2\left(\left(b_{pp}^{(k)}\right)^2 + \left(b_{qq}^{(k)}\right)^2 + \left(b_{pq}^{(k)}\right)^2 + \left(d_{pp}^{(k)}\right)^2 + \left(d_{qq}^{(k)}\right)^2 + \left(d_{pq}^{(k)}\right)^2\right). \end{aligned}$$

A.3. $|e_{pp}^{(k)}|^{1/2} = c^{(k)} \geq h^{(k)}$, $1 \leq p = p_k = q = q_k \leq n$, $\varphi_k = \varphi$.
Then the matrix $U = U_{pq}(\varphi)$ is of the form

$$U = \begin{pmatrix} I_n & S_p(\varphi) \\ 0 & I_n \end{pmatrix}, \tag{2.6}$$

where

$$S_p(\varphi) = \mathrm{diag}[I_{p-1}, \varphi, I_{n-p}]$$

and φ is computed by

$$\varphi = \frac{e_{pp}^{(k)}}{\max\left(|e_{pp}^{(k)}|, M_{pp}^{(k)}\right)} \tag{2.7}$$

and

$$M_{pp}^{(k)} = 2\sum_j \left(d_{pj}^{(k)}\right)^2 + 2\sum_{j\neq p}\left(a_{jp}^{(k)}\right)^2 + 4\left(a_{pp}^{(k)}\right)^2 + \tau^2\left(\left(b_{pp}^{(k)}\right)^2 + \left(d_{pp}^{(k)}\right)^2\right).$$

A.4. $|a_{pq}^{(k)} + a_{qp}^{(k)}| = h^{(k)} > c^{(k)}$, $1 \leq p = p_k < q = q_k \leq n$, $\varphi = \varphi_k$.
In this case we choose the matrix $U_k = U_{pq}(\varphi)$,

$$U_{pq}(\varphi) = \mathrm{diag}[T_{pq}(\varphi), T_{pq}(\varphi)], \tag{2.8}$$

where $T_{pq}(\varphi) \in \mathbb{R}^{n\times n}$ is a matrix of the form

$$T_{pq}(\varphi) = (t_{\beta\gamma}) = \begin{cases} t_{pp} = t_{qq} = \cos\ \varphi, \\ t_{pq} = -t_{qp} = -\sin\ \varphi, \\ t_{\beta\gamma} = \delta_{\beta\gamma}, \quad (\beta,\gamma) \notin \{(p,p),(p,q),(q,p),(q,q)\}. \end{cases}$$

The parameter φ is computed from the equation

$$\mathrm{tg}(2\varphi) = \frac{a_{pq}^{(k)} + a_{qp}^{(k)}}{a_{pp}^{(k)} - a_{qq}^{(k)}}.$$

A.5. $|b_{pq}^{(k)} + b_{qp}^{(k)}| = h^{(k)} > c^{(k)}$, $1 \le p = p_k < q = q_k \le n$, $\varphi = \varphi_k$.
We choose the matrix $U_k = U_{pq}(\varphi)$,

$$U_{pq}(\varphi) = \begin{pmatrix} C & -S \\ S & C \end{pmatrix}, \tag{2.9}$$

where $C,\ S \in \mathbb{R}^{n\times n}$ and

$$\begin{aligned} C &= \mathrm{diag}[I_{p-1}, \cos\varphi, I_{q-p}, \cos\varphi, I_{m-q}], \\ S &= (s_{\beta\gamma}) = \begin{cases} s_{pq} = s_{qp} = \sin\varphi, \\ s_{\beta\gamma} = 0, \quad (\beta,\gamma) \notin \{(p,q),(q,p)\}. \end{cases} \end{aligned}$$

In this case the parameter φ is computed from the equation

$$\mathrm{tg}(2\varphi) = \frac{b_{pq}^{(k)} + d_{qp}^{(k)}}{a_{pp}^{(k)} + a_{qq}^{(k)}}.$$

A.6. $|b_{pp}^{(k)} + b_{pp}^{(k)}| = h^{(k)} > c^{(k)}$, $1 \le p = p_k = q = q_k \le n$, $\varphi = \varphi_k$.
The matrix $U_k = U_{pq}(\varphi)$ has the form

$$U_{pq}(\varphi) = \begin{pmatrix} C & -S \\ S & C \end{pmatrix}, \tag{2.10}$$

where $C,\ S \in \mathbb{R}^{n\times n}$ and

$$\begin{aligned} C &= \mathrm{diag}[I_{p-1}, \cos\varphi, I_{m-p}], \\ S &= (s_{\beta\gamma}) = \begin{cases} s_{pp} = \sin\varphi, \\ s_{\beta\gamma} = 0, \quad (\beta,\gamma) \notin \{(p,p)\}, \end{cases} \end{aligned}$$

with the parameter φ computed now from the equation

$$\mathrm{tg}(2\varphi) = \frac{b_{pp}^{(k)} + d_{pp}^{(k)}}{a_{pp}^{(k)} + a_{pp}^{(k)}}.$$

We shall prove now the following

Lemma 2.1. Let p, q, $1 \le p \le q \le n$, and $H_{k+1} = U^{-1} H_k U$, where $U = U_{pq}(\varphi)$ is given by (2.2), (2.4) or (2.6); the parameter φ is computed by means of Eqs. (2.3), (2.5) or (2.7), respectively. Then

$$\|H_k\|^2 - \|H_{k+1}\|^2 \ge \frac{2}{\tau^2} \frac{(\tau^2 - \tau - 1)(c^{(k)})^4}{\max(2(c^{(k)})^2, M_{qp}^{(k)})} \ge \frac{1}{\tau^2} \frac{(c^{(k)})^4}{\|H_k\|^2}, \tag{2.11}$$

where $\tau \ge (1 + \sqrt{5})/2$.

Proof. Let us choose p, q so that $1 \le p < q \le n$. We shall provide a detailed proof of the lemma in the case **A.1.** only, since the reasoning in the rest of the cases is fully similar.

We compute φ from (2.3), choose the matrix U from (2.2) and construct $H_{k+1} = U^{-1} H_k U$. Then for $\Delta(\varphi)$ we get

$$\begin{aligned} \Delta(\varphi) &= \|H_k\|^2 - \|H_{k+1}\|^2 \\ &= -G\varphi^4 - Q\varphi^3 - W\varphi^2 + 4f_{pq}^{(k)}\varphi, \end{aligned}$$

where

$$\begin{aligned} G &= 2\left(a_{pq}^{(k)}\right)^2 + \left(b_{pp}^{(k)}\right)^2 + \left(d_{qq}^{(k)}\right)^2, \\ Q &= 4a_{pq}^{(k)}\left(a_{pp}^{(k)} - a_{qq}^{(k)}\right) - 4b_{pp}^{(k)}b_{pq}^{(k)} + 4d_{qq}^{(k)}d_{pq}^{(k)}, \\ W &= 2\sum_{i \ne q}\left(\left(a_{jq}^{(k)}\right)^2 + \left(b_{pj}^{(k)}\right)^2\right) + 2\sum_{j \ne p}\left(\left(a_{pj}^{(k)}\right)^2 + \left(d_{qj}^{(k)}\right)^2\right) \\ &+ \left(a_{pp}^{(k)} - a_{qq}^{(k)}\right)^2 + 4\left(b_{pq}^{(k)}\right)^2 + 4\left(d_{pq}^{(k)}\right)^2 \\ &+ 2b_{pp}^{(k)}b_{qq}^{(k)} + 2d_{pp}^{(k)}d_{qq}^{(k)} - 4a_{pq}^{(k)}a_{qp}^{(k)}. \end{aligned}$$

We find

$$\frac{|W|}{\max\left(2|f_{pq}^{(k)}|, M_{qp}^{(k)}\right)} \le \frac{|W|}{M_{qp}^{(k)}} \le 1.$$

Consider the inequality

$$2xy \le \frac{t^4 x^2 + y^2}{t^2} \tag{2.12}$$

which holds for all real numbers t, x, y; setting, in particular, $t = \sqrt{\tau}$, $x = \sqrt{2}|a_{pq}^{(k)}|$, $y = \sqrt{2}|a_{pp}^{(k)} - a_{qq}^{(k)}|$, we obtain

$$4a_{pq}^{(k)}\left(a_{pp}^{(k)} - a_{qq}^{(k)}\right) \le \frac{\tau^2 2\left(a_{pq}^{(k)}\right)^2 + 2\left(a_{pp}^{(k)} - a_{qq}^{(k)}\right)^2}{\tau}.$$

Similarly, we get

$$4b_{pp}^{(k)}b_{pq}^{(k)} \leq \frac{\tau^2\left(b_{pp}^{(k)}\right)^2 + 4\left(b_{pq}^{(k)}\right)^2}{\tau}.$$

Thus the expression $|Q|$ becomes

$$|Q| \leq \frac{\tau^2 2\left(a_{pq}^{(k)}\right)^2 + 2\left(a_{pp}^{(k)} - a_{qq}^{(k)}\right)^2}{\tau}$$

$$+ \frac{\tau^2\left(b_{pp}^{(k)}\right)^2 + 4\left(b_{pq}^{(k)}\right)^2}{\tau} + \frac{\tau^2\left(d_{qq}^{(k)}\right)^2 + 4\left(d_{pq}^{(k)}\right)^2}{\tau}.$$

Consequently,

$$\frac{|Q|}{\max\left(2|f_{pq}^{(k)}|, M_{qp}^{(k)}\right)} \leq \frac{|Q|}{M_{qp}^{(k)}} \leq \frac{1}{\tau}.$$

Similarly, we have

$$\frac{|G|}{\max\left(2|f_{pq}^{(k)}|, M_{qp}^{(k)}\right)} \leq \frac{|G|}{M_{qp}^{(k)}} \leq \frac{2\left(a_{pq}^{(k)}\right)^2 + \left(b_{pp}^{(k)}\right)^2 + \left(d_{qq}^{(k)}\right)^2}{M_{qp}^{(k)}} \leq \frac{1}{\tau^2}.$$

Using the above inequalities, we obtain

$$\begin{aligned}
\Delta(\varphi) &\geq 4f_{pq}^{(k)}\varphi - |W|\varphi^2 - |Q|\varphi^3 - |G|\varphi^4 \\
&\geq 2\varphi^2 \max\left(2|f_{pq}^{(k)}|, M_{qp}^{(k)}\right) - |W|\varphi^2 - |Q|\varphi^2 - |G|\varphi^2 \\
&= \max\left(2|f_{pq}^{(k)}|, M_{qp}^{(k)}\right)\left(2\varphi^2 - \frac{|W| + |Q| + |G|}{\max\left(2|f_{pq}^{(k)}|, M_{qp}^{(k)}\right)}\varphi^2\right) \\
&\geq \max\left(2|f_{pq}^{(k)}|, M_{qp}^{(k)}\right)\left(2\varphi^2 - (1 + \frac{1}{\tau} + \frac{1}{\tau^2})\varphi^2\right) \\
&= \max\left(2|f_{pq}^{(k)}|, M_{qp}^{(k)}\right)\left(\frac{\tau^2 - \tau - 1}{\tau^2}\right)\varphi^2
\end{aligned}$$

Then

$$\Delta(\varphi) \geq \frac{\tau^2 - \tau - 1}{\tau^2} \frac{4(f_{pq}^{(k)})^2}{\max\left(2|f_{pq}^{(k)}|, M_{qp}^{(k)}\right)}.$$

Since

$$\max\left(2|f_{pq}^{(k)}|, M_{qp}^{(k)}\right) < 4\left(\tau^2 - \tau - 1\right)||H_k||^2, \quad \tau > (1 + \sqrt{5})/2,$$

and therefore

$$\Delta(\varphi) \geq \frac{1}{\tau^2} \frac{(f_{pq}^{(k)})^2}{\|H_k\|^2} = \frac{1}{\tau^2} \frac{(c^{(k)})^4}{\|H_k\|^2}.$$

This completes the proof of the case **A.1**. The proofs of cases **A.2** and **A.3** are fully similar, as already pointed out. □

Lemma 2.2. Let p, q be natural numbers, $1 \leq p \leq q \leq n$, and $H_{k+1} = U^{-1} H_k U$, where H_k is a matrix from the sequence (2.1). Let the matrix $U = U_{p_k q_k}(\varphi_k)$ be given by (2.2), (2.4) or (2.6), and φ be computed by (2.3), (2.5) or (2.7), respectively. Then

$$|h_{rs}^{(k+1)} - h_{rs}^{(k)}| \leq 4|c_{pq}^{(k)}|^2$$

for $r, s = 1, \ldots, 2n$.

Proof. Let p, q be integers, $1 \leq p < q \leq n$, and let U be a matrix of the type (2.2) with φ computed by means of (2.3). Then

$$|h_{rs}^{(k+1)} - h_{rs}^{(k)}| = \begin{cases} |a_{rs}^{(k+1)} - a_{rs}^{(k)}|, & 1 \leq r, s \leq n, \\ |b_{rs-n}^{(k+1)} - b_{rs-n}^{(k)}|, & 1 \leq r \leq n,\ n+1 \leq s \leq 2n, \\ |d_{r-ns}^{(k+1)} - d_{r-ns}^{(k)}|, & n+1 \leq r \leq 2n,\ 1 \leq s \leq n, \\ |-a_{r-ns-n}^{(k+1)} + a_{r-ns-n}^{(k)}|, & n+1 \leq r, s \leq 2n. \end{cases}$$

For the expression $|a_{rs}^{(k+1)} - a_{rs}^{(k)}|$ we obtain in turn

$$|a_{rs}^{(k+1)} - a_{rs}^{(k)}| = \begin{cases} |a_{rp}^{(k)} + \varphi a_{rq}^{(k)} - a_{rp}^{(k)}|, & r = 1, \ldots, n, r \neq q, \\ |a_{qs}^{(k)} - \varphi a_{ps}^{(k)} - a_{qs}^{(k)}|, & s = 1, \ldots, n, s \neq p, \\ |a_{qp}^{(k)} - \varphi \left(a_{pp}^{(k)} - a_{qq}^{(k)} \right) - \varphi^2 a_{pq}^{(k)} - a_{qp}^{(k)}|, & r = p,\ s = q, \\ 0, & \text{otherwise.} \end{cases}$$

Hence

$$\begin{aligned} |a_{rp}^{(k+1)} - a_{rp}^{(k)}| &\leq |\varphi a_{rq}^{(k)}| \\ &\leq \frac{2|f_{pq}^{(k)}|}{\max\left(2|f_{pq}^{(k)}|, M_{qp}^{(k)}\right)} |a_{rq}^{(k)}| \\ &\leq 2|f_{pq}^{(k)}| = 2|c_{pq}^{(k)}|^2 \end{aligned}$$

for $r = 1, \ldots, n$ and $r \neq q$.

In the same manner it can be shown that

$$\begin{aligned}
|a_{qs}^{(k+1)} - a_{qs}^{(k)}| &\leq 2|f_{pq}^{(k)}| = 2|c_{pq}^{(k)}|^2, \qquad s = 1, \ldots, n, \;\; s \neq p, \\
|a_{qp}^{(k+1)} - a_{qp}^{(k)}| &\leq |\varphi||a_{pp}^{(k)} - a_{qq}^{(k)}| + |\varphi|^2|a_{pq}^{(k)}| \\
&\leq |\varphi| \left(|a_{pp}^{(k)} - a_{qq}^{(k)}| + |a_{pq}^{(k)}|\right) \\
&\leq 4|f_{pq}^{(k)}| = 4|c_{pq}^{(k)}|^2
\end{aligned}$$

and

$$\begin{aligned}
|b_{rs-n}^{(k+1)} - b_{rs-n}^{(k)}| &\leq 4|c_{pq}^{(k)}|^2, \\
|d_{r-ns}^{(k+1)} - d_{r-ns}^{(k)}| &\leq 4|c_{pq}^{(k)}|^2.
\end{aligned}$$

This completes the proof of the case **A.1**. The proofs of cases **A.2.** and **A.3.** are again similar and omitted. □

Theorem 2.3. The iterative method (2.1) has the following properties:

I. $C(H_k) \to 0, \quad k \to \infty$.

II. The symmetric matrix $\frac{1}{2}(H_k + H_k^T)$ tends to the diagonal matrix $\frac{1}{2}(H_0 + H_0^T)$, where $H_0 = H(A_0, B_0, D_0) = (h_{rs}^{(0)})$, and

$$\frac{1}{2}(H_0 + H_0^T) = \operatorname{diag}[h_{11}^{(0)}, \ldots, h_{2n2n}^{(0)}],$$

where $h_{ii}^{(0)}$ are the real parts of the eigenvalues of the matrix H.

III. Let p, q be natural numbers, $1 \leq p \neq q \leq 2n$, and $h_{pp}^{(0)} \neq h_{qq}^{(0)}$. Then

$$h_{pq}^{(k)} \to 0, \quad k \to \infty.$$

IV. Let p, q be natural numbers, $1 \leq p \neq q \leq 2n$, $h_{pp}^{(0)} = h_{qq}^{(0)}$, and for each t, $1 \leq t \leq 2n$, $t \neq p, q$, we have $h_{tt}^{(0)} \neq h_{pp}^{(0)}$. Then

$$h_{pq}^{(k)} \to h_{pq}^{(0)}, \quad k \to \infty,$$

where $h_{pq}^{(0)}$ is the imaginary part of the eigenvalues of H with real part $h_{pp}^{(0)}$.

Proof. I. We consider the sequence $\|H_k\|^2$, $k = 1, 2, 3, \ldots$ The similarity transformations with matrices of the form (2.8), (2.9) and (2.10) preserve the Euclidean norm and the similarity transformations with matrices of the form (2.2), (2.4) and (2.6) decrease or preserve the Euclidean norm implying that the sequence $\|H_1\|^2, \|H_2\|^2, \ldots$ is monotonically decreasing. Let for each matrix H_k a number α_k be introduced such that

$$\alpha_k = \begin{cases} 0, & \text{if } U_k \text{ is of the form (2.8), (2.9), (2.10),} \\ 1, & \text{if } U_k \text{ is of the form (2.2), (2.4), (2.6).} \end{cases}$$

The sequence of matrices $\{H_k\}$ is bounded. From this sequence we choose a convergent subsequence $\{H_s\}$, where $s \in S \subset N$ and N is the set of natural numbers. Suppose that $\{\alpha_s\}$ contains an infinite number of ones and $\{H_m\}$ is such a subsequence of $\{H_s\}$ that for each $m \in M \subset S \subset N$ we have $\alpha_m = 1$. Then from Lemma 1 it follows

$$\|H_m\|^2 - \|H_{m+1}\|^2 \geq \frac{1}{\tau^2} \frac{(c^{(m)})^4}{\|H\|^2}.$$

Hence $c^{(m)} \to 0$, $m \to \infty$. From the inequality $h^{(m)} \leq c^{(m)}$ it follows that $h^{(m)} \to 0$, $m \to \infty$.

Let $H_0 = H(A_0, B_0, D_0) = (h_{rs}^{(0)})$ be a limit of the sequence $\{H_m\}$. For $H_0 + H_0^T$ and $C(H_0)$ we obtain

$$\begin{aligned} H_0 + H_0^T &= H(A_0 + A_0^T, B_0 + D_0^T, D_0 + B_0^T) = (h_{rs}^{(0)} + h_{sr}^{(0)}), \\ C(H_0) &= H(F_0, E_0, E_0) = (c_{rs}^{(0)}), \end{aligned}$$

where

$$\begin{aligned} F_0 &= F_0^T = A_0 A_0^T + B_0 B_0 - A_0^T A_0 - D_0 D_0 = (f_{\beta\gamma}^{(0)}), \\ E_0 &= E_0^T = A_0 D_0 - B_0 A_0 - A_0^T B_0 + D_0 A_0^T = (e_{\beta\gamma}^{(0)}). \end{aligned}$$

Since H_0 is a limit, then if $r \neq s$, we have $h_{rs}^{(0)} + h_{sr}^{(0)} = 0$ and $c_{rs}^{(0)} = 0$. From $h_{rs}^{(0)} + h_{sr}^{(0)} = 0$, $r \neq s$, it follows that

$$a_{\beta\gamma}^{(0)} + a_{\gamma\beta}^{(0)} = 0, \quad \beta \neq \gamma, \tag{2.13}$$

$$b_{\beta\gamma}^{(0)} - d_{\gamma\beta}^{(0)} = 0, \quad \beta, \gamma = 1, \ldots, n. \tag{2.14}$$

For the elements $f_{\beta\gamma}^{(0)}$ of $C(H_0)$ we obtain

$$f_{\beta\gamma}^{(0)} = \sum_j \left(a_{j\beta}^{(0)} a_{j\gamma}^{(0)} - a_{\beta j}^{(0)} a_{\gamma j}^{(0)} + d_{\beta j}^{(0)} d_{j\gamma}^{(0)} - b_{\beta j}^{(0)} b_{j\gamma}^{(0)} \right).$$

From (2.13), (2.14) and $c_{rs}^{(0)} = 0$ we compute

$$f_{\beta\gamma}^{(0)} = 2a_{\beta\gamma}^{(0)} \left(a_{\beta\beta}^{(0)} - a_{\gamma\gamma}^{(0)} \right) = 0. \tag{2.15}$$

Consequently,

$$f_{\beta\beta}^{(0)} = 0, \quad \beta = 1, \ldots, n.$$

Similarly, we find

$$e_{\beta\gamma}^{(0)} = 2b_{\beta\gamma}^{(0)} \left(a_{\beta\beta}^{(0)} + a_{\gamma\gamma}^{(0)} \right) = 0, \quad \beta, \gamma = 1, \ldots, n. \tag{2.16}$$

Hence $c_{rs}^{(0)} = 0$ for each r and s, i.e. H_0 is a normal matrix and $H_0 + H_0^T$ is a diagonal matrix. Then the diagonal elements $h_{11}^{(0)}, \ldots, h_{2n\,2n}^{(0)}$ are the real parts of eigenvalues of H.

For the subsequence $\{H_m\}$ we have $C(H_m) \to 0$, $m \to \infty$, where $\|H_m\|^2 \to 2\sum_{j=1}^{n} |\nu_j|^2$, $\nu_j = \lambda_j + \mathrm{i}\mu_j$. Since the whole sequence $\|H_k\|$ is non-increasing, it follows that $\|H_k\| \to 2\sum_{j=1}^{n} |\nu_j|^2$, i.e. $C(H_k) \to 0$, $k \to \infty$. Hence the sequence H_k tends to a normal matrix.

Let the sequence $\{\alpha_k\}$ contains only a finite number of ones. Then there is a natural number s_0 and for each s, so that $s \geq s_0$, $\alpha_s = 0$, i.e. the matrices U_s are of the form (2.8) or (2.9) or (2.10). Then

$$h^{(s)} \to 0, \ s \to \infty.$$

Since $c^{(s)} \leq h^{(s)}$, then

$$c^{(s)} \to 0, \ s \to \infty.$$

Hence the convergent subsequence H_s has a limit $H_0 = H(A_0, B_0, D_0)$ with the properties (2.13) – (2.16). This proves I.

II. We will prove that $h^{(k)} \to 0$, $k = 1, 2, \ldots$, for the sequence $\{H_k\}$. In proving I we have found a subsequence $\{H_s\}$ of H_k. We consider the case when the sequence $\{\alpha_k\}$ contains an infinite number of both zeros and ones. Let $\{\alpha_p\}$ be a subsequence of $\{\alpha_k\}$. If $\alpha_p = 1$, then $h^{(p)} \to 0$, $p \to \infty$. We consider the sequence of all indices $k_1, \ldots, k_s, \ldots$, so that $\alpha_{k_s} = 0$ and $\alpha_{k_s - 1} = 1$ for $s = 1, 2, \ldots$ In the case $m = k_s$, according to Lemma 2 we obtain

$$\begin{aligned} |h^{(m)}| &\leq |h^{(m-1)}| + |h^{(m)} - h^{(m-1)}| \\ &\leq |h^{(m-1)}| + 8(c^{(m-1)})^4. \end{aligned}$$

Since $c^{(m-1)} \to 0$ and $h^{(m-1)} \to 0$ for $m = k_s$ and $s = 1, 2, \ldots$ $(m - 1 = k_s - 1)$, it follows that $h^{(m)} \to 0$. Let σ_k^2 denote a sum of the squares of off-diagonal elements in blocks of the symmetric matrix $H_k + H_k^T$. Then we have

$$h^{(k)} \leq \sigma_k^2 \leq 2n(2n-1)h^{(k)}.$$

For the subsequence $\{H_m\}$ from $h^{(m)} \to 0$ it follows that $\sigma_m^2 \to 0$.

Consider the indices $m + t$ of $\{\alpha_k\}$. For $m = k_s$ it is true for $\alpha_{m-1} = 1$, $\alpha_{m-1+t} = 0$, for $t = 1, 2, \ldots, p$ and $\alpha_{m+p} = 1$ for $s = 1, 2, \ldots$ For these indices the number sequence σ_{m+t}^2 is monotonically decreasing, because for the matrix $H_{m+t} + H_{m+t}^T$ a step is used from a modification of Jacobi's method for a symmetric Hamiltonian matrix [9].

It thus follows that $\sigma_k^2 \to 0$ for $k = 1, 2, \ldots$, and $h^{(k)} \to 0$ for the same k. Hence from I we obtain that each convergent subsequence of H_k has a limit with the properties (2.13) – (2.16) and its diagonal elements are the real parts of the eigenvalues of H.

III. Now we will prove that if $h_{pp}^{(0)} \neq h_{qq}^{(0)}$, $p \neq q$, then $h_{pq}^{(k)} \to 0$, $k \to \infty$. There are three possible cases.

Let p, q be natural numbers, $1 \leq p \neq q \leq n$. Then $h_{pp}^{(0)} = a_{pp}^{(0)}$ and $h_{qq}^{(0)} = a_{qq}^{(0)}$. Since $h_{pq}^{(k)} \to h_{pq}^{(0)}$, $a_{pp}^{(0)} \neq a_{qq}^{(0)}$, from (2.15) we have $a_{pq}^{(0)} = 0$, i.e. $h_{pq}^{(0)} = 0$.

Let p, q be natural numbers, $1 \le p \le n$, $n+1 \le q \le 2n$. Then $h_{pp}^{(0)} = a_{pp}^{(0)}$, $h_{qq}^{(0)} = -a_{q-nq-n}^{(0)}$, $h_{pq}^{(0)} = b_{pq-n}^{(0)}$. Since $h_{pq}^{(k)} \to h_{pq}^{(0)}$, from (2.16) when $p \neq q-n$ we have

$$e_{pq-n}^{(0)} = 2b_{pq-n}^{(0)} \left(h_{pp}^{(0)} - h_{qq}^{(0)} \right) = 0.$$

Hence $b_{pq-n}^{(0)} = 0$. When $p = q - n$, from (2.16) we obtain

$$e_{pp}^{(0)} = 4b_{pp}^{(0)} \, a_{pp}^{(0)} = 0.$$

If $a_{pp}^{(0)} = 0$, it follows that $h_{pp}^{(0)} = h_{qq}^{(0)}$, because $h_{pp}^{(0)} = a_{pp}^{(0)} = 0$, $h_{qq}^{(0)} = -a_{q-nq-n}^{(0)} = -a_{pp}^{(0)}$. Hence $b_{pp}^{(0)} = 0$.

In the case $n+1 \le p \le 2n$, $1 \le q \le n$, the proof is similar to that of the case $1 \le p \le n$, $n+1 \le q \le 2n$.

IV. Let $\{H_s\}$ be a convergent subsequence of $\{H_k\}$ with the limit

$$H_0 = H(A_0, B_0, D_0) = \left(h_{rs}^{(0)} \right).$$

The limit H_0 possesses the properties (2.13) – (2.16). Let $h_{pp}^{(0)} = h_{qq}^{(0)}$, $p \neq q$, and for each $t \neq p, q$, $h_{tt}^{(0)} \neq h_{pp}^{(0)}$. We choose the number t so that $1 \le t \le 2n$, $t \neq p, q$. Then, according to III, in the rows and columns of H_0 with numbers p, q there will exist only two nonzero off-diagonal elements $h_{pq}^{(0)}$, $h_{qp}^{(0)}$.

If $1 \le p, q \le n$, then the nonzero off-diagonal elements are $a_{pq}^{(0)}$, $a_{qp}^{(0)}$. Consequently, $a_{pp}^{(0)} \pm \mathtt{i}\, a_{pq}^{(0)}$ and $-a_{pp}^{(0)} \pm \mathtt{i}\, a_{pq}^{(0)}$ are eigenvalues of the Hamiltonian matrix H.

If $1 \le p \le n$, $n+1 \le q \le 2n$, $p \neq q-n$, then the nonzero off-diagonal elements are $b_{pq-n}^{(0)}$, $d_{q-np}^{(0)} = -b_{pq-n}^{(0)}$. This implies that $a_{pp}^{(0)} \pm \mathtt{i}\, b_{pq-n}^{(0)}$ are eigenvalues of the Hamiltonian matrix H. From the type of H_0 follows that $h_{q-nq-n}^{(0)} = h_{p+np+n}^{(0)}$. Hence $a_{q-nq-n}^{(0)} \pm \mathtt{i}\, b_{q-np}^{(0)}$ are eigenvalues of the Hamiltonian matrix H.

If $1 \le p \le n$, $n+1 \le q \le 2n$, $p = q - n$, from the type H_0 it follows that $h_{pp}^{(0)} = h_{qq}^{(0)} = 0$. Then the nonzero elements are $b_{pp}^{(0)}$, $d_{pp}^{(0)} = -b_{pp}^{(0)}$, and $\pm \mathtt{i}\, b_{pp}^{(0)}$ are eigenvalues of the Hamiltonian matrix H. □

3. APPLICATIONS AND NUMERICAL EXPERIMENTS

Numerical experiments for solving the spectral problem for Hamiltonian matrices and for numerical computing of the solution of the algebraic Riccati equation are performed and will be reported in this section. All numerical experiments were made on a PENTIUM computer using the algorithmic language *Turbo Pascal* and the real arithmetic having an 11 sedecimal digit mantissa. The code of our algorithm uses a cyclic choice on the pivot indices (p, q).

The presented method for computing the spectral problem of a Hamiltonian matrix is the Jacobi type method for solving the eigenproblem of real non-symmetric matrices. The reason is that in the Jacobi method for finding the eigenvalues only two rows and columns are involved in each iteration step of our method. The parallel implementation of our algorithm can be followed of those for Jacobi algorithm for symmetric eigenvalue on the hypercube or a linear array of processors [4] and on distributed memory multiprocessors [5].

3.1. THE SPECTRAL PROBLEM FOR HAMILTONIAN MATRICES

The code of our algorithm computes the eigenvalues of an $(n \times n)$-Hamiltonian matrix $H = H(A, B, D)$. Let us denote $\varepsilon = \max_i |\lambda_i - \tilde{\lambda}_i|$, where λ_i are the exact eigenvalues and $\tilde{\lambda}_i$ are the computed eigenvalues obtained by our algorithm.

Example 1 [1]. Consider the matrix

$$H = U \begin{pmatrix} A & 0 \\ 0 & -A^T \end{pmatrix} U^T, \tag{3.1}$$

where $A \in \mathbb{R}^{n \times n}$,

$$A = (a_{ij}) = \begin{cases} a_{11} = -10^3, \\ a_{ii} = n + 1 - i, \quad i = 2, \ldots, n-2, \\ a_{n-1n-1} = a_{nn} = 2, \\ a_{n-1n} = -a_{nn-1} = 1, \\ a_{ij} = 0, \quad \text{otherwise}, \end{cases}$$

and the matrix U is the product of randomly generated symplectic matrices of the form (2.2), (2.4), (2.8). The following results are obtained in this case:

TABLE 1

n	5	10	15	20	25
ε	$1.3245E-7$	$4.2331E-7$	$2.1289E-7$	$1.5673E-7$	$5.3289E-6$

Example 2. Consider the matrix (3.1), where A is a diagonal matrix with randomly chosen elements and the matrix U is the product of randomly generated symplectic matrices of the form (2.2), (2.4), (2.8). The results are shown in the following table:

TABLE 2

n	5	10	15	20	25
ε	$2.5463E-10$	$1.3568E-10$	$6.8452E-8$	$3.4562E-8$	$1.2344E-6$

Example 3. We have executed numerical experiments of random strict diagonal dominant Hamiltonian matrices using Byers' algorithm [2] and the algorithm proposed here. Table 3 displays the average number of sweeps necessary for convergence. Each trial includes 10 matrices of the different dimensions.

TABLE 3

n	Byers' algorithm	Our algorithm
10	14	12
15	15	15
20	18	16
30	19	16

We compare Byers' method with our method. Byers' algorithm computes $2n^2$ similarity transformations per sweep, the method proposed here computes $n(n+1)/2$ similarity transformations. Byers' algorithm makes $32n^3 + O(n^2)$ flops for computing the $2n^2$ transformations. Our algorithm makes $20n^3 + O(n^2)$ flops for computing the $n(n+1)/2$ transformations. Hence we obtain that one sweep of the Byers' algorithm is more expensive than a sweep of the algorithm proposed here. Our algorithm is faster than Byers' algorithm for the above set of examples (Example 3). Moreover, Byers' method uses complex arithmetic, while in our method real arithmetic is solely utilized.

In the case of a symmetric Hamiltonian matrix our method uses similarity transformations with orthogonal symplectic matrices of the form (2.2), (2.4), (2.6). The amount of work for performing the transformations per sweep is $12n^3 + O(n^2)$ flops. We have made numerical experiments for randomly generating symmetric Hamiltonian matrices for the same dimensions as in Example 3. We have obtained that the average number of sweeps of the method proposed here is equal to the average number of sweeps of Byers' method.

3.2. NUMERICAL SOLUTION OF THE ALGEBRAIC RICCATI EQUATION

The algorithm for computing the eigenvalues and eigenvectors of a real Hamiltonian matrix presented here can be successfully used to calculate the solution of the Riccati equation

$$L(X) = XBX - XA - A^T X - D = 0, \tag{3.2}$$

where $A \in \mathbb{R}^{n\times n}$, $B = B^T \in \mathbb{R}^{n\times n}$, $D = D^T \in \mathbb{R}^{n\times n}$ and B is a positive definite matrix, D is a positive semidefinite matrix.

The computation of the solution P of (3.2) leads to the solving of the spectral problem for the Hamiltonian matrix $H = H(A, -B, -D)$. An algorithm for computing the solution P is described in [8, 10]. The matrix $H = H(A, -B, -D)$ is reduced in Schur's form $\tilde{H}$ with the QR-algorithm

$$U^T H U = \tilde{H}, \quad U = \begin{pmatrix} U_{11} & U_{12} \\ U_{21} & U_{22} \end{pmatrix}.$$

For the solution P of the equation (3.2) we have $P = U_{21}U_{11}^{-1}$ [8, 10].

We propose the following algorithm for solving the algebraic Riccati equation (3.2). We compute the eigenvalues and eigenvectors of the Hamiltonian matrix $H = H(A, -B, -D)$ with the algorithm described in Section 2. The matrix U of the eigenvectors of H is partitioned into four $(n \times n)$-blocks

$$U = \begin{pmatrix} U_{11} & U_{12} \\ U_{21} & U_{22} \end{pmatrix}$$

and then we compute the solution $P = U_{21}U_{11}^{-1}$ of the matrix equation (3.2).

We have made numerical experimens for computing the solution of the equation (3.2). For these experiments three algorithms have been used. The first algorithm $W1$ uses the QR-method [8, 10]. The second algorithm $W2$ uses the iterative method described by Petkov and Ivanov [13]. This method computes the eigenvalues and eigenvectors of $H = H(A, -B, -D)$ and then the solution $P = U_{21}U_{11}^{-1}$ is found. The third algorithm $W3$ uses the iterative method wich solves the spectral problem of a Hamiltonian matrix. In the programs of algorithms $W2$ and $W3$ we use a cyclic choice on the pivot indices (p, q).

The matrices A, B, D, for which the solution of the equation (3.2) is computed, are the matrices from Example 1 and Example 2 from Section 3.1 and the examples described below. On each trial we compute the accuracy of the computed solution — $\|L(P)\|_\infty$.

The results from Example 1 and Example 2 are given in Table 4.

TABLE 4

Example	$W1$	$W2$	$W3$
Example 1 $n = 5$ $\|L(P)\|_\infty$	$1.0862E-7$	$1.8703E-7$	$6.2748E-8$
Example 1 $n = 10$ $\|L(P)\|_\infty$	$3.3222E-7$	$4.1609E-7$	$3.3191E-7$
Example 1 $n = 20$ $\|L(P)\|_\infty$	$7.9954E-4$	$3.1569E-4$	$1.4978E-4$
Example 2 $n = 5$ $\|L(P)\|_\infty$	$2.8741E-10$	$5.2502E-9$	$1.2554E-9$
Example 2 $n = 10$ $\|L(P)\|_\infty$	$2.3272E-5$	$5.0117E-8$	$2.6402E-7$
Example 2 $n = 20$ $\|L(P)\|_\infty$	$7.5153E-7$	$1.3549E-7$	$2.0637E-8$

Example 4. The blocks A, B and D in the Riccati equation are of the type

$$\begin{aligned} A &= (a_{ij}) = \begin{cases} ij, & \text{if } i = j, \\ i + j, & \text{if } i \neq j, \end{cases} \\ B &= \operatorname{diag}[1, 2^2, \ldots, n^2], \\ D &= \operatorname{diag}[1, 2, \ldots, n]. \end{aligned}$$

The results from this example are shown in the following table:

TABLE 5

n	$W1$	$W2$	$W3$
5 $\|\|L(P)\|\|_\infty$	$6.7767E-5$	$6.9849E-9$	$6.9028E-8$
10 $\|\|L(P)\|\|_\infty$	$1.3256E-3$	$1.6763E-8$	$2.5378E-8$
20 $\|\|L(P)\|\|_\infty$	$3.4268E-3$	$1.6938E-7$	$1.2096E-7$

Example 5 (Example 5 in [10]). We compute the solution of the Riccati equation with $n = 5, 10, 20$. The results are shown in the following table:

TABLE 6

n	$W1$	$W2$	$W3$
5 $\|\|L(P)\|\|_\infty$	$2.7048E-8$	$9.8542E-7$	$9.8556E-8$
10 $\|\|L(P)\|\|_\infty$	$2.1153E-8$	$1.1726E-9$	$1.3171E-9$
20 $\|\|L(P)\|\|_\infty$	$1.2149E-4$	$5.0361E-9$	$5.1435E-9$

Example 6 (Example 6 in [10]). We compute the solution of the Riccati equation with $n = 21$ and $q = r = 1$. For the correct results $x_{1n} = 1$ we receive the value $x_{1n} = 1.0792769258E+00$.

There are examples (Example 4 and Example 5) for which the iterative methods $W2$ and $W3$ for computing the solution of the Riccati equation are more accurate than the QR-method ($W1$).

4. CONCLUSION

We have presented and investigated a new method for solving the spectral problem of Hamiltonian matrices. The method is a generalization of the Jacobi-like method for arbitrary real matrices, as proposed by Veselić [15]. It allows us to construct a new algorithm for solving the algebraic Riccati equation. Our method preserves the special structure of a Hamiltonian matrix and uses less memory than the algorithm $W1$ (QR-method). The method offers simpler computational schemes and gives better options for parallel modifications.

We note finally that the algorithm proposed here can be modified as well for solving the spectral problem for a symplectic matrix. But we were not able to prove a convergence theorem in this case.

Acknowledgements. This work is partially supported under Grant No MM 521/95 by the Bulgarian Ministry of Education and Sciences.

REFERENCES

1. Bener, P., H. Faßbender. A restarted symplectic Lanczos method for the Hamiltonian eigenvalue problem (preprint). 1995.
2. Byers, R. A Hamiltonian Jacobi algorithm. *IEEE Trans. Aut. Control*, **35**, 1990, 566–570.
3. Eberlein, P. A Jacobi-like method for the automatic computation of eigenvalues and eigenvectors of an arbitrary matrix. *SIAM*, **10**, 1962, 74–88.
4. Eberlein, P. On one-sided Jacobi methods for parallel computation. *SIAM J. Algebraic Discrete Methods*, **8**, 1987, 790–796.
5. Eberlein, P., H. Park. Efficient implementation of Jacobi algorithms and Jacobi sets on distributed memory architectures. *J. of Parallel and Distributed Comp.*, **8**, 1990, 358–366.
6. Engwerda, J. On the existence of a positive definite solution of the matrix equation $X + A^T X^{-1} A = I$. *Linear Algebra Appl.*, **194**, 1993, 91–108.
7. Ikramov, H. Numerical solution of matrix equations and symplectic matrix algebra. *Computation processes and systems*, **5**, Moscow, 1987, 154-163 (in Russian).
8. Ikramov, H. Numerical solution of matrix equations. Science, Moskow, 1984 (in Russian).
9. Ivanov, I. Numerical solution of the eigenvalue problem of Hermitian Hamiltonian matrices. *Mathematics and Education in Mathematics*, Sofia, 1992, 60–65.
10. Laub, Al. A Schur method for solving algebraic Riccati equations. *IEEE Trans. Automat. Control.*, **23**, 1979, 913–921.
11. Loan, Ch. Van. A symplectic method for approximating all the eigenvalues of Hamiltonian matrix. *Linear Algebra Appl.*, **61**, 1984, 233–251.
12. Page, Ch., Ch. Van Loan. A Shur decomposition for Hamiltonian matrices. *Linear Algebra Appl.* **41**, 1981, 11–32.
13. Petkov, M., I. G. Ivanov. Numerical computation of eigenvalues of Hamiltonian matrices. *Mathematica Balkanica*, **11**, 1997, 339-346.
14. Stewart, S. A Jacobi-like algorithm for computing the Schur decomposition of a non-Hermitian matrix. *SIAM J. of Scientific and Statistical Comp.*, **6**, 1985, 853-864.
15. Veselić, K. A convergent Jacobi method for solving the eigenproblem of arbitrary real matrices. *Numer. Math.*, **25**, 1976, 179–184.
16. Voevodin, V. Numerical methods of the algebra. Moscow, 1966 (in Russian).
17. Xu Hong-guo, Lin-zhang Lu. Properties of a quadratic matrix equation and the solution of the continuous-time algebraic Riccati equation. *Linear Algebra Appl.*, **222**, 1995, 127–145.

Received February 10, 1999

Faculty of Mathematics and Informatics
Shoumen University
BG-9712 Shoumen, Bulgaria
E-mail: i.gantchev@fmi.uni-shoumen.acad.bg

ГОДИШНИК НА СОФИЙСКИЯ УНИВЕРСИТЕТ „СВ. КЛИМЕНТ ОХРИДСКИ“

ФАКУЛТЕТ ПО МАТЕМАТИКА И ИНФОРМАТИКА
Книга 2 — Приложна математика и информатика
Том 92, 1998

ANNUAIRE DE L'UNIVERSITE DE SOFIA „ST. KLIMENT OHRIDSKI“

FACULTE DE MATHEMATIQUES ET INFORMATIQUE
Livre 2 — Mathématiques Appliquée et Informatique
Tome 92, 1998

MATHEMATICAL MODELING OF THE MELT HEATING PROCESSES IN A FURNACE, PRODUCING FLAT GLASS

I. ZHELEVA, V. KAMBOUROVA, R. RUSEV, P. DIMITROV

The aim of this paper is to present a two-dimensional model of heat-transfer and transport processes in a glass melting furnace and to use this model for investigation of specifical temperature regimes for different heat flows as well as for the different effective thermal conductivity functions. The mathematical model is elaborated on the base of the real flat glass furnace working in Diamond Ltd in Razgrad. The appropriate numerical methods and their performance are discussed as well.

Keywords: Navier–Stokes equations, melt heating, numerical processes

1991/95 Math. Subject Classification: 76M20

1. INTRODUCTION

Processes taking place in a glass melting furnace producing flat glass are very complicated. In fact, there are five relatively separated physic-chemical processes — silication, refining fusion, degassing, homogenization and cooling, which are closely interconnected at very high temperature and practically occur simultaneously.

The outlet product of the furnace is a glass melt suitable for drawing. Its basic characteristic is its quality, defined by thermal and chemical homogeneity in the drawing volume and in time. Therefore it is very important for the quality of the flat glass that the temperature regime in the furnace be within given limits. Hence the automatic control of glass' quality is directly connected with the control of temperature distribution within the furnace.

The measurement of the temperature of the glass melt is very difficult however. The glass surface temperature is measured using pyrometers and the temperature of melt near the walls and the bottom is usually measured using very expensive special thermocouples. As a matter of fact, the temperature within the glass melt cannot be measured properly and it is practically impossible to have reliable information about it. Information for the temperature distribution in the glass melt can be obtained by mathematical modeling of heat transfer and transport phenomena taking place in the furnace. This information can be used for automatic control of the temperature regime in the furnace and for studying the energeting behaviour of the furnace.

The aim of this paper is to present a two-dimensional model of heat-transfer and transport processes in the furnace and to use this model for investigation of the temperature regime in the furnace for different heat flows as well as for the different effective thermal conductivity functions. The mathematical model is elaborated on the base of the real flat glass furnace working in Diamond Ltd in Razgrad. The appropriate numerical methods and their performance are discussed as well.

2. FORMULATION OF THE MATHEMATICAL PROBLEM

2.1. SCHEME OF THE FURNACE

The glass melting furnace is divided into two parts — a burning chamber and a tank. In this paper we will examine only the tank and will take into account the heat flow from the burning chamber to the glass surface as a boundary conditions on the melt glass surface.

The scheme of the tank and its geometric parameters and coordinate system are given in Fig. 1. The tank consists of two parts — a melting zone (I) and a cooling zone (II). The area HI (Fig. 1) is covered by a batch wedge with a small opening angle. Its length is also given in Fig. 1. The batch material is feed from the doghouse into the furnace with a given temperature.

The side below the batch wedge has a constant temperature which equals the melting temperature (Table 2). The melt batch enters the tank in this place with a given constant velocity v_0. All the heat flow towards the batch in the zone IH is spent for its melting. That is why the heat flow to the glass melt in this zone in fact is equal to zero. In the zone HG the glass melt is heated to the needed temperature for the chemical processes — silication, refining fusion, degassing and homogenization temperature. The glass melt is slightly cooled to the drawing temperature in the cooling zone (II). The homogeneous glass melt is drawn from four drawing machines. The place of the drawing machines is in the end of the cooling zone (area CK in Fig. 1).

2.2. BASIC EQUATIONS

On the base of physical properties of the melt we assume that the glass melt is incompressible Newtonian fluid and the process is steady [1, 2]. The mathematical

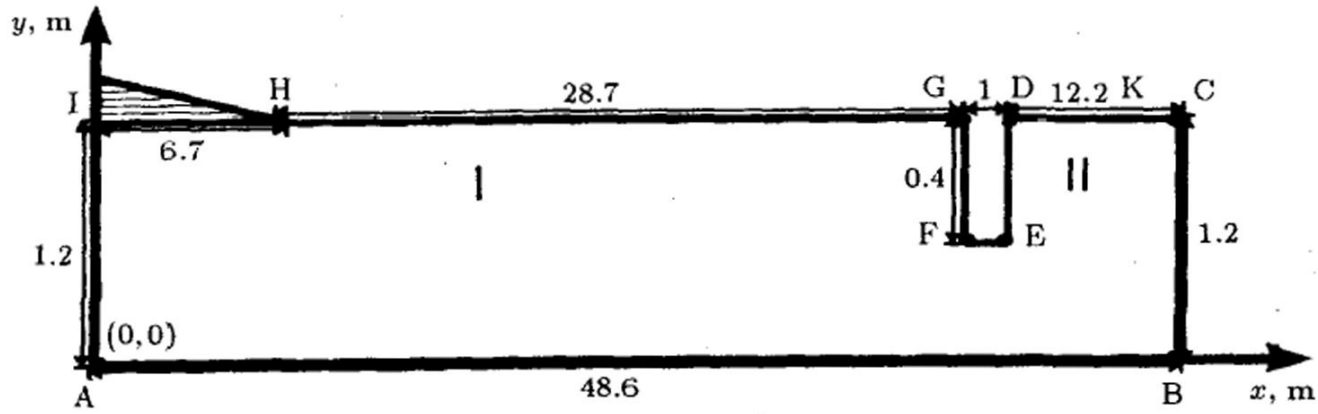

Fig. 1. Scheme of the glass melting furnace and coordinate system

model of the motion and the thermal behaviour of the glass melt are described by the mass, momentum and energy conservation equations that can be expressed as

$$\begin{aligned} &\operatorname{div} \boldsymbol{v} = 0, \\ &\rho(\boldsymbol{v} \cdot \nabla)\boldsymbol{v} = -\nabla P - \nabla \cdot \tau + \rho\beta(T - T_0)g, \\ &\rho C_p \boldsymbol{v} \cdot \nabla T = \nabla \cdot K_{\text{eff}} \nabla T, \end{aligned} \tag{1}$$

where ρ is the glass density, $\boldsymbol{v}(u, v)$ — the velocity vector with its rectangular coordinates, P — the pressure, τ — the viscous stress tensor, β — the volumetric coefficient of expansion, T and T_0 are the glass temperature and its reference value, g is the gravity acceleration, C_p — the specific heat, and K_{eff} — the effective thermal conductivity, which is a temperature function.

This system can be written in a dimensionless form:

$$\begin{aligned} &\operatorname{div} \boldsymbol{v} = 0, \\ &(\boldsymbol{v} \cdot \nabla)\boldsymbol{v} = -\nabla p + \frac{1}{\mathrm{Re}} \Delta \boldsymbol{v} + \widetilde{F}, \\ &\boldsymbol{v} \cdot \nabla \theta = \left(\frac{1}{\mathrm{Re}} \nabla\right) \cdot \frac{1}{\mathrm{Pr}} \nabla \theta, \end{aligned} \tag{2}$$

where θ, p and $\boldsymbol{v}$ are resp. dimensionless temperature, pressure and velocity; $\widetilde{F}$ is a function of the temperature; Re is the Reynolds number $\mathrm{Re} = L_1 \mathrm{v}_0/\nu$, L_1 and v_0 are resp. the scales of length and velocity, Pr is the Prandtl number $\mathrm{Pr} = \nu/a$, ν is the kinematic viscosity, $a = K_{\text{eff}}/(pC_p)$ is the coefficient of temperature conductivity. Therefore the Prandtl number is a function of the temperature.

The Navier–Stokes equations can be written in an equivalent form, without pressure p, which is more suitable for numerical calculations, namely,

$$\begin{aligned} &u\frac{\partial \omega}{\partial x} + v\frac{\partial \omega}{\partial y} = \frac{1}{\mathrm{Re}}\left(\frac{\partial^2 \omega}{\partial x^2} + \frac{\partial^2 \omega}{\partial y^2}\right) + \widetilde{F}, \\ &\frac{\partial^2 \psi}{\partial x^2} + \frac{\partial^2 \psi}{\partial y^2} = \omega. \end{aligned} \tag{3}$$

Here ψ is the stream-function and ω is the vorticity:

$$\omega = \frac{\partial u}{\partial y} - \frac{\partial v}{\partial x}. \tag{4}$$

In turn, the components of the velocity vector u and v in a rectangular coordinate system (Fig. 1) can be written as functions of the stream function:

$$u = \frac{\partial \psi}{\partial y}, \quad v = -\frac{\partial \psi}{\partial x}. \tag{5}$$

2.3. BOUNDARY CONDITIONS FOR THE TEMPERATURE AND THE STREAM-FUNCTION

The boundary conditions for the temperature and stream-function are given in Table 1. The special feature of Navier–Stokes equations is that the boundary conditions are given only for the stream-function and for the vorticity they must be calculated on the base of the values of the stream-function on the boundaries. The boundary conditions are described in details in [4].

Table 1

	Dimensionless Temperature θ	Dimensionless Stream-function ψ
Front wall; IA	$\frac{\partial \theta}{\partial x} = \frac{U_{w_1} L_0}{K_{\text{eff}}}(\theta - \theta_a)$	$\psi = 0; \quad \frac{\partial \psi}{\partial x} = 0$
Back wall; BC	$-\frac{\partial \theta}{\partial x} = \frac{U_{w_2} L_0}{K_{\text{eff}}}(\theta - \theta_a)$	$\psi = 0; \quad \frac{\partial \psi}{\partial x} = 0$
Bottom; AB	$\frac{\partial \theta}{\partial y} = \frac{U_b L}{K_{\text{eff}}}(\theta - \theta_a)$	$\psi = 0; \quad \frac{\partial \psi}{\partial y} = 0$
Shield assembly wall; DE	$\frac{\partial \theta}{\partial x} = 0$	$\psi = C_1 = \text{const}; \quad \frac{\partial \psi}{\partial x} = 0$
Shield assembly wall; EF	$\frac{\partial \theta}{\partial y} = 0$	$\psi = C_1 = \text{const}; \quad \frac{\partial \psi}{\partial y} = 0$
Shield assembly wall; FG	$\frac{\partial \theta}{\partial x} = 0$	$\psi = C_1 = \text{const}; \quad \frac{\partial \psi}{\partial x} = 0$
Top surface; HI	$\frac{\partial \theta}{\partial y} = 0$	$\psi = \int\limits_{L-L_5}^{L} V\,dx; \quad \frac{\partial \psi}{\partial y} = 0$
Top surface; GH	$\frac{\partial \theta}{\partial y} = q_T \frac{L_0}{K_{\text{eff}} T_0}$	$\psi = C_1 = \text{const}; \quad \frac{\partial \psi}{\partial y} = 0$
Top surface; KD	$\frac{\partial \theta}{\partial y} = q_C \frac{L_0}{K_{\text{eff}} T_0}$	$\psi = C_1 = \text{const}; \quad \frac{\partial \psi}{\partial y} = 0$
Place of drawing machines; CK	$\frac{\partial \theta}{\partial y} = q_C \frac{L_0}{K_{\text{eff}} T_0}$	$\psi = -\int\limits_{0}^{L_1} V\,dx; \quad \frac{\partial \psi}{\partial y} = 0$

The following notations are used in Table 1: U_i is the heat transfer coefficient of solid surfaces (the front wall, the back wall and the bottom), whose values are different for the different surfaces; θ_a is the dimensionless ambient temperature; q_T, q_C are the heat flows, entering the glass melt surface.

2.4. HEAT FLOW AT THE GLASS SURFACE

For the present purposes, we shall formulate a model for the heat flow, using experimental data for the distribution of the fuel flux from the burners. The heat losses due to outlet combustion gasses are taken into account. The distribution of the heat flow entering the glass melt surface along the x-direction and its approximation are given in Fig. 2. The distribution is approximated by the function

$$\log q = 1.0489 + 0.0173x - 0.0000770258x^2. \tag{6}$$

In the model 14 different functions are to be chosen. The criterion for the best choice is a maximal correlation coefficient and a minimal mean square error.

3. NUMERICAL PROCEDURE

A 5-point approximation is used for the solution of the system of partial differential equations. The grid is non-uniform and it is concentrated in critical areas (near the walls, the bottom and the top surface). The number of points used in the numerical solution is 497 in the x-direction and 23 in the y-direction.

An alternating direction implicit method is used for numerical calculation of the Navier – Stokes equations written in term of stream-function and vorticity and of heat transfer equation. This algorithm is described in detail in [3]. This method is iterative and is based on the solving of the tri-diagonal matrix.

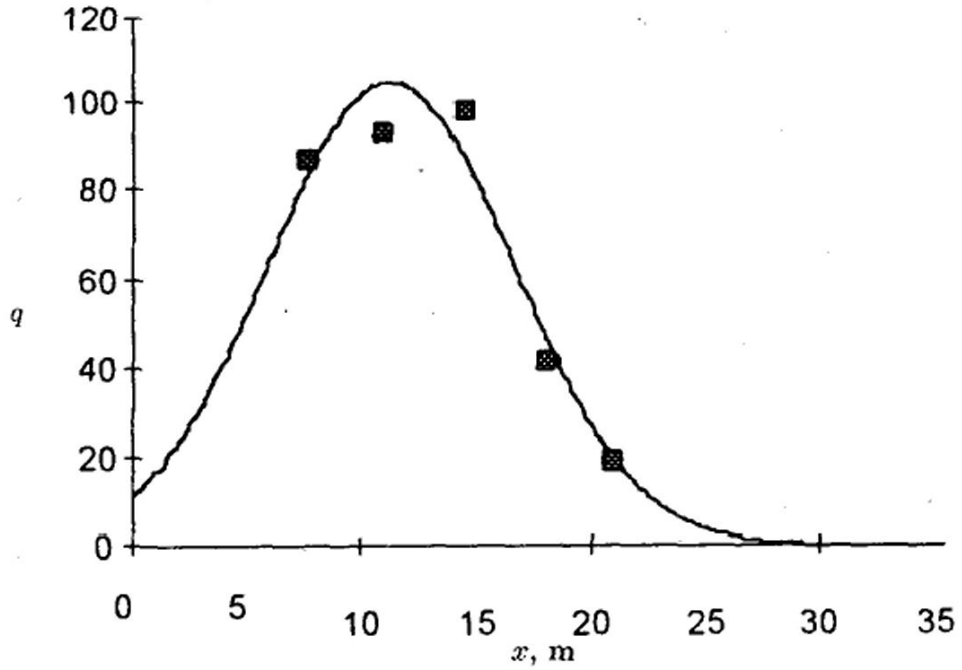

Fig. 2. Distribution of the heat flow in the x-direction and its approximation

4. NUMERICAL RESULTS AND DISCUSSION

We shall use the present mathematical model for an investigation of the heat regime in the glass melt and the influence of the heat flow upon the temperature distribution. Usually, the temperature of the glass melt which enters into the cooling zone is controlled by a change of the fuel flux from the last two couples of burners. That is why we shall study first of all the influence of the heat flow from these burners upon the temperature of the glass melt.

4.1. MODEL PARAMETERS

The tank size considered (Fig. 1) has 48.6 m length and 1.2 m depth. The geometrical dimensions of the melting and cooling zones of the tank are given in Table 2. The heat transfer coefficients of the bottom and front and back walls are given in the same table. The thermophisycal properties for the glass melt (density, specific heat, kinematical viscosity and effective thermal conductivity) for the flat glass melt are taken from the literature [1, 2] and they are summarized in Table 2.

Table 2

Parameters	Value
Density, ρ	2320 kg/m^3
Specific heat, C_p	1256 J/(kg.K)
Kinematics viscosity, ν	0.0101 m^2/s
Effective thermal conductivity, K_{eff}	$5.386 - 2.168 \times 10^{-2}T + 2.058 \times 10^{-5}T^2$
Prandtl number, Pr	$29430.6/K_{\text{eff}}$
Reynolds number, Re	0.0222
Melting temperature, T_m	1100 K
Ambient temperature, T_a	350 K
Heat transfer coefficient of the walls and the bottom, U_1	4 W/(m^{2}K)
Melting zone: length, IG	35.4 m
depth, IA	1.2 m
Length of the batch, IH	0.3 m
Cooling zone: length, CD	12.2 m
depth, BC	1.2 m
Shield assembly: length, EF	1.0 m
depth, GF	0.35 m

The basic simulation is calculated for the parameters given in Table 2 and for the heat flow approximation, given by the function (6) (see Fig. 2). The temperature distribution for the basic simulation is plotted in Fig. 3 and the stream-function field is shown in Fig. 4.

The maximal temperature is calculated at the top surface in the melting zone for $x = 11.1$ m and it equals 2020.4 K. The maximal temperature gradient is in the same area. The minimal temperature in the tank is 1262 K and it is calculated near the front wall and the bottom. The maximal temperature gradient is in the area with maximal heat flow ($x = 11$ m). In this area the difference between temperature at the top and at the bottom is 600 K. The same difference in the area with minimal heat flow ($x = 0$ m) is 50 K. The temperature gradient in the cooling zone near the back wall is only 20 K.

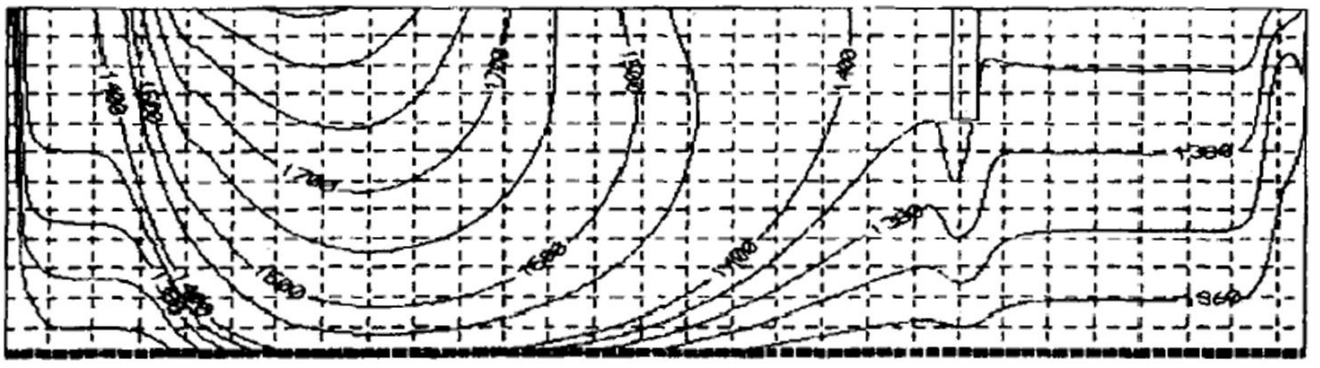

Fig. 3. Temperature distribution for the basic simulation

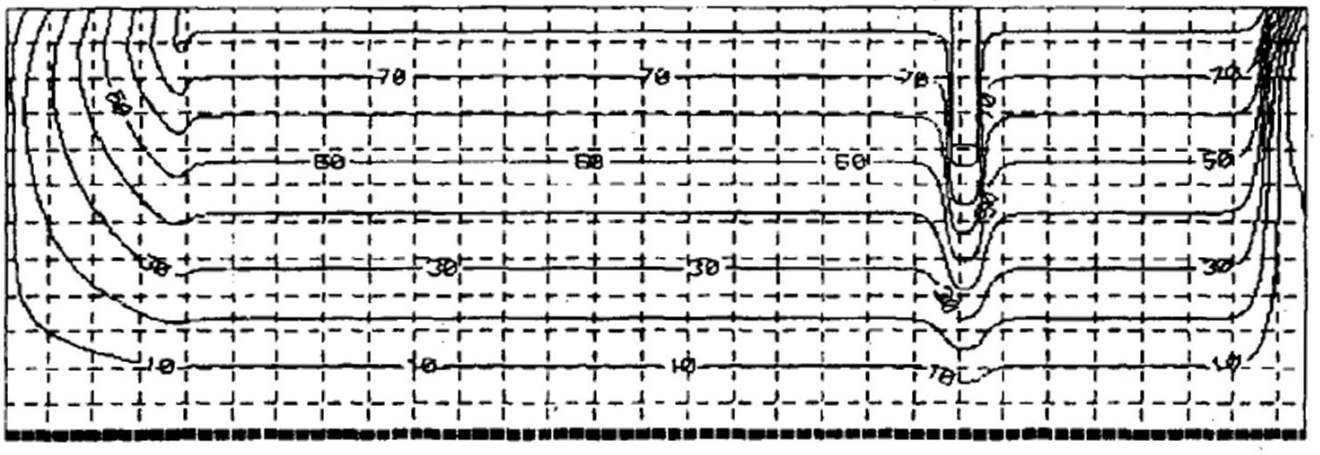

Fig. 4. Stream-function field for the basic simulation

4.3. INVESTIGATION OF THE INFLUENCE OF THE HEAT FLOW

We shall investigate the decrease and the increase of the heat flow from the last two couples of burners by 10 and 20 percent. Different variants for the distribution of the heat flow and its approximation are given in Table 3.

Table 3

Variant	Heat flow from V-th couple of burners, %	Heat flow from VI-th couple of burners, %	Approximation of the heat flow
1	100	100	$\log q = 1.0489 + 0.0173x - 0.0000770258x^2$
2	100	90	$\log q = 0.9695 + 0.018744x - 0.000082906x^2$
3	100	80	$\log q = 0.8806 + 0.020343x - 0.000089479x^2$
4	90	90	$\log q = 1.0272 + 0.017879x - 0.000080401x^2$
5	80	90	$\log q = 1.0918 + 0.016913x - 0.000077602x^2$
6	90	80	$\log q = 0.9384 + 0.019479x - 0.000086975x^2$
7	80	80	$\log q = 1.0030 + 0.018513x - 0.000084176x^2$
8	90	100	$\log q = 1.1067 + 0.016486x - 0.000074522x^2$
9	80	100	$\log q = 1.7113 + 0.015482x - 0.000071723x^2$

The decreasing of the heat flow from the last couple of burners (VI) by 10 percent leads to decrease of the temperature of the glass melt surface with 12 K. 20 percent decreasing of the heat flow leads to decrease of the temperature with 24 K (see Fig. 5). The maximal change of the temperature is near the top surface in the zone under the VI-th couple of burners ($x = 17$ to 25m). Little changes of the fuel flux can lead to a smooth change of the temperature of this zone.

The influence of the heat flow from the V-th couple of burners upon the temperature is given in Fig. 6. This couple of burners dislocates at $x = 17.9$ m. That is why the maximal differences in the temperatures are observed for $x = 15$ m and they are 12.3 K for 10 percent decreasing of the heat flow and 25.7 K for 20 percent decreasing.

Maximal temperature differences when the heat flow decreases from V-th and VI-th couples of burners (variants 1, 4, 5, 6 and 7) are observed for $x = 15$ m to 25 m, too. The decreasing of the heat flow from the V-th and VI-th couples of burners leads to decrease of the temperature at the top surface maximum with 36 K and its temperature difference is under the VI-th couple of burners ($x = 19$ m). Therefore the change of the heat flow from the V-th and VI-th couples of burners can be used for the automatic control of the temperature distribution at the end of the melting zone and in the cooling zone.

4.4. COMPARISON BETWEEN THE COMPUTED RESULTS AND THE EXPERIMENTAL DATA FOR THE TEMPERATURE

The results from the mathematical model could be compared with measurements in the working furnace which is described in this study. The measurement of the temperature of the glass melt is very difficult. The glass surface temperature

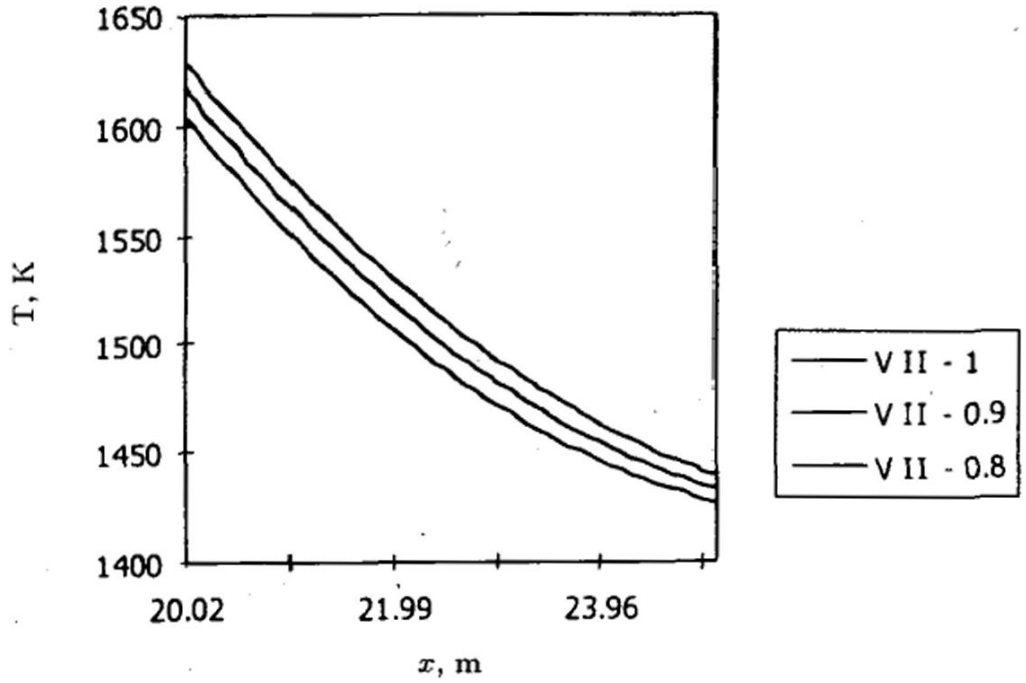

Fig. 5. Temperature of the glass melt at the top surface for variants 1, 2 and 3 (Table 3)

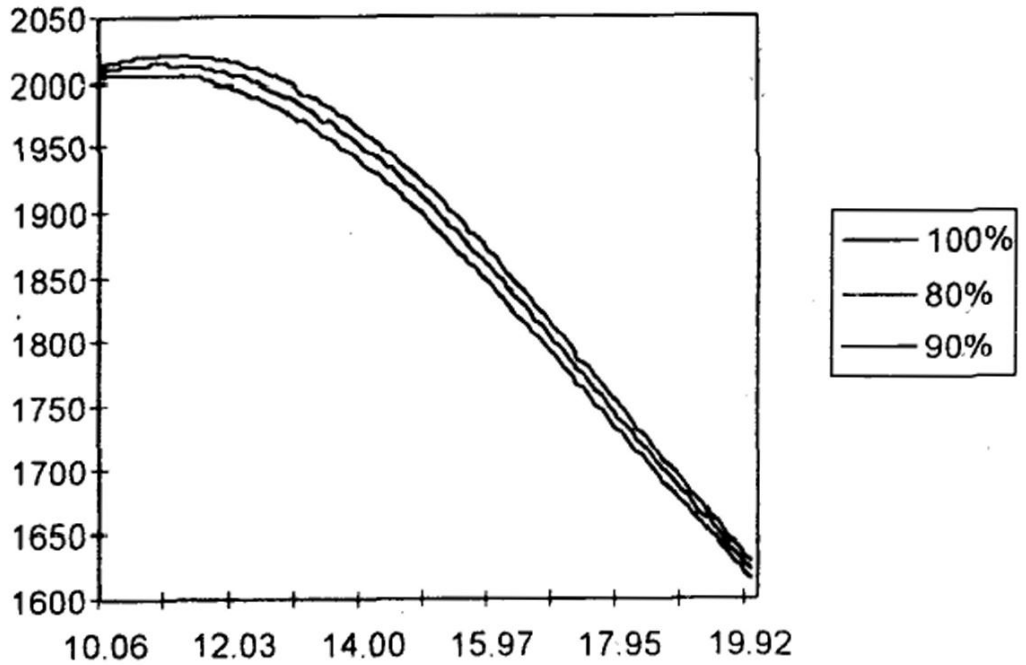

Fig. 6. Temperature of the glass melt at the top surface for variants 1, 8 and 9 (Table 3)

is measured using pyrometers, which leads to considerable errors. As a matter of fact, the temperature within the glass melt cannot be measured and it is practically impossible to have reliable information about it.

The experimental data and computed results for the basic variant for the surface temperature are shown in Fig. 7. It is well seen that the results for the surface temperature agree very well with the measured temperature. The comparison of the simulation results and real data shows that the model gives good results for

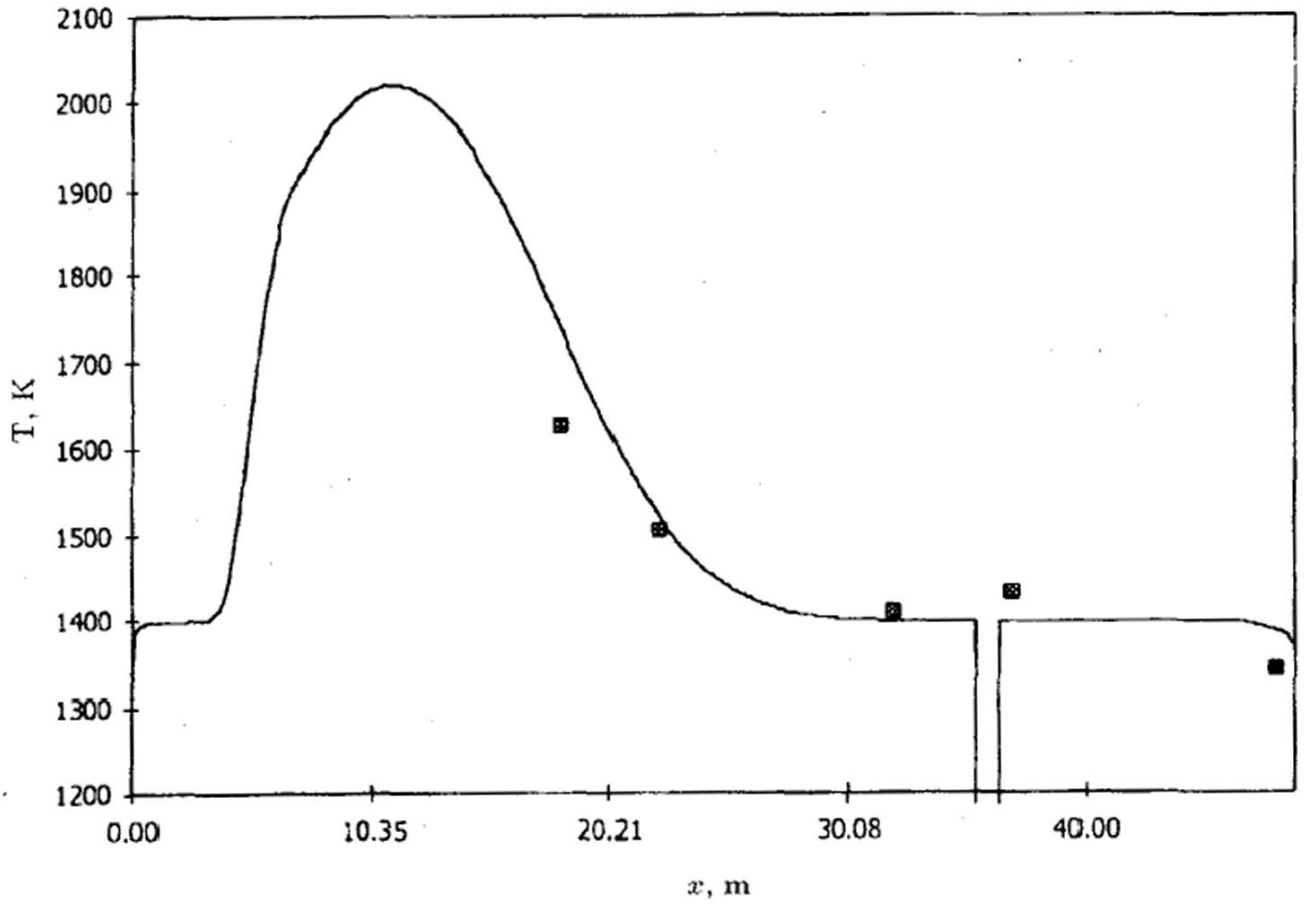

Fig. 7. Computed results and experimental data for the surface temperature

the surface temperature in the melting and cooling zones. The maximal difference is in the zone of burners.

5. CONCLUDING REMARKS

A simplified, but effective mathematical model of the heat transfer and transport processes in the glass melting furnace is presented. The numerical solution uses a finite differences method. The influence of the heat flow from the last two couple of burners is investigated. The comparison of the calculated results and the real data demonstrates a good agreement. The present model offers as well the possibility of computing the appropriate spatial distributions of the temperature and velocity fields. It is possible to study, in particular, the energetical behaviour of the furnace and the influence of its technological parameters upon the temperature and velocity distributions. The approach presented here can be also modified in order to include more specific details of the heat flow and heat transfer in the combustion chamber for calculating more precisely the boundary conditions and the temperature on the glass melting surface. For example, if the air-bubbles motion is taken into account, we can obtain more precise results for the temperature and velocity fields within the furnace.

Acknowledgements. The support of the Bulgarian National Fund of Scientific Research under Grant No MM510/1995 is gratefully acknowledged.

REFERENCES

1. Horvath, Z., G. Hilbig. Mathematical model for fuel-heated glass melting tanks. *Glastech. Ber.*, **61**, 10, 1988, 277–282.
2. Roache, P. Computation fluid dynamics. Hermosa Publishers, Albuquerque, 1976.
3. Ungan, A., R. Viskanta. Three dimensional numerical modeling of circulation and heat transfer in a glass melting tank. *Glastech. Ber.*, **60**, 1987, No 3, 71–78; No 4, 115–124.
4. Zheleva, I., V. Kambourova, P. Dimitrov, R. Rusev. Numerical algorithm for studying heat transfer in a glass melting furnace. In: *Lecture notes in Computer Science*, First International Workshop on Numerical Analysis and its Applications, WNAA'96, 1996, 589–595.
5. Zheleva, I., V. Kambourova, P. Dimitrov, R. Rusev. Identification of the heat transfer in a glass melting furnace. In: *Lecture notes in Computer Science*, First International Workshop on Numerical Analysis and its Applications, WNAA'96, 1996, 596–605.

Received February 10, 1999
Revised March 27, 1999

University of Rousse
Technology College - Razgrad
BG-7200 Razgrad, P.O. Box 110
Bulgaria
E-mail: ihtbt@mbox.digsys.bg

ГОДИШНИК НА СОФИЙСКИЯ УНИВЕРСИТЕТ „СВ. КЛИМЕНТ ОХРИДСКИ“

ФАКУЛТЕТ ПО МАТЕМАТИКА И ИНФОРМАТИКА
Книга 2 — Приложна математика и информатика
Том 92, 1998

ANNUAIRE DE L'UNIVERSITE DE SOFIA „ST. KLIMENT OHRIDSKI“

FACULTE DE MATHEMATIQUES ET INFORMATIQUE
Livre 2 — Mathématiques Appliquée et Informatique
Tome 92, 1998

FIELD FLUCTUATIONS AND THE EFFECTIVE BEHAVIOUR OF MICRO-INHOMOGENEOUS SOLIDS

K. Z. MARKOV and K. S. ILIEVA

The problem of predicting the effective mechanical properties and response of micro-inhomogeneous solids is revisited. The aim is to highlight the influence of field fluctuations which, as a rule, is neglected by the numerous existing theories that interconnect the micro- and macro-behaviour of such solids. The key observation is that in a heterogeneous solid of random constitution when, say, macroscopic quantities like the mean stress tensor are prescribed, fluctuations always create regions in which considerably higher stresses appear. In these regions either plastic flow or a certain kind of deterioration takes place, which affects the macroscopic behaviour of the solid. The result is that the latter should start exhibiting deviations from linear response from the very beginning of straining, despite the assumed linearity of its constituents. A quantitative approach that takes into account the field fluctuations is proposed and outlined in the lecture. For the simplicity sake, the scalar conductivity problem for a dilute dispersion of spheres, possessing properties different from those of the matrix, is employed in order to illustrate better the basic ideas. The progressive deviation from linearity, when the macroscopic "straining" increases, indeed shows up clearly in the performed analysis.

Keywords: random media, effective properties, fluctuations, excursion sets

1991/1995 Math. Subject Classification: 73B35, 73S10, 60G60

1. INTRODUCTION

The problem of predicting the macroscopic mechanical behaviour of a solid with a given internal structure is the central problem of micromechanics, see, e.g.

[1], where the well-known approximate theories, like self-consistent one, differential scheme, effective fields approach, etc., are considered in detail and compared to experimental findings. If the microstructure is periodic, there exist rigorous mathematical theories that yield numerical algorithms for evaluating the macroproperties [2]. For a random solid, the problem is considerably more complicated. In the simplest linear scalar case it is posed in the following typical manner [3].

Assume that $\kappa(\mathbf{x})$ is the known random field of conductivity coefficient for the medium. (For a two-phase one, $\kappa(\mathbf{x})$ takes the values κ_m or κ_f depending on whether $\mathbf{x}$ lies in the matrix or in a particle, respectively.) The temperature field, $\theta(\mathbf{x})$, in such a medium is governed by the equations

$$\nabla \cdot \mathbf{q}(\mathbf{x}) = 0, \quad \mathbf{q}(\mathbf{x}) = \kappa(\mathbf{x})\nabla\theta(\mathbf{x}), \quad \langle \nabla\theta(\mathbf{x}) \rangle = \mathbf{G}, \tag{1.1}$$

where $\mathbf{q}(\mathbf{x})$ is the (opposite) heat flux, $\mathbf{G}$ denotes the prescribed macroscopic temperature gradient. Hereafter $\langle \cdot \rangle$ signifies ensemble averaging. The problem (1.1) should be solved in statistical sense — for a given (infinite) hierarchy of multipoint moments $\langle \kappa(\mathbf{x}_1) \ldots \kappa(\mathbf{x}_q) \rangle$, $q = 1, 2, \ldots$, one should find the similar hierarchy of *all* multipoint moments of $\theta(\mathbf{x})$ and the joint moments of $\theta(\mathbf{x})$ and $\kappa(\mathbf{x})$. In particular among the latter, one of the simplest is of special interest, namely,

$$\mathbf{Q} = \langle \kappa(\mathbf{x})\nabla\theta(\mathbf{x}) \rangle = \kappa^* \mathbf{G}, \tag{1.2}$$

since it defines the well-known effective conductivity, κ^*, of the medium. The definition (1.2) of κ^* reflects the familiar "homogenization" of the problem under study, in the sense that from a macroscopic point of view, when only the macroscopic values of the flux $\mathbf{Q}$ and of the temperature gradient $\mathbf{G}$ are of interest, the medium behaves as if it were homogeneous with a certain macroscopic conductivity κ^*. This interpretation explains why the value κ^* and its counterparts, say, the effective elastic moduli, have been extensively studied in the literature on micro-inhomogeneous and composite materials, see [1] and the references therein.

However, κ^* is only a *tiny* part of the full statistical solution of the random problem (1.1). Moreover, its evaluation *cannot* be torn away from the full statistical solution of (1.1), i.e., of specifying the *whole* infinite hierarchy of multipoint moments, as argued, e.g., in [3 – 5] et al. (The latter fact explains the failure of all schemes that try to determine solely the effective property κ^* without trying to solve the whole stochastic problem (1.1).) Besides, there are plenty of reasons why one should pay much more attention to other statistical characteristics of random fields like $\theta(\mathbf{x})$ in (1.1), that appear in problems in random heterogeneous media. For instance, one of the most important quantities is often the variance of local fields, connected with the square of its fluctuation, see, e.g., [6 – 7] for more details and references. For transport-like problems of the type of (1.1), the fluctuations of the local fields are of primary importance when there exists a transition to nonlinearity (or a deterioration starts) after a certain threshold. Their effect will then consists in a deviation from a linear response, however small is the macroscopic "loading" $\mathbf{G}$, compared to the respective threshold value.

The aim of the present work is to quantify this statement to a certain extent, sketching very briefly a theory that describes such an effect in a highly idealized

situation and thus will stimulate, hopefully, further interest and research. The core of the approach is a combination between the functional (Volterra-Wiener) series method, proposed and used in the last years by one of the authors in the study of micro-inhomogeneous solids [4, 5], and Vanmarcke's theory [8] of excursion sets for random fields.

2. STATISTICAL SOLUTION OF EQ. (1.1) FOR A DILUTE DISPERSION

To illustrate the basic ideas, assume that the medium is a random dispersion of spheres — a typical representative of the wide and important class of particulate micro-inhomogeneous media, extensively studied in the literature.

Let $\eta_f = nV_a$, $V_a = \frac{4}{3}\pi a^3$, be the volume fraction of the spheres, n be their number density, a be the spheres' radius. In this case the random temperature field, $\theta(\mathbf{x})$, that solves the problem (1.1), can be conveniently constructed by means of the functional series approach, see [5, 6]. In particular, in the dilute limit $\eta_f \ll 1$, $\theta(\mathbf{x})$ has the form of the truncated functional series

$$\theta(\mathbf{x}) = \mathbf{G}\cdot\mathbf{x} + \int T_1(\mathbf{x}-\mathbf{y})\,\psi'(\mathbf{y})\,d\mathbf{y} + o(\eta_f), \tag{2.1}$$

where $\psi'(\mathbf{x}) = \psi(\mathbf{x}) - n$ is the fluctuating part of Stratonovich's random density field $\psi(\mathbf{x}) = \sum_\alpha \delta(\mathbf{x}-\mathbf{x}_\alpha)$, see [9]. The integrals hereafter are over the entire space $\mathbb{R}^3$ if the integration domain is not explicitly indicated. In Eq. (2.1)

$$T_1(\mathbf{x}) = 3\beta\mathbf{G}\cdot\nabla\varphi(\mathbf{x}), \quad \beta = \frac{[\kappa]}{\kappa_f + 2\kappa_m}, \tag{2.2}$$

$[\kappa] = \kappa_f - \kappa_m$, is the "single-sphere" field, i.e. the disturbance to the temperature field $\mathbf{G}\cdot\mathbf{x}$ in the homogeneous matrix of conductivity κ_m, introduced by a single spherical inclusion of conductivity κ_f; $\varphi(x)$ is the Newtonian potential for the latter inclusion. Recall that

$$\nabla\nabla\varphi(\mathbf{x}) = -\frac{1}{3}\begin{cases}\mathbf{I}, & \text{if } |\mathbf{x}| < a,\\ \dfrac{1}{\rho^3}(\mathbf{I} - 3\mathbf{e}_r\mathbf{e}_r), & \text{if } |\mathbf{x}| > a,\end{cases} \tag{2.3}$$

$\rho = r/a$, $\mathbf{e}_r = \mathbf{r}/r$, $r = |\mathbf{x}|$ and $\mathbf{I}$ is the unit 2nd-rank tensor.

The representation (2.1) allows us to obtain all statistical characteristics of the field $\theta(\mathbf{x})$, asymptotically correctly to the order $O(\eta_f)$. In particular, we shall need in what follows the full statistical information about the random variable

$$\tau = \tau(0) = |\nabla\theta(0)|^2, \tag{2.4}$$

i.e. its probability distribution function. Using Eq. (2.1) and the formula

$$\langle\psi'(\mathbf{y}_1)\psi'(\mathbf{y}_2)\rangle = n\delta(\mathbf{y}_1 - \mathbf{y}_2) + o(n), \tag{2.5}$$

see [9], yields after some algebra:

$$\tau = \widetilde{G}^2 + 3\beta \mathbf{G} \cdot \int \widetilde{\mathbf{T}}(\mathbf{y}) \cdot \mathbf{G}\, \psi'(\mathbf{y})\, d\mathbf{y} + o(\eta_f), \tag{2.6}$$

$$\widetilde{G}^2 = G^2(1 + 3\beta^2 \eta_f), \quad \widetilde{\mathbf{T}}(\mathbf{y}) = \nabla\nabla\varphi(\mathbf{y}) \cdot \big(2\mathbf{I} + 3\beta \nabla\nabla\varphi(\mathbf{y})\big), \tag{2.7}$$

having employed some results of [5]. In a similar way, other statistical characteristics of the field $\theta(\mathbf{x})$ can be obtained in a closed form. Details can be found, e.g., in [5] and [10] (in the last paper the effective conductivity κ^* of the dispersion is rigorously evaluated to the order $O(\eta_f^2)$, using the truncated functional series approach).

3. THE IDEALIZED MODEL

Hereafter we shall consider a highly idealized situation in which there exists a threshold G_0 of the temperature gradient with the following properties: if $|\nabla\theta(\mathbf{x})| < G_0$ locally (at the point $\mathbf{x}$ that is), then both the matrix and the particles behave linearly, obeying the Fourier law, see Eq. (1.1).

If, however, $|\nabla\theta(\mathbf{x})| \geq G_0$, the constituents become nonconducting, i.e. both κ_f and κ_m vanish. In other words, the following constitutive equation is adopted:

$$\mathbf{q}(\mathbf{x}) = \widetilde{\kappa}(\mathbf{x}) \nabla\theta(\mathbf{x}), \quad \widetilde{\kappa}(\mathbf{x}) = \begin{cases} \kappa(\mathbf{x}), & \text{if } |\nabla\theta(\mathbf{x})| < G_0, \\ 0, & \text{if } |\nabla\theta(\mathbf{x})| \geq G_0, \end{cases} \tag{3.1}$$

$$\kappa(\mathbf{x}) = \begin{cases} \kappa_m, & \text{if } \mathbf{x} \in \text{matrix}, \\ \kappa_f, & \text{if } \mathbf{x} \in \text{spheres}. \end{cases}$$

The model is not claimed to have any specific physical meaning — its sole role here is to illustrate the basic ideas and techniques as simply as possible. The generalizations to more realistic situations when, say, plastic flow and/or damaging take place, according to certain well-known criteria, can be performed along a similar line of reasoning (provided the volume fraction η_f of the inhomogeneities is small enough).

As it follows from Eqs. (2.2) and (2.3), the "stress-concentration factor" for the single-spherical inhomogeneity, in the scalar conductivity context under discussion, does not exceed 2, whatever the values of κ_f and κ_m. In other words, if $G = |\mathbf{G}|$ is the magnitude of the temperature gradient at infinity, the magnitude of this gradient within or around the single inhomogeneity does not exceed $2G$. This means that if $G < 2G_0$, the linear Fourier law in (1.1) is applicable throughout the whole infinite space, comprising the matrix with the single spherical inhomogeneity. In turn, for the considered dilute dispersion the spheres are, as a rule, far one from

another[1] and hence each one can be considered as single, immersed into the infinite matrix constituent. This means that under the condition

$$G < G_0/2, \tag{3.2}$$

the linear equation (1.1) for the temperature field $\theta(\mathbf{x})$ is still applicable, despite the obvious strong nonlinearity of the model (3.1). The applicability of this linear equation does not mean, though, that the effective behaviour of the solid will be linear even in the region (3.2). The reason is that however small is the mean gradient's magnitude G, the solution $\theta(\mathbf{x})$ of Eq. (1.1), being *random*, will always exhibit fluctuations, some of which will be big enough to generate regions in which the local gradient $|\nabla\theta(\mathbf{x})| \geq G_0$. These are just the so-called *excursion* sets to be discussed in the next section.

4. THE EXCURSION SETS FOR THE RANDOM FIELD $|\nabla\theta(\mathbf{x})|^2$

Let $f(\mathbf{x})$ be a random field, whose realizations are defined over the region $\Omega \subset \mathbb{R}^3$. The sets $\Omega_A = \{\mathbf{x} \in \Omega \mid f(\mathbf{x}) \geq A\}$ are called excursion for the field $f(\mathbf{x})$, [8, 11]. A problem of central importance for many applications concerns a more detailed description of these sets and, in particular, estimates of their mean volume Ω_A/Ω.

In general, such questions are very hard since the answers should involve the multipoint statistics of $f(\mathbf{x})$. Comparatively simple results are achieved for infinite regions $\Omega = \mathbb{R}^3$ under the assumption that the field $f(\mathbf{x})$ is Gaussian, see again [8, 11]. The latter assumption unfortunately is not appropriate for the fields that, like $\theta(\mathbf{x})$, emerge as solution of the random equations of the type of (1.1) in media of particulate microstructure, see [4] for a more detailed discussion. For arbitrary (statistically homogeneous) random fields convenient, though approximate, results are given by Vanmarcke [8, Ch. 4], and they will turn out very useful for our study, as we shall see in a moment.

Namely, Vanmarcke observed that if the excursion value A is considerably higher than the mean value of the field (say, two or three time at least, which as a matter of fact is just our case, as it follows from Eq. (3.2)), the excursion sets have a simple structure — they represent well separated areas in $\mathbb{R}^3$, whose volume fraction, η_A, is just the complementary cumulative distribution function $F_f^c(A)$ of the random variable $f = f(0)$. More precisely,

$$\eta_A = \lim_{\Omega \to \mathbb{R}^3} \Pr\{f \geq A\} = F_f^c(A) = 1 - F_f(A). \tag{4.1}$$

The result (4.1), though not mentioned explicitly in [8], immediately follows from the formulae (4.6.4) of the same book.

[1] More precisely, this is true if the inclusions are "well-separated"; however, there can exist realizations of the arrays of spheres when they form clusters. The latter may result in a considerable increase of the local temperature gradient, as pointed out in the final section.

Note that due to the assumed statistical homogeneity, all random variables $f(\mathbf{x})$, $\mathbf{x} \in \mathbb{R}^3$, possess one and the same probability distribution function $F_f(A)$ and hence it suffices to take $\mathbf{x} = 0$, i.e. to consider the random variable $f = f(0)$ only.

It is to be also noted that the formula (4.1) has a simple and clear interpretation: it states that for high enough excursion levels the multipoint statistics of the random field $f(\mathbf{x})$ *does not* influence the volume fraction η_A of the respective excursion sets. This volume fraction is thus specified by the "one-point" statistics only, i.e. by the cumulative distribution function (c.d.f.) $F_f(A) = \Pr\{f < A\}$ of the random field $f(\mathbf{x})$ in a fixed point $\mathbf{x}$ (which can always be chosen at the origin due to the assumed statistical homogeneity). With this interpretation, the formula (3.2) becomes natural enough.

For the dispersion under study which obeys the constitutive law (3.1), the quantity of central interest is just the field $|\nabla\theta(\mathbf{x})|^2$ since its excursion sets above the level G_0^2 will represent, so to say, the "plastified" region of the volume fraction η_{G_0} of the composite. According to (4.1),

$$\eta_A = F_\tau^c(G_0) = 1 - F_\tau(G_0), \tag{4.2}$$

where τ is the random variable, introduced in Eq. (2.4), and $F_\tau(A)$ is its cumulative distribution function. These regions will cover both matrix and inhomogeneities with probabilities η_m and η_f, respectively. Hence the dispersion under study will become a three-phase material, comprising:

- phase '1' — matrix of conductivity κ_m and volume fraction $\eta_m(1 - \eta_{G_0})$;
- phase '2' — inhomogeneities of conductivity κ_f and volume fraction $\eta_f(1 - \eta_{G_0})$;
- phase '3' — nonconducting excursion sets ("plastified" regions) of volume fraction η_{G_0}.

Since the dispersion is dilute, $\eta_f \ll 1$, we can imagine that the foregoing three-phase material can be adequately homogenized in the following simple and obvious way. First, the matrix with the inhomogeneities is replaced by a homogeneous medium of effective conductivity

$$\kappa^* = \kappa_m(1 + 3\beta\eta_f) + o(\eta_f), \tag{4.3}$$

without paying attention to the excursion regions. In the next step we introduce the latter (whose conductivity is zero) into the already homogenized medium of conductivity κ^* and get a material of conductivity

$$\widetilde{\kappa}^* = \kappa^* \frac{1 - \eta_{G_0}}{1 + \eta_{G_0}/2} = \kappa_m(1 + 3\beta\eta_f)(1 - \frac{3}{2}\eta_{G_0}), \tag{4.4}$$

so that, eventually,

$$\frac{\widetilde{\kappa}^*}{\kappa_m} = 1 + 3\beta\eta_f - \frac{3}{2}\eta_{G_0} + o(\eta_f). \tag{4.5}$$

In the last formulae (4.3) to (4.5) we have applied the well-known Maxwell (or Clausius-Mossotti) result for predicting the effective conductivity of a dispersion, which is exact in the dilute limit. Also, we have tacitly assumed that the applied gradient G is considerably smaller than the limit one G_0, so that the volume fraction η_{G_0} of the "plastified" regions is small as well; the parameter β in the foregoing relations is defined in Eq. (2.2). The latter assumption is in full agreement with the one that assures the applicability of Vanmarcke's formula (4.1), so that

$$\eta_{G_0} = F_\tau(G_0). \tag{4.6}$$

Hence, as it follows from Eqs. (4.5) and (4.6), to predict the effective behaviour of the dispersion under study, with fluctuations of the appropriate random fields taken into account, it is necessary that the c.d.f. $F_\tau(A)$ of the random variable τ, defined by Eq. (2.4), be evaluated.

5. EVALUATION OF THE DISTRIBUTION FUNCTION $F_\tau(A)$

Let

$$f_\tau(u) = \frac{\mathrm{d}F_\tau(u)}{\mathrm{d}u} \tag{5.1}$$

be the probability density function (p.d.f.) of the random variable $\tau = |\nabla\theta(0)|^2$. Then the moments of τ read

$$t_p = \langle \tau^p \rangle = \int_0^\infty u^p f_\tau(u)\,\mathrm{d}u. \tag{5.2}$$

The integration is over $(0, \infty)$ since, obviously, the random variable τ is nonnegative and hence both $f_\tau(u)$ and $F_\tau(u)$ vanish if $u < 0$.

To find the moments t_p, the representation (2.6) of τ is to be used together with the formulae

$$\langle \psi'(\mathbf{y}_1)\ldots\psi'(\mathbf{y}_k)\rangle = n\delta(\mathbf{y}_1 - \mathbf{y}_2)\ldots\delta(\mathbf{y}_1 - \mathbf{y}_2) + o(n), \tag{5.3}$$

$k = 2, 3, \ldots$, which generalize Eq. (2.5) in an obvious manner, see [9]. The final result reads

$$\begin{aligned}
&t_0^{(1)} = t_1^{(1)} = 0, \quad t_p = \widetilde{G}^{2p} + \eta_f t_p^{(1)} G^{2p}, \\
&\widetilde{G}^{2p} = G^{2p}(1 + 6p\beta^2\eta_f), \\
&t_p^{(1)} = \sum_{k=0}^{p} \frac{1}{3^k} C_k^p \left[(\beta - 2)^k + 3I_k\right]^k, \quad p \geq 2, \\
&I_k = \frac{1}{3}\int_0^1 \mathrm{d}z \int_1^\infty \frac{\mathrm{d}\rho}{\rho^{6k-2}} \left[\beta + 2\rho^3 + 3(\beta - 2\rho^3)z^2\right]^k \\
&\quad = \frac{1}{3}\sum_{l=0}^{k} \frac{2^l \beta^{k-1}}{2k - l - 1} C_l^k \int_0^1 (1 + 3z^2)^{k-l}(1 - \beta z^2)^l\,\mathrm{d}z.
\end{aligned} \tag{5.4}$$

Knowledge of the moments t_p allows us to evaluate the Laplace transform of $f_\tau(u)$:

$$\overline{f}_\tau(s) = \mathcal{L}[f_\tau](s) = \int_0^\infty \mathrm{e}^{-su} f_\tau(u)\,\mathrm{d}u = \overline{f}_\tau^{(0)}(s) + \eta_f \overline{f}_\tau^{(1)}(s),$$

$$\overline{f}_\tau^{(0)}(s) = \sum_{p=0}^{\infty} \frac{(-1)^p}{p!}(\widetilde{G}^2 s)^p = \mathrm{e}^{-\widetilde{G}^2 s}, \tag{5.5}$$

$$\overline{f}_\tau^{(1)}(s) = \sum_{p=0}^{\infty} \frac{(-1)^p}{p!}(\widetilde{G}^2 s)^p t_p^{(1)}.$$

Hence

$$f_\tau(u) = \delta(u - \widetilde{G}^2) + \eta_f f_\tau^{(1)}(u), \tag{5.6}$$

where

$$f_\tau^{(1)}(u) = \mathcal{L}^{-1}[\overline{f}_\tau^{(1)}](u) \tag{5.7}$$

is the inverse Laplace transform of the function $\overline{f}_\tau^{(1)}(s)$, defined in the last line of Eq. (5.5). Let us recall that all the foregoing formulae hold in the dilute limit, i.e. they are correct to the order $O(\eta_f)$ only.

Note that the formula (5.6) is fully natural — if $\eta_f = 0$, then $\widetilde{G} = G$, see Eq. (2.7), the medium is homogeneous so that $\nabla\theta(\mathbf{x}) \equiv \mathbf{G}$ and thus $\tau = |\nabla\theta(\mathbf{x})|^2 \equiv G^2$.

6. DISCUSSION

The formulae (5.4) — (5.7) specify, at least in principle, the function $f_\tau(u)$ and hence its primitive $F_\tau(u)$, see Eq. (5.1), i.e. the needed cumulative distribution function of the random variable τ, defined in Eq. (2.4). It is easily seen that the function $F_\tau(u)$ depends on the nondimensional ratio u/G^2, i.e. $F_\tau = F_\tau(u/G^2)$, and as a c.d.f. it monotonically increases, tending to 1 when $u/G^2 \to \infty$. The formulae (4.5) and (4.6) now give

$$\frac{\widetilde{\kappa}^*}{\kappa_m} = 1 + 3\beta\eta_f - \underline{\frac{3}{2}\left(1 - F_\tau(G_0^2/G^2)\right)} + o(\eta_f) \tag{6.1}$$

and the underlined term is just the result of fluctuations of the temperature gradient. When $G \to 0$, i.e. at $G \ll G_0$, κ^* tends to its classical value $\kappa_m(1 + 3\beta\eta_f)$, predicted by the Maxwell formula in the dilute limit. Therefore, the values of the effective conductivity and, more general, of the effective properties for a composite, represent but tangents to the appropriate "stress-strain" curves at the onset of loading. When G increases, the underlined term in Eq. (6.1) increases as well thus leading to progressively bigger deviation from the classical linear behaviour.

The aforesaid means that no matter how small is the macrostrain "loading" ($\mathbf{G}$ in our simplified context), imposed upon a micro-inhomogeneous medium, there will always appear zones of "plastic" yielding or deteriorated ones, due to fluctuations of the appropriate random fields. Hence such a random medium should show deviation

from linear behaviour from the *very onset* of loading. This represents the central conclusion of our study, which we have tried to quantify in the proposed scheme.

It is curious to point out immediately that the nonlinearity of stress-strain curves, even for very small strain, is an experimentally observed feature of all solids, as specially emphasized by Bell [12] as a result of his extensive and detailed review of experimental data in the last 300 years. (See, e.g., his words at the end of Ch. 2.6, p. 30: "One might dismiss nonlinearity of the response functions observed in the experiments of Hodgkinson and Dupin as merely an interesting historical development in the fields of solid mechanics, were it not for the fact that by the end of the 19th century the increasing accuracy of measurements and improved experiments demonstrated that that was indeed the precise manner in which such solids deformed.")

The proposed scheme possesses, however, certain drawbacks which should be explicitly pointed out and which make it only approximate, even in the simplest dilute case under study. The point is the following: When using ensemble averaging, one should consider a multitude of spatial realizations of the array of spheres in the dispersion. When the spheres in a given realization are "well-separated," then each one can be indeed treated as single, immersed into unbounded matrix material. The behaviour of both spheres and matrix is then linear under the condition (3.2). There will be however specific realizations of a "clustering" type, so to say, when some of the spheres are close one to another; in this case the "stress-concentration" factor may become much higher than 2. For these realizations the "plastic" or deteriorated zones will be considerable and the behaviour will not be linear already. Moreover, the number of such "clustering" realizations is not negligible, even at small G/G_0, since they influence the overall response when averaging over the set of all realizations. This means that the basic equation (1.1) should be considered as a nonlinear and random one, with $\kappa(\mathbf{x})$ replaced by $\widetilde{\kappa}(\mathbf{x})$, see Eq. (3.1), whatever the value of G. The representation (2.1), which has served as a basis of our analysis, can be viewed then as a certain approximation which allows solely to highlight the role of the fluctuations on the overall behaviour of the composite. Despite this, its adoption seems unavoidable in the proposed scheme, because it is not clear, at least to the authors, how a problem of the type (1.1) with a discontinuous coefficient $\widetilde{\kappa}(\mathbf{x})$ can be efficiently treated in the random case.

Acknowledgements. The support of the Bulgarian Ministry of Education and Science under Grant No MM 805-98 is gratefully acknowledged.

REFERENCES

1. Nemat-Nasser, S., M. Hori. Micromechanics: Overall Properties of Heterogeneous Solids. Elsevier, 1993.
2. Sanchez-Palencia, E. Non-Homogeneous Media and Vibration Theory. Springer-Verlag, New York, 1980.
3. Beran, M. Statistical Continuum Theories. John Wiley, New York, 1968.
4. Christov, C. I., K. Z. Markov. Stochastic functional expansion for random media with perfectly disordered constitution. *SIAM J. Appl. Math.*, **45**, 1985, 289-311.

5. Markov, K. Z. On the factorial functional series and their application to random media. *SIAM J. Appl. Math.*, **51**, 1991, 128–139.
6. Beran, M., J. J. McCoy. Mean field variation in random media. *Q. Appl. Math.*, **28**, 1970, 245–258.
7. Bobeth, M., G. Diener. Field fluctuations in multicomponent mixtures. *J. Mech. Phys. Solids*, **34**, 1986, 1–17.
8. Vanmarcke, E. Random Fields: Analysis and Synthesis. MIT Press, Cambridge, Massachusetts and London, England, 1983.
9. Stratonovich, R. L. Topics in Theory of Random Noise. Vol. 1. Gordon and Breach, New York, 1963.
10. Markov, K. Z. On the heat propagation problem for random dispersions of spheres. *Math. Balkanica* (*New Series*), **3**, 1989, 399–417.
11. Adler, R. The Geometry of Random Fields. John Wiley, New York, 1981.
12. Bell, J. F. Mechanics of solids. I. Experimental foundations of mechanics of solids. In: Encyclopedia of Physics, Vol. VIa/1, ed. Truesdell, C., Springer-Verlag, Berlin-Heidelberg-New York, 1973.

Received February 25, 1999

Keranka S. ILIEVA
Faculty of Mathematics and
Informatics, Shoumen University
BG-9712 Shoumen
Bulgaria

Konstantin Z. MARKOV
Faculty of Mathematics and Informatics,
"St. Kliment Ohridski" University of Sofia
BG-1164 Sofia, Bulgaria
E-mail: kmarkov@fmi.uni-sofia.bg

ГОДИШНИК НА СОФИЙСКИЯ УНИВЕРСИТЕТ „СВ. КЛИМЕНТ ОХРИДСКИ“

ФАКУЛТЕТ ПО МАТЕМАТИКА И ИНФОРМАТИКА
Книга 2 — Приложна математика и информатика
Том 92, 1998

ANNUAIRE DE L'UNIVERSITE DE SOFIA „ST. KLIMENT OHRIDSKI“

FACULTE DE MATHEMATIQUES ET INFORMATIQUE
Livre 2 — Mathématiques Appliquée et Informatique
Tome 92, 1998

A NOTE ON THE BULK MODULUS OF A BINARY ELASTIC MIXTURE

M. K. KOLEV and K. Z. MARKOV

The Hashin-Shtrikman and Walpole bounds on the effective bulk modulus of a binary elastic mixture are revisited. A simple method of derivation is given as a generalization of the approach, recently proposed by one of the authors in the absorption and scalar conductivity problems for a two-phase medium.

Keywords: two-phase random media, effective bulk modulus, variational estimates, Hashin-Shtrikman and Walpole bounds

1991/1995 Math. Subject Classification: 73B35, 73S10, 60G60

The aim of this note is to present and discuss a simple derivation of the well-known two-point estimates on the effective bulk modulus of a binary elastic mixture, due to Hashin and Shtrikman [1] and Walpole [2]. The basic idea is a straightforward generalization of the approach, used by one of the authors in the absorption and scalar conductivity cases [3].

Assume that the mixture is statistically homogeneous and isotropic. Let

$$\chi_i(x) = \begin{cases} 1, & \text{if } x \in \Omega_i, \\ 0, & \text{otherwise,} \end{cases} \tag{1}$$

be the characteristic function of the region Ω_i, occupied by one of the constituents, labelled 'i', $i = 1, 2$, so that $\chi_1(x) + \chi_2(x) = 1$. Hereafter, all quantities, pertaining to the region Ω_1 or Ω_2, are supplied with the subscript '1' or '2', respectively.

The statistical properties of the medium follow from the set of multipoint moments of one of the functions $\chi_i(x)$, say $\chi_2(x)$, for definiteness, or, which is the

same, by the volume fraction $\eta_2 = \langle \chi_2(x) \rangle$ of the phase '2', and the multipoint moments

$$M_2(x) = \langle \chi_2'(0)\chi_2'(x) \rangle\,, \; M_3(x,y) = \langle \chi_2'(0)\chi_2'(x_1)\chi_2'(y) \rangle\,, \ldots\,, \tag{2}$$

with $\chi_2'(x) = \chi_2(x) - \eta_2$ being the fluctuating part of the field $\chi_2(x)$, see, e.g. [4]. The angled brackets $\langle \cdot \rangle$ hereafter denote ensemble averaging. One point could be taken at the origin, because of the assumed statistical homogeneity, as already done in (2).

Assuming also the constituents isotropic, the fourth-rank tensor of elastic moduli of the medium, $\mathbf{L}(x)$, is a random field of the familiar form

$$\begin{aligned} &\mathbf{L}(x) = 3k(x)\mathbf{J}' + 2\mu(x)\mathbf{J}'', \\ &k(x) = k_1\chi_1(x) + k_2\chi_2(x) = \langle k \rangle + [k]\chi_2'(x), \\ &\mu(x) = \mu_1\chi_1(x) + \mu_2\chi_2(x) = \langle \mu \rangle + [\mu]\chi_2'(x), \end{aligned} \tag{3}$$

where k and μ stand, as usual, for the bulk and shear modulus, respectively. The square brackets denote the jumps of the appropriate quantities, say, $[k] = k_2 - k_1$, $[\mu] = \mu_2 - \mu_1$, etc. In Eq. (3), $\mathbf{J}'$ and $\mathbf{J}''$ are the basic isotropic fourth-rank tensors with the Cartesian components

$$J'_{ijkl} = \frac{1}{3}\delta_{ij}\delta_{kl}, \quad J''_{ijkl} = \frac{1}{2}\Big(\delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk} - \frac{2}{3}\delta_{ij}\delta_{kl}\Big). \tag{4}$$

The displacement field $u(x)$ in the medium, at the absence of body forces, is governed by the well-known equations

$$\begin{aligned} &\nabla \cdot \boldsymbol{\sigma}(x) = 0, \\ &\boldsymbol{\sigma}(x) = \mathbf{L}(x) : \boldsymbol{\varepsilon}(x) = k(x)\theta(x)\mathbf{I} + 2\mu(x)\mathbf{d}(x), \\ &\boldsymbol{\varepsilon} = \frac{1}{2}(\nabla u + u\nabla), \quad \mathbf{d}(x) = \boldsymbol{\varepsilon}(x) - \frac{1}{3}\theta(x)\mathbf{I}, \end{aligned} \tag{5}$$

where $\boldsymbol{\sigma}$ denotes the stress tensor, $\boldsymbol{\varepsilon}$ is the small strain tensor, generated by the displacement field $u(x)$, $\mathbf{d}$ is the strain deviator, and $\theta = \operatorname{tr}\boldsymbol{\varepsilon}$ is the volumetric strain. The colon designates contraction with respect to two pairs of indices and $\mathbf{I}$ is the unit second-rank tensor.

The system (5) is supplied with the condition

$$\langle \boldsymbol{\varepsilon}(x) \rangle = \mathbf{E}, \tag{6}$$

prescribing the macroscopic strain tensor $\mathbf{E}$, imposed upon the medium.

Recall [4] that the random problem (5), (6) is equivalent to the variational principle of classical type:

$$\begin{aligned} &W[\boldsymbol{\varepsilon}(x)] = \langle \boldsymbol{\varepsilon}(x) : \mathbf{L}(x) : \boldsymbol{\varepsilon}(x) \rangle \to \min, \\ &\min W = \mathbf{E} : \mathbf{L}^* : \mathbf{E}. \end{aligned} \tag{7}$$

The energy functional W is considered over the class of random fields $u(x)$ that generate strain fields $\varepsilon(x)$, complying with the condition (6). In Eq. (7), $\mathbf{L}^*$ is the tensor of effective elastic moduli for the medium which, in the isotropic case under study, has the form

$$\mathbf{L}^* = 3k^*\mathbf{J}' + 2\mu^*\mathbf{J}'', \tag{8}$$

where k^* and μ^* are the effective bulk and shear modulus of the mixture, respectively.

Consider, guided by [3], the class of trial fields for the variational principle (7):

$$\mathcal{K}^{(1)} = \left\{ \widetilde{u}(x) \mid \widetilde{u}(x) = \mathbf{E}\cdot x - \alpha \int \nabla G(x-y)\chi_2'(y)\, d^3y \right\}, \tag{9}$$

having assumed that $\mathbf{E}$ is spherical

$$\mathbf{E} = \frac{1}{3}\mathbf{I}, \quad G(x) = \frac{1}{4\pi|x|}, \tag{10}$$

and α is an adjustable scalar parameter. Hereafter the integrals are over the whole $\mathbb{R}^3$ if the integration domain is not explicitly indicated.

The energy functional W, when restricted over $\mathcal{K}^{(1)}$, becomes a quadratic function of α:

$$\begin{aligned} &W[\widetilde{u}(x)] = A - 2B\alpha + C\alpha^2, \quad A = \langle k \rangle, \quad B = [k]M_2(0), \\ &C = \langle \lambda \rangle M_2(0) + [\lambda]M_3(0,0) + 2\langle \mu \rangle P_2 + 2[\mu]P_3, \end{aligned} \tag{11}$$

with the dimensionless statistical parameters for the medium, defined as follows:

$$\begin{aligned} P_2 &= \iint \nabla\nabla G(y_1) : \nabla\nabla G(y_2)\, M_2(y_1 - y_2)\, d^3y_1 d^3y_2, \\ P_3 &= \iint \nabla\nabla G(y_1) : \nabla\nabla G(y_2) M_3(y_1, y_2)\, d^3y_1 d^3y_2; \end{aligned} \tag{12}$$

$\lambda = k - \frac{2}{3}\mu$ is the familiar Lamé constant.

Note that for the isotropic binary medium under study

$$M_2(0) = \langle \chi_2'^2(0) \rangle = \eta_1\eta_2, \quad M_3(0) = \langle \chi_2'^3(0) \rangle = \eta_1\eta_2(\eta_1 - \eta_2). \tag{13}$$

Moreover, the parameter P_2 can be easily evaluated, having integrated by parts and noting that $G(x)$ is the well-known Green function for the Laplacian:

$$P_2 = M_2(0) = \eta_1\eta_2. \tag{14}$$

The variational principle (7), together with (11), implies

$$k^* \le W[\widetilde{u}(x)] = A - 2B\alpha + C\alpha^2, \quad \forall\alpha. \tag{15}$$

In particular, at $\alpha = 0$ one has

$$k^* \leq \langle k \rangle \tag{16}$$

which, obviously, is the elementary (Voigt) bound on k^*.

Next, optimizing (15) with respect to α, one gets another estimate on k^*:

$$k^* \leq A - \frac{B^2}{C}, \tag{17a}$$

i.e.

$$k^* \leq \langle k \rangle - \frac{\eta_1 \eta_2 [k]^2}{\langle \lambda + 2\mu \rangle + ([\lambda] + 2[\mu] I_3)(\eta_1 - \eta_2)}, \tag{17b}$$

where

$$I_3 = \frac{1}{\eta_1 \eta_2 (\eta_1 - \eta_2)} P_3 \tag{18}$$

is the statistical parameter that appears in the perturbation expansion of κ^* for a weakly inhomogeneous medium, see [5], and also in the Beran's bound on the effective conductivity constant [6]. A simple check shows that (18) coincides with the upper bound on k^*, due to Beran and Molyneux (BM) [7].

The main problem in specifying the bound (17b) is just the three-point parameter I_3 whose evaluation for special and realistic random constitution is nontrivial. Recall that in many cases it is more convenient to employ, instead of I_3, the Torquato-Milton parameter ζ_1, see [8, 9], defined as a certain integral, similar to P_3 (see, e.g. the Torquato review [10]). Without going into detail, we shall only point out the formula

$$3(\eta_2 - \eta_1) I_3 = 2\zeta_1 + 3\eta_1 - \eta_2. \tag{19}$$

The bound (18) should be at least as good as the elementary bound (16) (since the energy functional is minimized over a broader class of trial fields). This implies that

$$C > 0, \quad AC - B^2 \geq 0, \tag{20}$$

because $k^* \geq 0$. Since $A = \langle k \rangle > 0$, $C \geq B^2/A > 0$, which means that the second inequality in (20) is the stronger one. Using the expressions for A, B and C from (11), we can write the latter in the form

$$\langle \lambda + 2\mu \rangle + ([\lambda] + 2[\mu] I_3)(\eta_1 - \eta_2) - \frac{[k]^2}{\langle k \rangle} \eta_1 \eta_2 \geq 0. \tag{21}$$

The inequality (21) should hold for every "realistic" choice of the elastic moduli of the constituents (i.e. for which the appropriate elastic energy is positive-definite). This implies

$$\frac{1}{3}\eta_1 - \eta_2 \leq (\eta_1 - \eta_2) I_3 \leq \eta_1 - \frac{1}{3}\eta_2. \tag{22}$$

Note that (22) drastically simplifies when the parameter ζ_1 is used instead of I_3, see (19). Namely, it states then that $0 \leq \zeta_1 \leq 1$, which is a well-known fact [8, 9].

However, keeping I_3 in the BM-bound (17b) has its advantages. Namely, by means of (22) we can exclude the product $(\eta_1 - \eta_2)I_3$ from this bound. Depending on the sign of $[\mu] = \mu_2 - \mu_1$, we should use to this end the upper or lower bound (22). The final result reads

$$\begin{aligned} k^* &\le \langle k \rangle - \frac{\eta_1 \eta_2 [k]^2}{\lambda_1 + 2\mu_1 + \eta_1 [k]}, \quad \text{if } \mu_2 \le \mu_1, \\ k^* &\le \langle k \rangle - \frac{\eta_1 \eta_2 [k]^2}{\lambda_2 + 2\mu_2 - \eta_2 [k]}, \quad \text{if } \mu_2 \ge \mu_1. \end{aligned} \tag{23}$$

In the so-called "well-ordered" case, when $(k_2 - k_1)(\mu_2 - \mu_1) > 0$, the first of the estimates (23) coincides with the Hashin-Shtrikman (HS) bound on k^*, provided that not only $\mu_2 \le \mu_1$, but also $k_2 \le k_1$, see [1]. This unnecessary restriction was removed by Walpole [2]. It is easily seen that our bounds (23) are just the Walpole bounds in which no requirements are put on the sign of $k_2 - k_1$.

The derivation of the lower bound, corresponding to (23), is fully similar. In this case we write the elastic energy (7) by means of the stress tensor:

$$\begin{aligned} W[\boldsymbol{\sigma}(x)] &= \left\langle \boldsymbol{\sigma}(x) : \mathbf{L}^{-1}(x) : \boldsymbol{\sigma}(x) \right\rangle \to \min, \\ \min W &= \boldsymbol{\Sigma} : \mathbf{L}^{*-1} : \boldsymbol{\Sigma}. \end{aligned} \tag{24}$$

The functional W is considered now over the class of trial fields such that

$$\nabla \cdot \boldsymbol{\sigma}(x) = 0, \quad \langle \boldsymbol{\sigma}(x) \rangle = \boldsymbol{\Sigma}, \tag{25}$$

with a prescribed macrostress tensor $\boldsymbol{\Sigma}$ imposed upon the medium.

The functional W in (24) is minimized now over the class of trial fields

$$\mathcal{N}^{(1)} = \left\{ \tilde{\boldsymbol{\sigma}}(x) \,\middle|\, \tilde{\boldsymbol{\sigma}}(x) = \boldsymbol{\Sigma} + \alpha \left[\int \nabla\nabla G(x - y) \chi_2'(y)\, d^3 y + \mathbf{I} \chi_2'(y) \right] \right\} \tag{26}$$

with the spherical $\boldsymbol{\Sigma} = \frac{1}{3}\mathbf{I}$ and an adjustable scalar parameter α, $G(x)$ being the function defined in (10). The straightforward manipulations are omitted and the final result reads

$$\begin{aligned} k^* &\ge \langle k \rangle - \frac{\eta_1 \eta_2 [k]^2}{\lambda_2 + 2\mu_2 - \eta_2 [k]}, \quad \text{if } \mu_2 \le \mu_1, \\ k^* &\ge \langle k \rangle - \frac{\eta_1 \eta_2 [k]^2}{\lambda_1 + 2\mu_1 + \eta_1 [k]}, \quad \text{if } \mu_2 \ge \mu_1. \end{aligned} \tag{27}$$

The inequalities (27), combined with (23), are just the Walpole bounds on the effective bulk modulus of a binary mixture, see [2] and also [11], which are a direct generalization of the Hashin-Shtrikman result with the condition of "well-orderness" removed. Here we have demonstrated how this classical estimate shows up simply and naturally within the frame of the general method recently developed by one of the authors [3] in the absorption and scalar conductivity contexts.

Acknowledgements. The support of the Bulgarian Ministry of Education and Science under Grant No MM 805-98 is gratefully acknowledged.

REFERENCES

1. Hashin, Z., Shtrikman, S. A variational approach to the theory of the elastic behaviour of multiphase materials. *J. Mech. Phys. Solids*, **11**, 1963, 127–140.
2. Walpole, L. On the overall elastic moduli of composite materials. *J. Mech. Phys. Solids*, **17**, 1969, 235–251.
3. Markov, K. Z. On the correlation functions of two-phase random media and related problems. *Proc. Roy. Society London A*, **455**, 1999, 1069–1086.
4. Beran, M. Statistical Continuum Theories. John Wiley, New York, 1968.
5. Brown, W. F. Solid mixture permittivities. *J. Chem. Phys.*, **23**, 1955, 1514–1517.
6. Beran, M. Use of a variational approach to determine bounds for the effective permittivity of a random medium. *Nuovo Cimento*, **38**, 1965, 771–782.
7. Beran, M., J. Molyneux. Use of classical variational principles to determine bounds for the effective bulk modulus in heterogeneous medium. *Q. Appl. Math.*, **24**, 1966, 107–118.
8. Torquato, S. Microscopic approach to transport in two-phase random media. Thesis, State University of New York at Stony Brook, 1980.
9. Milton, G. W. Bounds on the electromagnetic, elastic and other properties of two-component composites. *Phys. Review Lett.*, **46**, 1981, 542–545.
10. Torquato, S. Random heterogeneous media: Microstructure and improved bounds on effective properties. *Appl. Mech. Rev.*, **44**, 1991, 37–76.
11. Walpole, L. Elastic behavior of composite materials: Theoretical foundations. In: *Adv. Appl. Mechanics*, C. Yih, ed., vol. 21, Academic Press, New York, 1981, 169–242.

Received March 23, 1999

Mikhail K. KOLEV
Faculty of Mathematics and
Informatics, East-West University
Blagoevgrad, Bulgaria
E-mail: mkkolev@aix.swu.bg

Konstantin Z. MARKOV
Faculty of Mathematics and Informatics
"St. Kliment Ohridski" University of Sofia
BG-1164 Sofia, Bulgaria
E-mail: kmarkov@fmi.uni-sofia.bg

ERRATA

In the article "A POLYNOMIAL PROBLEM" by Pavel G. Todorov, vol. **91**, 1, 1999, 21–32, on the right-hand side of equation (6) on page 22 to the determinant it has to be added " $= 0$".

Submission of manuscripts. The *Annuaire* is published once a year, in two parts: Part I. Mathematics and Mechanics, and Part II. Applied Mathematics and Informatics. No deadline exists. Once received by the editors, the manuscript will be subjected to rapid, but thorough review process. If accepted, it is immediately sheduled for the nearest forthcoming issue. No page charge is made. The author(s) will be provided with a total of 30 free of charge offprints of their paper.

The submission of a paper implies that it has not been published, or is not under consideration for publication elsewhere. In case it is accepted, it implies as well that the author(s) transfers the copyright to the Faculty of Mathematics and Informatics at the "St. Kliment Ohridski" University of Sofia, including the right to adapt the article for use in conjuction with computer systems and programs and also reproduction or publication in machine-readable form and incorporation in retrieval systems.

Instructions to Contributors. Preferences will be given to papers, not longer than 15 to 20 pages, written preferably in English and typeset by means of a TEX system. A simple specimen file, exposing in detail the instruction for preparing the manuscripts, is available upon request from the electronic address of the Editorial Board. Two copies of the manuscript should be submitted. Upon acceptance of the paper, the authors will be asked to send by electronic mail or on a diskette the text of the papers and the appropriate graphic files (in any format like `*.tif`, `*.pcx`, `*.bmp`, etc.).

The manuscripts should be prepared for publication in accordance with the instructions, given below.

The manuscripts must be *typed* on one side of the paper in double spacing with wide margins. On the *first* page the author should provide: a title, name(s) of the author(s), a short abstract, a list of keywords and the appropriate 1995 Mathematical Subject Classification codes (primary and secondary, if necessary). The affiliation(s), including the electronic address, is given at the end of the manuscripts.

Figures have to be inserted in the text near their first reference. If the author cannot supply and/or incorporate the graphic files, drawings (in black ink and on a good quality paper) should be enclosed separately. If photographs are to be used, only black and white ones are acceptable.

Tables should be inserted in the text as close to the point of reference as possible. Some space should be left above and below the table.

Footnotes, which should be kept to a minimum and should be brief, must be numbered consecutively.

References must be cited in the text in square brackets, like [3], or [5, 7], or [11, p. 123], or [16, Ch. 2.12]. They have to be numbered either in the order they appear in the text or alphabetically. Examples (please note order, style and punctuation):

For books: Obreshkoff, N. Higher algebra. Nauka i Izkustvo, 2nd edition, Sofia, 1963 (in Bulgarian).

For journal articles: Frisch, H. L. Statistics of random media. *Trans. Soc. Rheology*, 9, 1965, 293–312.

For articles in edited volumes or proceedings: Friedman, H. Axiomatic recursive function theory. In: *Logic Colloquim 95*, eds. R. Gandy and F. Yates, North-Holland, 1971, 188–195.