On machines as living things*

By Le Hoi

I. Introduction

As it is known Von Neumann in [9] considered environment as tessellation structure. The tessellation is a mathematical system to model a behaviour and structure of uniformly interconnected identical finite automata, processing information as the result of local functions acting simultaneously throughout the array on the states of the interconnected automata. Von Neumann [9], J. Thatcher [8] E. F. Codd [4], A. Smith [7] and M. A. Arbib [1—3] considered machines only self-reproducing in tessellation without metabolism, adaptation, evolution etc.

Here we consider environment as modular space.

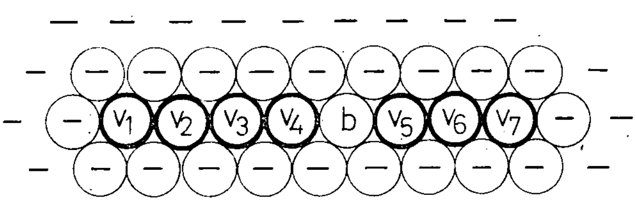

Fig. 1

In Figure 1 v_i representing a module (in state $v_i \in V$) of "solid sub-volume" is considered as a "molecule" of the solid sub-volume embedded in "fluid environment". Moreover, — representing a "raw module" is considered as a free molecule in fluid environment. Every module can change its state depending on its present state and the state of its neighbourhood. But the difference between Von Neumann's tessellation and our modular space is that positions of modules in tessellation are fixed,

* Presented at the Conference on Automata and Formal Languages, Szeged (August 30—September 2, 1977).

but modules in our modular space can move depending on their present neighbourhoods.

For one-dimensional solid volume, we denote its configuration in the environment (like in Fig. 1) by $(\overline{v_1\, v_2\, v_3\, v_4\, b\, v_5\, v_6\, v_7})$ or shortly by the word $v_1\, v_2\, v_3\, v_4\, b\, v_5\, v_6\, v_7$. The ring $(\overline{\quad\quad})$ indicates raw modules surrounding the solid volume.

II. Main problems

We define, formalise and construct some kind of universal environments enough for those machines (solid volumes embedded in them), which in the environment not only compute all of the partial recursive functions and are self-reproducing as in Von Neumann's tessellation, but also have some important other characteristics such as growth, death, adaptation and mutation.

III. Main notion and main results

It is shown that the environments which can be formalized by the so-called "parallel exchanging system" (P. E. System) are enough for our above mentioned requirements.

Definition. An RE-System is a triple $S=\langle V, F, b\rangle$ where V is a vocabulary, $b\in V$ is called the "blank", F is a finite non-empty set of productions of the form $(\alpha\bar{v}\beta, \gamma)$, $v\in V$; $\alpha, \beta, \gamma\in V^*$ (that means, v in the neighbourhood $(\alpha^-\beta)$ is replaced by γ) with the following conditions

1) If $(\alpha\bar{v}\beta, \gamma_1)\in F$ and $(\alpha\bar{v}\beta, \gamma_2)\in F$ then $\gamma_1=\gamma_2$.
2) If $(\alpha_1\bar{v}\beta_1, \gamma_1)\in F$ and $(\alpha_2\bar{v}\beta_2, \gamma_2)\in F$ then $|\alpha_1|=|\alpha_2|$ and $|\beta_1|=|\beta_2|$ ($|\alpha|$ is the length of α).
3) Productions of b are only of the forms $(\alpha\bar{b}\beta, v)$, $|v|=1$ or $(\alpha\bar{b}\beta, b^n)$ where $\forall_m\, \alpha\neq b^m$ or $\forall_m\, \beta\neq b^m$ and $n, m\in\{0, 1, 2, 3, \dots\}$.

When defining relations on V^* "x directly generates y", written $x\vdash_S y$, "x generates y in k-step", written $x\overset{k}{\underset{S}{\Rightarrow}} y$, and "$x$ terminally generates y" written $x \mid\!\underset{S}{\Rightarrow} y$, productions are applied simultaneously.

Theorem 1. The class of "stability function" $h\colon V^*\to V^*$ $(y=h(x)$ iff $x\mid\!\underset{S}{\Rightarrow} y)$ in all PE-Systems $S=\langle V, F, b\rangle$ is a proper subclass of partial recursive functions on V^*.

We can formalise the required environments $E=\langle A, X, F, b\rangle$ as a special kind of PE-System $S=\langle V, F, b\rangle$ where $V=A\cup X$, $b\in X$ and

$$(\alpha\bar{v}\beta, \gamma)\in F \quad |\alpha|\leqq 2, \quad |\beta|\leqq 2, \quad |\gamma|\leqq 2.$$

Definitions. A modular machine (M-machine) Z in an environment $E=\langle A, X, F, b\rangle$ has the following elements:
— the signal to begin working a_0
— the signal to stop working $*$

— the body α containing a program
— an input tape x, output tape y
where a_0, $*$, α are distinguished strings of modules in states from A; x, y are strings of modules in states from X.

When beginning to work the modular machine Z has an initial configuration interpreted as the string of modules $a_0\alpha x$ surrounded by raw modules. Denote this by $(\overline{a_0\alpha x})$. Each module of the machine can change its state to a determined state; or can either become a raw module and go off the machine, or can change its state and simultaneously splice (take in) one raw module $\ominus$ above it onto its left. The behaviour of modules defined by productions F is such that the initial configuration $(\overline{a_0\alpha x})$ can enter the "terminal static configuration" $(\overline{\beta b * \gamma y})$ with $\beta, \gamma \in A^*$, $b=$blank, $y \in X^*$, $a_0 \not\subset \gamma$, $* \not\subset \gamma$ and if $\beta \vdash_E \beta'$ $(*\gamma) \vdash_E (*\gamma)'$ then $\beta b * \gamma y \vdash_E \beta' b(*\gamma)'y$. Furthermore, the machine is always surrounded by raw modules as a solid volume embedded in liquid environment E, and we write

$$a_0\alpha x \underset{E}{\Rightarrow} \beta b * \gamma y.$$

In this case we say the M-machine $Z=a_0\alpha$ or $Z=\langle \alpha, a_0, * \rangle$ in $E=\langle A, X, F, b\rangle$ (denoted by $\langle \alpha, a_0, * \rangle$ in $\langle A, X, F, b\rangle$) transforms x into y (or computes $y=F_z^b(x)$), reproduces β and modifies the program in α to the program in γ, and also write

$$a_0\alpha x \underset{F_z^b}{\Longrightarrow} \beta b * \gamma y$$

if $\beta b * \gamma y$ is the first configuration of this form derived from $a_0\alpha x$.

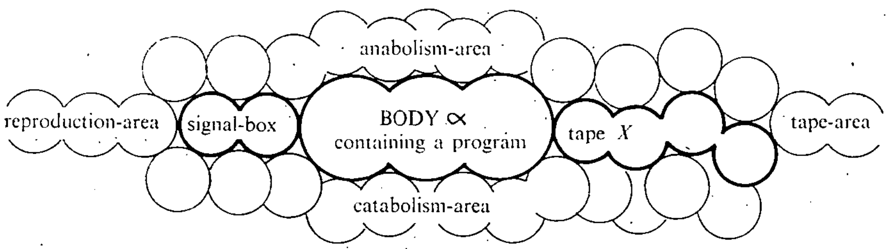

Fig. 2
M-machine Z in E

If product β also is a modular-machine then we say that machine $a_0\alpha$ is a computation-organism (C-organism). If product β equals $a_0\alpha$ or $*\alpha$ then we say that machine $a_0\alpha$ in E is *self-reproducing*. If $|\gamma|>|\alpha|$ and $a_0\alpha$ is also a C-organism in E then $\langle \alpha, a_0, * \rangle$ in E *is growing*. If $|\gamma|<|\alpha|$ then C-organism $\langle \alpha, a_0, * \rangle$ in E is *degenerating*. A C-organism $\langle \alpha, a_0, * \rangle$ in $\langle A, X, F, b\rangle$ is said *to die by* x after computing $y=F_Z^b(x)$ if $a_0\alpha x \overset{Z}{\Rightarrow} *\alpha b * \gamma y$ but $\forall x' \in X^*$: $a_0\gamma x' \not\Rightarrow a_0\gamma x'$, that is $Z=a_0\alpha$ is no longer active after interacting with x.

If γ is a function of y (and α) such that $a_0\gamma$ is still an M-machine then $Z=a_0\alpha$ is said to be *adaptive*. If $\langle \alpha, a_0, * \rangle$ is a C-organism and $\exists x \in X^*$ such that β is also an M-machine but $\beta \neq a_0\alpha$ and $\beta \neq *\alpha$ then $\langle \alpha, a_0, * \rangle$ in E is an *M-machine with mutation*.

Theorem 2. There exists a universal environment $E_{c.u}$ in the sense that for every partial recursive function f we can construct an M-machine Z in $E_{c.u}$ to compute f.

Corollary. The class of partial recursive functions coincides with that of parallelly computable functions of modular-machines.

Some theorems show an existence and how to construct the universal environments for growing machines, for self-reproducing, for degenerating, for going to death after a number of computations or for all of them.

Notation. Let $Z=\langle \alpha_0, a_0, *\rangle$ in $\langle A, X, F, b\rangle$ be an adaptive machine and

$$a_0\alpha_0 x_1 \underset{F^b_{\alpha_0}}{\Longrightarrow} \beta_1 b * \gamma_1 y_1$$

$$a_0\gamma_1 x_2 \underset{F^b_{\gamma_1}}{\Longrightarrow} \beta_2 b * \gamma_2 y_2, \ldots, a_0\gamma_{n-1} x_n \underset{F^b_{\gamma_{n-1}}}{\Longrightarrow} \beta_n b \gamma_n y_n,$$

and $a_0\gamma_n$ be a M-machine in $\langle A, X, F, b\rangle$. Then we denote the M-machine $a_0\gamma_n$ in E by $Z(x_1, x_2, \ldots, x_n)$.

Theorem 3. There exists a universal environment for adaptive C-organisms Z's in which every $Z(x_1, x_2, \ldots, x_n)$ also is adaptive and self-reproducing if domain of $F^b_{Z(x_1, x_2, \ldots, x_n)}$ is non-empty.

Theorem 4. There exists a universal environment for adaptive C-organism with mutations Z's (i.e., Z is adaptive and also is with mutation), and if $a_0\alpha x \overset{Z}{\Rightarrow} *\beta b * \gamma y$ then $a_0\beta$ and $a_0\gamma$ also are adaptive C-organisms with mutation (if their domains, Dom F^b_Z, are non-empty) and β is a function of (α, y).

Two last theorems say that by "adaptation" and "mutation" C-organisms in evolution modify their programs in α depending on α and new situation y in the environment and then transmit the new genetic programs in γ to their offspring β.

IV. Conclusion

By tessellation structure, Von Neumann, Thatcher, Codd, Smith, Arbib were concerned with only self-reproducing machines. Professor Pawlak [5] introduced the model of stored program computer only with modification of instructions. René Thom's theory of development and morphogenesis concernes the systematic continuous-topological approach (cf. [6]). Here, by means of PE-System, we introduced a new mathematical model of computing machines not only self-reproducing but with some other essential characteristics of living things, and we showed universal environments for such machines. Since modules in tessellation can not move, self-reproductions and movements in tessellation are rather of configurations, pictures (of machines) than of machines themselves. In our modular space, self-reproduction, adaptation, movement are of modular-machines themselves.

DAI HOC THUONG NGHIEP
(Ng. th. Minh)
HANOI, VIETNAM

References

[1] ARBIB, M. A., *Brains, machines and mathematics,* McGraw-Hill Book Company, New York, 1964.

[2] ARBIB, M. A., A simple self-reproducing universal automata, *Information and Control,* v. 9, No. 2, 1966.

[3] ARBIB, M. A., *Theories of abstract automata,* 1969.

[4] CODD, E. F., Propagation, computation, construction in 2-dimensional cellular spaces, *Technical Publication,* University of Michigan, 1965; Reprinted in 1968 as Cellular Automata, New York.

[5] PAWLAK, Z., A mathematical model of digital computers, *Proc. Conf. on Theory of Automata and Form. Lang.*, Springer-Verlag, Bonn, 1973.

[6] RENE THOM, Stabilité structurelle et morphogenese, *Essai d'une théorie génerale des modeles,* W. A. Benjamin, Inc., Reading, Mass., 1972.

[7] SMITH, A. R., Simple computation-universal cellular spaces and self-reproduction, *Conf. Record. IEEE 9th Ann. Symp.* Switching and Automata Theory, 1968.

[8] THATCHER, J. W., Universality in the Von-Neumann cellular model, *Technical Report,* University of Michigan.

[9] VON NEUMANN, Theory of self-reproducing automata, A. W. Burks Ed., 1966.

(Received Oct. 31, 1977)