Bimachines: an introduction

Pedro V. Silva

University of Porto

June 12, 2006
This is part of a joint work with John Rhodes summarized in the preprint *An algebraic analysis of Turing machines and Cook’s Theorem leading to a profinite fractal differential equation and a random walk on a deterministic Turing machine*.

The authors thank Jean-Camille Birget for his advice and helpful comments.
lp-mappings

$A_1, A_2$ - finite nonempty alphabets

$A_1^+ - free$ semigroup on $A_1$

$\alpha : A_1^+ \rightarrow A_2^+$ is an lp-mapping if

$$|\alpha(w)| = |w|$$

for every $w \in A_1^+$.
Factorizing the output

For $i = 1, \ldots, |w|$, write

$$w = \lambda_i(w) \cdot \sigma_i(w) \cdot \mu_i(w)$$

with $|\lambda_i(w)| = i - 1$, $\sigma_i(w) \in A_1$.

**Example:**

<table>
<thead>
<tr>
<th></th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>d</th>
<th>a</th>
<th>b</th>
<th>c</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\lambda_5(w)$</td>
<td>a</td>
<td>b</td>
<td>c</td>
<td>d</td>
<td>a</td>
<td>b</td>
<td>c</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>d</th>
<th>a</th>
<th>b</th>
<th>c</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\sigma_5(w)$</td>
<td>a</td>
<td>b</td>
<td>c</td>
<td>d</td>
<td>a</td>
<td>b</td>
<td>c</td>
</tr>
<tr>
<td>$\mu_5(w)$</td>
<td>a</td>
<td>b</td>
<td>c</td>
<td>d</td>
<td>a</td>
<td>b</td>
<td>c</td>
</tr>
</tbody>
</table>
A domain extension

The lp-mapping $\alpha : A_1^+ \rightarrow A_2^+$ induces a mapping

$$A_1^* \times A_1 \times A_1^* \rightarrow A_2$$

$$(u, a, v) \mapsto \sigma_{|u|+1} \alpha(uav)$$

<table>
<thead>
<tr>
<th>u</th>
<th>a</th>
<th>v</th>
</tr>
</thead>
<tbody>
<tr>
<td>(\alpha) \downarrow</td>
<td></td>
<td></td>
</tr>
<tr>
<td>\ldots</td>
<td>b</td>
<td>\ldots</td>
</tr>
</tbody>
</table>

$\alpha : A^+ \rightarrow A'^+$ is uniquely determined by

$\alpha(\_, \_, \_) : A^* \times A \times A^* \rightarrow A'$ and vice-versa.
A **right A-automaton** is a triple $\mathcal{A}_R = (I_R, Q_R, S_R)$ where

- $Q_R$ is a set
- $I_R \in Q_R$
- $S_R$ is an $A$-semigroup acting on $Q_R$ on the right, so

$$(q_R s_R) s'_R = q_R (s_R s'_R).$$

The action is *proper* ($I_R \notin I_R S_R$) but not necessarily *faithful* (different elements of $S_R$ may have the same action).

The action of $S_R$ on $Q_R$ induces an obvious action of $A^*$ on $Q_R$. 
Morphisms of right automata

Let

\[ A_R = (I_R, Q_R, S_R), \quad A'_R = (I'_R, Q'_R, S'_R) \]

be right \( A \)-automata. A morphism \( \varphi : A_R \to A'_R \) is defined, whenever \( S'_R \) is a quotient of \( S_R \), via a mapping \( \varphi : Q_R \to Q'_R \) such that

\[ \varphi(I_R) = I'_R; \]
\[ \varphi(q_R u) = \varphi(q_R) u \quad \text{for all} \quad q_R \in Q_R \quad \text{and} \quad u \in A^+. \]

This is equivalent to say that there exists a mapping on the states and an \( A \)-semigroup morphism preserving initial state and the action.
A left $A$-automaton is a triple $A_L = (S_L, Q_L, I_L)$ where

- $Q_L$ is a set
- $I_L \in Q_L$
- $S_L$ is an $A$-semigroup acting on $Q_L$ on the left, so

$$s_L(s'_L q_L) = (s_L s'_L)q_L.$$ 

The action is proper ($I_L \notin S_L I_L$) but not necessarily faithful.

The action of $S_L$ on $Q_L$ induces an obvious action of $A^*$ on $Q_L$. 

Pedro V. Silva  
Bimachines: an introduction
An $A_1, A_2$-bimachine is a structure of the form

$$\mathcal{B} = ((I_R, Q_R, S_R), f, (S_L, Q_L, I_L)),$$

where

- $(I_R, Q_R, S_R)$ is a right $A$-automaton;
- $(S_L, Q_L, I_L)$ is a left $A$-automaton;
- $f : Q_R \times A \times Q_L \rightarrow A'$ a full map (the output function).

We say that $\mathcal{B}$ is finite if both state sets and semigroups are finite.
Morphisms of bimachines

Let

\[ B = ((l_R, Q_R, S_R), f, (S_L, Q_L, l_L)) \]
\[ B' = ((l'_R, Q'_R, S'_R), f', (S'_L, Q'_L, l'_L)) \]

be \( A_1, A_2 \)-bimachines. We say that \( \varphi : B \to B' \) is a morphism of \( A_1, A_2 \)-bimachines if \( \varphi = (\varphi_R, \varphi_L) \), where

- \( \varphi_R : (l_R, Q_R, S_R) \to (l'_R, Q'_R, S'_R) \) is a morphism of right \( A \)-automata;
- \( \varphi_L : (S_L, Q_L, l_L) \to (S'_L, Q'_L, l'_L) \) is a morphism of left \( A \)-automata;
- \( \forall u, v \in A^* \forall a \in A \quad f'(l'_R u, a, vl'_L) = f(l_R u, a, vl_L) \).
From bimachines to lp-mappings

We associate an lp-mapping

$$\alpha_B : A_1^+ \rightarrow A_2^+$$

to the $A_1, A_2$-bimachine

$$\mathcal{B} = ((I_R, Q_R, S_R), f, (S_L, Q_L, I_L))$$

by

$$\alpha_B(u, a, v) = f(I_R u, a, v I_L) \quad (u, v \in A^*, a \in A).$$

Thus

$$\alpha_B(w) = \prod_{i=1}^{\mid w \mid} f(I_R \lambda_i(w), \sigma_i(w), \mu_i(w) I_L).$$
Let $\alpha : A_1^+ \to A_2^+$ be an lp-mapping.

**Proposition.** There exists an $A_1, A_2$-bimachine $B_\alpha$ such that:

(i) $\alpha B_\alpha = \alpha$.

(ii) If $B'$ is a trim $A, A'$-bimachine such that $\alpha B' = \alpha$, then there exists a (surjective) morphism $\varphi : B' \to B_\alpha$.

(iii) Up to isomorphism, $B_\alpha$ is the unique trim $A, A'$-bimachine satisfying (ii).

We can view $B_\alpha$ as the *minimum* bimachine of $\alpha$. 
The block product

Let

\[ \mathcal{B}^{(i)} = ((I_R^{(i)}, Q_R^{(i)}, S_R^{(i)}), f^{(i)}, (S_L^{(i)}, Q_L^{(i)}, I_L^{(i)})) \]

be an \( A_i, A_{i+1} \)-bimachine for \( i = 1, 2 \). We shall define an \( A_1, A_3 \)-bimachine

\[ \mathcal{B}^{(2)} \Box \mathcal{B}^{(1)} = \mathcal{B}^{(21)} = ((I_R^{(21)}, Q_R^{(21)}, S_R^{(21)}), f^{(21)}, (S_L^{(21)}, Q_L^{(21)}, I_L^{(21)})) \]

called the \textit{block product} of \( \mathcal{B}^{(2)} \) and \( \mathcal{B}^{(1)} \).

The block product turns out to be a construction on bimachines appropriate do deal with composition.
The block product construction involves sets of mappings whose domain is often a direct product of the form $Q_R^{(1)} \times Q_L^{(1)}$. We shall use the notation $q_R^{(1)} g q_L^{(1)} = g(q_R^{(1)}, q_L^{(1)})$ for $g \in U Q_R^{(1)} \times Q_L^{(1)} = Q_R^{(1)} U Q_L^{(1)}$, $q_R^{(1)} \in Q_R^{(1)}$ and $q_L^{(1)} \in Q_L^{(1)}$.

To be consistent, we shall write maps with domains of type $Q_R^{(1)}$ on the right and type $Q_L^{(1)}$ on the left.
The big semigroups

We define

\[ S^{(21)}_R = \begin{pmatrix} S_L^{(1)} & 0 \\ Q^{(1)}_R & S_R^{(2)} S_L^{(1)} \\ S_R^{(1)} & S_R^{(1)} \end{pmatrix}. \]

\( S^{(21)}_R \) is a semigroup for the product

\[ \begin{pmatrix} s_L^{(1)} & 0 \\ g & s_L^{(1)} \end{pmatrix} \begin{pmatrix} s'_L^{(1)} & 0 \\ g' & s'_L^{(1)} \end{pmatrix} = \begin{pmatrix} s_L^{(1)} s'_L^{(1)} & 0 \\ g s'_L^{(1)} + s_R^{(1)} g' & S_R^{(1)} s'_R^{(1)} \end{pmatrix}, \]

where

\[ q_R^{(1)} (g s'_L^{(1)} + s_R^{(1)} g') q_L^{(1)} = (q_R^{(1)} g (s'_L^{(1)} q_L^{(1)})) + ((q_R^{(1)} s_R^{(1)})) g' q_L^{(1)}). \]
The states

Let

\[ Q_{R}^{(21)} = Q_{R}^{(2)} Q_{L}^{(1)} \times Q_{R}^{(1)}. \]

It will be often convenient to represent the elements of \( Q_{R}^{(21)} \), termed \( R \)-generalized 2 step crossing sequences, as \( 1 \times 2 \) matrices.

Let

\[ I_{R}^{(21)} = (\gamma_{0}^{(21)}, I_{R}^{(1)}), \]

where \( \gamma_{0}^{(21)} \in Q_{R}^{(2)} Q_{L}^{(1)} \) is defined by \( \gamma_{0}^{(21)}(q_{L}^{(1)}) = I_{R}^{(2)}. \)
The semigroup $S_R^{(21)}$ acts on $Q_R^{(21)}$ on the right by

$$
\begin{pmatrix}
\gamma & q_R^{(1)} \\
0 & s_R^{(1)}
\end{pmatrix}
\begin{pmatrix}
s_L^{(1)} \\
g
\end{pmatrix}
= \begin{pmatrix}
\gamma s_L^{(1)} \cdot q_R^{(1)} g & q_R^{(1)} s_R^{(1)}
\end{pmatrix},
$$

where

$$
(\gamma s_L^{(1)} \cdot q_R^{(1)} g)(q_L^{(1)}) = \gamma (s_L^{(1)} q_L^{(1)}) \cdot q_R^{(1)} g q_L^{(1)}.
$$

This is again a form of matrix multiplication (but we refrain from using $+$ for the action).
Cutting to generators

The semigroup $S_R^{(21)}$ is not an $A_1$-semigroup, so let

$$\eta_R : A^+ \to S_R^{(21)}$$

be the homomorphism defined by

$$\eta_R(a) = \begin{pmatrix} a_{S_L^{(1)}} & 0 \\ g_a^{(1)} & a_{S_R^{(1)}} \end{pmatrix},$$

where

$$q_R^{(1)} g_a^{(1)} q_L^{(1)} = (f^{(1)}(q_R^{(1)}, a, q_L^{(1)}))_{S_R^{(2)}}$$

for all $q_R^{(1)} \in Q_R^{(1)}$ and $q_L^{(1)} \in Q_L^{(1)}$. We define

$$S_R^{(21)} = \eta_R(A^+).$$
For $w \in A^+$, we may write

$$\eta_R(w) = \begin{pmatrix}
  w S^{(1)}_L & 0 \\
  g^{(1)}_w & w S^{(1)}_R
\end{pmatrix}$$

for some $g^{(1)}_w \in Q^{(1)}_R S^{(2)}_R Q^{(1)}_L$. 
Dually, we define

\[ Q_L^{(21)} = Q_L^{(1)} \times Q_R^{(2)}, \]

\[ S_L^{(21)} = \left\{ \begin{pmatrix} w_{S_L^{(1)}} & 0 \\ h_{(1)} & w_{S_R^{(1)}} \end{pmatrix} \mid w \in A_1^* \right\} \]

with \( h_w^{(1)} \in Q_R^{(1)} S_L^{(2)} Q_L^{(1)} \). The action is defined by

\[
\begin{pmatrix} s_L^{(1)} & 0 \\ h & s_R^{(1)} \end{pmatrix} \begin{pmatrix} q_L^{(1)} \\ \delta \end{pmatrix} = \begin{pmatrix} s_L^{(1)} q_L^{(1)} \\ hq_L^{(1)} \cdot s_R^{(1)} \delta \end{pmatrix}.
\]
The output function $f^{(21)} : Q_R^{(21)} \times A_1 \times Q_L^{(21)} \rightarrow A_3$ is defined by

$$f^{(21)}(\begin{pmatrix} \gamma & q_R^{(1)} \end{pmatrix}, a, \begin{pmatrix} q_L^{(1)} \\ \delta \end{pmatrix})$$

$$= f^{(2)}(\gamma(aq_L^{(1)}), f^{(1)}(q_R^{(1)}, a, q_L^{(1)}), (q_R^{(1)} a)\delta).$$

This completes the definition of the bimachine $B^{(2)} \Box B^{(1)}$.

If $B^{(2)}$ and $B^{(1)}$ are both finite, so is $B^{(2)} \Box B^{(1)}$. 
The mappings $g_u^{(1)}$ and $h_u^{(1)}$

Let $w \in A^+$, $q_R^{(1)} \in Q_R^{(1)}$ and $q_L^{(1)} \in Q_L^{(1)}$. Write

$$z = \prod_{i=1}^{\mid w \mid} f^{(1)}(q_R^{(1)} \lambda_i(w), \sigma_i(w), \mu_i(w)q_L^{(1)}).$$

Then

$$q_R^{(1)} g_w^{(1)} q_L^{(1)} = z_{S_R^{(2)}},$$

$$q_R^{(1)} h_w^{(1)} q_L^{(1)} = z_{S_L^{(2)}}.$$
Composition

**Theorem.** Let $\mathcal{B}^{(1)}$ be an $A_1, A_2$-bimachine and let $\mathcal{B}^{(2)}$ be an $A_2, A_3$-bimachine. Then $\alpha_{\mathcal{B}^{(2)}} \Box_{\mathcal{B}^{(1)}} = \alpha_{\mathcal{B}^{(2)}} \alpha_{\mathcal{B}^{(1)}}$.

The block product of faithful bimachines is not necessarily faithful.
Additional properties

**Proposition.** Let $B^{(1)}$ be an $A_1, A_2$-bimachine and let $B^{(2)}$ and $B'^{(2)}$ be $A_2, A_3$-bimachines. Let $\varphi^{(2)} : B^{(2)} \rightarrow B'^{(2)}$ be a morphism. Then there exists a morphism $\varphi^{(21)} : B^{(2)} \boxtimes B^{(1)} \rightarrow B'^{(2)} \boxtimes B^{(1)}$ naturally induced by $\varphi^{(2)}$.

**Proposition.** Let $B^{(i)}$ be an $A_i, A_{i+1}$-bimachine for $i = 1, 2$. Then there exist canonical surjective homomorphisms

$$\xi^{(21)}_R : \left( I^{(21)}_R, Q^{(21)}_R, S^{(21)}_R \right) \rightarrow \left( I^{(1)}_R, Q^{(1)}_R, S^{(1)}_R \right),$$

$$\xi^{(21)}_L : \left( S^{(21)}_L, Q^{(21)}_L, I^{(21)}_L \right) \rightarrow \left( S^{(1)}_L, Q^{(1)}_L, I^{(1)}_L \right).$$
And then there were three...

Let
\[ B^{(i)} = ((I_R^{(i)}, Q_R^{(i)}, S_R^{(i)}), f^{(i)}, (S_L^{(i)}, Q_L^{(i)}, I_L^{(i)})) \]
be an \( A_i, A_{i+1} \)-bimachine for \( i = 1, 2, 3 \). Write
\[ B^{(3(21))} = B^{(3)} \Box (B^{(2)} \Box B^{(1)}), \quad B^{((32)1)} = (B^{(3)} \Box B^{(2)}) \Box B^{(1)}. \]

We can get associativity at the semigroup level (for three bimachines, but not necessarily for four bimachines!).

**Lemma.** \( S_R^{(3(21))} \cong S_R^{((32)1)} \) and \( S_L^{(3(21))} \cong S_L^{((32)1)} \).
A cardinality argument

We have

\[ |Q_R^{(3(21))}| > |Q_R^{((32)1)}|. \]

Follows from

\[ a^{bc^d} e^b d > (ac^e)^b d. \]

Similarly,

\[ |Q_L^{(3(21))}| > |Q_L^{((32)1)}|. \]

Let \( \varphi_R : Q_R^{(3(21))} \rightarrow Q_R^{((32)1)} \) be defined as follows. Given
A morphism

\[
(\gamma^{(3(21))}, (\gamma^{(21)}, q^{(1)}_R)) \in Q_R^{(3)} Q_L^{(21)} \times (Q_R^{(2)} Q_L^{(1)} \times Q_R^{(1)}) = Q_R^{(3(21))},
\]

we set

\[
(\gamma^{(3(21))}, (\gamma^{(21)}, q^{(1)}_R)) \varphi_R = (\gamma^{((32)1)}, q^{(1)}_R) \in Q_R^{(32)} Q_L^{(1)} \times Q_R^{(1)} = Q_R^{((32)1)},
\]

where

\[
\gamma^{((32)1)}(q^{(1)}_L) = (\beta^{(1)}_{q_L}, \gamma^{(21)}(q^{(1)}_L)) \in Q_R^{(3)} Q_L^{(2)} \times Q_R^{(2)} = Q_R^{(32)}
\]

and

\[
\beta^{(1)}_{q_L}(q^{(2)}_L) = \gamma^{(3(21))}(q^{(1)}_L, \overline{q^{(2)}_L}),
\]
where $q^{(2)}_L \in Q_R^{(1)} Q^{(2)}_L$ is the constant mapping with image $q^{(2)}_L$.

**Theorem.** $(B^{(3)} \Box B^{(2)}) \Box B^{(1)}$ is a quotient of $B^{(3)} \Box (B^{(2)} \Box B^{(1)})$.

We shall choose bracketing from left to right, that is, priority is assumed to hold from left to right:

$((\ldots (B^{(n)} \Box B^{(n-1)}) \Box B^{(n-2)}) \Box \ldots ) \Box B^{(1)}$. 
We are interested in deterministic Turing machines that halt for all inputs, particularly those that can solve NP-complete problems. In comparison with the most standard models, our model presents three particular features:

- the “tape” is potentially infinite in both directions and has a distinguished cell named the origin;
- the origin contains the symbol # until the very last move of the computation, and # appears in no other cell;
- the machine always halts in one of a very restricted set of configurations.
Our model (formal)

Our deterministic Turing machine is then a quadruple of the form 
\[ T = (Q, q_0, A, \delta) \] where

- \( Q \) is a finite set (set of states) containing the initial state \( q_0 \);
- \( A \) is a finite set (restricted tape alphabet) containing the special symbols \( B \) (blank), \( B' \) (pseudoblank), \( Y \) (yes), \( N \) (no), \( G \) (garbage) and \( \# \) (origin);
- \( \delta \) is a union of full maps

\[
Q \times (A \setminus \{\#\}) \rightarrow Q \times (A \setminus \{\#, B, Y, N, G\}) \times \{L, R\},
\]

\[
Q \times \{\#\} \rightarrow (Q \times \{\#\} \times \{L, R\}) \cup \{Y, N, G\}.
\]
Symbols

We write \( A^\circ = A \setminus \{\#, B\} \).

Since the machine is not allowed to write blanks (we must consider space functions), we shall use the pseudoblock as a substitute to keep the final configurations simple.

Note that, in the final move of a computation, the control head is removed from the tape and so we allow \( \{Y, N, G\} \) in the image of \( \delta \). The symbols \( Y, N, G \) are used to classify the final configurations: for a TM solving a certain problem,

- \( Y \) will stand for *correct input, acceptance*,
- \( N \) for *correct input, rejection*,
- \( G \) for *incorrect input*.
We intend to work exclusively with words, hence we exchange the classical model of “tape” and “control head” by a purely algebraic formalism. Let

\[ A' = A \cup \{a^q \mid a \in A, q \in Q\} \]

be the extended tape alphabet.

The exponent \(q\) on a symbol acknowledges the present scanning of the corresponding cell by the control head, under state \(q\).
Example

\[ q \]

\[ \downarrow \]

\[ \ldots \ B \ B \ c \ c \ b \ a \ # \ c \ B \ \ldots \]

\[ BBccba\#cB \]
Let \( \text{for} : A'^+ \rightarrow A^+ \) and \( \exp : A'^+ \rightarrow (\mathbb{N}, +) \) be the “forgetting” and “counting” homomorphisms defined by

\[
\text{for}(a) = a, \quad \text{for}(a^q) = a, \quad \exp(a) = 0, \quad \exp(a^q) = 1.
\]

We define

\[
\text{ID} = B^* \{ w \in A'^+ \mid \text{for}(w) \in (\{1\} \cup B)(A^o)^*(\{1, \#\}) \cdot (A^o)^* (\{1\} \cup B), \ \exp(w) \leq 1 \} B^*.
\]

\( \overline{\text{ID}} \) is the set of all nonempty factors of words in \( \text{ID} \).
The one-move mapping

The Turing machine $T$ induces a mapping $\beta : \overline{ID} \to \overline{ID}$ as follows: Let $w \in \overline{ID}$. If $|\exp(w)| = 0$, let $\beta(w) = w$. Suppose now that $w = ua^q v$ with $a \in A$ and $q \in Q$.

- if $\delta(q, a) = b \in \{Y, N, G\}$, let $\beta(w) = ubv$;
- if $\delta(q, a) = (p, b, R)$ and $c$ is the first letter of $v = cv'$, let $\beta(w) = ubc^p v'$;
- if $\delta(q, a) = (p, b, R)$ and $v = 1$, let $\beta(w) = ubB^p$;
- if $\delta(q, a) = (p, b, L)$ and $c$ is the last letter of $u = u' c$, let $\beta(w) = u' c^p b v$;
- if $\delta(q, a) = (p, b, R)$ and $u = 1$, let $\beta(w) = B^p b v$. 
Normalized TM

Given $w \in ID$, the sequence $(\beta^n(w))_n$ is eventually constant if and only if $T$ stops after finitely many moves if and only if $\beta^m(w) \in A^+$ for some $m \in \mathbb{N}$. In this case, we write

$$\lim_{n \to \infty} \beta^n(w) = \beta^m(w).$$

We say that our deterministic Turing machine (TM) is normalized if

- $(\beta^n(w))_n$ is eventually constant for every $w \in ID$;
- $\lim_{n \to \infty} \beta^n(w) \in B^*B'^*\{Y,N,G\}B'^*B^*$ for every $w \in ID$.

In view of our stopping conventions, this implies in particular that the symbol $Y$, $N$ or $G$ must be precisely at the origin.
The space and time functions for the normalized TM $\mathcal{T}$ can be naturally defined by

$$s_{\mathcal{T}} : ID \to \mathbb{N} \quad w \mapsto | \lim_{n \to \infty} \beta^n(w) |,$$

$$t_{\mathcal{T}} : ID \to \mathbb{N} \quad w \mapsto \min \{ m \in \mathbb{N} : \beta^m(w) = \lim_{n \to \infty} \beta^n(w) \}.$$ 

Any deterministic (multi-tape) Turing machine solving a problem with space and time complexities of order $s(n)$ and $t(n)$ (not less than linear) can be turned into a normalized TM with space and time functions of order $s(n)$ and $(t(n))^2$, respectively.
The one-move $I^p$-mapping

$\beta_0 : ID \rightarrow ID$ is defined by

$$
\beta_0(w) = \begin{cases} 
\beta(w) & \text{if } |\beta(w)| = |w| \\
w' & \text{if } |\beta(w)| = |w| + 1 \text{ and } \beta(w) = w'B^p; \\
w' & \text{if } |\beta(w)| = |w| + 1 \text{ and } \beta(w) = B^p w'.
\end{cases}
$$

Alternatively, we can say that $\beta_0(w)$ is obtained from $\beta(BwB)$ by removing the first and the last letter.

We can deduce $\beta(w)$ from $\beta_0(BwB)$ and, more generally, $\beta^n(w)$ from $\beta_0(B^n wB^n)$. 
Let $\iota_B : ID \rightarrow (A' \setminus \{B\})^+$ be the mapping that removes all blanks from a given $w \in ID$.

**Proposition.** Let $T$ be a normalized TM with one-move mapping $\beta$. Let $w \in \overline{ID}$ be such that $\text{for}(w) \in (A \setminus \{B\})^+$. Then

(i) $\lim_{n \to \infty} \beta^n(w) = \lim_{n \to \infty} \iota_B(\beta_0^n(B^n wB^n))$;

(ii) $s_T(w) = |\lim_{n \to \infty} \iota_B(\beta_0^n(B^n wB^n))|$

(iii) $t_T(w) = \min\{m \in \mathbb{N} : \iota_B(\beta_0^m(B^m wB^m)) = \lim_{n \to \infty} \iota_B(\beta_0^n(B^n wB^n))\}$. 

Pedro V. Silva  
Bimachines: an introduction
We extend $\beta_0 : \overline{ID} \to \overline{ID}$ to an lp-mapping $\beta_0 : A'^+ \to \overline{ID}$ by composing $\beta_0$ with $\Delta : A'^+ \to ID$ defined by

$$\Delta(w) = \begin{cases} w & \text{if } w \in \overline{ID} \\ GB'^{|w|-1} & \text{otherwise.} \end{cases}$$

So far, we have associated to the normalized TM $T$ an lp-mapping $\beta_0$ encoding the full computational power of $T$ with space and time functions equivalent to those of $T$.

We define next a canonical finite bimachine matching $\beta_0$ in $\overline{ID}$. 
The one-move bimachine

The \( A', A' \)-bimachine

\[ B_T = ((I_R, Q_R, S_R), f, (S_L, Q_L, I_L)) \]

is defined as follows:

- \( Q_R = A' \cup \{I_R\} \), \( Q_L = A' \cup \{I_L\} \);
- \( S_R = A' \) is a right zero semigroup (\( ab = b \));
- \( S_L = A' \) is a left zero semigroup (\( ab = a \));
- the action \( Q_R \times S_R \rightarrow Q_R \) is defined by \( q_R a = a \);
- the action \( S_L \times Q_L \rightarrow Q_L \) is defined by \( aq_L = a \).
The one-move bimachine

For the output function, let us write $I'_{R} = B$ and $q'_{R} = q_{R}$ for every $q_{R} \in Q_{R} \setminus \{I_{R}\}$. Similarly, we define $q'_{L}$. Given $q_{R} \in Q_{R}$, $a \in A'$ and $q_{L} \in Q_{L}$, let

$$f(q_{R}, a, q_{L}) = \beta_{0}(q'_{R}, a, q'_{L}).$$

If $q_{R}aq_{L} \in \overline{TD}$, then $q_{R}aq_{L}$ will encode the situation of three consecutive tape cells at a certain moment. Then $f(q_{R}, a, q_{L})$ describes the situation of the middle cell after one move of $T$.

**Proposition.** Let $T$ be a normalized TM with one-move lp-mapping $\beta_{0}$. Then $\alpha_{B_{T}}(w) = \beta_{0}(w)$ for every $w \in \overline{TD}$. 