ON THE (GENERALIZED) POST CORRESPONDENCE PROBLEM WITH LISTS OF LENGTH 2

A. Ehrenfeucht
Department of Computer Science
University of Colorado at Boulder
Boulder, Colorado
U.S.A.

G. Rozenberg
Institute of Applied Mathematics
and Computer Science
University of Leiden
Leiden, The Netherlands.

The Post Correspondence Problem, considered first by E.Post in [P], is perhaps the most useful problem as far as undecidable properties of formal languages are concerned (see, e.g., [H], [HV] and [S1]).
It can be formulated as follows.

Definition. Let Σ be an alphabet and let h,g be two homomorphisms of Σ^*. The *Post Correspondence Problem* (PCP for short) is to determine whether or not there exists a word w in Σ^+ such that h(w) = g(w). If $\Sigma = n$ then we say that we deal with the *Post Correspondence Problem of length* n (PCP(n) for short). □

The set of solutions of an instance of PCP (that is the set of all words satisfying the equation h(w) = g(w)) is referred to as an *equality language*. The "descriptional power" of PCP stems from the fact that it is able to code computations by arbitrary Turing Machines. This is reflected in the fact that equality languages form a natural base in several characterizations of the class of recursively enumerable languages and its various subclasses (see, e.g. [BB], [C], [ER] and [S2]).

One particular aspect of PCP attracted quite a lot of attention. Since it is such a simply formulated problem of such a strong descriptional power it forms an excellent framework for an attempt to formulate a boundary between "decidable" and "undecidable" (or "computable" and "noncomputable"). In other words one would like to establish as small as possible u such that PCP(u) is undecidable and as big as possible bound ℓ such that PCP(ℓ) is decidable.

The best possible u so far is 10, which is derivable from a result of Matijasevic (see [C1]). As far as ℓ is concerned the only available (trivial) observation until now was the fact that PCP(1) is decidable. To establish whether or not PCP(2) is decidable turned out to be a challenging open problem. There are also several results available which establish the decidability or undecidablility of PCP not depending on the length but rather on other, more structural properties of the homomorphisms involved. For example, in [Le] it is proved that PCP remains undecidable when the involved homomorphisms are codes. Several interesting results concerning PCP can be found in [CK] and [KS].

In this paper we consider a more general version of PCP(2) which is defined as follows.

Definition. Let Σ, Δ be alphabets, h,g be two homomorphisms from Σ^* into Δ^* and let a_1,a_2,b_1,b_2 be words over Δ. The *Generalized Post Correspondence Problem* (GPCP for short) is to determine whether or not there exists a word w in Σ^+ such that $a_1h(w)a_2 = b_1g(w)b_2$. If $\#\Sigma = n$ then we say that we deal with the *Generalized Post Correspondence Problem of length* n (GPCP(n) for short). □

Note that if we set $a_1 = a_2 = b_1 = b_2 = \Lambda$ then GPCP(n) reduces to PCP(n).

We prove that GPCP(2) is decidable. Our solution of this result is rather involved and so in this extended abstract we can merely indicate some more important constructs and reductions used in the solution.

In addition to standard language-theoretic notation and terminology we will use also the following notation: for a word x, $|x|$ denotes its length and for words x,y we write xPREFy if either x is a prefix of y or y is a prefix of x.
Clearly in considering an instance $a_1h(x)a_2=b_1g(x)b_2$ of GPCP(2) one can restrict oneself to an alphabet {0,1}, words a_1,a_2,b_1,b_2 over {0,1} and h,g which are nonerasing endomorphisms of $\{0,1\}^*$.

Definition. Let f be an endomorphism of $\{0,1\}^*$.
(1). f is *marked* if f(0) and f(1) have different first letters.
(2). f is *periodic* if f(0) f(1) = f(1) f(0). □

Definition. Let $I = (h,g,a_1,a_2,b_1,b_2)$ be an instance of GPCP(2).
(1). I is *marked* if both h and g are marked.
(2). I is *periodic* if either h or g is periodic. □

Definition. Let $I = (h,g,a_1,a_2,b_1,b_2)$ be an instance of GPCP(2).
(1). I is *marked* if both h and g are marked.
(2). I is *periodic* if either h or g is periodic. □

First we get a rather easy result.

Theorem 1. It is decidable whether or not an arbitrary periodic instance of GPCP(2) has a solution. □

Then we get our first reduction theorem.

Theorem 2. There exists an algorithm which given an arbitrary instance I of GPCP(2) that is not periodic, produces a positive integer D and a finite set MAR(I) of marked instances of GPCP(2) such that I has a solution if and only if either I has a solution not longer than D, or there exists a $J \in MAR(I)$ such that J has a solution. □

Hence we can restrict our attention to marked instances of GPCP(2) only. The *equality collector* of a given instance of GPCP(2) is the very basic construct of our solution and it is defined, in several stages, as follows.

Definition. Let h,g be marked homomorphisms from $\{0,1\}^*$ into $\{0,1\}^*$ and let $\alpha,\beta \in \{0,1\}^*$. For a nonnegative integer i we define $(\alpha,\beta)^{(i)}_{h,g}$ inductively as follows.

0: $(\alpha,\beta)^{(0)}_{h,g} = (h,\Lambda)(g,\Lambda)$.

The h-*projection* of $(\alpha,\beta)^{(0)}_{h,g}$, denoted by $((\alpha,\beta)^{(0)}_{h,g})_h$ or simply by $(\beta,\alpha)^{(0)}_h$, whenever g is understood, is defined by $(\alpha,\beta)^{(0)}_h = \Lambda$.

The g-*projection* of $(\alpha,\beta)^{(0)}_{h,g}$, denoted by $((\alpha,\beta)^{(0)}_{h,g})_g$ or simply by $(\alpha,\beta)^{(0)}_g$ whenever h is understood, is defined by $(\alpha,\beta)^{(0)}_g = \Lambda$.

i + 1: $(\alpha,\beta)^{(i+1)}_{h,g}$ is defined if and only if

$\alpha h((\alpha,\beta)^{(i)}_h)$PREF $\beta\, g((\alpha,\beta)^{(i)}_g)$ and $\alpha\, h((\alpha,\beta)^{(i)}_h) \neq \beta\, g((\alpha,\beta)^{(i)}_g)$.

If $(\alpha,\beta)^{(i+1)}_{h,g}$ is defined and $c \in \{0,1\}$ then

(a). if $\alpha\, h((\alpha,\beta)^{(i)}_h)c$ *pref* $\beta\, g((\alpha,\beta)^{(i)}_g)$ then $(\alpha,\beta)^{(i+1)}_{h,g} = (\alpha,\beta)^{(i)}_{h,g}(h,c\text{-}ind(h))$,

and

(b). if $\beta\, g((\alpha,\beta)^{(i)}_g)c$ *pref* $\alpha\, h((\alpha,\beta)^{(i)}_h)$ then $(\alpha,\beta)^{(i+1)}_{h,g} = (\alpha,\beta)^{(i)}_{h,g}(g,c\text{-}ind(g))$.

If $(\alpha,\beta)^{(i+1)}_{h,g}$ is defined then the h-*projection* of it and the g-*projection* of it are defined by:

if (a) holds then $(\alpha,\beta)^{(i+1)}_h = (\alpha,\beta)^{(i)}_h c\text{-}ind(h)$ and $(\alpha,\beta)^{(i+1)}_g = (\alpha,\beta)^{(i)}_g$,

and

if (b) holds then $(\alpha,\beta)^{(i+1)}_h = (\alpha,\beta)^{(i)}_h$ and $(\alpha,\beta)^{(i+1)}_g = (\alpha,\beta)^{(i+1)}_g c\text{-}ind(g)$.

For $i \geq 0$ we say that

$(\alpha,\beta)^{(i)}$ is *successful* if $\alpha\, h((\alpha,\beta)^{(i)}_h) = \beta\, g((\alpha,\beta)^{(i)}_g)$, and

$(\alpha,\beta)^{(i)}$ *blocks* if it is not true that $\alpha\, h((\alpha,\beta)^{(i)}_h)$PREF $\beta\, g((\alpha,\beta)^{(i)}_g)$. □

Definition. Let h,g be marked homomorphisms from $\{0,1\}^*$ into $\{0,1\}^*$ and let $\alpha,\beta \in \{0,1\}^*$. The (α,β)-*sequence* (*with respect to* h,g), denoted by $(\alpha,\beta)_{h,g}$, is defined as follows.

(a). Assume that $i \geq 0$ is such that $(\alpha,\beta)^{(i)}_{h,g}$ is successful (note that i is unique). Then $(\alpha,\beta)_{h,g} = (\alpha,\beta)^{(i)}_{h,g}$ and we say that $(\alpha,\beta)_{h,g}$ is *successful*.

(b). Assume that $i \geq 0$ is such that $(\alpha,\beta)^{(i)}_{h,g}$ blocks (note that i is unique). Then $(\alpha,\beta)_{h,g} = (\alpha,\beta)^{(i)}_{h,g}$ and we say that $(\alpha,\beta)_{h,g}$ *blocks*.

(c). If there is no i satisfying either (a) or (b) then $(\alpha,\beta)_{h,g}$ is the infinite (to the right) word over the alphabet $\{(h,\Lambda), (h,0), (h,1), (g,\Lambda), (g,0), (g,1)\}$ such that for each $i \geq 0$, $(\alpha,\beta)^{(i)}_{h,g}$ is its prefix.

The h-*projection* of $(\alpha,\beta)_{h,g}$, denoted by $((\alpha,\beta)_{h,g})_h$ or simply by $(\alpha,\beta)_h$ whenever g is understood, is defined by:

if (a) holds then $(\alpha,\beta)_h = (\alpha,\beta)_h^{(i)}$.

if (b) holds then $(\alpha,\beta)_h = (\alpha,\beta)_h^{(i)}$, and

if (c) holds then $(\alpha,\beta)_h$ is the infinite (to the right) word over {0,1} such that for each $i \geq 0$, $(\alpha,\beta)_h^{(i)}$ is its prefix.

The g-*projection* of $(\alpha,\beta)_{h,g}$, denoted by $((\alpha,\beta)_{h,g})_g$ or simply by $(\alpha,\beta)_g$ whenever h is understood, is defined by:

if (a) holds then $(\alpha,\beta)_g = (\alpha,\beta)_g^{(i)}$,

if (b) holds then $(\alpha,\beta)_g = (\alpha,\beta)_g^{(i)}$, and

if (c) holds then $(\alpha,\beta)_g$ is the infinite (to the right) word over {0,1} such that for each $i \geq 0$, $(\alpha,\beta)_g^{(i)}$ is its prefix. □

Definition. Let (h,g) be an ordered pair of marked homomorphisms such that both the sequence $(h(0),g(\mu(0)))_{h,g}$ and the sequence $(h(1),g(\mu(1)))_{h,g}$ are successful. Then the *equality collector of* (h,g), denoted as *ecol* (h,g), is the pair of homomorphisms $(\overline{h},\overline{g})$ on $\{0,1\}^*$ defined by
$\overline{h}(0) = 0(h(0),g(\mu(0)))_h$, $\overline{h}(1) = 1(h(1),g(\mu(1)))_h$,
$\overline{g}(0) = \mu(0)(h(0),g(\mu(0)))_g$ and $\overline{g}(1) = \mu(1)(h(1),g(\mu(1)))_g$,
where for $i,j \in \{0,1\}$ $\mu(i) = j$ if and only if the first letters of h(i) and g(j) are identical. □

In the sequel given a pair of homomorphisms (h,g) we will use the "bar notation" $(\overline{h},\overline{g})$ to denote *ecol* (h,g).

Definition. Let $I = (h,g,a_1,a_2,b_1,b_2)$ be a marked instance of GPCP(2) such that both the sequence $(h(0),g(\mu(0)))_{h,g}$ and the sequence $(h(1),g(\mu(1)))_{h,g}$ are successful. The *tail equation of* I, denoted as $E_{Tail(I)}$, is the equation
$h(x)a_2 = g(y)b_2$
in variables x, y ranging over $\{0,1\}^*$. A pair of words (u,w) is called a *short solution* of $E_{Tail(I)}$ if $h(u)a_2 = g(w)b_2$ and moreover,
$|h(u)a_2| \leq |a_2b_2| + |h\overline{h}(0)| + |h\overline{h}(1)|$. The set of all short solutions of $E_{Tail(I)}$ is denoted by $sol(E_{Tail(I)})$. □

The notion of the equality collector is extended now to instances of GPCP(2) as follows.

Definition. Let $I = (h,g,a_1,a_2,b_1,b_2)$ be a marked instance of GPCP(2) such that the sequence $(a_1,b_1)_{h,g}$ is successful, the sequence $(h(0),g(\mu(0)))_{h,g}$ is successful, the sequence $(h(1),g(\mu(1)))_{h,g}$ is successful and $sol(E_{Tail(I)}) \neq \emptyset$. Then an *equality collector of* I, denoted *ecol* I, is an instance $J = (\overline{h},\overline{g},\overline{a}_1,u,\overline{b}_1,w)$ of GPCP(2) such that $(\overline{h},\overline{g}) = ecol(h,g)$, $\overline{a}_1 = (a_1,b_1)_h,b_1 = (a_1,b_1)_g$ and $(u,w) \in sol(E_{Tail(I)})$. The set of all equality collectors of I is denoted by ECOL(I). □

Definition. If I is a marked instance of GPCP(2) such that ECOL(I) $\neq \emptyset$ then we say that I is *successful*; otherwise we say that I is *unsuccessful.* □

The following result "justifies" the use of ECOL transformation as a tool in solving the GPCP(2).

Theorem 3. Let I be a marked instance of GPCP(2) such that ECOL(I) is not empty. One can effectively compute a positive integer constant C such that: I has a solution if and only if either I has a solution not longer than C or there exists a J in ECOL(I) such that J has a solution. □

We can also handle the "unsuccessful situation".

Theorem 4. It is decidable whether or not an arbitrary unsuccessful instance of GPCP(2) has a solution. □

We will use the notation *trace* (h,g) to denote the sequence (h,g), *ecol* (h,g), $ecol^2$ (h,g),...., . If this sequence is infinite then it turns out to be ultimately periodic. We use *thres* (h,g) to denote the length of its threshold part and *per* (h,g) to denote the length of its period part.

In our solution of the GPCP(2) (in the case of marked instances) we will iteratively apply the ECOL transformation until we reach the "stable situation" which is formally defined as follows.

Definition. Let $I = (h,g,a_1,a_2,b_1,b_2)$ be an instance of marked GPCP(2) such that *trace* (h,g) is infinite and let *thres* (h,g) = r.
Then $ecol^{r+1}$ (h,g) is called *stable.*
We say that $J = (\hat{h},\hat{g},\hat{a}_1,\hat{a}_2,\hat{b}_1,\hat{b}_2)$ is a *stable version* of I whenever $J \in ECOL^{r+1}(I)$; the set of all stable versions of I is denoted by STABLE(I). We also say then that J is a *stable instance* of GPCP(2) (with respect to I). □

Our next step is to demonstrate that if one considers the decidability status of stable instances of GPCP(2) then it suffices to consider nine "quite concrete" cases. (In what follows, for a word x such that $|x| \geq 2$, we use *two*(x) to denote the prefix of x consisting of the first two letters of x.)

Theorem 5. There exists an algorithm which given an arbitrary stable instance $I = (h,g,a_1,a_2,b_1,b_2)$ of GPCP(2) decides whether or not it has a solution, unless I belongs to one of the following nine categories.

$I \in CAT_1$ if
$h(0) = 0$, $h(1) = 1\alpha$, where $\alpha \in \{0,1\}^+$, and
$g(0) = 0\beta$, $g(1) = 1$, where $\beta \in \{0,1\}^+$, and
For $i \in \{0,1\}$, $I \in CAT_{2,i}$ if
$h(0) = 0$, $h(1) = 1\alpha$, where $\alpha \in \{0,1\}^+$, and
$g(i) = 0\beta$, $g(1-i) = 1\gamma$, where $\beta,\gamma \in \{0,1\}^+$.
For $i \in \{0,1\}$, $I \in CAT_{3,i}$ if
two$(h(0)) = 00$, *two*$(h(1)) = 10$,
two$(g(i)) = 00$, *two*$(g(1-i)) = 10$.
For $i \in \{0,1\}$, $I \in CAT_{4,i}$ if
two$(h(0)) = 01$, *two*$(h(1)) = 10$,
two$(g(i)) = 01$, *two*$(g(1-i) = 10$.
For $i \in \{0,1\}$, $I \in CAT_{5,i}$ if
two$(h(0)) = 00$, *two*$(h(1)) = 11$,
two$(g(i)) = 00$, *two*$(g(1-i)) = 11$.

Now we demonstrate that in considering the decidability status of GPCP(2) it suffices to consider six categories which quite precisely describe the exact pattern of images of homomorphisms involved in an instance of GPCP(2).

We start by defining six (regular) languages.
For $i \in \{0,1\}$, $A_i = i^+$, $B_i = i(1-i)^*$ and $C_i = i((1-i)i)^*\{\Lambda,(1-i)\}$.

Theorem 6. There exists an algorithm which given an arbitrary stable instance $I = (h,g,a_1,a_2,b_1,b_2)$ of GPCP(2) decides whether or not I has a solution, unless I belongs to one of the following six categories .
For $i \in \{0,1\}$, $I \in CL_{A_i}$ if
$h(0) \in A_0$, $h(1) \in A_1$, $g(i) \in A_0$ and $g(1-i) \in A_1$.
For $i \in \{0,1\}$, $I \in CL_{B_i}$ if
$h(0) \in A_0$, $h(1) \in B_1$, $g(i) \in A_0$ and $g(1-i) \in B_1$.
For $i \in \{0,1\}$, $I \in CL_{C_i}$ if
$h(0) \in C_0$, $h(1) \in C_1$, $g(i) \in C_0$ and $g(1-i) \in C_1$. □

Then it turns out that we can also handle the remaining six categories.

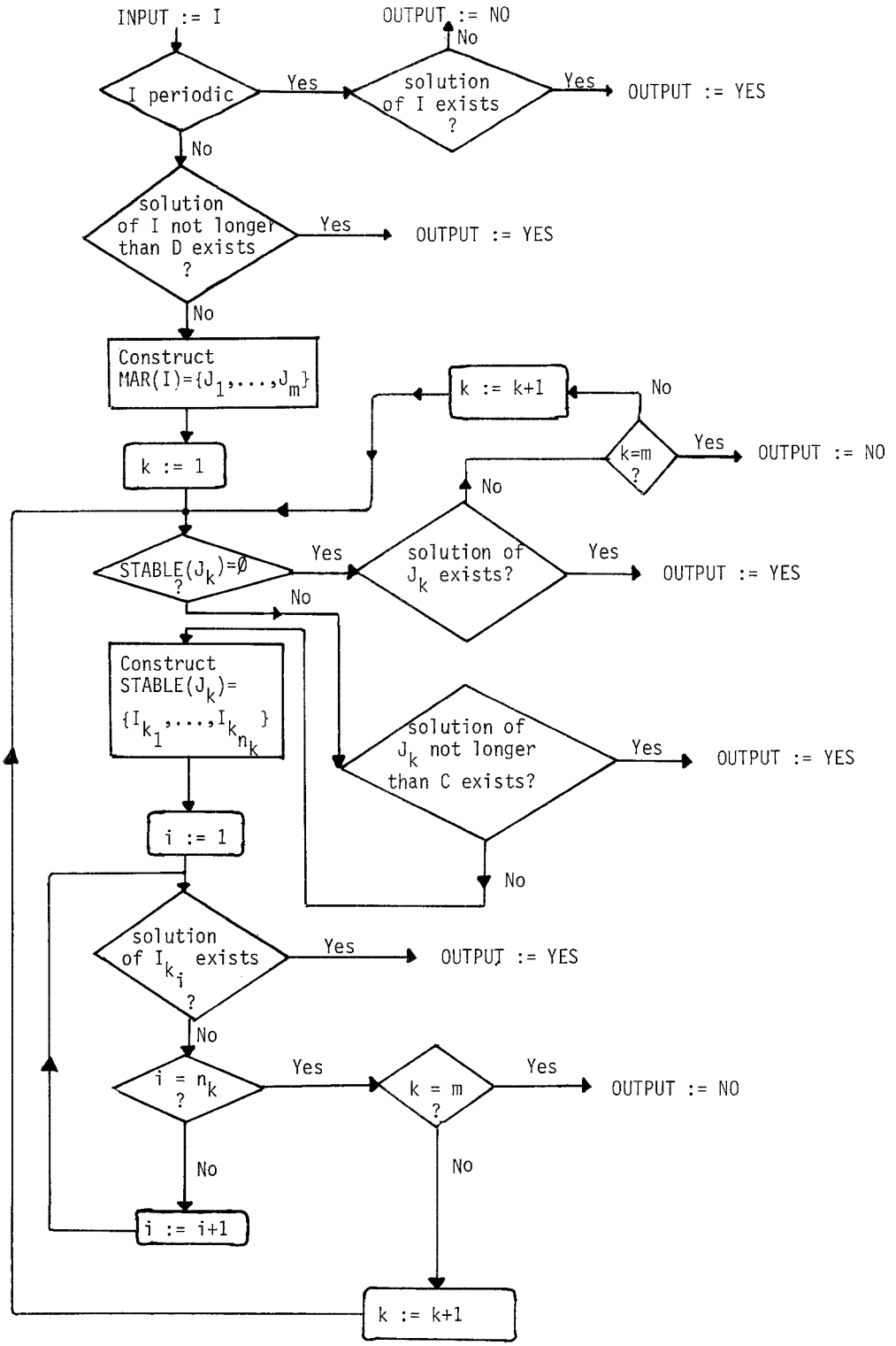
Theorem 7. It is decidable whether or not an arbitrary stable instance I of GPCP(2) such that $I \in CL_{A_i} \cup CL_{B_i} \cup CL_{C_i}$ where $i \in \{0,1\}$, has a solution. □

Combining the above results we can finally prove our main result.

Theorem 8. It is decidable whether or not an arbitrary instance of GPCP(2) has a solution. □

Actually, the following algorithm given an arbitrary instance I of GPCP(2) gives answer YES if I has a solution and answer NO if I has no solution.

(In the following flowchart of our algorithm, D and C are effectively computable constants and the set MAR(I) is the effectively computable set referred to in the statement of Theorem 2).


INPUT := I
OUTPUT := NO
No
I periodic
Yes
solution of I exists ?
Yes
OUTPUT := YES
No
solution of I not longer than D exists ?
Yes
OUTPUT := YES
No
Construct MAR(I)={J1,...,Jm}
k := k+1
No
k := 1
k=m ?
Yes
OUTPUT := NO
No
STABLE(Jk)=∅ ?
Yes
solution of Jk exists?
Yes
OUTPUT := YES
No
Construct STABLE(Jk)= {Ik1,...,Iknk}
solution of Jk not longer than C exists?
Yes
OUTPUT := YES
No
i := 1
solution of Iki exists ?
Yes
OUTPUT := YES
No
i = nk ?
Yes
k = m ?
Yes
OUTPUT := NO
No
No
i := i+1
k := k+1

Corollary. It is decidable whether or not an arbitrary instance of PCP(2) has a solution. □

Remark. A simpler proof of Theorem 8 was obtained recently. It is presented in [EKR]

Acknowledgments. The authors are indebted to D. Janssens and R. Verraedt for useful comments concerning the first draft of this paper. The authors gratefully acknowledge support under National Science Foundation grant number MCS 79-03838.

References

[BB] Book, R.V. and Brandenburg, F.J., Equality sets and complexity classes, *SIAM J. of Comp.*, to appear.

[C] Culik, K., II, A purely homomorphic characterization of recursively enumerable sets, *J. of the ACM*26, 345-450, 1979.

[Cl] Claus, V., Die Grenze zwischen Entscheidbarkeit und Nichtentscheidbarkeit, Fernstudienkurs für die Fernuniversität Hagen, Open University Hagen, 1979.

[CK] Culik, K., II and Karhumaki, J., On the equality sets for homomorphisms on free monoids with two generators, University of Waterloo, Techn.Rep. CS-79-17, 1979.

[ER] Engelfriet, J. and Rozenberg, G., Fixed point languages, equality languages and representations of recursively enumerable languages, *J. of the ACM*, to appear.

[EKR] Ehrenfeucht, A., Karhumaki, J. and Rozenberg, G., The (Generalized) Post Correspondence Problem with lists of length 2 is decidable, manuscript.

[H] Harrison, M.A., *Introduction to formal language theory*, Addison-Wesley Publ., 1978.

[HU] Hopcroft, J.E. and Ullman, J.D., *Introduction to Automata Theory, Languages and Computation*, Addison-Wesley Publ., 1979.

[KS] Karhumaki, J. and Simon, I., A note on elementary homomorphisms and the regularity of equality sets, *Bulletin of the EATCS* 9, 1979.

[Le] Lecerf, Y., Recursive insolubilité de l'equation generale de diagonalisation de deux momomorphisms de monoides libres $\mathbb{Y}x = \mathbb{Y}x$, *Comptes rendus* 257, 2940-2943, 1963.

[P] Post, E.L., A variant of a recursively unsolvable problem, *Bull. of the Am. Math. Soc.*, 52, 264-268, 1946.

[S1] Salomaa, A., *Formal languages*, Academic Press, 1973.

[S2] Salomaa, A., Equality sets for homomorphisms on free monoids, *Acta Cybernetica*-4, 127-139, 1978.