INFORMATIQUE THÉORIQUE ET APPLICATIONS

FRANZ J. BRANDENBURG
JÜRGEN DASSOW

Efficient reductions of picture words

Informatique théorique et applications, tome 27, nº 1 (1993), p. 49-56.
<http://www.numdam.org/item?id=ITA_1993__27_1_49_0>

Informatique théorique et Applications/Theoretical Informatics and Applications
(vol. 27, n° 1, 1993, p. 49 à 56)

EFFICIENT REDUCTIONS OF PICTURE WORDS (*)

by Franz J. BRANDENBURG ([1]) and Jürgen DASSOW ([2])

Communicated by J. BERSTEL

Abstract. – *We introduce a new system of reduction rules on picture words. A picture word is a string over the alphabet* $\{u, d, l, r\}$ *and describes a walk on the grid or, equivalently, an Euler tour on a grid graph. In comparison with reduction systems of Gutbrod* [4] *and Séébold and Slowinski* [10] *our reductions are intuitive and very efficient and construct picture descriptions of length at most two times the size of the drawn picture in quadratic time. However, the reduced descriptions are not necessarily of minimal length. Moreover, it is shown, that the families of regular and context-free picture languages are not closed under our reductions and under minimization.*

Résumé. – *Nous introduisons un nouveau système de règles de réduction de mots de figures. Un mot de figure est une chaîne de caractères pris dans* $\{u, d, l, r\}$ *qui décrit un chemin sur une grille ou, de façon équivalente un cycle eulérien dans le graphe de la grille. Par rapport aux systèmes de réduction de Gutbrod* [4] *et Séébold, Slowinski* [10], *nos réductions sont intuitives et très efficaces et construisent des représentations d'image de taille au plus le double de la taille de l'image en temps quadratique. Toutefois, les descriptions réduites ne sont pas nécessairement de longueurs minimale. De plus, on prouve que les familles de languages d'images réguliers et context-frees ne sont pas fermées pour nos séductions ni pour la minimisation.*

1. INTRODUCTION

The set of symbols $\{u, d, l, r\}$ can be seen as basic commands for a plotter pen, a cursor, a laser or an electron ray with the obvious meaning to draw a line of unit length from the current position in the direction up, down, left, and right, respectively. A picture word $w \in \{u, d, l, r\}^*$ is then a program for such a device. It defines a walk on the grid; its trace is the connected boundary of the picture $p(w)$, or equivalently, an Euler tour on a grid graph. For example, *urdl* describes the unit square with the lower left corner as start and end points. The same picture is described by e. g. *urdlurdl*, *urdulrdl* and

(*) Received January 1991, accepted June 1992.
([1]) Fakultät für Mathematik und Informatik, Universität Passau, W 8390 Passau.
([2]) Technische Universität Magdeburg, Fakultät für Mathematik, O 3010 Magdeburg.

urduldrulrdl. Here the plotter pen draws some pieces of the picture two or more times, sometimes clockwise and sometimes anti-clockwise.

Our aim is to eliminate such duplications, and to establish more compact descriptions. This is achieved in two ways. First, we introduce a system of reduction rules R. R is simple and intuitive and basically changes the order of the traversal of cycles. R reduces a picture word w to a set of equivalent, R-irreducible picture words. Each of these has the same length, which is at most two times the size of the drawn picture. This bound is optimal. The reduction is effective and very fast operating in time $O(|w|^2)$.

However, R-irreducible picture words are not necessarily of minimal length. Using classical methods from graph theory, minimal picture words for a picture p can be constructed in time $O(|p|^3)$ as shortest Chinese Postman tours [2].

Finally, we show that our reductions do not preserve the regularity and the context-freeness of picture languages. In fact, there is a regular set L such that the set of R-irreducible picture words of L is not context-free. The same non-closure result holds for the set of minimal picture words.

Our investigations can be seens as improvements of some previous results on picture languages. Maurer *et al.* [8] have proved that the length of a shortest description w of a drawn picture p is bounded by two times the size of p, and that this bound is optimal. However, their proof is by contradiction and is nonconstructive. Our reduction system is effective and constructs a picture word satisfying this bound for any given picture description.

Concerning rewriting systems on picture words, our reduction system is a generalization of the finite system of retreat deletions of Hinz [6]. Gutbrod [4] and Séébold and Slowinski [10] have recently introduced rewriting introduced rewriting systems which generate all resp. all minimal descriptions of a given picture. Our reduction rules can be deduced from their systems. However, they give no time bounds for the construction of a minimal picture word. Hence, our approach is the first introducing complexity into the construction of reduced and minimal picture descriptions.

2. PICTURE WORDS

Let us recall some elementary notions on picture words and the way they are used to describe pictures. For more technical details we refer to [8].

Let $\{u, d, l, r\}$ denote the picture description alphabet corresponding to the directions up, down, left, and right, respectively. A word $w \in \{u, d, l, r\}^*$

is called a *picture word*. It is now used to define a walk on the grid or in the discrete Cartesian plane.

For a grid point (x, y) the *up-successor* is $\mathrm{pos}(u, (x, y))=(x, y+1)$. Similarly, define the down, left and right-successors. For a picture $wz \in \{u, d, l, r\}^*$ and a grid point (x, y) define the *position* of the cursor after processing the word wz starting from (x, y) by

$$\mathrm{pos}(wz, (x, y))=\mathrm{pos}(z, \mathrm{pos}(w, (x, y))), \qquad \text{where} \quad \mathrm{pos}(\lambda, (x, y))=(x, y)$$

for the empty word λ. Observe that a picture word is executed left to right. We standardize picture words w to start drawing from the origin $(0, 0)$ and define $\mathrm{pos}(w)=\mathrm{pos}(w, (0, 0))$. This leads to the standard representative of an equivalence class of pictures, as defined in [8].

Next, we shall see pictures from a graph theoretic point of view. This introduces new tools for the analysis of pictures words. With each picture word $w=a_1 \ldots a_n$, $a_i \in \{u, d, l, r\}$, $i=1, \ldots, n$, we associate an undirected *graph* $g(w)$ with vertices defined by the positions reached by w and weighted edges between adjacent vertices.

$$g(w)=(V, E) \qquad \text{with} \qquad V=\{\mathrm{pos}(a_1 \ldots a_i) \mid 0 \leqq i \leqq n\},$$
$$\mathrm{E}=\{\{\mathrm{pos}(a_1 \ldots a_{i-1}), \mathrm{pos}(a_1 \ldots a_i)\} \mid 1 \leqq i \leqq n\}$$

and for each edge $e \in E$, the weight $f(e)$ is the number of passes over e while processing w. For a visualization we identify the graph $g(w)$ with its natural drawing on the grid.

For integers m and n, define a translation on grid points by $t_{m,n}(x, y)=(x+m, y+n)$. The translation $t_{m,n}$ canonically extends to grid lines and to grid graphs. The set of mappings $t_{m,n}$ with integers m and n defines an equivalence relation on grid graphs g and on tuples (g, s, e), such that g and g' resp. (g, s, e) and (g', s', e') are *translation equivalent*, if there are integers m and n, such that the natural grid drawing of g is mapped into the natural grid drawing of g', *i.e.*, $g'=t_{m,n}(g)$ and $s'=t_{m,n}(s)$, $e'=t_{m,n}(e)$. *See* [8] for details.

For a picture word w, define the standard (drawn) picture by $p(w)=(g(w), (0, 0), \mathrm{pos}(w))$, and define the (drawn) picture of w by the equivalence class of pictures, which are translation equivalent to $p(w)$. $p(w)$ is the standard representative of this class. Similarly, the graph $g(w)$ is the standard basic picture of w and the set of translation equivalent graphs is the basic picture of w.

To simplify terminology we shall address a translation equivalence class by its standard representative and refer to $p(w)$ and $g(w)$ as the (drawn) picture and the basic picture of a picture word w.

Two picture words w and w' are *equivalent*, if they describe the same picture, *i.e.*, $p(w)=p(w')$. Similarly, the equivalence of basic pictures is defined.

Let $|w|$ denote the *length* of a string w. *The size* $|p|$ of a picture p is the number of edges of the graph (V, E) associated with p. Notice that $|V|-1 \leqq |E| \leqq 2|V|$ with $|p|=|E|$, where $|V|$ and $|E|$ denote the sizes of the sets of vertices and edges. Moreover, $|w| = \sum_{e \in E} f(e)$ and $|p| \leqq |w|$ for every picture word w describing p.

3. REDUCTIONS

Next, we define four reduction rules on picture words. These rules change the order of repeated traversals of cycles and shorten the description.

It is intuitively clear that (u, d) and (l, r) are pairs of inverses, so that $u=d^{-1}$, etc. This extends canonically to picture words, so that for $w=xy$, $w^{-1}=y^{-1}x^{-1}$ is the *inverse* of w. The inverse of w draws $p(w)$ backwards from $\mathrm{pos}(w)$ to the origin.

A picture word w is *a cycle* if $\mathrm{pos}(w)=(0, 0)$. For $a \in \{u, d, l, r\}$, w is an *open a-cycle*, if wa is a cycle. Note, that if w is a cycle, then so are w^{-1} and ww. If w is an open a-cycle, then aw is a cycle and w^{-1} is an open a^{-1}-cycle.

Now we define our reduction rules. Rule R_3 defined below generalizes reductions from [5].

Definition: Let $a \in \{u, d, l, r\}$, C and C' open a-cycles, D an open a^{-1}-cycle, and O and O' cycles. Define the rules

$$R_1\colon a\,C'\,a\,C\,a \rightarrow a\,C'\,C^{-1}$$
$$R_2\colon a\,C\,a\,O\,a^{-1} \rightarrow C^{-1}\,O\,a^{-1}$$
$$R_3\colon a\,O\,a^{-1}\,O'\,a \rightarrow O'\,a\,O$$
$$R_4\colon a\,O\,a^{-1}\,D\,a^{-1} \rightarrow a\,O\,D^{-1}$$

Let R be the infinite system of all rewriting rules obtained from R_1, R_2, R_3, R_4, $a \in \{u, d, l, r\}$ and the infinitely many open a-cycles C, C', D and cycles O, O'.

R is a rewriting system and defines a reduction relation on picture words by $w \rightarrow_R w'$ iff $w = zxz'$, $w' = zyz'$ and $x \rightarrow_R y$ is a rule from R. Then $x \rightarrow_R y$ is said to be applicable to w. Let $\rightarrow_R^*$ denote the reflexive, transitive closure of the reduction $\rightarrow_R$.

A picture word w is *R-irreducible*, if no rule from R is applicable to w.

Let IRR (w) denote the set of *R-irreducible picture words* of w, *i.e.*,

$$\mathrm{IRR}(w) = \{ w' \mid w \rightarrow_R^* w' \text{ and } w' \text{ is } R\text{-irreducible} \}.$$

Observe, that IRR (w) is not necessarily unique, Hence, R is not a confluent rewriting system. For example, let $w = u\,C\,u\,O\,dO'\,u$ with $C = rdl$, $O = lurd$ and $O' = ldru$. Then $w \rightarrow_R urdldluruuldr$ by R_1, $w \rightarrow_R rullurddlurdu$ by R_2 and $w \rightarrow_R urdlldruulurd$ and $w \rightarrow_R ldruurdlulurd$ by R_3, and these words are irreducible.

The following properties are easily established:

LEMMA 1: *Each rule of R preserves pictures and decreases the length of picture words by two*, i.e., *if $w \rightarrow_R w'$ then $p(w) = p(w')$ and $|w| = |w'| + 2$.*

COROLLARY 1: *If $v \in \mathrm{IRR}(w)$ then $p(v) = p(w)$ and $|v| \leqq |w|$.*

LEMMA 2: *If w is R-irreducible then every edge e of the associated weighted graph $g(w)$ has weight $f(e) \leqq 2$.*

Proof: Assume the contrary and let e be the first edge of $g(w)$ with $f(e) \geqq 3$, when edges of $g(w)$ are traversed according ro w. Without loss of generality let $w = a\,x\,a'\,y\,a''\,z$, where a, a' and a'' correspond to the first three traversals of e. Then $a', a'' \in \{a, a^{-1}\}$.

If $a' = a$, then x is an open a-cycle. If $a'' = a$, then y is an open a-cycle, and R_1 is applicable to w; otherwise, if $a'' = a^{-1}$, then y is a cycle and R_2 is applicable. If $a' = a^{-1}$, then x is a cycle. If $a'' = a$ then y is a cycle and R_3 is applicable; otherwise, if $a'' = a^{-1}$, then y is an open a^{-1}-cycle and R_4 is applicable. In each case, w is not R-irreducible.

LEMMA 3: *It is linear time decidable whether a picture word w is R-irreducible.*

Proof: From w construct the weighted graph $g(w)$ in $|w|$ steps by a two-dimensional Turing machine. This device can be simulated in time $O(|w|)$ on a RAM using results from [9]. w is R-irreducible, if no edge e has weight $f(e) \geqq 3$.

LEMMA 4: *For every picture word w an R-irreducible picture word $v \in \mathrm{IRR}(w)$ can be computed in time $O(|w|^2)$.*

Proof: As above, construct the directed weighted graph $g(w)$ from w by a two-dimensional Turing machine. For an edge e with $f(e) \geqq 3$ apply an appropriate rule from R to three symbols from w associated with traversals of e. This application takes at most $|w|$ steps, and R reduces the length by two by Lemma 1. Hence, after at most $O(|w|^2)$ steps we are at an R-irreducible word.

We summarize these facts and obtain:

THEOREM 1: *For every picture word w there is an R-irreducible picture word $v \in \mathrm{IRR}(w)$ with $p(w)=p(v)$ and $|v| \leqq 2|p(w)|$. v can effectively be computed in time $O(|w|^2)$. All R-irreducible words of w are of the same length.*

Recall that Maurer *et al.* [8] have proved that the length of a shortest description of a picture is bounded by two times its size, and that this bound is optimal. However, their proof is by contradiction and is nonconstructive. To the contrary, our reduction system R efficiently constructs a picture word satisfying this bound starting from any given picture description.

Next, we take a closer look at minimal picture words. A word w is called a minimal picture word for a picture p if $p(w)=p$ and $|w| \leqq |w'|$ for all w' with $p(w')=p$. Such words are not necessarily obtained by our reduction system. However, they can be constructed as Chinese Postman tours.

THEOREM 2: *R-irreducible picture words are not necessarily of minimal length.*

Proof: For any $i>1$ consider $w_i=u^i r d^i l u^i r d^i$. Each w_i is R-irreducible; however $p(w_i)=p(v_i)$ with $v_i=u^i r d^i l r$ and $|v_i| < |w_i|$. In fact, for any $\varepsilon>0$ there is an i such that $|w_i|/|v_i| \geqq 2-\varepsilon$.

THEOREM 3: *For every picture word w a minimal picture word v with $p(w)=p(v)$ can be computed in time $O(\max\{|w|, |g(w)|^3\})$.*

Prof: First, construct the graph $g(w)=(V, E)$ from w in time $O(|w|)$. Then by Edmonds and Johnson's algorithm [2] compute a shortest Chinese Postman tout on $g(w)$ from the start point to the end point, which takes time $O(|V|^3)$. This tour defines a minimal picture word. The time bound now follows from $|V|-1 \leqq |E|$ and $|E| = |g(w)| \leqq |w|$.

Notice that the construction of minimal and irreducible picture words can be made canonical by introducing elementary orders. This leads to unique representatives. Define an arbitrary order on the symbols describing directions, $u<r<d<l$, say, and for a picture p fix its start and end points. This order induces a canonical lexicographic order on picture words. In particular, cycles with the same entry are traversed according to the induced order.

Obviously, for every picture word w there is a unique picture word v describing the same picture such that v is of minimal length and is the lexicographic first. This v can be obtained by an appropriate Chinese Postman tour.

Unique reductions by R can be obtained by imposing a leftmost condition. A rewriting step $zxz' \rightarrow_R zyz'$ by a rule $x \rightarrow y$ of R is *leftmost*, if no rule of R is applicable to a proper prefix of zx. It is easily seen that for every picture word there is a unique R-irreductible picture word obtained by leftmost rewriting steps.

Finally, we show that neither reductions by R nor minimizations preserve the regularity and the context-freeness of picture languages. This demonstrates that our reduction rules and the minimization are based on non-trivial computations, which cannot be done e. g. by pushdown machines.

THEOREM 4: *There are regular sets L_1 and L_2 such that the set of R-irreducible strings* $\mathrm{IRR}(L_1) = \{\mathrm{IRR}(w) \mid w \in L_1\}$ *and the set of minimal picture words* $\min(L_2) = \{v \mid w \in L_2,\ p(w) = p(v)$ *and v is of minimal length*$\}$ *are not context-free.*

Proof: Define $Q = u^+ r^+ d^+ l^+$ and $K = \{u^k r^n d^m l^p \mid k, m, n, p \geqq 1,\ k \geqq m$ and $n \geqq p\}$.

If $L_1 = Q^3$ then $K = \mathrm{IRR}(L_1) \cap Q$.

This follows from the fact that rules from R are applicable only if an edge of the associated graph is traversed at least three times and the intersection with Q forces a single traversal.

If $L_2 = Q^2$ then $K = \min(L_2) \cap Q$. To see this observe that words $w = u^k r^n d^m l^p u^{k'} r^{n'} d^{m'} l^{p'}$ from L_2 describe two cycles, which are not necessarily closed and may not interfere with each other. However, the selection of pictures described by words from Q implies that the cycles must overlap, so that $n = p = n'$ and $k' = m = m'$. Then $\min(w) = u^{k''} r^n d^m l^{p''}$ with $k'' = \max\{k, k'\}$ and $p'' = \max\{p, p'\}$.

Using the Pumping Lemma for the context-free languages K can be shown non-context-free and the closure of the context-free languages under intersection with regular sets then proves that $\mathrm{IRR}(L_1)$ and $\min(L_2)$ are non-context-free.

REFERENCES

1. A. V. AHO. J. E. HOPCROFT and J. D. ULLMAN, The Design and Analysis of Computer Algorithms, *Addison Wesley Publishing Company*, Reading, Mass., 1974.

2. J. EDMONDS and E. L. JOHNSON, Matching, Euler Tours and the Chinese Postman, *Math. Programming*, 1973, *5*, pp. 88-124.

3. S. EVEN, Graph Algorithms, *Computer Science Press,* 1979.

4. R. GUTBROD, A Transformation System for Generating Description Languages of Chain Code Pictures, *Theoret. Comput. Sci.,* 1989, *68*, pp. 239-252.

5. F. HINZ, Regular Chain Code Picture Languages of Nonlinear Descriptional Complexity, *Lecture Notes in Computer Science,* 1986, *233*, pp. 414-421.

6. F. HINZ, Classes of Picture Languages that Cannot be Distinguished in the Chain Code Concept and Deletion of Redundant Retreats, *Lecture Notes in Computer Science,* 1989, *349*, pp. 132-143.

7. J. E. HOPCROFT and J. D. ULLMAN, Introduction to Automata Theory, Languages and Computation, *Addison-Wesley Publishing Company,* 1979.

8. H. A. MAURER, G. ROZENBERG and E. WELZL, Using String Languages to Describe Picture Languages, *Inform. Contr.,* 1982, *54*, pp. 155-185.

9. A. SCHÖNHAGE, Real-Time Simulation of Multidimensional Turing Machines by Storage Modification Machines, *SIAM J. Comput.,* 1980, *9*, pp. 490-508.

10. P. SÉÉBOLD and K. SLOWINSKI, Minimizing Picture Words, *Lecture Notes in Computer Science,* 1990, *464*, pp. 234-243.

11. I. H. SUDBOROUGH and E. WELZL, Complexity and Decidability for Chain Code Picture Languages, *Theoret. Comput. Sci.,* 1985, *36*, pp. 172-202.