Extended abstract

The problem of designing an efficient algorithm for deciding whether a given polygon tiles the plane becomes tractable when restricted to polyominoes, that is, subsets of the square lattice \( \mathbb{Z}^2 \) whose boundary is a non-crossing closed path (see [11] for more on tilings and [6] for related problems). Here, we consider tilings obtained by translation of a single polyomino, called exact in [14]. Paths are conveniently described by words on the alphabet \( \{0, 1, 2, 3\} \), representing the elementary grid steps \( \{\rightarrow, \uparrow, \leftarrow, \downarrow\} \). Beauquier and Nivat [1] characterized exact polyominoes by showing that the boundary word \( b(P) \) of such a polyomino satisfies the equation \( b(P) = X \cdot Y \cdot Z \cdot \hat{X} \cdot \hat{Y} \cdot \hat{Z} \), where \( \hat{W} \) is the path traveled in the direction opposite to that of \( W \) (the paths \( W \) and \( \hat{W} \) are said homologous). From now on, this condition is referred to as the BN-factorization. In this factorization, one of the variables may be empty, in which case \( P \) is called a square, and hexagon otherwise. Note that a single polyomino may lead to several distinct tilings of the plane: for instance the \( n \times 1 \) rectangle does it in \( n - 1 \) distinct ways as a hexagon (see Figure 1).

![Figure 1: The three hexagonal tilings of the 4×1 rectangle.](image)

However, it was recently established [4] that an exact polyomino tiles the plane as a square in at most two distinct ways. A polyomino having exactly two distinct square tilings is called double square [13] and there is a linear time algorithm to find all the square factorizations from its boundary word [9].

*With the support of NSERC (Canada)
†Corresponding author.
squares have a peculiar combinatorial structure and motivated developments in
equations on words involving periodicities and palindromes [2, 3]. Christoffel
and Fibonacci tiles were introduced in [5] as examples of infinite families of
double squares (see Figure 2) but do not characterize completely the class of
double square tiles (see Figure 3).

In this extended abstract, we address this problem by showing that every
double square can be reduced to a composed cross pentamino by using some
reduction operators (Theorem 19). We show that these operators are invertible
which allows one to generate any double square tile from a composed cross
pentamino. Thus, an algorithm for the generation of double square tiles is
proposed. The reduction method allows to show that all double squares reducing
to the cross pentomino $\dagger$ have a BN-factorization consisting of palindromes, a
weaker form of a conjecture stated in Provençal thesis [13].

Some problems remain open. Although we proved that four reduction oper-
ators reduce any double square, we believe that only two of them are necessary.
1 Preliminaries

The usual terminology and notation on words is from Lothaire [12]. An alphabet \( \mathcal{A} \) is a finite set whose elements are letters. A finite word \( w \) is a function \( w : [1, 2, \ldots, n] \to \mathcal{A} \), where \( w_i \) is the \( i \)-th letter, \( 1 \leq i \leq n \). The length of \( w \), denoted by \( |w| \), is the integer \( n \). The length of the empty word is 0. The free monoid \( \mathcal{A}^* \) is the set of all finite words over \( \mathcal{A} \). The reversal of \( w = w_1w_2\cdots w_n \) is the word \( \bar{w} = w_nw_{n-1}\cdots w_1 \). A word \( u \) is a factor of another word \( w \) if there exist \( x, y \in \mathcal{A}^* \) such that \( w = xuy \). We denote by \( |w|_u \) the number of occurrences of \( u \) in \( w \). Two words \( u \) and \( v \) are conjugate, written \( u \equiv v \) or sometimes \( u \equiv_i v \), when \( x, y \) are such that \( u = xy \) and \( v = yx \). Conjugacy is an equivalence relation ant the class of a word \( w \) is denoted \([w]\).

In this paper, the alphabet \( \mathcal{F} = \{0, 1, 2, 3\} \) is considered as the additive group of integers mod 4. Basic transformations on \( \mathcal{F} \) are rotations \( \rho^i : x \mapsto x+i \) and reflections \( \sigma_i : x \mapsto i-x \), which extend uniquely to morphisms (w.r.t concatenation) on \( \mathcal{F}^* \). Given a nonempty word \( w \in \mathcal{F}^* \), the first differences word \( \Delta(w) \in \mathcal{F}^* \) of \( w \) is

\[
\Delta(w) = (w_2-w_1) \cdot (w_3-w_2) \cdots (w_n-w_{n-1}).
\]

One may verify that if \( z \in \mathcal{F}^* \), then \( \Delta(wz) = \Delta(w)\Delta(wnz_1)\Delta(z) \). Words in \( \mathcal{F}^* \) are interpreted as paths in the square grid as usual (See Figure 4), so that we indistinctly talk of any word \( w \in \mathcal{F}^* \) as the path \( w \).

Moreover, the word \( \bar{w} := \rho^2(\bar{w}) \) is homologous to \( w \), that is, described in direction opposite to that of \( w \) (see Figure 4). A word \( u \in \mathcal{F}^* \) may contain factors in \( \mathcal{C} = \{02, 20, 13, 31\} \), corresponding to cancelling steps on a path. Nevertheless, each word \( w \) can be reduced in a unique way to a word \( w' \), by sequentially applying the rewriting rules in \( \{u \mapsto \varepsilon | u \in \mathcal{C}\} \). The reduced word \( w' \) of \( w \) is nothing but a word in \( \mathcal{P} = \mathcal{F}^* \setminus \mathcal{F}^*\mathcal{C}\mathcal{F}^* \). We define the turning number\(^1 \) of \( w \) by \( T(w) = (|\Delta(w')|_1 - |\Delta(w')|_3)/4 \).

A path \( w \) is closed if it satisfies \( |w|_0 = |w|_2 \) and \( |w|_1 = |w|_3 \), and it is simple if no proper factor of \( w \) is closed. A boundary word is a simple and closed path, and a polyomino is a subset of \( \mathbb{Z}^2 \) contained in some boundary word. It is convenient to represent each closed path \( w \) by its conjugacy class \([w]\), also called circular word. An adjustment is necessary to the function \( T \), for we take

\[^1\)In [7, 8], the authors introduced the notion of winding number of \( w \) which is \( 4T(w) \).
into account the closing turn. The first differences also noted $\Delta$ is defined on any closed path $w$ by setting

$$\Delta([w]) \equiv \Delta(w) \cdot (w_1 - w_n),$$

which is also a closed word. By applying the same rewriting rules, a circular word $[w]$ is circularly-reduced to a unique word $[w']$. If $w$ is a closed path, then the turning number of $w$ is $T(w) = \frac{|\Delta([w'])|_1 - |\Delta([w'])|_3}{4}$. It corresponds to its total curvature divided by $2\pi$. And clearly, the turning number $T(w)$ of a closed path $w$ belongs to $\mathbb{Z}$ (see [7, 8]). In particular, the Daurat-Nivat relation [10] is rephrased as follows.

**Proposition 1.** The turning number of a boundary word $w$ is $T(w) = \pm 1$.

Now, we may define orientation: a boundary word $w$ is positively oriented (counterclockwise) if its turning number is $T(w) = 1$. As a consequence, every square satisfies the following factorization

**Proposition 2.** Let $w \equiv XY\hat{X}\hat{Y}$ be the boundary word of a square, then

$$\Delta([w]) \equiv \Delta(X) \cdot \alpha \cdot \Delta(Y) \cdot \alpha \cdot \Delta(\hat{X}) \cdot \alpha \cdot \Delta(\hat{Y}) \cdot \alpha,$$

where $\alpha = 1$ if $w$ is positively oriented, $\alpha = 3$ otherwise.

The following result is easy to check.

**Proposition 3.** Let $w \equiv XY\hat{X}\hat{Y}$ be an oriented boundary word of a square. Then $\text{Fst}(X) = \text{Lst}(X)$ and $\text{Fst}(Y) = \text{Lst}(Y)$.

## 2 Admissible solutions

In this section, we introduce the useful notion of admissible solution in order to describe all double squares. Its definition is motivated by the following result stating that the BN-factorizations of a double square must alternate.

**Lemma 4.** [9, 13] If an exact polyomino satisfies $AB\hat{A}\hat{B} \equiv_d XY\hat{X}\hat{Y}$, then the factorization must alternate, i.e., $0 < d < |A| < d + |X|$.

Hence, we must have the situation depicted in Figure 5. Moreover, it is useful to encode double squares while keeping track of their (two) factorizations. For that purpose, we refine the BN-factorization as follows.

![Figure 5: Finer factorization of a double square.](image_url)
Definition 5. An admissible solution is an 8-tuple \((w_i)_{i \in [0..7]}\), \(w_i \in \mathcal{F}^+\), such that \(|w_i| = |w_{i+4}|\) for \(i \in \{0, 1, 2, 3\}\) and

(i) \(w_0w_1 = w_4w_5\); (iii) \(w_2w_3 = w_6w_7\);
(ii) \(w_1w_2 = w_5w_6\); (iv) \(w_3w_4 = w_7w_0\).

Observe that every admissible solution \((w_i)_{i \in [0..7]}\) is uniquely determined by the words \(w_0, w_1, w_2\) and \(w_3\). The length of a solution \(S = (w_i)_{i \in [0..7]}\) is naturally defined as \(|S| = |w_0w_1 \cdots w_7|\).

Example 6. Clearly, each double square factorization yields an admissible solution. Indeed, consider the double square given in Figure 6: the black and white dots together with the ending arrow uniquely determine the admissible solution

\((3, 03010303, 01030, 10103010, 1, 21232121, 23212, 32321232)\).

Figure 6: A double square and its admissible solution. The black and white dots distinguish the two BN-factorizations. The boundary of the polyomino is traveled counter-clockwise and ends with the triangular arrow.

In what follows we exhibit the properties satisfied by admissible solutions. To fix the notation, hereafter \(S = (w_i)_{i \in [0..7]}\) denotes an admissible solution, and all indices are taken in \(\mathbb{Z}_8\). The first result concerns periodicity.

Lemma 7. For all \(i\), \(d_i = |w_{i+1}| + |w_{i+3}|\) is a period of \(w_i\). Hence, there exist \(u_i, v_i\) and \(n_i\) such that

\[
\begin{align*}
\widehat{w_{i-3}w_{i-1}} &= u_i v_i \quad (2) \\
w_i &= (u_i v_i)^{n_i} u_i \quad (3) \\
\widehat{w_{i+1}w_{i+3}} &= v_i u_i \quad (4)
\end{align*}
\]

where \(0 \leq |u_i| < d_i\). Moreover, \(d_0 = d_2 = d_4 = d_6\) and \(d_1 = d_3 = d_5 = d_7\).

A direct consequence is that the periods extend.

Corollary 8. For all \(i\), \(d_i\) is a period of \(w_{i-1}w_iw_{i+1}\) \(\square\)

With the notation of Lemma 7, we have the following commuting properties.

Lemma 9. For all \(i\), we have

\[
\begin{align*}
u_i v_i \cdot w_i &= w_i \cdot v_i u_i, \quad (5) \\
w_i \cdot u_{i+1} v_{i+1} &= u_{i+5} v_{i+5} \cdot w_{i}, \quad (6) \\
v_{i-1} u_{i-1} \cdot w_i &= w_i \cdot v_{i+3} u_{i+3}. \quad (7)
\end{align*}
\]
With the notation of Lemma 7,

**Lemma 10.** For all $i$, the following properties hold

\[
\begin{align*}
\hat{w}_i u_{i+1} &= \hat{w}_{i+5} w_{i+4} \\
u_i w_{i+1} &= \hat{w}_{i+5} u_{i+4} \\
\hat{w}_i v_{i+3} &= v_i + 7 \hat{w}_{i+4} \\
\hat{v}_i w_{i+3} &= \hat{w}_{i+7 + 4}
\end{align*}
\]

**Proof.** The results follow from the decomposition in Lemma 7. We obtain (8) and (10) by comparing the suffixes and prefixes of the following equality

\[
\hat{w}_{i+4} \cdot \hat{w}_{i+4} \cdot v_{i+3} = \hat{w}_{i+1} \hat{w}_{i+3} w_{i+1} \hat{w}_{i+4} = v_i u_i w_{i+1} = v_i \hat{w}_{i+3} \cdot w_{i-1} u_i,
\]

and doing a shift on the indices. We obtain (9) and (11) similarly from

\[
\hat{w}_{i-3} u_{i+4} \cdot v_{i+4} w_{i-1} = \hat{w}_{i-3} w_{i-1} \hat{w}_{i-3} w_{i-1} = u_i v_i u_i v_i = u_i w_{i+1} \cdot \hat{w}_{i+3} v_i.
\]

\[\square\]

With the notation of Lemma 7, we have

**Lemma 11.** $n_i \neq 0 \implies n_{i+1} = n_{i+3} = n_{i+5} = n_{i+7} = 0$.

**Lemma 12.** Assume that $d_i = |w_{i+1}| + |w_{i+3}|$ divides $|w_i|$, i.e. $u_i$ is empty.

Let $g = \gcd(|w_{i+2}|, d_{i+2})$. Then

(i) $w_{i+1} = \hat{w}_{i+5}$ and $w_{i+3} = \hat{w}_{i+7},$

(ii) there exist two words $p, q \in \mathcal{F}^+$ and $k, \ell \in \mathbb{N}$ such that

\[
w_{i+1} w_{i+2} w_{i+3} = p^k \text{ and } w_{i+6} = \hat{p}^\ell
\]

\[
w_{i+5} w_{i+6} w_{i+7} = q^k \text{ and } w_{i+2} = \hat{q}^\ell
\]

where $|p| = |q| = g$ and $\ell = |w_{i+2}|/g$,

(iii) $p w_{i+1} = w_{i+1} \hat{q}$ and $\hat{q} w_{i+3} = w_{i+3} p$.

**Proof.** (i) From Lemma 7, we have that

Then $w_{i+1} = \hat{w}_{i-3}, w_{i+3} = \hat{w}_{i+5}$ and $w_{i+3} = \hat{w}_{i-1} = \hat{w}_{i+7}$.

(ii) Using assertion (i), we can write

\[
w_{i+1} w_{i+3} = \hat{w}_{i+5} = \hat{w}_{i+5} + w_{i+6} = w_{i+6} w_{i+5} = w_{i+6} + 1 w_{i+3}.
\]

Since this equation has the form $ab = ba$, with $a = w_{i+1} w_{i+3}$ and $b = \hat{w}_{i+6}$, we have from Lothaire [12] that there exists $p \in \mathcal{F}^*$ such that

\[
ab = w_{i+1} w_{i+2} w_{i+3} = \hat{w}_{i+6} \hat{w}_{i+5} w_{i+3} = p^k
\]
with $|p| = \gcd(|b|, |a|) = g$. In particular, $w_{i+6} = \hat{p}^\ell$. To prove that there exists $q \in \mathcal{F}^*$ such that $w_{i+5}w_{i+6}w_{i+7} = q^k$ with $|q| = g$ and that $w_2 = \hat{q}^\ell$, it suffices to increase all indices in the precedent proof by four.

(iii) To prove the equality $pw_{i+1} = w_{i+1}\hat{q}$, by (ii) we have $w_{i+2} = \hat{q}^\ell$ and $w_{i+1}w_{i+2}$ has period $g$ with $\text{Pref}_g(w_{i+1}w_{i+2}) = p$ and $\text{Suff}_g(w_{i+1}w_{i+2}) = \hat{q}$. Then

$$pw_{i+1}w_{i+2} = w_{i+1}w_{i+2}\hat{q} = w_{i+1}\hat{q}^{\ell+1} = w_{i+1}\hat{q}w_{i+2}.$$ Comparing the prefixes of length $g + |w_{i+1}|$ on both sides of this equality, we obtain that $pw_{i+1} = w_{i+1}\hat{q}$. The proof that $\hat{q}w_{i+3} = w_{i+3}p$ is similar to the previous one. Since $w_{i+2}w_{i+3}$ has period $g$ with $\text{Pref}_g(w_{i+2}w_{i+3}) = \hat{q}$ and $\text{Suff}_g(w_{i+2}w_{i+3}) = p$, \hfill $\square$

The turning number of an admissible solution $S = (w_i)_{i \in [0..7]}$ is naturally defined from the circular word it defines: $\mathcal{F}(S) = \mathcal{F}(w_0w_1w_2w_3w_4w_5w_6w_7)$. Proposition 3 translates directly as follows for admissible solutions.

**Lemma 13.** $\mathcal{F}(S) = \pm 1$ if and only if $\text{Fst}(w_i) = \text{Lst}(w_{i+1})$ for all $i$.

Under some conditions, we may guarantee that some admissible solutions do not yield double squares. More precisely:

**Proposition 14.** Assume that there exists $i \in [0..7]$ such that $|w_i| + |w_{i+2}| = |w_{i+1}| + |w_{i+3}|$. Then $\mathcal{F}(S) \notin \{-1, 1\}$.

*Proof.* Let $d = |w_i| + |w_{i+2}| = |w_{i+1}| + |w_{i+3}|$. We first show that there exists $j \in [0..7]$ such that $|w_{j-1}w_j| \geq d$ and $|w_jw_{j+1}| \geq d$. Arguing by contradiction, assume that the contrary holds. This implies that there exists $k \in [0..7]$ with $|w_k| + |w_{k+1}| < d$ and $|w_{k+2}| + |w_{k+3}| < d$. Thus,

$$2d = |w_k| + |w_{k+1}| + |w_{k+2}| + |w_{k+3}| < 2d,$$

which is absurd. Now, we know from Lemma 12 that the words $x = w_{j-2}w_{j-1}w_j$, $y = w_{j-1}w_jw_{j+1}$ and $z = w_jw_{j+1}w_{j+2}$ all have period $d$. Moreover, $x$ has a suffix of length at least $d$ that is a prefix of $y$, and $y$ has a suffix of length at least $d$ that is a prefix of $z$, so that the period $d$ propagates on the whole word $w_{j-2}w_{j-1}w_jw_{j+1}w_{j+2}$. First, since $|w_{j-2}w_{j-1}w_jw_{j+1}| = 2d$, we have $\text{Fst}(w_{j-2}) = \text{Fst}(w_{j+2})$. On the other hand, $w_{j+2}w_{j+3} = w_{j-1}w_{j-2}$ implies $\text{Fst}(w_{j+2}) = \text{Lst}(w_{j-1})$. To conclude, we proceed again by contradiction. Assume that $\mathcal{F}(S) \in \{-1, 1\}$. Then Lemma 13 applies. In particular, $\text{Lst}(w_{j-1}) = \text{Fst}(w_{j-2})$. Gathering these three equalities, we obtain

$$\text{Fst}(w_{j-2}) = \text{Fst}(w_{j+2}) = \text{Lst}(w_{j-1}) = \text{Fst}(w_{j-2}),$$

which is impossible. Hence, $\mathcal{F}(S) \notin \{-1, 1\}$. \hfill $\square$
3 Reduction of solutions

Let $S$ be the set of admissible solutions. To describe the structure of double squares, we consider invertible functions acting on $S$. Below, we describe each of them and show their action on double squares. Let $S = (w_i)_{i \in [0..7]}$ be an admissible solution with $g = \gcd(|w_2|, d_2)$, $p = \text{Pref}_g(w_1w_2w_3)$ and $q = \text{Pref}_g(w_5w_6w_7)$. We define the following operators:

- **shrink** ($S$) = $(w_0(v_0u_0)^{-1}, w_1, w_2, w_3(v_4u_4)^{-1}, w_5, w_6, w_7)$,
- **l-shrink** ($S$) = $(p^{-1}w_0, p^{-1}w_1, w_2, w_3, q^{-1}w_4, q^{-1}w_5, w_6, w_7)$,
- **r-shrink** ($S$) = $(w_0q^{-1}, w_1, w_2, w_3p^{-1}, w_4p^{-1}, w_5, w_6, w_7^{-1})$,
- **swap** ($S$) = $(\widehat{w}_4, (v_1u_1)^n v_1, \widehat{w}_6, (v_3u_3)^n v_3, \widehat{w}_0, (v_5u_5)^n v_5, \widehat{w}_2, (v_7u_7)^n v_7)$.

![Figure 7](image1.png)

Figure 7: $S' = \text{extend}_1(S)$ is obtained from $S$ by extending $w_1$ and $w_5$. As for $S' = \text{swap}(S)$, we have $w'_0 = \widehat{w}_4, w'_2 = \widehat{w}_6, w'_4 = \widehat{w}_0$ and $w'_6 = \widehat{w}_2$.

![Figure 8](image2.png)

Figure 8: The operators $\text{r-shrink}$ and $\text{r-extend}$ modify $w_7$ and $w_0$, while $\text{l-shrink}$ and $\text{l-extend}$ modify $w_0$ and $w_1$.

The basic operators shrink, l-shrink, r-shrink and swap are generalized to act on any $w_i$ by using a shift operator. Let $\text{shift}$ be the operator defined by

$$\text{shift}(S) = (w_1, w_2, w_3, w_4, w_5, w_6, w_7, w_0).$$

It is obvious that $\text{shift}(S)$ is admissible. Then for each $i \in [0..7]$ and every $\varphi \in \{\text{swap}, \text{shrink}, \text{l-shrink}, \text{r-shrink}\}$, we define the operator $\varphi_i(S)$ as

$$\varphi_i(S) = \text{shift}^{-i} \circ \varphi \circ \text{shift}^i(S).$$

The reason for shifting back is simply to keep fixed the positions of other factors. In particular, $\varphi_0(S) = \varphi(S)$. With the notation of Lemma 7, we have:

\[ \text{shift}(S) = (w_1, w_2, w_3, w_4, w_5, w_6, w_7, w_0). \]
Proposition 15. The following properties hold.

(i) If $|w_i| > d_i$, then SHRINK$_i(S)$ is admissible.
(ii) If $|w_i| = d_i$ and $|w_{i+1}| > g$, then L-SHRINK$_i(S)$ is admissible.
(iii) If $|w_i| = d_i$ and $|w_{i+7}| > g$, then R-SHRINK$_i(S)$ is admissible.
(iv) If $u_{i+1}, u_{i+3}, u_{i+5}$ and $u_{i+7}$ are nonempty, then SWAP$_i(S)$ is admissible.

When the conditions described in (i), (ii) or (iii) hold, there must be local periodicity in the neighborhood of $w_i$ (Lemma 8). Consequently, the action of the operators SHRINK, L-SHRINK and R-SHRINK results in removing an occurrence of this period as shown in Figure 7 and 8.

As for SWAP, these operators are defined from the relations between the $w_i$’s and the periods $u_j, v_j$ (Lemma 10). The operators SHRINK, L-SHRINK and R-SHRINK are all invertible (see Section 4 for the definition of the respective inverses EXTEND, L-EXTEND and R-EXTEND).

Proposition 16. The following properties hold.

(i) If $|w_i| > d_i$, then $|\text{SHRINK}_i(S)| < |S|$.
(ii) If $|w_i| = d_i$ and $|w_{i+1}| > g$, then $|\text{L-SHRINK}_i(S)| < |S|$.
(iii) If $|w_i| = d_i$ and $|w_{i+7}| > g$, then $|\text{R-SHRINK}_i(S)| < |S|$.
(iv) If $u_{i+1}, u_{i+3}, u_{i+5}$, $u_{i+7}$ are nonempty then

$$|u_{i+1}| + |u_{i+3}| < |u_{i+1}| + |u_{i+3} \iff |\text{SWAP}_i(S)| < |S|.$$ 

Lemma 17. The turning number $\mathcal{T}$ is invariant under the operators SHIFT, SHRINK, L-SHRINK, R-SHRINK and SWAP.

The cross pentamino $\bigotimes$ is the smallest non-trivial double square. Up to conjugacy and reversal, its admissible solution is

$$b(\bigotimes) \equiv (0, 10, 1, 21, 2, 32, 3, 03).$$

Before proving Theorem 19, let us define what we mean by reduction of solution. Let $S$ and $S'$ be two admissible solutions such that $S' = \varphi_n \circ \cdots \circ \varphi_2 \circ \varphi_1(S)$ where the $\varphi_i$’s are operators on solutions. Let $S_k = \varphi_k \circ \varphi_{k-1} \circ \cdots \circ \varphi_1(S)$, so that $S_0 = S$ and $S_n = S'$. Then we say that $S$ reduces to $S'$ if $|S_k| < |S_{k-1}|$ for all $k \in [1..n]$.

Proposition 18. Let $S$ be such that $\mathcal{T}(S) = \pm 1$. Then $S$ is a composed cross pentamino or one of SHRINK, SWAP, L-SHRINK or R-SHRINK reduces $S$. 
Proof. If there is $i \in \{0, 1, 2, 3\}$ such that $|w_i| > d_i$, then $S$ reduces to $\text{SHRINK}_i(S)$. Also, if there is $i$ such that $|u_{i+1}| + |u_{i+3}| > |v_{i+1}| + |v_{i+3}|$, then $S$ reduces to $\text{SWAP}_i(S)$. Hence, if neither $\text{SHRINK}$ nor $\text{SWAP}$ can be applied on $S$, then the $n_i$’s are necessarily equal to 0 or 1.

The case $(n_0, n_1, n_2, n_3) = (0, 0, 0, 0)$ is impossible. Indeed, suppose that we are in this situation, that is $w_i = u_i$ for all $i$. Since $\text{SWAP}_i(S)$ does not reduce $S$, we know that $|u_{i+1}| + |u_{i+3}| \leq |v_{i+1}| + |v_{i+3}|$ for all $i$. Using the equality $|v_i| = |w_{i-1}| + |w_{i+1}| - |u_i|$, this implies that $|u_i| + |u_{i+2}| \leq |u_{i+1}| + |u_{i+3}|$ for all $i$. Then we deduce that $|u_0| + |u_2| = |u_1| + |u_3|$. But from Lemma 14, this implies that $\mathcal{F}(S) \neq \pm 1$ which is a contradiction.

If $n_i = 1$, then $u_i = \varepsilon$ and from Lemma 11, we have that $n_{i+1} = n_{i+3} = 0$. Let $g = \gcd(|w_{i+2}|, |w_{i+1}| + |w_{i+3}|)$. We know from Corollary 8 that $w_{i+1}w_{i+2}w_{i+3}$ has period $g$. If $n_{i+2} = 0$, we have in particular that $|w_{i+2}| < |w_{i+1}| + |w_{i+3}|$. This implies that $g < |w_{i+1}|$ or $g < |w_{i+3}|$. From Proposition 15, we have that $S$ reduces to $L$-$\text{SHRINK}_i(S)$ in the first case, and to $R$-$\text{SHRINK}_i(S)$ in the other one.

Consider now $(n_0, n_1, n_2, n_3) = (1, 0, 1, 0)$. In this case the solution has the form $(u_1 \widehat{u}_3, u_1, u_3\widehat{u}_1, u_3, \widehat{u}_1 u_3, \widehat{u}_1, \widehat{u}_3 \widehat{u}_1, \widehat{u}_3)$, which is a composed cross pentamino under the morphism $0 \rightarrow u_1, 1 \rightarrow u_3, 2 \rightarrow \widehat{u}_1$ and $3 \rightarrow \widehat{u}_3$. The case $(n_0, n_1, n_2, n_3) = (0, 1, 0, 1)$ is similar. \hfill \Box

Algorithm 1 Reduction of a double square tile

1: function REDUCE($S$)
2: Input: a solution $S = (w_0, w_1, w_2, w_3, w_4, w_5, w_6, w_7)$
3: Output: an ordered list of operators.
4: L $\leftarrow ()$
5: while there is no $i$ s.t. $|w_i| = d_i$ and $|w_{i+2}| = d_{i+2}$ do
6: if there is $i$ such that $|w_i| > |w_{i-1}| + |w_{i+1}|$ then
7: $S$ $\leftarrow$ $\text{SHRINK}_i(S)$, $L$ $\leftarrow$ $L + (\text{SHRINK}_i)$
8: else if there is $i$ such that $|u_{i+1}| + |u_{i+3}| > |v_{i+1}| + |v_{i+3}|$ then
9: $S$ $\leftarrow$ $\text{SWAP}_i(S)$, $L$ $\leftarrow$ $L + (\text{SWAP}_i)$
10: else $\triangleright$ There is $i$ s.t. $n_i = 1$ and $n_{i+1} = n_{i+2} = n_{i+3} = 0$
11: $g \leftarrow \gcd(|w_{i-1}| + |w_{i+1}|, |w_{i+2}|)$
12: if $g < |w_{i+1}|$ then
13: $S$ $\leftarrow$ $L$-$\text{SHRINK}_i(S)$, $L$ $\leftarrow$ $L + (L$-$\text{SHRINK}_i)$
14: else $\triangleright g < |w_{i+3}|$
15: $S$ $\leftarrow$ $R$-$\text{SHRINK}_i(S)$, $L$ $\leftarrow$ $L + (R$-$\text{SHRINK}_i)$
16: end if
17: end if
18: end while
19: return $L$ $\triangleright S$ corresponds to a composed cross pentamino
20: end function
Theorem 19. Every double square reduces to a composed cross pentamino.

Proof. Let $S$ be the solution of a double square. From Theorem 1, the turning number of $S$ is $\pm 1$. Hence, from Proposition 18, either $S$ is a composed cross pentamino or $S$ can be reduced by a reduction operator which preserves the turning number (Lemma 17). Then, Proposition 18 can be applied again. Since the length of the solution gets strictly smaller at each reduction, Fermat’s infinite descent principle applies. It follows that the number of iterations is finite and $S$ reduces to a composed cross pentamino.

Algorithm 1 contains the pseudo code for the Reduction and Figure 9 illustrates the execution of the reduction on a double square tile.

4 Generation of solutions

The previous section culminated with Theorem 19 stating that every double square reduces to a composed cross pentamino. Then, it becomes very natural...
to ask whether this leads to an algorithm that generates all double squares by inverting the reduction operators. In this section, we introduce some new operators on admissible solutions that are inverses of the reduction operators. Moreover, we give some relations between these operators and we provide an algorithm that generates all double squares up to a given perimeter length.

Figure 10: Subtree of the space of admissible solutions generated when starting from the cross pentamino.

Let \( S = (w_i)_{i \in [0..7]} \) be an admissible solution. Let \( g = \gcd(|w_2|, d_2) \), \( p = \text{Pref}_g(w_1w_2w_3) \) and \( q = \text{Pref}_g(w_5w_6w_7) \). We define

\[
\text{EXTEND}(S) = (w_0(v_0u_0), w_1, w_2, w_3, w_4(u_4v_4), w_5, w_6, w_7),
\]

\[
\text{L-EXTEND}(S) = (pw_0, pw_1, w_2, w_3, qw_4, qw_5, w_6, w_7),
\]

\[
\text{R-EXTEND}(S) = (w_0q, w_1, w_2, w_3p, w_4p, w_5, w_6, w_7q).
\]

**Proposition 20.** Let \( S = (w_i)_{i \in [0..7]} \) be an admissible solution and let \( p, q \in F^* \) defined as above.

(i) Then \( \text{EXTEND}(S) \) is admissible.

(ii) If \( |w_0| = d_0 \) then \( \text{L-EXTEND}(S) \) is admissible.

(iii) If \( |w_0| = d_0 \) then \( \text{R-EXTEND}(S) \) is admissible.

All our operators are invertible, as shown by the next proposition.
Proposition 21. Let \( S = (w_i)_{i \in [0..7]} \) be a solution.

(i) \( \text{SHIFT}^8(S) = S \)

(ii) \( \text{SHRINK} \circ \text{EXTEND}(S) = S \) and if \( |w_0| > d_0 \), then \( \text{EXTEND} \circ \text{SHRINK}(S) = S \).

(iii) If \( |w_0| = d_0 \) then \( \text{L-SHRINK} \circ \text{L-EXTEND}(S) = S \). If in addition \( |w_1| > g \), then \( \text{L-EXTEND} \circ \text{L-SHRINK}(S) = S \).

(iv) If \( |w_0| = d_0 \) then \( \text{R-SHRINK} \circ \text{R-EXTEND}(S) = S \). If in addition \( |w_7| > g \), then \( \text{R-EXTEND} \circ \text{R-SHRINK}(S) = S \).

(v) If \( u_{i+1}, u_{i+3}, u_{i+5} \) and \( u_{i+7} \) are nonempty, then \( \text{SWAP}^2_i(S) = S \).

In order to provide a more efficient generation algorithm, it is worth mentioning that the operators \( \text{EXTEND}, \text{SWAP}, \text{L-EXTEND} \) and \( \text{R-EXTEND} \) satisfy commuting properties (see Figure 11).

Figure 11: Two distinct ways of generating the same double square tile. The diagram commutes in virtue of Proposition 22(iii) and (iv).

Proposition 22. Let \( \varphi \in \{\text{EXTEND}, \text{SWAP}, \text{L-EXTEND}, \text{R-EXTEND}\} \) and \( i \in \mathbb{Z}_8 \).

(i) \( \varphi_i = \varphi_{i+4} \);

(ii) \( \text{SWAP}_i = \text{SWAP}_{i+2} \);

(iii) \( \text{EXTEND}_{i+2} \circ \text{EXTEND}_i = \text{EXTEND}_i \circ \text{EXTEND}_{i+2} \);

(iv) \( \text{EXTEND}_{i+1} \circ \text{SWAP}_i = \text{SWAP}_i \circ \text{EXTEND}_{i+1} \);

(v) \( \text{L-EXTEND}_i \circ \text{R-EXTEND}_i = \text{R-EXTEND}_i \circ \text{L-EXTEND}_i \).

Based on the preceding results, Algorithm 2 allows to generate all double squares of perimeter at most \( n \). Notice that it may be improved significantly by using Proposition 22. More precisely, it is possible to avoid exploring all paths involving commuting operators by choosing precedence on the operators. For instance, we could avoid using the operator \( \text{EXTEND}_2 \) if the last applied operator is either \( \text{EXTEND}_0 \) or \( \text{SWAP}_1 \), i.e. these two last operators would precede \( \text{EXTEND}_2 \). We also believe that some operators might be superfluous, as
discussed in the last section. Figure 10 illustrates a partial trace of Algorithm 2 when starting with the cross pentamino.

It is not clear what is the complexity of Algorithm 2. Indeed, except for some conjectures that we state in the last section, we do not know exactly how many admissible solutions yield double squares. On the other hand, our algorithm is clearly more effective than the naive strategy of enumerating all words of length \( n \) on \( \mathcal{F} \) and check if it describes a double square tile. A fine analysis of Lines 5 and 8 of Algorithm 2 would be also useful. Finally, it would not be hard to enumerate double squares according to perimeter length: it suffices to make \( Q \) a priority heap.

**Algorithm 2** Generation of double squares

1: function Generate\((n)\)
2: Input: \( n \), the maximum perimeter of the generated double squares.
3: Output: the set of all double squares of perimeter at most \( n \).
4: \( T \leftarrow \emptyset \)
5: \( Q \leftarrow \{P : P \text{ is a composed cross pentamino of size at most } n\} \)
6: while \( Q \neq \emptyset \) do
7: \( t \leftarrow \text{Pop}(Q) \)
8: if \([t]\) is a polyomino then \( T \leftarrow T \cup \{[t]\} \)
9: \( C \leftarrow \{\text{EXTEND}_i(t) : i = 0, 1, 2, 3\} \)
10: \( C \leftarrow C \cup \{\text{SWAP}_i(t) : i = 0, 1\} \)
11: \( C \leftarrow C \cup \{\text{L-EXTEND}_i(t) : i = 0, 1, 2, 3 \text{ and } |w_i| = d_i\} \)
12: \( C \leftarrow C \cup \{\text{R-EXTEND}_i(t) : i = 0, 1, 2, 3 \text{ and } |w_i| = d_i\} \)
13: \( Q \leftarrow Q \cup \{c \in C : |t| < |c| \leq n\} \)
14: end while
15: return \( T \) \quad \triangleright \text{\( T \) contains all tiles of size at most } n
16: end function

5 Concluding remarks and open problems

Although we have described an algorithm to generate double squares, there are still some improvements that remain to be done. For instance, we observed that all admissible solutions whose generation use at least one operator among R-SHRINK and L-SHRINK are always self-crossing. Hence, we conjecture that only two of the reduction operators suffice for reducing any double square, so that only the operators EXTEND and SWAP would be needed for generation purposes.

**Conjecture 23.** Let \( S \) be an admissible solution coding a double square. Then

(i) \( S \) is a composed cross pentamino or

(ii) \( \text{SHRINK}_i(S) \) is a double square smaller than \( S \) or

(iii) \( \text{SWAP}_i(S) \) is a double square smaller than \( S \).
Moreover, it was conjectured in [13] that, given a prime double square and its admissible solution $S$, the factor $w_iw_{i+1}$ is a palindrome for all $i \in \mathbb{Z}_8$. Although we do not solve that problem here, we obtain a result strongly suggesting that this property holds:

**Proposition 24.** Let $w \equiv AB\bar{A}\bar{B} \equiv XY\bar{X}\bar{Y}$ be the boundary of a double square. If $w$ reduces to the prime cross pentamino, then $A$, $B$, $X$ and $Y$ are palindromes.

As a last remark, we conjecture that the operators EXTEND and SWAP preserve primality. More precisely:

**Conjecture 25.** Let $D$ be a double square tile. If $D$ is prime, then $D$ reduces to the prime cross pentamino.

### References


