Theoretical Computer Science 586 (2015) 2–11

Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

No easy puzzles: Hardness results for jigsaw puzzles

Michael Brand

Faculty of IT, Monash University, Clayton, VIC 3800, Australia

ARTICLE INFO

Article history:
Received 18 September 2014
Received in revised form 25 January 2015
Accepted 10 February 2015
Available online 24 February 2015

Keywords:
Jigsaw puzzle
Parsimonious testing
Communication complexity
Subgraph isomorphism

ABSTRACT

We show that solving (bounded-degree) jigsaw puzzles requires $\Theta(n^2)$ edge matching comparisons both in the worst case and in expectation, making all jigsaw puzzles as hard to solve as the trivial upper bound. This result applies to bounded-degree puzzles of all shapes, whether pictorial or apictorial. For non-bounded degree puzzles, we show that $\Omega(n \log n)$ is a tight bound.

1. Introduction

Jigsaw puzzles [25] are among the most popular forms of puzzles. Fig. 1 gives a few examples of their variations. A canonical jigsaw puzzle [33] is one where the pieces are square-like and are joined together in a grid-like fashion via tabs and pockets along their edges. These tabs and pockets can be of arbitrary shape. Fig. 1(a) demonstrates a canonical puzzle. It differs from standard jigsaw puzzles only in that it is apictorial [8], meaning that it has no guiding image. Fig. 1(b) also demonstrates an apictorial jigsaw puzzle, however, unlike the first puzzle, it is not canonical: the pieces fit together in a scheme different to the canonical grid scheme. Fig. 1(c) demonstrates that puzzle schemes need not even be planar. It depicts a partially-assembled 27-tile puzzle that can be assembled into a $3 \times 3 \times 3$ cube.

The study of jigsaw puzzles in computer science began with [8], where the problem was investigated in terms of whether machine vision techniques are able to determine if two edges match. This problem was considered to have uses, e.g., in piecing together archaeological artefacts, and, indeed, has since been put to such use (see, e.g., [18]). Later improvements concentrated on better edge-shape representations (e.g., [21,26,31]), use of pictorial data (e.g., [33]), better match quality metrics (e.g., [9,27,32]), etc.

These papers all address the first of three sub-problems, which are normally tackled jointly, which form jigsaw puzzle solving. We refer to it as the “tile matching” problem. Suppose now that we take this problem as solved, that is to say, that we are given a constant-time Oracle function that is able to provide a precise Boolean answer regarding whether two tiles match. Then, we are faced with the second question of which tile pairs to run this Oracle function on. We call this the problem of “parsimonious testing”. Lastly, taking this second problem as solved (e.g., by supposing that one has already applied the Oracle function to every possible pair), one is faced with the problem of finding a mapping that would match the tile pairing data with the puzzle shape. This is the “bijection reconstruction” problem. It is an instance of the well-studied problem of graph isomorphism (see, e.g., [1,2,20,24]), or, in some contexts, an instance of the well-studied problem of subgraph isomorphism (see, e.g., [4,10,14,29]).

E-mail address: michael.brand@alumni.weizmann.ac.il.

http://dx.doi.org/10.1016/j.tcs.2015.02.030
0304-3975/

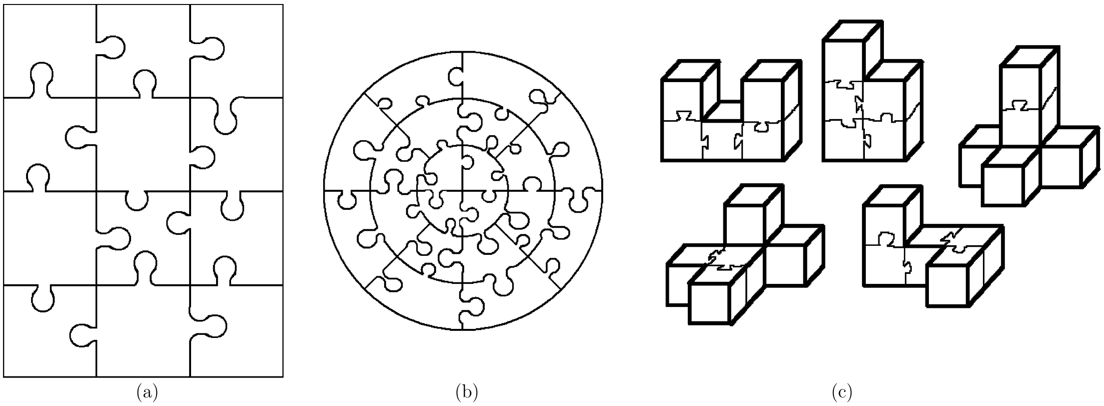

Fig. 1. Apictorial jigsaw puzzle variations: (a) a canonical puzzle, (b) a non-canonical puzzle, (c) a partially-assembled non-planar puzzle.

In contrast to its popular siblings, the second of the three sub-problems introduced, the problem of parsimonious testing, has not received much attention in the literature. In practice, however, it appears to be quite important: the solutions proposed to tile matching demonstrate that the more one improves one's method for tile-match assessment, the more time-consuming it becomes. There is, therefore, an incentive to minimise the number of tile comparisons, or otherwise to exploit trade-offs between tile-match accuracy and the number of tile pairs that need to be tested.

This paper closes this gap in the literature by focusing on the problem of parsimonious testing. To be able to study it in isolation from the two problems flanking it, we consider the following model, which describes the issue as a problem in communication [22]. In this model, two entities, I and O, are tasked with solving a puzzle. Entity I is an infinitely powerful computer, able to solve, for example, subgraph isomorphism and related problems in constant time, but it does not have any information regarding tile shapes and colours. Entity O, on the other hand, has perfect information regarding the puzzle, including which tiles match and how. For the puzzle to be considered solved, however, the solution must be communicated from O to I. The puzzle is solved by entity I making Oracle calls to entity O, in which I queries O using a fixed communication protocol. The question is how many queries does I require in order to solve the puzzle. We call this the communication complexity of jigsaw puzzles.

The remainder of this paper is arranged as follows. In Section 2, we give a formal definition of the model. In Section 3 we then prove the claim that bounded-degree jigsaw puzzles, regardless of their shape, are always as hard as the trivial upper bound (up to a multiplicative constant). In Section 4 we prove related results regarding other variations of the jigsaw puzzle problem. A short conclusions section follows.

2. Formal definition of the model

We use the following model to describe a jigsaw puzzle.

Definition 1. A *jigsaw puzzle* is a tuple $\langle T, P, E_p, Q\rangle$.

Here, T is the set of *tiles* and P is the set of *positions* to place them in. We refer to $n \stackrel{\text{def}}{=} |T| = |P|$ as the *size* of the puzzle, and assume $n > 1$.

E_p is a relation over P that describes which positions are adjacent. We refer to the (undirected) graph $\langle P, E_p\rangle$ as the *shape* of the puzzle. Our only requirement from this graph is that it is connected. We refer to its maximum vertex degree as the *degree* of the puzzle.

The last element in the tuple defining a jigsaw puzzle, Q, is a set of queries. These can be of either of two types, as follows.

Match queries: Does tile $x \in T$ fit to tile $y \in T$? This query corresponds to a test whether the tabs and pockets of two tiles match.

Positional queries: Does tile $x \in T$ fit to position $p \in P$? This query corresponds to a test whether the image portion on tile x matches the portion of the guiding image covered by position p.

Definition 2. A jigsaw puzzle is called *pictorial* if Q is the set of all possible queries. It is called *apictorial* if Q is the set of all possible match queries.

Definition 3. A *solution* to a jigsaw puzzle $\langle T, P, E_p, Q\rangle$ is a bijection, π, from T to P.