



Approximating the longest paths in grid graphs

Wen-Qi Zhang, Yong-Jin Liu*

Tsinghua National Laboratory for Information Science and Technology, Department of Computer Science and Technology, Tsinghua University, China

ARTICLE INFO

Article history:

Received 11 January 2011

Received in revised form 29 May 2011

Accepted 6 June 2011

Communicated by D.-Z. Du

Keywords:

Long paths

Square-free 2-factor

Grid graphs

ABSTRACT

In this paper, we consider the problem of approximating the longest path in a grid graph that possesses a square-free 2-factor. By (1) analyzing several characteristics of four types of cross cells and (2) presenting three cycle-merging operations and one extension operation, we propose an algorithm that finds in an n -vertex grid graph G , a long path of length at least $\frac{5}{6}n + 2$. Our approximation algorithm runs in quadratic time. In addition to its theoretical value, our work has also an interesting application in image-guided maze generation.

© 2011 Elsevier B.V. All rights reserved.

1. Introduction

Finding the longest path in a graph is a classical problem and have attracted considerable attentions in theoretical computer science. Given that Hamiltonian path (or circle) problem is $\mathcal{NP}$ -complete as a special case, it is only practical to design computer programs that find approximate solutions to the longest path in a graph. For general undirected graphs, Karger et al. [1] have shown that, unless $\mathcal{P} = \mathcal{NP}$, for any $\varepsilon < 1$, the problem of finding a path of length $n - n^\varepsilon$ is $\mathcal{NP}$ -hard. Bazgan et al. [2] further revealed that for any $\varepsilon > 0$, the longest path problem is not constantly approximated even in cube Hamiltonian graphs.

Many approximation algorithms have been proposed for the longest path problem. Monien [3] proposed an exponential time algorithm that finds in a graph, a long path of length $O(\log L / \log \log L)$, where L is the length of the longest path. For a general undirected graph, Bjorklund and Husfeldt [4] improved the results by presenting a polynomial algorithm that finds a path of length $\Omega((\log L / \log \log L)^2)$ with the performance ratio $O(n(\log \log n / \log n)^2)$, where n is the vertex number in the graph. Several special graphs have also been considered. Feder et al. [5] showed that for sparse graphs such as 3-connected cubic n -vertex graphs, there is a polynomial time algorithm that can find a cycle of length at least $n^{(\log_3 2)/2}$. Chen et al. [6] showed that, for a 3-connected n -vertex graph with bounded degree d , there is a cubic algorithm which can find a cycle of length at least $n^{(\log_b 2)/2}$, where $b = 2(d - 1)^2 + 1$. This result was improved in [7], showing that for the same graph, there is a cubic algorithm which can find a cycle of length at least $n^{(\log_b 2)/2} / 2 + 3$, where $b = \max\{64, 4d + 1\}$.

In this paper, we consider the approximate longest path problem in grid graphs. Itai et al. [8] first proved that determining whether a general grid graph is Hamiltonian is $\mathcal{NP}$ -complete. Later in 1997, Umans and Lenhart [9] showed that Hamiltonian cycles can be identified in solid grid graphs (a special type of grid graphs without holes) in polynomial time. Their algorithm is based on a 2-factor of the graph and runs in $O(n^3)$ time. For general grid graphs with initial square-free 2-factors, in this paper, we show that a long path of length at least $\frac{5}{6}n + 2$ can be found in an n -vertex grid graph in quadratic time.

In addition to theoretical contributions, our work on approximating longest paths in grid graphs has also an interesting application, maze design. In the history of art creation and design, mazes have found diverse applications in visual art (e.g.,

* Corresponding author. Tel.: +86 10 62784141.

E-mail address: liuyongjin@tsinghua.edu.cn (Y.-J. Liu).

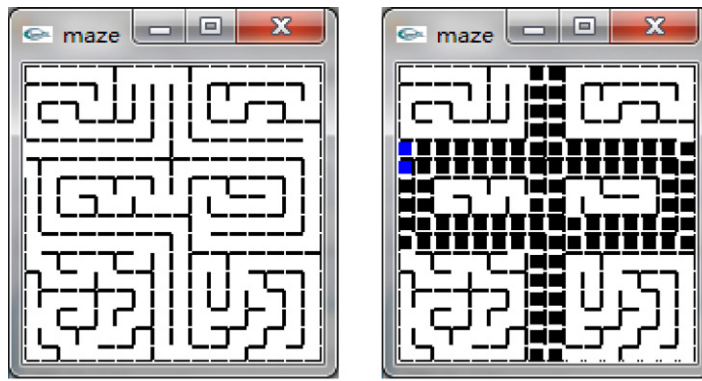


Fig. 1. A simple perfect maze embedded in 16×16 lattices, whose solution encodes the shape of a Chinese character “Zhong”.

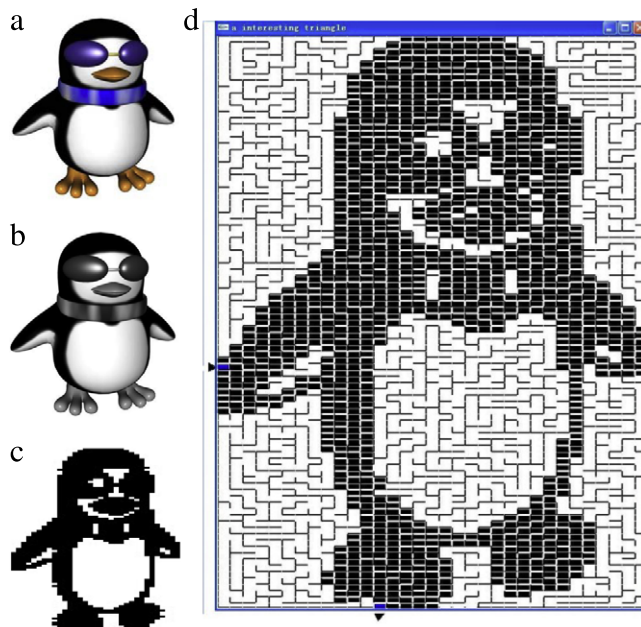


Fig. 2. Image-guided maze generation. (a) An original color image. (b) The gray-scale image. (c) The black-and-white image after halftoning. (d) The maze whose solution embeds the pictorial information.

stylized line drawings), architectural decoration (e.g., the herb garden hedge maze in UK) and cultural and religious symbols (e.g., decoration in bronze wares in China Shang Dynasty), etc. Designing a meaningful and interesting maze is not an easy task. Usually wealth of experience and long-term training are needed. In this paper, we show that using the approximating longest path in grid graphs, a perfect maze can be constructed with the aid of a computer that can encode any predefined pictorial information.

A perfect maze is defined as a maze which has one and only one path from any point in the maze to any other point. A simple example is illustrated in Fig. 1. Between the entrance and exit points that are both on the maze boundary, there is a unique path. When all the lattices passed through by the solution path are painted in black, the encoded shape appears. To use computer to develop a maze whose solution shows predefined pictorial information, an image-guided maze design method can be used as follows. First the user chooses an interesting image as a reference (ref. Fig. 2(a)). If the chosen image is a colored one, it is converted to a gray-scale image (ref. Fig. 2(b)) using a standard computer vision technique [10]. Then a halftoning technique (Chapter 3 in [11]) is applied to convert the gray-scale image into a binary (i.e., black-and-white) image (ref. Fig. 2(c)), in which the black pixels are all connected. Finally, the binary image is embedded into a rectangle of squared lattices (ref. Fig. 2(d)) and the 4-connectivity property of black pixels defines a sparse graph with bounded degree $d \leq 4$. If a Hamiltonian path exists in the graph, then the maze solution is exactly the binary image. Our proposed algorithm finds a long path in the graph that passes through as many as possible lattices so that the painted path shows a very closed image of the binary image.

To complete the maze design, given the approximate longest path as the solution, we start iteratively at each node in the solution and apply the depth-first search in the whole square lattice graph. Finally, the maze pattern is obtained so that

Download English Version:

<https://daneshyari.com/en/article/438957>

Download Persian Version:

<https://daneshyari.com/article/438957>

[Daneshyari.com](https://daneshyari.com)