Adaptive Windows Layout based on Evolutionary Multi-Objective Optimization

Rui Chen, College of Computer Science, Sichuan University, Chengdu, China
Tiantian Xie, College of Computer Science, Sichuan University, Chengdu, China
Tao Lin, College of Computer Science, Sichuan University, Chengdu, China
Yu Chen, Department of Computer Science, Sichuan University for Nationalities, Kangding, China

ABSTRACT

Large displays are becoming more and more pervasive, and their larger screen sizes provide an opportunity for users to see more information in more windows at the same time, but at the cost of managing a larger amount of screen space. Some methods of automatic window management have been studied with some success. However, they mainly focus on utilization of empty space and ignore preservation of the mental map of users, which tends to cause user confusion in practical use. This paper presents a genetic multi-objective optimization algorithm to automatically manage window layout with the aim of balancing the two aspects: utilization of screen space and mental map preservation. Results show that the method is capable of generating suitable window layouts for users and takes a key step toward developing an automatic windows manager.

Keywords: Adaptive Windows, Genetic Algorithm, Mental Map, Multi-Objective, Windows Layout Optimization

INTRODUCTION

The last decade has witnessed an unprecedented growth of information, display size and multi-task interfaces. The larger screen size provides an opportunity for users to see more information in more windows at the same time, but at the cost of managing a larger amount of screen space. For example, when using some information-intensive software such as visual analytics and stock market quotation analysis, users often need frequent window operations (i.e., add, move, delete and resize) to obtain the desired information. These operations are not relevant to the task and, as a result, drag users out of their task domain. Users’ mental resources consumed by window operations make no contribution to task performing. Additionally, when a window

DOI: 10.4018/jthi.2013070105

Copyright © 2013, IGI Global. Copying or distributing in print or electronic forms without written permission of IGI Global is prohibited.
operation is completed, a recall process is necessary before users can return into their task context and proceed. This is somewhat similar to process scheduling performed by computer operating systems. Consequently, such methods of manually managing windows not only induce users’ attention to be interrupted, thereby decreasing their overall task performance (Bly & Rosenberg, 1986; Bury, Davies, & Darnell, 1985; Davies, Bury, & Darnell, 1985; Johnson-Laird, 1985; Kandogan & Shneiderman, 1997; Sandberg-Diment, 1984), but also dissipate valuable mental resources. Research on automatic window layout optimization aiming to reduce the amount of needed window operations throughout the task performing process is thus of great importance.

Improving the interaction with multi-window interfaces has been a challenge for HCI and interface researchers and some efforts have been made (Beaudouin-Lafon, 2001; Hutchings & Stasko, 2002a, 2002b, 2004; Trivedi, Lai, & Zhang, 2001). They can generally be split into two sub-categories: (1) automatically optimizing window layout, in which interfaces focus on helping users effectively utilize display space usually by maximizing visual scope and minimizing the overlap between windows, and (2) developing more efficient window operations, in which interfaces focus on considering whether the standard window operations could be improved or expanded to better help users efficiently operate the windows in the context of multi-tasks. This study lies in the former sub-category and, specifically, focuses on the integration of automatic window layout optimization and users’ mental map preservation as well as their balance.

Some researchers (Funke, Neal, & Paul, 1993; Lüders, Ernst, & Stille, 1995; Trivedi et al., 2001) have explored the method of automatic window layouts with some success. Nevertheless, they have rarely been adopted in practical multi-window interfaces, since most studies focus their attention on ways to better exploit empty display space and ignore the fact that the destruction of user’s mental map of window layout induced by window deployment adjustment tends to cause users confusion and frustration. The term mental map refers to the structural cognitive information a user creates internally by observing the layout of the graph (Purchase, Hoggan, & Görg, 2007). This internal cognitive structure represents the user’s underlying understanding of the information. Several studies (Coleman & Parker, 1996; Misue, Eades, Lai, & Sugiyama, 1995; Purchase, 1998; Purchase, et al., 2007; Purchase & Samra, 2008) have suggested that it is important for dynamic graph layout to preserve consistent mental map of users, otherwise confusion may result. As window layout can also be treated as graph layout consisting of rectangles, the suggestions are appropriate for the domain of automatic window layout as well. However, how to integrate mental map preservation into the traditional method of automatic window layout is still a great challenge.

To reduce user’s manual adjustment to a window layout, the following three aspects may be considered: making use of empty screen space, minimizing overlap between windows and preservation of mental map. Better utilization of empty screen space and less overlap lead to neater window layouts with larger windows, which are capable of showing larger amount of information to users at the same time and, therefore, greatly reduce the probability that users manually adjust the layout. By means of preservation of mental map, extra cognitive load of mental map reconstruction is eliminated. Users would not be interrupted and are kept in their task domain, thereby achieving higher overall task performance. However, the three aspects are in conflict, among which a compromise needs to be achieved. Multi-objective optimization algorithms are commonly used to obtain the tradeoff among several conflicting optimization objectives, ensuring that performance improvement of a certain objective would not cause performance decrease of any other objectives (Coello, Lamont, & Van Veldhuizen, 2007). For this reason, they are very appropriate choices for the problem domain.
Related Content

An Office on the Go: Professional Workers, Smartphones and the Return of Place
www.irma-international.org/article/office-professional-workers-smartphones-return/49667/

A Motive Analysis as a First Step in Designing Technology for the use of Intuition in Criminal Investigation
www.irma-international.org/article/motive-analysis-first-step-designing/2934/

Interpretive Flexibility Along the Innovation Decision Process of the UK NHS Care Records Service (NCRS): Insights from a Local Implementation Case Study
www.irma-international.org/chapter/interpretive-flexibility-along-innovation-decision/7284/

Habits of the Mind: Challenges for Multidisciplinary Engagement
www.irma-international.org/chapter/habits-mind-challenges-multidisciplinary-engagement/45318/
Adaptation and Personalization of User Interface and Content


www.irma-international.org/chapter/adaptation-personalization-user-interface-content/22263/