2016 5th International Workshop on Games and Software Engineering

Serious Games for NP-hard Problems: Challenges and Insights

Christina Chung, Asako Matsuoka, Yueti Yang
University of Toronto
{chr.chung, asako.matsuoka, elsie.yang}@mail.utoronto.ca

Julia Rubin
MIT
mjulia@csail.mit.edu

Marsha Chechik
University of Toronto
chechik@cs.utoronto.edu

ABSTRACT

This paper describes the on-going work of developing a serious game (a.k.a "game with a purpose") to solve the NP-hard problem of n-way merging. We outline the challenges that were encountered while designing the game, steps that we took to overcome these challenges and results of the preliminary evaluation of our current game design. We hope our experience will be useful for those developing serious games to solve other computationally expensive problems.

CCS Concepts

• Computing methodologies • Applied computing→Computer games • Information systems→Massively multiplayer online games • Theory of computation→Design and analysis of algorithms.

Keywords

N-way merge; model merging; serious games; NP-hard problems.

1. INTRODUCTION

Serious games (a.k.a. "games with a purpose") [1] have recently emerged as a promising way of engaging people in solving problems that computers cannot efficiently solve. Most serious games developed until now focus on solving cognitive problems, e.g., tagging images for enhancing image search [2], locating objects in images [3] and guessing search queries that yield a given web page [4].

Recently, a few approaches also explored the idea of using games for solving computationally intensive problems that are NP-complete or undecidable, e.g., graph pebbling [5], SAT [6], type-checking [7] and aligning protein sequences [8, 9, 10]. As such problems are deemed insolvable by computer algorithms in an efficient manner, the conventional approach is to develop approximate or heuristic solutions. Earlier work demonstrated that creating serious games to solve such problems could be a promising alternative.

In this paper, we explore the applicability and suitability of serious games to find an efficient solution for the *n-way merging problem* (NwM) [11]: combining multiple models into one by grouping elements with similar attributes. In our earlier work [11], the last two authors showed that this problem is equivalent to the weighted set packing problem and is thus NP-hard. We also showed that no existing solution with a fixed approximation factor is applicable to solve the NwM problem for inputs of realistic size and complexity. We thus devised a heuristic solution that, despite the lack of a fixed approximation factor, outperforms other possible approaches that are used to solve the NwM problem in practice.

While exploring the applicability of serious games to solve the NwM problem, we faced a set of obstacles that might arise when developing serious games for other problems of the same kind. The goal of this paper is to highlight these obstacles and to present our solutions to some of them.

The rest of the paper is structured as follows. Section 2 gives the overview of our approach. Section 3 summarizes the challenges that we faced in the game's development process and outlines potential solutions. Section 4 presents a preliminary evaluation, while Section 5 concludes the paper.

2. OVERVIEW

N-way model merging. NwM [11] is the process of combining multiple models into one by grouping elements with similar attributes. Figure 1(a) shows an example of three similar yet somewhat different UML models (M1, M2, and M3) of a hospital system. These models are simplified versions of their "real" counterparts developed by independent stakeholders and that now need to be consolidated into a single-copy representation [11]. Model M1 contains a single "Care Taker" element (element #1); model M2 contains elements "Physician" and "Nurse" (#2 and #3), and model M3 contains a single element "Nurse" (#4).

The three models are combined into a model M1 + M2 + M3 by merging elements from distinct models that are "alike". Similarity of model elements is established by considering similarity of their attributes. In this example, attributes are UML element names, attributes, and relationships with other model elements.

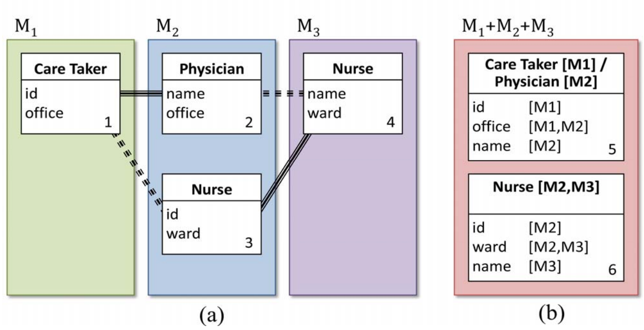

Figure 1. (a) Models M1, M2, M3 are combined into (b) a merged model M1+M2+M3 through the process of NwM.

GAS'16, May 16 2016, Austin, TX, USA
© 2016 ACM. ISBN 978-1-4503-4160-8/16/05 $15.00
DOI: http://dx.doi.org/10.1145/2896958.2896963

An optimal solution to NwM produces a result in which the chosen groupings have the highest degree of similarity. The optimal solution for the example in Figure 1 is to match and merge elements #3 and #4 (they have two attributes in common: their name, and the attribute "ward") and elements #1 and #2 (they both have the attribute "office"). The merged result is shown in Figure 1(b). Matching element #1 with #3, and element #2 with #4 is an alternative but non-optimal solution.

A realistic Hospital example, analyzed in [11], contains eight different models, with the number of elements in each model ranging between 18 and 38 (27.63 on average), and the number of attributes per element ranging between 1 and 15 attributes (4.76 on average). Exploring ways to efficiently perform NwM becomes increasingly important for large and complex software models, such as the Hospital model, as it is infeasible to do this manually or employ a brute-force approach in which all possible combinations are considered. In this work, we report on building a serious game for NwM in hope that it will produce better results than the heuristic approach proposed in [11].

Figure 2. People are able to quickly make out that the 5th and 6th dogs are similar, while the 1st and 2nd are not. Source: http://www.doggyhq.co.uk/media/wysiwyg/dogs-in-row-2.jpg.

Rationale and design of the game. The goal of our game is to make humans solve an NP-hard problem in a fun and engaging manner. Therefore, the problem must be presented in a way that allows non-experts to tackle it effectively. At the same time, we need to leverage human abilities that will make it possible for them to arrive at high-quality solutions in a reasonable time. Several studies show that humans are skilled at visual comparisons [12, 13]. With high accuracy, people are often able to make snap judgments about whether objects are similar to each other. This can be illustrated in Figure 2. With just a quick glance, most humans are able to conclude that the fifth and sixth dogs are similar, while the first two dogs are distinct. Our approach takes advantage of this human ability by requiring players to perform NwM using visual comparisons. In essence, we are building a matching game where every element of input software models is represented by a game character. Each character visually encodes the properties of its corresponding elements. As a result, similar elements are represented by visually similar characters while dissimilar elements – by visually dissimilar ones.

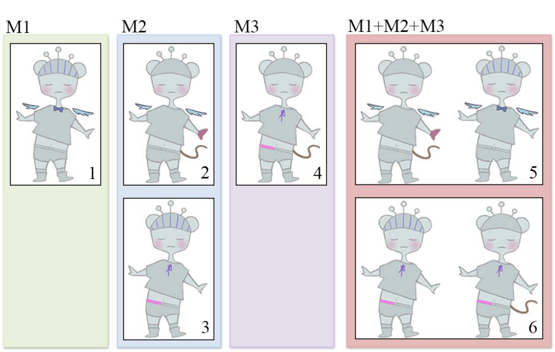

Figure 3. Alien encoding of the models in Figure 1. M1 + M2 + M3 depicts the optimal merge.

For example, the alien characters in Figure 3 encode the Hospital model elements in Figure 1. Features (wings, tail, etc.) are added to the basic alien character (in grey) according to the attributes of the model element that each feature represents: the helmet represents 'id', wings represent 'office', the tail represents 'name', and the belt represents the 'ward' attribute.

Players match characters that look alike and compete for the best score. The score is calculated using the formula in [11] and is high if the matched elements share a large number of common attributes. Thus, matching visually similar elements contributes to obtaining a higher score.

3. CHALLENGES

This section describes the challenges that we encountered during the game design process.

3.1 Object Encoding

One major challenge is the selection of objects that encode model elements in the game. Suboptimal object selection would lead to players failing to notice object similarities / differences and, as a result, grouping objects that are meant to be dissimilar. Moreover, the objects should be able to encode a large number of distinct attributes. To encode the Hospital model, 162 different attributes need to be encoded as features in the game! To address this challenge, we experimented with multiple designs, each adopting a certain encoding strategy and performed user studies to determine suitability of these designs to the NwM problem.

Entities as sets of features. Our initial idea was to represent models as *sets* of attributes. The first design used was mochi ("Japanese rice balls") sets – see Figure 4(a). Each mochi was representing one model attribute. Mochi sets were considered similar if they had identical mochis in common. In Figure 4(a), the two mochi sets with fives elements are similar because they both have four mochis in common. The mochi set with two elements is not similar the other two sets because there are no elements in common between them. After testing this design internally within the group, we realized that making visual comparisons with mochi sets was difficult since they looked too similar to each other.

The next design involved using bacteria sets – see Figure 4(b). This was an attempt to resolve the issue regarding the mochi set design, by making each bacterium more distinct from one another. While the bacteria design was an improvement, the process of comparing bacteria sets was inefficient. We had to explicitly count the number of bacteria that were common between sets, rather than being able to take a glance and quickly jump to a conclusion as in the "dogs example" of Figure 2.

Entities with superimposed features. We also experimented with entities that had *superimposed* features. This led to the alien design from in Figure 4 (c). Recall that there are up to 162 model attributes in the software models. As such, the chosen game object was required to have 162 features to encode these. Identifying 162 distinct features of an alien was difficult (and took the first author almost the entire summer!). The features had to be made small so that any occurring combination of these features (between 1 and 15, per number of model element attributes) can be superimposed on a single alien.

Our experiments with this encoding showed that since the plain alien template – an alien without any attributes – already had some features present, e.g., a helmet and shorts, test subjects had a hard time determining whether or not alien features were intended to be model attributes or template attributes. They thus deemed dissimilar elements to be similar. To avoid this problem, we further modified the alien design so that the plain alien template was greyed out (Figure 4(e)), making it clear which alien parts were meant to be attributes and which were not.

Another attempt to improve the alien design was to create "alien families", shown in Figure 4(d). In the alien families design, the 162 features were spread over four entities (a mom, a dad, a daughter and a son). Since each entity needed to encode only around 40 properties, they could then be made bigger and more noticeable, and the plain alien family template could be made completely bare.

Conducting user studies. Each of these designs had its own advantages and disadvantages. A large-scale University of Toronto ethics-approved user study was conducted in summer 2015 to determine which objects would be most suitable for the game. A full description of the protocol and results are available at http://www.cs.toronto.edu/~chechik/games_protocol.pdf. The study considered the grey alien design (Figure 4(e)), bacteria sets (Figure 4(b)), and alien families (Figure 4(d)). Our goal was to determine whether participants would correctly identify the most similar pair when presented with multiple objects on the screen. The results from 164 participants revealed that people performed best on grey aliens, and second-best on bacteria sets. However, we believe that people did not perform as well on bacteria sets due to the tiring effect as they were presented last in the questionnaire. Therefore, we will consider both bacteria sets and grey aliens as our game objects.

In a separate study with approximately 200 participants, we determined which alien features were most noticed by players. The study presented participants with a series of questions, asking them to identify a certain number of alien properties, working against a timer. Our assumption was that the order in which participants selected properties would determine which properties are most noticeable. Results of the study suggested that participants first noticed properties around the face and then moved down in a systematic manner. Furthermore, properties that were bright and prominent tended to be chosen first, even when they were not part of the face region. Using these findings, the alien design was refined by making features in the lower half of the alien brighter and more prominent than those in the face, with the intention that this would prevent players from neglecting lower-located features.

Future work. In the future, the alien design will be refined further by enlarging features that are not readily noticeable. In order to prevent players from incorrectly matching aliens that are meant to be dissimilar, alien features will also be redesigned to make them as distinct as possible. Prior to beginning the game, players will also be trained in an alien matching tutorial.

3.2 Layout Design

The next challenge involved designing a suitable game layout. For instance, how should the aliens be placed on the screen? Fitting all of the aliens on a single screen is clearly an issue for larger models, as aliens would be too small to see in detail, and displaying a full screen of aliens may overwhelm players. On the other hand, dividing aliens into separate screens may cause players to only match those appearing in their current screen while neglecting others. It was also not clear whether aliens should remain stationary or move around in the game. Keeping aliens stationary could lead to the players matching only those situated in some part of a screen while ignoring others. Moving them around may frustrate players and make the game a bad experience. As a result, we experimented with a variety of prototypes to determine what would be most suitable.

Moving aliens design. The first approach was to make the aliens float in the screen. A prototype of this game layout was not built, but a sketch is shown in Figure 5(a). We decided against it due to the difficulty of matching objects which are in constant motion and because of the scalability of this layout for large models, if the goal is to avoid overlapping bubbles. Another problem was that not all of the bubbles could be shown on the screen simultaneously, thus potentially hindering the players' potential to produce optimal groupings.

Row design. The next layout design was comprised of rows of aliens (see Figure 5(b)). The design's intention was to keep the

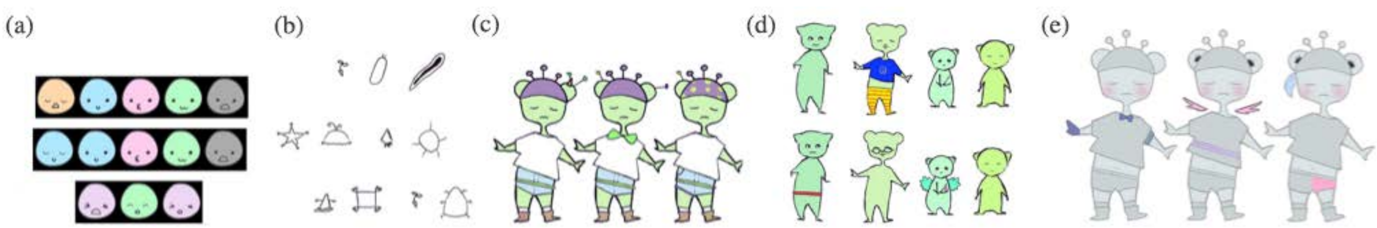

Figure 4. Game objects created: (a) Mochi sets; (b) Bacteria sets; (c) Aliens; (d) Alien families; (e) Greyed-out aliens.

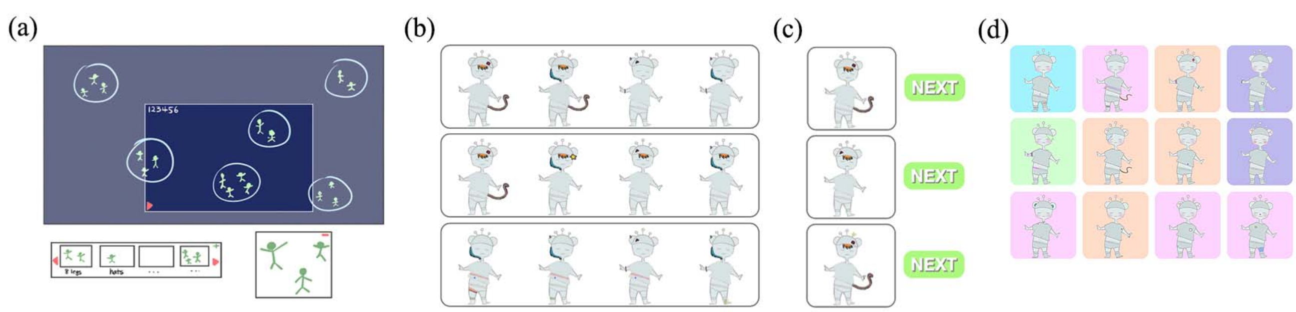

Figure 5. Game layouts created: (a) floating aliens layout; (b) row layout; (c) row layout with "next" button; (d) randomized display.

aliens stationary. Each row was made up of aliens belonging to the same input model. However, this design posed the issue of users employing a greedy algorithm to match aliens (i.e., matching aliens within the same part of the screen, such as first aliens in each row), which is proven to be ineffective for solving NwM [11]. To prevent players from employing a greedy approach to matching, the aliens presented in each row were randomized (see Figure 5(c)). Players could then browse through the rows by clicking a "next" button, which would generate a new set of randomly selected aliens for the corresponding model.

Randomized display. Eventually, we realized that the game layout did not need to group together aliens that came from the same models. In fact, players only need to match aliens that they consider similar; the underlying game mechanism should prevent them from matching aliens from the same model since this operation is illegal in NwM [11]. As such, grouping is unnecessary. The final prototype involved presenting all the aliens on the screen randomly, without associating them to a particular model (see Figure 5(d)). To make it clear to players that they are not to group aliens of the same model, each alien is displayed in a colored square. Players are disallowed to group aliens that have the same colored square.

Future work. We believe that the randomized display in Figure 5(d) is the most promising layout. It successfully prevents players from employing a greedy algorithm, as the aliens are randomly presented on the screen. The aliens are also stationary, preventing the distraction and frustration of players. Furthermore, it allows users to compare enlarged images of aliens, and so it resolves the problem of presenting many aliens on the screen without compromising detail. However, the layout still has the issue of not displaying all of the aliens on the same screen, which may cause players to neglect alien combinations that they do not see. A potential solution may be to regenerate the game screen at random intervals, or to display all of the aliens within a single screen. We intend to explore this approach as part of future work.

3.3 Engaging Players

Solving NwM and many other NP-hard problems is not an inherently interesting task. Thus, to create a successful game-based solution, developers must find ways to engage players.

Inspiration for player engagement had come from *Candy Crush Saga*, a viral matching game that has a similar theme to ours. Reasons for *Candy Crush Saga's* success are its captivating sound effects, nice visuals, and the competition that it promotes among its players. Similar features are already added to our design (score, sounds, winners board), and others are forthcoming. Furthermore, we hope to be able to create a backstory about the aliens and bacteria and the need to match them in order to motivate players to perform the relatively boring task of n-way model merging.

4. PRELIMINARY EVALUATION

The preliminary game is available at http://matchaliens.herokuapp.com/. Even though it is still work in process, we encourage people to play the current version to evaluate its fitness in solving NwM on a small-scale example. At the time of writing, two out of forty participants were able to achieve the optimal score of 6669 on a game consisting of nine aliens (three models; three elements per model; and three attributes per element). Furthermore, players who played the game multiple times were able to consistently improve their score in the game. Player scores were able to beat the heuristic algorithm in [11], which obtained a score of 6666. With this encouraging result, we plan to further extend and improve the game to evaluate it on larger case studies.

In future work, we would like to compare the time it takes for humans to reach the optimal solution in comparison to the brute-force approach.

5. CONCLUSION

Serious games may be a promising alternative to conventional approaches of solving computationally expensive problems. This paper reports on the on-going work of developing a serious game to solve NwM, the task of combining several models into one. The primary challenge involved in developing serious games for computationally expensive problems stems from HCI. These problems must be abstracted into representations that can be easily understood by non-experts. This involves creating a game interface that is intuitive, enjoyable, and playable while still meeting the goal of the research. In our case, great difficulty was faced in meeting these characteristics as shown in our numerous trials and errors. In hindsight, it seems that an essential part in overcoming these challenges is to establish synergies between software engineering and HCI. All in all, we hope that the examples drawn from our experiences will provide insight for those who may consider developing serious games.

6. ACKNOWLEDGMENTS

We gratefully acknowledge Si Hua Cao Liu and Angel You for their contributions to this research, and the Natural Sciences and Engineering Research Council of Canada for funding support.

7. REFERENCES

[1] Von Ahn, L. 2006. "Games with a Purpose". *IEEE Computer 39(6)*.

[2] Von Ahn, L. & Dabbish, L. 2004. "Labeling Images with a Computer Game". In *Proc. of CHI'04*.

[3] Von Ahn, L., Liu, R., & Blum, M. 2006. "Peekaboom: a Game for Locating Objects in Images". In *Proc. of CHI'06*.

[4] Ma, H., Chandrasekar, R., Quirk, C., & Gupta, A. 2009. "Page Hunt: Improving Search Engines Using Human Computation Games". In *Proc. of SIGIR'09*.

[5] Cusack, C., Largent, J., Alfuth, R., & Klask, K. 2010. "Online Games as Social-Computational Systems for Solving NP-complete Problems". In *Proc. of Meaningful Play'10*. 5.

[6] DeOrio, A., & Bertacco, V. 2009. "Human Computing for EDA". In *Proc. of DAC'09*.

[7] Dietl W., Dietzel S., Ernst M., Mote N., Walker B., Cooper S., Pavlik T. & Popovic Z. 2012. "Verification Games: Making Verification Fun". In *Proc. of FTfJP@ECOOP*.

[8] Hess, M., Wiemeyer J., Hamacher K., & Goesele M. 2014. "Serious Games for Solving Protein Sequence Alignments - Combining Citizen Science and Gaming". In *Proc. of GameDays'14*.

[9] Cooper, S., Khatib, F., Treuille, A., Barbero, J., Lee, J., Beenen, M., Leaver-Fay, A., Baker, D., Popovi´c, Z., et al. 2010. "Predicting Protein Structures with a Multiplayer Online Game." *Nature 466(7307)*.

[10] Kawrykow, A., Roumanis, G., Kam, A., Kwak, D., Leung, C., Wu, C., Zarour, E., Sarmenta, L., Blanchette, M., & Waldisp¨uhl, J. 2012. "Phylo: A Citizen Science Approach for Improving Multiple Sequence Alignment". *PLoS ONE 7(3)*.

[11] Rubin, J. and Chechik, M. 2013. N-way Model Merging. In *Proc. of ESEC/FSE'13*.

[12] Lamba H., Sarkar A., Vatsa M., Singh R., & Noore A. 2011. "Face Recognition for Look-Alikes: A Preliminary Study". In *Proc. of IJCB'11*.

[13] Sinha P., Balas B., Ostrovsky Y., & Russell R. 2006. "Face Recognition by Humans: Nineteen Results All Computer Vision Researchers Should Know About". *IEEE, 94(11)*.