Balancing and Transposition of Maps for Location-based Games

Luís Fernando Maia
Federal Institute of Education - IFMA
Caxias-MA, Brazil
luis.maia@ifma.edu.br

Windson Viana
GREaT - MDCC
Federal University of Ceará - UFC
Fortaleza-CE, Brazil
windson@great.ufc.br

Fernando Trinta
GREaT - MDCC
Federal University of Ceará - UFC
Fortaleza-CE, Brazil
fernandotrinta@great.ufc.br

***Abstract*—Developing worldwide Location-Based Games (LBGs) is a challenging task due to the need to deploy game instances in multiple locations. Therefore, current LBGs are not available in many areas, especially small cities and poor neighborhoods of big cities. In this thesis, we introduced measurements to estimate game balancing in modern LBGs. We also proposed a new optimization algorithm that aims at transposing maps of LBGs with minimal variations in their game balancing. To validate the proposed approach, we designed four LBGs and conducted experiments with subjects. Results indicate that the proposed approach contributes to mitigate critical challenges in the development of LBGs, both by saving resources and allowing players to compete more fairly.**

***Index Terms*—Pervasive computing, Internet of Things, Artificial intelligence**

I. Introduction

Smartphones have impacted the way people play games and have boosted the game industry due to its immense potential as a gaming platform. Additionally, it has allowed the implementation of new game genres, such as Location-based Games (LBGs), which can be defined as games that use players' location to modify the game state during runtime.

Lately, both industry and academia have focused efforts on the development of LBGs, e.g. *Ingress*, *Pokémon GO*, *Ghostbusters World*, *The Walking Dead: Our World*, and *Harry Potter: Wizards Unite*. This market has shown great potential, with some games generating more than $2 billion dollars in revenue[1].

Furthermore, LBGs are being used in tourism, education, and health. Still, there are a myriad of applications that can benefit from the use of LBGs, ranging from teachers designing games that require students to visit historical places, crowd-sensing for companies or governments, businesses promoting advertising campaigns, among others [1], [2].

Despite all of its potential, LBGs still represent just a fraction of the mobile gaming industry. In fact, only a few game studios are currently venturing into this area, especially due to the high cost and time required to produce these type of games. For instance, Niantic and Google spent roughly $30 million dollars in the development of *Pokémon GO* [2].

[1]https://www.superdataresearch.com/3-reasons-why-mobile-ar-developers-should-focus-on-ads-and-not-in-app-purchases/

[2]https://gamerant.com/pokemon-go-30-million-dollar-investment/

Designing an LBG is very difficult because developers have to create games to be available in a vast number of places, and maps of LBGs must fulfill requirements that suite aspects of both real and virtual environments, such as avoiding private properties and unreachable areas. Consequently, there are two main challenges that hinder the mass production of new LBGs. First, the need to map the game to countless places in such a way the generated instances can preserve their game balancing; and second, the high costs and time required to develop worldwide LBGs [3].

These challenges have impaired many game studios and independent developers from creating more games. In this context, we proposed an alternative to alleviate this problem called *Transposition of maps for LBGs*. It consists of using Procedural Content Generation (PCG) to create maps of LBGs according to the location of each player. Another crucial feature addressed in this thesis concerns the balancing of transposed maps. A proper map transposition has to simultaneously allow the game to be played in multiple places and provide equivalent balancing in every instance of the game to ensure fairness among the players.

The main contributions of this work are summarized below:

- A PCG approach to tackle the transposition of maps for LBGs while focusing on maintaining game balance;
- A simplified game model that can represent several types of LBGs;
- Measurements to gauge game balancing in LBGs (Internal Difficulty Level - $\mathfrak{I}$ and Minimum Balance Difference - $\mathfrak{M}$);
- Formulation of the transposition of maps for LBGs as a Graph Matching Problem;
- A novel algorithm called Parallel Weighted Ullmann (PWU);
- Experiments to validate the proposed approach with several subjects.

As a result, this research has been involved in the publication of 4 (four) papers in prestigious conferences and journals. The game model used to represent the games was originated from the work [4], awarded as one of the best papers in the 2017 Brazilian Symposium on Computer Games and Digital Entertainment. Later, the game model was also used in [5],

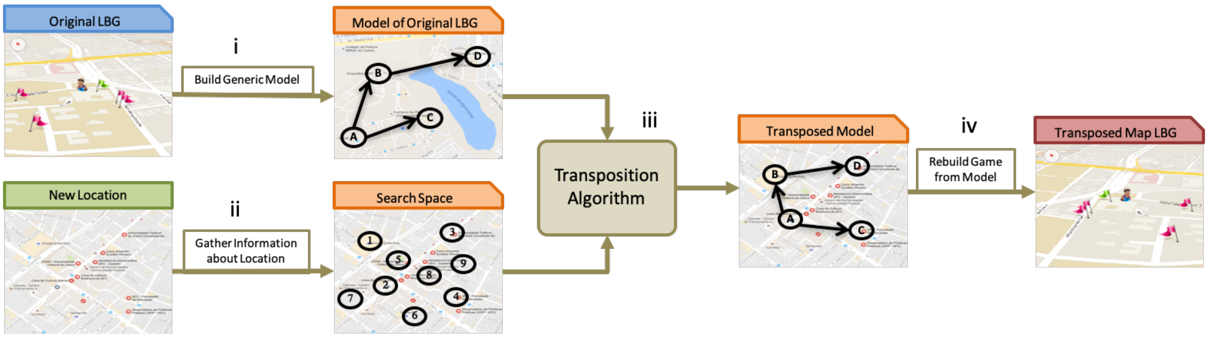


Fig. 1. Execution pipeline for the proposed approach.

published in the Journal of Entertainment Computing. The game balancing measurements ($\mathfrak{I}$ and $\mathfrak{M}$) were first published in the IEEE Conference on Computational Intelligence and Games (CIG) [6], along with the use of MCTS (Monte Carlo Tree Search) to improve game balancing in the game *Pokémon GO*. The MCTS algorithm was also used to transpose the map of a game in [7], published in the Journal of Entertainment Computing.

Additionally, a paper was submitted to the Journal of Pervasive and Mobile Computing to introduce the overall approach, an article detailing the PWU algorithm is expected to be submitted to the Journal of Computational Optimization and Applications, and an LBG is currently being developed using of the methods presented in this thesis.

II. Balancing and Transposition of Maps for LBGs

The proposed approach contains the following steps: (*i*) build a generic model of the original LBG, (*ii*) gather information about locations in the area where the transposition must take place, (*iii*) apply the transposition algorithm to a set of Points of Interest (POI) in the new location, and (*iv*) rebuild the game map from the resulting game model to obtain the transposed instance of the LBG map. Fig. 1 presents the pipeline of execution.

A. Game Model

This work presents a game model based on weighted directed graphs that is capable of representing several types of LBGs using a compact data structure. The method relies on weights to estimate game balancing and perform transposition, and this model has successfully been used in [7] to showcase the transposition of LBGs.

The proposed game model converts LBGs into weighted directed graphs using a straight mapping. In this case, the model is defined as a graph $G = (V, E, W)$, where V is a set of nodes representing specific locations (places to visit), E is a set of edges ($E \subset V \times V$) symbolizing an existing path between places, and W is the set of edge weights ($W : e_{i,j} \rightarrow \mathbb{N}+$) typifying the cost to move along the path [5].

Accordingly, nodes can be used to replace geographic coordinates, places, or interaction points in a game. In contrast, edges are relevant for determining existing paths between nodes, and weights can be associated with distinct characteristics depending on the game. For instance, a fitness game may decide to use footsteps, calories burned, or heartbeats as weights, whereas competitive games can use time, speed and distance.

Furthermore, we can build a game model for a specific region using data gathered from public APIs, such as POIs, paths, and the cost to move between locations. Nowadays, there are many public APIs that provide a plethora of information about POIs (e.g. *Google Places*, *Foursquare Venues Service*, and *Nominatim*) and routes (e.g. *Google Distance Matrix*, *Mapbox*, and *Microsoft Bing Route Data*).

B. Measuring Game Balancing in LBGs

We devised two measurements to gauge the game balancing of maps of LBGs and thus assess the effectiveness of the transposition. First, the Internal Difficulty Level ($\mathfrak{I}$), evaluates the equality of the costs to move between locations within a game (Equation 1).

$$\mathfrak{I} = \frac{\sum_{x=1}^{|V|} \sum_{y=1}^{|V|} W(V_x, V_y)}{|V|}, \quad W(V_x, V_y) \neq 0. \tag{1}$$

where $W(V_x, V_y)$ is the cost to move from node V_x to V_y, and $|V|$ is the number of nodes in the graph [6].

The second metric is called Minimum Balancing Difference ($\mathfrak{M}$) and depicts the best similarity between the paths of two games (G_A and G_B), as shown in Equation 2.

$$\mathfrak{M} = min \sum_{x=1}^{|V_A|} \sum_{y=1}^{|V_B|} |W_A(V_{Ax}, V_{Ay}) - W_B(V_{Bx}, V_{By})| \tag{2}$$

where $W_A(V_{Ax}, V_{Ay})$ and $W_B(V_{Bx}, V_{By})$ represent the weights of paths connecting nodes V_{Ax} to V_{Ay} and V_{Bx} to V_{By}, respectively [6].

C. Problem Formulation

As depicted in Fig. 1, the transposition algorithm takes two game models as input, one representing the original version of the game - here called the target model $G_T = (V_T, E_T, W_T)$ - and another characterizing the search space $G_S = (V_S, E_S, W_S)$ [7]. The purpose of the transposition algorithm is to create a bijective mapping between locations from the original game and the search space ($\mathcal{F} : V_T \rightarrow V_R$).

Consequently, the resulting game model $G_R = (V_R, E_R, W_R)$ is composed of transposed locations selected from the search space ($V_R \subset V_S, E_R \subset E_S$, and $W_R \subset W_S$).

Ideally, the transposition algorithm will produce a transposed map where the cost to move between each location has an equivalent cost of the corresponding path in the original map. Thus, the ideal transposed map would satisfy the following relation:

$$\sum_{v_x \in V_T} \sum_{v_y \in V_T} (W_T(v_x, v_y) - W_R(\mathcal{F}(v_x), \mathcal{F}(v_y)) = 0$$

However, the cost to move between locations in distinct regions is seldom equal, therefore the transposition algorithm must operate to ensure the game balancing between G_T and G_R to be as similar as possible. This challenge is formulated as an optimization problem by defining a cost function $\mathfrak{T}$, which must be minimized to make G_R as similar as possible to G_T.

III. Parallel Weighted Ullmann

In theory, the transposition algorithm should select the best solution from the many candidates. However, in practice, the time and effort to process all solutions increase exponentially. Consequently, we implemented a set of algorithms conceived to handle the broad range of game models according to their sizes, namely Parallel Brute Force (PBF), Monte Carlo Tree Search (MCTS), and Genetic Algorithm. Moreover, we devised a novel algorithm called Parallel Weighted Ullmann (PWU) to tackle the transposition problem in a faster way.

The PWU is based on the algorithm proposed by Ullmann that combines a pruning analysis with a depth first tree search to eliminate inadequate solutions from the search space [8]. Despite being widely referred and used in many applications, Ullmann's algorithm is not suited to work with weighted graphs. Hence, the PWU alters the refinement of the search space to make the pruning process to include a restriction based on the difference between the weights of vertices.

For instance, given the vertices $v_T \in V_T$ and $v_S \in V_S$, to define whether they can be matched it is necessary to examine if the weight w_T of each edge e_T has at least one corresponding edge e_S with similar weight w_T. This approach can be implemented by linking a threshold $\tau \in \mathbb{R}^+$ to each edge to define an acceptable percentage of similarity defined by the relation:

$$\mid w_T - w_S \mid \leq \tau * w_T \quad (3)$$

In a nutshell, the PWU algorithm presents a novel approach to prune large branches of a search tree before focusing on cases that can be addressed individually, then it uses a parallel approach to select the best among the remaining solutions in the search tree. The PWU has complexity $O(N|V_S| + \mathcal{M})$, where N is the number of iterations the algorithm performs when pruning the search tree and $\mathcal{M}$ is the number of solutions remaining to be processed in parallel.

IV. Evaluation

Two evaluations were conducted to validate the transposition of maps of LBGs. The first (Quality Evaluation) devoted to compare the algorithms using $\mathfrak{I}$ and $\mathfrak{M}$, and the second (User Evaluation) aiming at assessing whether the proposed method can successfully generate games in multiple places.

A. *Quality Evaluation*

We conducted the quality evaluation with four distinct algorithms: MCTS, GA, PWU, and a Parallel Brute Force (PBF) approach. The evaluation consisted in applying the transposition algorithms to 30 randomly generated pairs of graphs (G_T,G_S). The final result was given by the average $\mathfrak{M}$ and the turnaround time gauged for each pair.

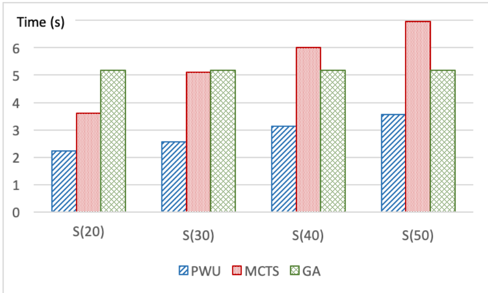


Fig. 2. Chart depicting the average time in seconds to process graphs with $G_T = 10$ and varying sizes of G_S.

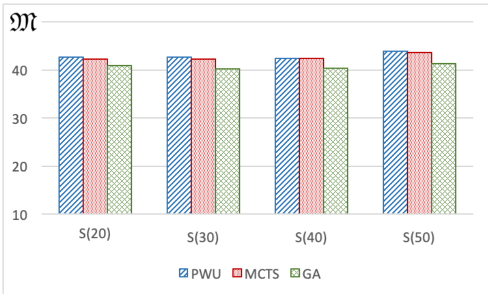


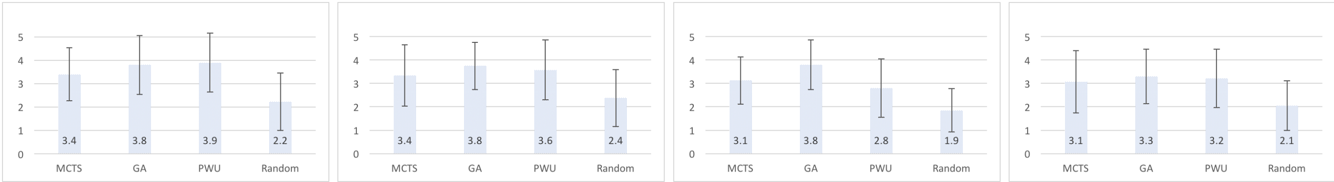
Fig. 3. Comparison between the average $\mathfrak{M}$ for $G_T = 10$ and G_S with different sizes.

The results showed that all algorithms outputted similar $\mathfrak{M}$ (Fig. 3) for graphs with size 10, however, PWU delivered faster performance (Fig. 2). Conversely, when applied to graphs with size 20 and 30, the GA was the only algorithm capable of delivering satisfying results.

B. *User Evaluation*

Another evaluation was conducted with 40 subjects to assess whether the proposed approach works appropriately in multiple places despite the style and size of the games. Hence, four LBGs were developed, each containing distinct characteristics regarding mechanics, the number of places to visit (size), strategy, and game flow.

During the test, users were asked to rate (using a Likert Scale) the similarity between a designed LBG and a set of



(a) (b) (c) (d)

Fig. 4. Average score for the transposed maps of the games.

game maps generated by distinct transposition algorithms. Then, a statistical investigation was conducted to determine whether $\mathfrak{M}$ (and hence the outcome of each algorithm) is linked to transpositions deemed successful.

Results show the transposition algorithms were better evaluated than the random selection of points in all cases, and the following null hypothesis was tested:

H_0: The proposed method has no significant effect on the score of transposed maps.

A One-way ANOVA was run with the rates provided by users for each game, and the analysis indicated that the hypothesis H_0 must be rejected in all cases, as $F > F_{critical}(2.66)$ and $p - value < 0.05$ for each game tested. The data also allowed us to link higher scores to smaller values of $\mathfrak{M}$, thus indicating that the metric is functional to assess the game balancing between LBGs.

The tests required the transposition of 800 game maps and the analysis of more than 3.000 POIs. Among all these numbers, there has not been a single case where the method failed to generate the transposed version of a map. Furthermore, the average time spent for the methods to calculate the transposition did not exceed 5.5 seconds, hence allowing the algorithms to be used on demand.

V. Conclusion

The thesis was approved in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Federal University of Ceará (UFC). The work was supervised by the professors Dr. Fernando Trinta and Dr. Windson Viana, both members of the Group of Computer Networks, Software Engineering and Systems (GREat).

The main focus of the work is to address the challenge of transposing maps of LBGs while focusing on maintaining the original game balancing. There are many motivations behind this work, such as the rising popularity of LBGs and the benefits they bring to health and entertainment, the challenge of designing better, cheaper and fairer games, and the possibility to increase the reach of LBGs to as many places as possible.

The work has the potential to enhance the development of LBGs by reducing the cost and complexity associated with it. Moreover, the proposed approach enables these games to deliver new modes of competition. Since the possibility to create maps with similar game balancing enables players to compete or interact within the same virtual environment regardless of their location.

It is also possible to adapt this approach to include graph consensus, so it will feasible to use multiple features when determining game balancing. As a result, the transposition could benefit from this trait to create maps that can be balanced according to multiple aspects, such as distance, time, calories, etc.

Finally, there are social aspects linked to the adoption of this method in the development of LBGs as most of these games do not include POIs in impoverished neighborhoods and rural areas. Hence, this work also tackles this issue by shifting the focus from playing where the game is deployed to deploying where the players are.

References

[1] C. Arkenson, Y.-Y. Chou, C.-Y. Huang, and Y.-C. Lee, "Tag and seek: a location-based game in tainan city," in *Proceedings of the first ACM SIGCHI annual symposium on Computer-human interaction in play*, (Toronto, ON, CA), pp. 315–318, ACM, ACM, 2014.

[2] I. Rubino, C. Barberis, J. Xhembulla, and G. Malnati, "Integrating a location-based mobile game in the museum visit: Evaluating visitors' behaviour and learning," *Journal on Computing and Cultural Heritage (JOCCH)*, vol. 8, no. 3, p. 15, 2015.

[3] T. Tregel, L. Raymann, S. Göbel, and R. Steinmetz, "Geodata classification for automatic content creation in location-based games," in *Serious Games* (M. Alcañiz, S. Göbel, M. Ma, M. Fradinho Oliveira, J. Baalsrud Hauge, and T. Marsh, eds.), (Cham), pp. 212–223, Springer International Publishing, 2017.

[4] C. Ferreira, L. F. Maia, C. Salles, F. Trinta, and W. Viana, "A model-based approach for designing location-based games," in *2017 16th Brazilian Symposium on Computer Games and Digital Entertainment (SBGames)*, (Curitiba, PR, BR), pp. 29–38, 2017.

[5] L. F. M. Silva, C. Noleto, M. Lima, C. Ferreira, C. Marinho, W. Viana, and F. Trinta, "LAGARTO: A location based games authoring tool enhanced with augmented reality features," *Entertainment Computing*, vol. 22, pp. 3–13, 2017.

[6] L. F. Maia, W. Viana, and F. Trinta, "Using monte carlo tree search and google maps to improve game balancing in location-based games," in *2017 IEEE Conference on Computational Intelligence and Games (CIG)*, (NY, NY, USA), pp. 215–222, IEEE Computer Society, Aug 2017.

[7] C. Ferreira, L. F. Maia, C. de Salles, F. Trinta, and W. Viana, "Modelling and transposition of location-based games," *Entertainment Computing*, vol. 30, p. 100295, 2019.

[8] J. R. Ullmann, "An algorithm for subgraph isomorphism," *J. ACM*, vol. 23, pp. 31–42, Jan. 1976.