Meaningful Choice in Strategic Unit Selection: A Case Study of Unit Rankings in StarCraft II

Shaun Bangay
School of Information Technology
Deakin University, Geelong, Australia
shaun.bangay@deakin.edu.au

Owen F. Makin
School of Information Technology
Deakin University, Geelong, Australia
owen.makin@deakin.edu.au

ABSTRACT

Trading card games challenge players to select a card from their personal deck to compete against cards from an opponent's deck with the outcome determined by rules specific to the game. Players desire that the cards in their decks offer meaningful choice relative to those held by the opponent since one player dominating removes all challenge from the competition. The issue of determining the existence and extent of meaningful strategies during competitive selection processes is common to range of other contexts, including picking units for combat in real-time strategy games such as StarCraft II. The approach described models game outcomes as a skew-symmetric matrix and presents an algorithm for excluding dominated and dominating units, and then further ranks the remaining meaningful choice options. A metric: band size quantifies the degree to which subsets of units can still contribute to meaningful game play. This process is applied to a single unit combat scenario using the StarCraft II rules to identify and rank a core set of 39 combat units that only offer meaningful choice within a limited neighbourhood of 12 units around each unit.

CCS Concepts

•Human-centered computing → Interaction design theory, concepts and paradigms; •Computing methodologies → Causal reasoning and diagnostics;

Author Keywords

StarCraft; meaningful choice; strategy; balance; unit selection; game design

INTRODUCTION

Players of interactive games are continually confronted with choices ranging from selecting appearance of their avatar, providing responses to in-game choices relating to the deployment of their unit(s), and to applying accumulated resources to directing the development of their in-game characters. Some of these choices are purely cosmetic while others are meaningful and affect the outcome of the game. Strategic game play then involves not only anticipating the consequences of meaningful choices, but also assessing the extent to which any choice will be meaningful within the current context.

This paper investigates how meaningful choice can be quantified within one particular game context: the core trading card mechanic of selecting a specific unit from a set of candidates which is then pitted against that of the other player in a two unit competition. The case study provided investigates the particular context of a *sub-game* of StarCraft II where two opposing units fight to the death in an otherwise unremarkable environment but according to the rules already defined by the StarCraft II combat engine. This scenario is representative of a key element of a number of similar well known and popular commercial games, involves competition between two units that are potentially unbalanced, asymmetrical [18] or orthogonally differentiated [42, 29]; with a non-trivial and largely opaque conflict simulation not amenable to systematic analysis but that is also the subject of detailed anecdotal analysis by players and popularly employed for developing strategies for artificial intelligence based game play [46].

The unit selection problem is the basis for many trading card style games, where each player must select a single card from their hand of cards during each round. It is core to the game design process where units must be divided into different factions (such as the Terran, Protoss and Zerg of StarCraft) or where units must be ranked according to the phases of game play represented as a technology tree [46]. Unit selection is a metaphor for strategic choice, in situations where outcomes can be directly measured for one strategy pitted against another. Meaningful game design requires intrinsic balance between the available strategies even in the absence of players. Complex game mechanics incorporate strategic selection, such as multi-unit combat scenarios in RTS games which often degenerate into combat between single units along a battle front (although clearly this is not the only form of interaction between units possible in such cases). This analysis involves only consideration of competitive outcomes so this process can also be applied to players themselves, through data collected in ranking systems and thus being applicable to the matchmaking problem [23]. Unit selection also applies outside traditional game settings, for example selecting candidates from a party list to contest a particular election, or deciding on which products to stock when shelf space may be limited.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
CHI PLAY 2018, October 28–31, 2018, Melbourne, VIC, Australia.
Copyright is held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 978-1-4503-5624-4/18/10 ...$15.00.
https://doi.org/10.1145/3242671.3242690

BACKGROUND

Motivating players is a significant challenge in game design, typically involving intrinsic and extrinsic rewards [41]. Satisfaction has been linked to meaningful choice and autonomy across several studies [6, 10, 44], although these tend to use scenarios where choice is clearly linked to perceived consequence. The significance of particular strategic decisions in complex games are often masked by additional factors such as variations in the skills of competing players (dominating over choice of champion during individual matches in League of Legends [23]), or differences due to randomness, timing or other confounding factors. This affects players and game designers jointly who then have to rely on data collected after the game's release, or anthropomorphized insights assembled by experts within player communities [23][1]. Procedurally generated elements of game content benefit from models that predict their effect on game play [15].

This paper measures the extent to which the effect of access to particular groups of play pieces can affect the degree to which a player can control the game outcome. This supports the design process but also provides the starting point for analysis of more complex game mechanics derived from choice of individual strategic elements. This work standardizes the representation of the game structure but also incorporates considerations relevant to game design and the player experience.

Formal models of games and strategy

Abstract representations of games allow reasoning about their strategic properties. Game theory normal form allows dominant strategies and equilibria to be identified [34]. Decision trees represent strategies as sequences of tactical decisions [46]. The matrix elements used in game theory define the expected *payoff* for each combination of strategies, compared to the formulation used in this paper which uses the game *outcome*.

Feedback mechanisms control the impact of player actions, with negative feedback loops used to keep the game competitive by limiting the benefit of player advantages [38]. Positive feedback mechanisms amplify the effects of minor player advantages [37] while negative feedback limits the benefit of early lead. Stability can also be achieved by ensuring that the game has multiple Nash equilibria [30], particularly if these are robust under small perturbations. Combinations of individual game units or alliances formed between players can lead to unexpected emergent formations [36]. The effects of the feedback loops in the internal game economy can be explored by simulation once interactions between units have been quantified [11]. A Petri net derived model represents each component as a source, sink, converter or trader [11, 32]. Even relatively simple models can provide insight into the relationships between attributes that affect choice of available strategies [39][4].

The random walk view of games [13] divides the game into phases separated by reversals. Designers can ensure this by providing one side with an initial advantage which decays with time. Strategies then involve ensuring that the game lasts long enough that the game can be ended during the period when the player's pieces have reached their peak. Technology tree design sets the time before maximum strength units can be created and uses cool down timers that limit reuse. Evidence of long term play benefiting better players has been found for an analysis of strategies in StarCraft [12].

Relationships between agents is the basis for balance theory [19]. This concept can be generalized using a graph representation with edges labeled $+$ or $-$ (a signed graph) according to the nature of the relationship. The sign of the product of links in a cycle is significant [7], and in the original formulation results in being able to partition a balanced graph into two subsets of nodes with only positive edges internal to each set, and only negative edges between sets [7, 17, 9]. The latter representation identifies two opposing cliques; a common scenario in competitive games, and informs the process of forming subgroups, clusters or coalitions. The nature of the relationship represented is amenable to alternative interpretations, ranging from like/dislike [19] but also a higher/lower rank [27] similar to the win/loss concept applied in this paper which aims to represent stable cycles of *dominance* (status). As with game theory and the work in this paper, matrix representations of the graph structure facilitate various forms of analysis. The Laplacian matrix for a signed graph, defined as the difference between the diagonal degree matrix and the adjacency matrix, uses eigenvalue analysis to determine degree of balance [25].

Strategic analysis of games and game play

The strategies possible in any game can be the same for each player in symmetrical games [33], but are more interesting where players have different sets of assets encouraging strategic innovation [40, 37, 39]. All players in a multi-player game need to be able to select strategies that ensure fair chances of winning and hence need equivalent resources and play pieces available to each player. Communities in massively multiplayer online games will quickly identify a dominant element or strategy [16][34] which in turn forces all players to use this strategy or lose [26]. Elements of the game that are underutilized point to potential areas of imbalance [22]. Strategic analysis may be based solely on designer intuition [39], patching and player feedback, analysis of actions recorded from game play [21] or using state space searches e.g. human directed binary search [39] or optimization algorithms e.g. particle swarms [45].

Orthogonal differentiation of units has been suggested as an approach to avoiding dominant strategies [42]. A set of orthogonal basis vectors is created with each vector specifying an attribute unique to one of the units. No unit is equivalent to a combination of other units in the set. Assuming each attribute corresponds to a particular strategic opportunity the existence of an orthogonal basis ensures that any strict subset of units can be countered by sufficiently many units derived from one of the remaining basis elements [42, 29].

Relationships between individual strategic options can be manipulated or analyzed to predict game outcomes. A cyclic ordering between competing elements automatically avoids

[1] `http://liquipedia.net/starcraft/Category:Strategies`

dominant strategies [37, 39] as every unit can be countered by at least one other. Analysis focuses on relationships between *game outcomes* for individual units rather than on their defining attributes. These relations can be explicitly defined for all pairs of units [37] or inverted to produce attribute values using simulation and search [28]. Unit attributes (such as speed, range and firepower) affect strategic choice when they relate to resources required to acquire the unit [37]. Pairwise balance is computed by ensuring a mixed strategy with zero net payoff [37]. Ensuring cyclic ordering becomes difficult for large cycles involving many types of unit, or for large scale conflicts involving more than a pair of units.

Quantifying the value of a particular strategy through analysis of game play data allows elements of the player's skill to be included [31]. Player abilities are incorporated into the choice of strategies in a fighting game [1] by using reinforcement learning to select strategies optimized for scores within 10% of the player's. Human players noted variety in the strategies used by an adaptive agent in comparison to agents trained to find a single optimal strategy. Regulation of game challenge allows for dynamic difficulty adjustment [2] without removing the significance of player prior choices and consequent impact on autonomy and experience of competing players [3].

Developing measures of strategic benefit

Assigning value to particular strategies provides ways of representing and reasoning about choice. Such objective goals or fitness functions are used to iteratively refine the structure of a game [8]. Metrics based on simulations of individual game mechanics and on statistics collected from intermediate and completed versions of the games represent the value of outcomes associated with tactics and strategies [16]. A uniform frequency distribution over the usage of the different game elements indicates that they are all equally useful and hence balanced. While it can be convenient to have explicit measures such as player's score provided by the game to quantify value [5], heuristics based on smaller scale interactions (such as number of individual conflicts won during a battle) can provide an estimate of this [1]. Massively multi-player online games have very large numbers of players offering opportunities for collusion and alliances to affect the outcome [30].

Modelling players as random processes and using summary statistics such as win rates over a set of games provide an indication of any potential bias towards one side [22, 20] which can lead to a perception of imbalance [13]. The probability of the frequent changes in leadership is low [14] with the more likely outcome being that the player that first gains the lead will retain it. Regular changes of which player is in the lead or which is gaining would be perceived as balanced [20] represented as the number of zero crossings in the relative score or its derivative. Periodic measures of relative advantage quantify the effect of feedback mechanisms that encourage equilibrium throughout the game.

METHOD

Let us assume the existence of a set of units $\mathscr{H} = \{A_i\}$ where $i \in \{1,\dots,n\}$. An individual unit, A_i, may be a game piece, such as a Marine or Zergling in the StarCraft context, but could equally be a description of any item capable of comparison with another unit. Traditionally units have been described in terms of their attributes or properties [29] with the outcome dependent on some black box process that is sensitive to these attributes. We assume we have access to the outcomes for any competition between two units represented as a competitive game function $f : \mathscr{H} \times \mathscr{H} \mapsto \{-1,0,+1\}$ which maps two units A_i and A_j to $f(A_i,A_j)$ where the outcome is either $+1$ to represent A_i wins against A_j, 0 to represent a draw, or -1 to represent a loss. Other options for the range of the function can be considered, such as a continuous value representing the extent of the win, such as the final health value of the winning unit [24].

Let H_1 and H_2 be two non-empty subsets of $\mathscr{H}$. These could represent hands of cards held by each player. We wish to establish the extent to which one of the players is able to affect the outcome of the game by selecting a single unit from H_1 to be played against a unit selected by the other player from H_2.

In the simplest cases where we have a dominant unit A_k that beats all other units ($f(A_k,A_i) = +1, \forall i \neq k$) then there is no meaningful choice for the second player as the first player need only select unit A_k in order to win every time [18]. However intelligent game design is unlikely to allow such scenarios and we would expect that for every unit A_k there will be at least one other unit that it loses to and one other unit that it defeats. This condition is still insufficient to ensure a meaningful choice for the player as it does not exclude the existence of cliques of units that adhere to this property internally but that collectively still dominate all remaining units [17]. A rational player would then confine their choice to one of the units within this dominant clique.

The goal of this analysis is to produce a measure indicating the degree of meaningful choice within a competitive game function, and to determine where meaningful choice may exist relative to any set of units that an opponent may be able to deploy.

Skew symmetry

Intuitively we would expect that $f(A_i,A_j) = -f(A_j,A_i)$ i.e. that if unit A_i wins against A_j then unit A_j would lose against A_i, or that units that draw would do so irrespective of the order in which they participate in the competition.

If this condition holds or is enforced then:

- the game function can be represented by a $n \times n$ skew symmetric matrix where the entry at position(i,j) is $f(A_i,A_j)$, providing a lookup table representing the outcome of any unit competing against all others.
- $f(A_i,A_i) = 0$, and so all competitions involving an instance of a particular unit competing against another instance of the same type would end in a draw. Thus the diagonal elements are all 0 and thus contain no useful information.
- the symmetry means that the upper (or lower) triangular portions of the matrix would be sufficient to completely define the game.

If this condition does not hold then this also has an impact on the concept of meaningful choice for individual players. There may be various reasons why skew symmetry is not present, equivalent to the game function being able to differentiate between the first and second player. The game may have a turn based aspect to it, where the first player takes an action before the second providing some benefit to the first (e.g. able to select best vantage point or get first choice of available resources) or to the second (able to adapt and respond to the first player's strategy). This may even be implicit in the combat simulation due to processing of internal lists of units in a set order. Skew symmetry may also appear to be violated due to stochastic elements within the game simulation. Effects of random fluctuations need to be distinguished from a consistent bias.

Where skew symmetry is absent, the upper and lower triangular portions of the matrix can be separated into two different matrices, each of which can then be extended into a skew symmetric matrix. This is equivalent to creating two versions of the game corresponding to alternative orderings of the two units. Analysis then proceeds as for the skew symmetric case, for each of these alternatives.

Detecting absence of skew symmetry can be done easily where evaluation of the game function produces consistent deterministic results by directly comparing each matrix element (i,j) with its corresponding (j,i). Simulations that incorporate varying degrees of randomness are interesting with respect to their outcome over many games and so comparison of the mean outcome is required.

Lemma 1. *If $\exists A_i : f(A_i,A_i) \neq 0$ then the game function f is not skew symmetric.*

If there is a unit that does not draw against itself then skew symmetry is violated. Thus the *absence* of skew symmetry can often be quickly confirmed by finding such a unit. The effect of randomness in the outcomes of stochastic game functions is addressed by collecting multiple samples of game output for one unit played against itself. For any given unit A_i,the outcome $f(A_i,A_i)$ is repeatedly evaluated. Skew symmetry implies that the mean of this value should converge to 0. This can be trivially achieved if all individual outcomes result in a draw (outcome 0). When a non-zero outcome x is achieved, we expect that it should be matched with a corresponding outcome of $-x$ elsewhere in the outcome sequence. Thus outcomes $+1$ and -1 should occur with probability 0.5, once the 0 outcomes have been discarded. A two-tailed binomial test is used to identify the significance of any deviation from this distribution.

It is possible that all of the diagonal elements in the game matrix are indistinguishable from 0 but the game is still not skew symmetric. A more thorough test checks all combinations of units to identify any signs of bias. Skew symmetry can be verified by explicitly testing for cases where the outcome does not change sign when the order of combat is reversed (where $f(A,B) \neq -f(B,A)$). This approach suffers from the limitation that the number of trials required is $O(n^2)$ rather than $O(n)$ for the previous test (where n is the number of units in the system), potentially reducing the number of samples of each outcome that can be collected while simultaneously impacting on any significance levels of tests required to compare the two distributions. We elect to rather devise a measure of bias (lack of skew symmetry) which can be visualized to identify trends across units to distinguish a particular flaw in interactions between two units from a systematic flaw across many units.

Each pairwise interaction between units can be classified in terms of a bias measure $b_{A_i,A_j} = f(A_i,A_j) + f(A_j,A_i)$ which should evaluate to 0 for well behaved game functions. A normalized bias measure, that indicates how far the unit is from achieving skew symmetry, can then be assessed for each unit by combining the results for that unit interacting with all other units:

$$bias_{A_i} = \frac{\sqrt{\sum_{j=1}^{n} b^2_{A_i,A_j}}}{n} \tag{1}$$

Unit categorization

For the remaining steps, assume that f is skew-symmetric. As mentioned previously, meaningful choice is not possible with where we have dominant units. There are several other categories of unit that can also be pruned from consideration at this stage.

Definition 2. A unit A_i is k^+ balanced if $\exists A^+ = \{a_1,\ldots,a_p\}$ such that $f\left(A_i,A_{a_j}\right) = +1, \forall j \varepsilon \{1,\ldots,p\}$ and $p \geq k$ i.e. the unit A_i wins against at least k other units.

A similar definition applies to k^- balanced units, which lose to at least k other units.

Definition 3. A *meaningful choice* is a unit A_i that is both 1^+ balanced and 1^- balanced.

A meaningful choice clearly wins against at least one other unit, and loses to at least one other unit. This excludes in particular:

- dominating units: that are 1^+ balanced and do not lose to any other unit. Selecting these units represents a dominant strategy.
- dominated units: that are 1^- balanced and do not beat any other unit. There is no gain possible by selecting these units.
- passive units: units that draw with every other unit.

Further analysis is applied to only those units identified as meaningful choices, with a process for pruning outlined in Algorithm 1. The next stage derives a measure that reflects the degree of meaningful choice available, and presents an algorithm for providing a canonical ranking of these remaining units.

A measure of choice

The remaining units in the game are all meaningful choices once dominating, dominated and passive units have been pruned. The i^{th} row in the matrix represents the set of outcomes between unit A_i and all other units. While none of these will be dominating units, it is possible to identify the most dominant units by selecting the rows that have the most $+1$ entries. The structure of the matrix could be rearranged to

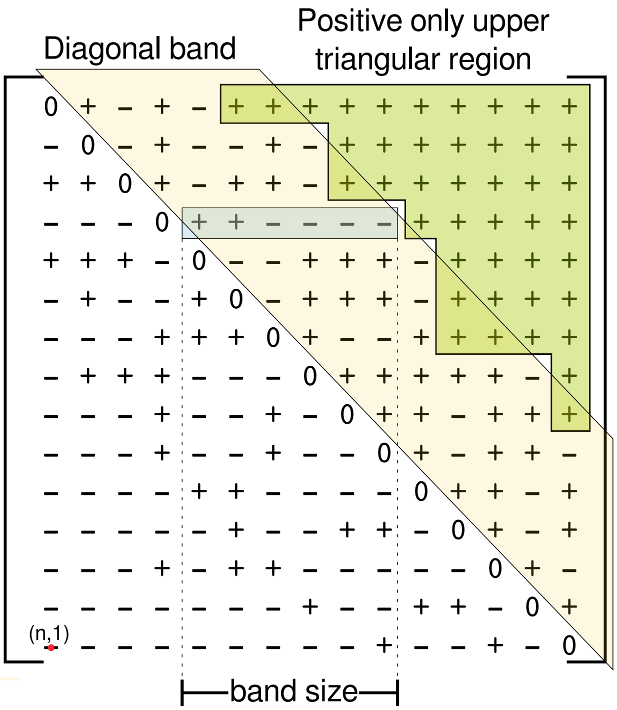

Figure 1. The skew-symmetric outcomes matrix can be manipulated by reordering the list of units. Maximizing the positive only upper triangular region and minimizing the size of the diagonal band provides a ordering of units with meaningful interactions with only a small set of their immediate neighbours.

place one of these most dominant units in the first row without changing the nature of the game by just relabeling the units (corresponding to reordering the list of units). Similar relabeling can be used to shuffle all the $+1$ entries to the right so that from some position j and beyond to the right, the first row of the matrix only consists of $+1$. After the relabeling, unit A_1 dominates all A_m for $m \geq j$. If one player selects A_1 then the other player loses meaningful choice if they are restricted to units $\{A_j, A_{j+1}, \ldots, A_n\}$.

If this process could be repeated in a greedy fashion for the remaining rows then it would provide a ranking of all the units. The extent of dominance would indicate the degree of meaningful choice if the players are restricted to choose from within a subset of the units. However the definition of meaningful choice also ensures that every unit has at least one counter-unit and so the process is not so straightforward because improving the arrangement to benefit one unit can worsen the situation for others. However we embrace the principle demonstrated and restrict the -1 upper triangular entries to the narrowest band along the diagonal of the game matrix, as shown in Figure 1 to maximize the degree to which dominance is presented. The width of this band at any row would represent the number of remaining units that are not completely dominated and hence provides a measure of the degree of meaningful choice left to this unit. The maximum width across all rows then represents the potential for meaningful choice across all units.

If we assume that units are provided in an ordered list $L = [A_1, A_2, \ldots, A_n]$ then any skew symmetric matrix representing the game function is equivalent to a reordering of the list. Swapping two elements in the list is equivalent to exchanging the corresponding rows and the corresponding columns, and can alter the width of the diagonal band of meaningful choice. Once a matrix is achieved that maximizes the degree of dominance present in the game then we achieve a characteristic ordering of the units that represents the dominance patterns within the game.

A measure of meaningful choice thus involves optimizing the form of the matrix through manipulating the order in which units are added. The optimization criterion is the band size of the matrix, and the optimal value of this fitness function becomes the measure of choice within the game.

The desired matrix should have each row end in the maximum numbers of +1s, but need have no more than all of the rows above it. Thus we only count trailing +1 in each row that also have +1 values in the column above, as shown in the positive only upper triangular region in Figure 1.

The matrix should maximize the number of elements in the upper triangular region of the matrix (containing top-right corner) that is exclusively +1. Various scoring criteria can either favour long +1 rows at the top (often a case of a few units that dominate most others), or moderate length rows throughout (most units dominate the last few) or some mix in between. Since there are usually multiple valid maximal configurations we also minimize the width of the diagonal band in the upper triangular region that remains to minimize the extent of influence across all units. This can be measured as the maximum distance from the diagonal elements to the last -1 entry in the same row. This quantity is defined as the band size β for the game as shown in Figure 1. Band size is too coarse on its own and not provide sufficient resolution to discriminate between similar configurations.

An objective function representing a combination of the two has proven to provide useful insights. Let T be the set of matrix elements that are $+1$, have only $+1$ elements to their right in the row, and have only $+1$ elements above and to the right in all previous rows. Let $d(i, j)$ between the Manhattan distance between element at position (i, j) and the position $(n, 1)$ (bottom, left hand element of the matrix). The objective function used for matrix M is:

$$C(M) = \sum_{(i,j) \in T} \frac{d(i,j)}{2n} - n^2 \beta(M)$$

The normalization constants ensure that band size is the primary criteria, with dominance relationships used to direct and refine the optimization. Skew symmetry ensures that the lower triangular portion of the matrix need not be considered.

Process

Measurement of meaningful choice involves:

1. Validation of skew symmetry. If this is not present then one player may have an advantage merely due to the order in which they play, inhibiting meaningful choice.

2. Skew symmetry is enforced, by setting $f(A_i,A_i) = 0$, and $f(A_i,A_j) = -f(A_j,A_i)$.

3. Units that are dominating, dominated or passive are removed using Algorithm 1.

4. Let $L = [A_1,A_2,\ldots,A_n]$ be the ordering of the list of units that produces matrix $M' = arg\,min(C(M))$. The band size of the game is $\beta(M')$. The order provided by L ranks the units, with meaningful choice only existing for units within $\beta(M')$ units above and below any given unit. Units earliest in the ranking dominate units later in the ranking beyond this limit.

CASE STUDY

The StarCraft II combat simulator determines combat outcomes for 69 player identified unit variations characterized by at least 65 attributes [29]. The game is widely played both socially and professionally, and is also the subject of annual competitions to test artificial intelligence systems built to play strategically [35]. Such research aims to identify strategies as sequences of tactics within the game that would produce the most effective artificial players. These efforts have resulted in access being granted to various programming interfaces culminating in the StarCraft II API[2] [43]. This opens up opportunities for research into the properties of a complex balanced competition with diverse agent characteristics as this API allows individual units to compete against one another using the StarCraft II game engine in the absence of confounding factors of terrain, other units or player interactions. The data presented in this paper was created using Linux package version 3.16.1.

The API identifies 200 different units distributed across 4 factions: Terrain, Zerg, Protoss and Neutral. Neutral elements are predominantly modifiable elements of the terrain such as buildings and resource sources. The API was used to collect data on the outcomes of combat between pairs of units. An empty map was used to avoid any effects of terrain or obstacles. For combat involving any given pair of units, one of each is spawned at on the circumference of the largest circle fitting into the map. Positioning is random to avoid any directional bias but opponents are always positioned directly opposite one another on the circle to ensure starting separation is constant. The entire process is repeated multiple times to determine the aggregated behaviour.

When the game starts, the two units are both instructed to head towards the center of the circle using their SMART ability. This is chosen to allow maximum inherent unit behaviour. Those units able to move will head for the center, attempt to combat the other if possible, and flee if necessary. Pursuit of a fleeing unit is possible but pursuing units may break off and return to occupy the targeted position if their victim gets out of range. The simulation ends when one unit is destroyed, or after a fixed number (3000) of iterations of the game loop. Since the simulation does not run in real-time, each game typically takes less than 10 s.

Results measured include:

- the outcome of the game reported as a member of the set $\{-1,0,+1\}$ corresponding to lose, draw and win from the perspective of the first unit of the pair.
- the change of health of the units involved, the duration of the game (in game loop cycles) and the game loop cycle at which a unit last took damage. This helps to identify whether the simulation was terminated before a conclusive outcome was achieved, or whether no further interaction between the units is likely.

The aim of this experiment is to gather data and apply the analysis process described to derive insight into the strategic options available to units within StarCraft II. Hypotheses which will be tested are:

H1: Unit combat within StarCraft II is symmetrical and balanced from the perspective that each player has equal opportunity for victory from the outset of a game.

H2: All combat units within StarCraft II are meaningful choices.

H3: The ranking of units within StarCraft II is evenly distributed across the different factions.

H4: The ranking of units within StarCraft II corresponds to their availability during the game (as determined by prerequisites within the tech tree, and resource requirements).

Skew symmetry

Lack of skew symmetry can be proven by finding a unit A_i for which the relationship: $f(A_i,A_i) \neq 0$. We collected a sample of StarCraft II competitions involving a two player game with each player using the **same** unit. Data sets of at least 550 games for each of the 200 units are used in this analysis.

The majority (124) of the units result in a draw outcome for every round played and so automatically $f(A_i,A_i) = 0$ for these units. The remainder are able to win and lose against a completely identical opponent presumably due to elements of randomness being present in the simulation.

A further 22 units only have outcomes only from the set $\{-1,1\}$ (i.e. they never draw) and so all 550 samples contribute directly to the binomial test. The initial data set identifies bias in 1 unit at a 2% significance level. However given the number of units being tested a much higher standard of significance is required to discriminate against random fluctuations. This is confirmed when repeating the process with a second and third data set fails to flag the same unit. Hence we conclude that there is no evidence of lack of skew symmetry in these cases.

The remaining 54 units have situations in which usually a draw occurs, but do also manage to win or lose. The binomial test is only applied to the win/lose cases, which means that the number of samples available is substantially reduced. Despite this a number of the units shown in Table 1 indicate a significant bias almost always in favour of the first unit in the pair. Three units appear consistently when a significance threshold of 0.1% is applied even when the experiment is repeated on two additional data sets and are marked in bold in Table 1.

[2] https://github.com/Blizzard/s2client-proto

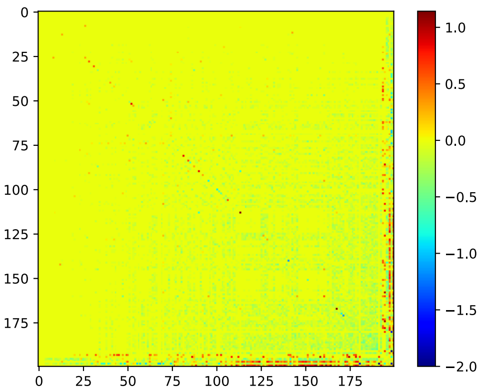

Figure 2. Bias measures for every pair of units, for all units sorted by their normalized bias measures.

This provides the evidence required to invalidate hypothesis **H1**. This is likely the result of the game simulation providing an advantage to the first unit added. This might be the result of processing the outcomes of combat actions in order, leading to the extinction of the second unit before the consequence of its reciprocal action can be recorded. These units may have limited impact on the game play since the game outcome is still a draw in the majority of cases. However it does suggest that internal engine implementation processes are affecting outcomes and that a more detailed investigation into unit symmetry is justified.

A data set involving combat between every possible pair (200×200) of units was gathered. This was repeated to ensure that every combination was represented 14 times. The bias measures as per (1) for the top 5 units with the highest values are listed in Table 2. These are small values suggesting an imbalance may be non-significant and due to random effects within a small sample. Of the 5 units listed, 3 are later categorized as significantly dominated and so the imbalance may be related to the mechanism used to ensure their early demise. In contrast the other 2 occupy the middle ranking according to degree of meaningful choice offered.

The extent to which skew symmetry is violated is shown visually in Figure 2 by visualizing the values of b_{A_i,A_j}. This reveals more systematic patterns that are inconsistent with skew symmetry. Visual inspection shows 3 regions. The units with $b_{A,B} = 0$ could be the units which draw with all others. Investigations suggest that the set of units with minor bias values are the result of limited numbers of samples, or due to units that do not offer meaningful choice (e.g. draw in most games, but spontaneously suicide in others). This bias affects interactions with many other units. The third category are the heavily biased units (Table 2) showing this behaviour consistently and severely across large numbers of other units. This provides further evidence to refute hypothesis **H1**.

Unit categorization

The process of determining meaningful choice units is an iterative one as depicted in Algorithm 1. This involves remov-

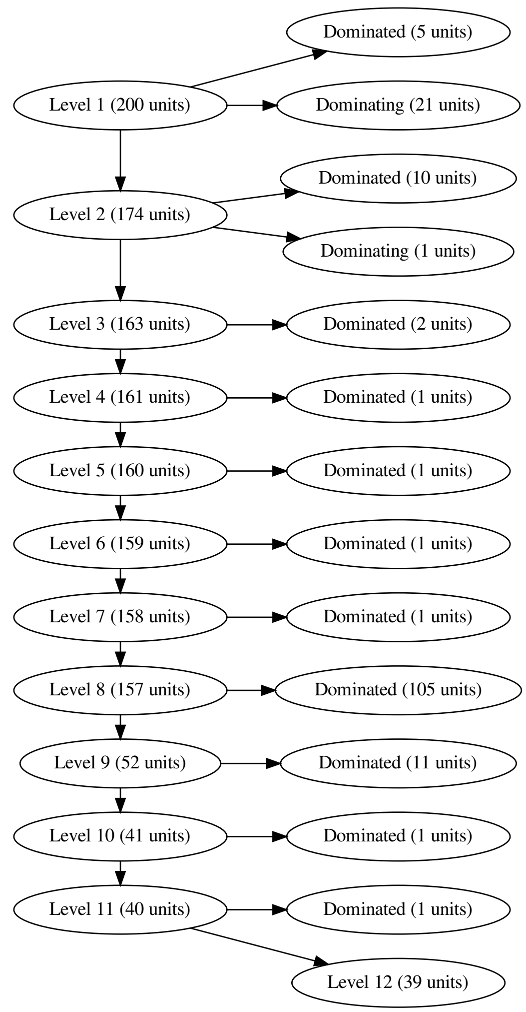

Figure 3. The pruning of different unit categories while extracting the core set of units that provide meaningful choice is represented in this tree structure.

ing all units that are identified as dominating, dominated and passive until no more units in these category remain. The remaining units are all meaningful choices. The StarCraft II data set is regularized for this process so that only win, loss or draw outcomes are present. The outcome of each combination of units is replaced with the sign of the average outcome: $f'(A_i,A_j) = sign\left(\overline{f(A_i,A_j)}\right)$, and $f'(A_i,A_i) = 0$ so that the function f' maps to $\{-1,0,+1\}$. This data requires 11 iterations to reach equilibrium and produce the meaningful choice core. The nature of the units removed during each iteration are summarized in Figure 3.

The units removed are both dominated and dominating, with no passive units in this sequence. It turns out that several StarCraft II units require particular support from other units and will die without any intervention, either immediately or once some initial resource level runs out. These are clearly

Unit	Win	Draw	Lose	Two-tailed p-value
ZERG_BANELINGCOCOON	25	524	1	0.000%
ZERG_BROODLORDCOCOON	20	529	1	0.002%
ZERG_CHANGELING	17	532	1	0.014%
ZERG_EGG	18	530	2	0.040%
ZERG_LURKERMPEGG	23	520	7	0.522%
ZERG_ULTRALISKCAVERN	13	510	27	3.848%
PROTOSS_TWILIGHTCOUNCIL	43	480	27	7.224%
PROTOSS_TEMPEST	267	56	227	7.921%

Table 1. Results of bias testing for 2-player games with both using the same unit. These cases result in win, draw or loss (from the perspective of the first unit in the pair). Several of these units indicate significant bias in favour of the first unit of the pair beyond the 1% threshold. Those that are consistent across three repetitions of the experiment are marked in bold.

Unit A_i	$bias_{A_i}$
ZERG_LURKERMPEGG	0.0286
ZERG_LARVA	0.0268
PROTOSS_DISRUPTOR	0.0225
ZERG_SPORECRAWLER	0.0178
ZERG_SPINECRAWLER	0.0156

Table 2. Bias measure for the 5 units with the largest values, corresponding to the right hand columns and bottom rows in Figure 2.

Algorithm 1 The algorithm for categorization and pruning of passive, dominated and dominating units. The priority list when provided aggressively removes any units of the higher priority categories first ensuring that the remaining units classified into the low priority category are unambiguously in that category.

```
// classify units into categories
function classify (unitList)
  p = d = D = C = []
  for u in unitList
  loseToSome = (u is 1-balanced)
  winOverSome = (u is 1+balanced)
  if loseToSome and winOverSome then
    C.append (u) // meaningful choice core
  if loseToSome and not winOverSome then
    d.append (u) // dominated
  if not loseToSome and winOverSome then
     D.append (u) // dominating
  if not loseToSome and not winOverSome then
    p.append (u) // passive
  return (p, d, D, C)

// prune list to return meaningful choice core.
function prune (unitList, havePriority, priorityList)
  (p, d, D, C) = identifyGrouping (unitList)
  while p or d or D are not empty
    if havePriority
      g = highest priority and non empty list in {p, d, D}
        according to priorityList
      unitList.remove (g)
    else
      unitList.remove (p)
      unitList.remove (d)
      unitList.remove (D)
  return unitList
```

dominated and cause other passive units such as buildings to be flagged as dominating these units. The first 7 iterations remove units that suicide after increasing intervals. The bulk of the scenery is removed at iteration 8 when the passive structures that have proven indestructible are removed, followed by the helpless buildings that can be destroyed by other units. The remaining iterations similarly progressively prune units that are only capable of beating other easily beaten (dominated) units. The iteration sequence provides a ranking within units of the same category and the number of steps involved indicates a substantial hierarchy exists when units interact in this way.

Units may vary between the passive and dominated, and passive and dominating categories depending on the order in which units are removed but the same set of meaningful choices will be identified. A priority removal option in Algorithm 1 allows each iteration to remove only one of the passive, dominated and dominating categories per iteration in that order. Prioritizing the different categories provides a way of identifying minimal sets of units that are clearly only in the lowest priority category. When using prioritized removal removing passive or dominated first, the key dominating units are the TERRAN_BATTLECRUISER and TERRAN_KD8CHARGE, followed by the PROTOSS_MOTHERSHIP, with units classified as dominant using other orderings being marked as passive in this case. The dominated units remaining after first removing the passive and dominating units include the burrowed forms of many units as the weakest (most dominated) and the ZERG_BROODLING as the most powerful, but still dominated unit.

The process results in only 39 of the original 200 units that can be considered as meaningful choices in a one-on-one unit combat scenario. The units are shown in Figure 4. This outcome does also impact on hypothesis **H2**, with much fewer units than expected remaining as meaningful choices. Some of the static combat units will be passive or dominated due to the nature of the specific combat scenario used to generate this data set as these will be unable to move into combat range, or be vulnerable to units with greater range. Units representing an intermediate stage of another unit may also be helpless for a period, or lack critical support infrastructure. Such units only become relevant in group settings and become meaningful choices in such cases. Analysis of the dependencies required to fully enable a particular strategy is a direction to be pursued in further developing this work.

A measure of choice

The gradient descent algorithm is used to perform the optimization process to identify configurations with minimal band size. During each iteration the degree of change in the fitness function $C(M)$ is calculated for each pair of unit swaps in the ordered list, and the greatest gradient is selected. Randomization is used when local minima are encountered. For any given ordering, the band size is only calculated for the upper triangular portion of the matrix, despite previous results indicating that the game function does not produce a perfectly skew symmetric matrix.

An ordering and properties of the resulting StarCraft II balanced unit set are shown in Figure 4. The bandsize for this set of 39 units is 12 units. Thus for any unit, it is vulnerable to at most 12 other units, and these are the 12 that immediately follow the unit. Any unit more than 12 units further down the list will be defeated. This unit can only effectively challenge units 12 places prior to them in the list. The bandsize is an upper bound on the zone of potential balanced competition. Many of these units have smaller zones of non-trivial behaviour. Conversely, those units that achieve the maximum bandsize can be identified (where the red and light blue regions coincide). These are units that are most vulnerable to "significantly inferior" units, as determined by this ranking.

Some units can be exchanged without affecting the objective value (dark blue markers). This suggests the ordering of these units is not critical. The order of the list is not related to the number of units that the unit wins against. One of the weakest units (Zergling) consistently ends up much higher up the ranking than other units with similar numbers of wins.

The ordering of the list shows some separation of the factions involved, in contradiction to hypothesis **H3**. Many Terran units receive the lowest ranking, with heavy Protoss units prevalent in leading positions. Zerg units span the midrange. This could be a strategy to help distinguish the various factions. The Protoss faction is popularly viewed as a race with a tendency towards more expensive and powerful units. The surprising result is the lower ranking of Terran units which are expensive relative to similar Zerg units. This might explain why professional players tend to avoid playing the Terrain race [46] with statistics indicating a roughly 25:35:35 breakdown between Terran:Zerg:Protoss among grand-master players.

The ranking has the potential to inform the order in which units are accessed. All players should ideally start with units ranked within a band size of each other to ensure fairness. Access to higher order units (the technology tree) would need to ensure that the advances in ranking should be achieved in comparable time and cost increments. The ranking can be compared with the technology trees for StarCraft II[3] to assess the extent to which the analysis matches the intentions of the game designer. Each faction has a separate tree so comparisons are only valid within a faction, and are confined to units in both the meaningful choice core and the technology tree. For Zerg and Protoss, the ranking corresponds well to the progression in the technology tree, with units in the same tree rank being well within the band size from one another, although the ZERG_RAVAGER is more highly ranked than the technology tree would indicate. The Terran side reveals units out of order, such as the lowest ranked TERRAN_HELLION which requires more pre-requisites than other higher ranked units in the technology tree. Many Terran units are similarly ranked suggesting that the initial technology tree is intended to facilitate diversity rather than specialist progression. The ranking achieved from single unit combat outcomes shows surprising levels of support for hypothesis **H4**, particularly given that the combat scenario tested ignores the resource costs for each unit.

This analysis is confined to single unit-to-unit competition. Investigating strategies where one side grows number of low ranked units while others work on fewer numbers of higher ranked units are planned for future work into effects of cooperation between units.

[3]http://us.battle.net/sc2/en/game/race/terran/techtree/lotv, http://us.battle.net/sc2/en/game/race/zerg/techtree/lotv and http://us.battle.net/sc2/en/game/race/protoss/techtree/lotv

DISCUSSION

The process described focuses on one key mechanic of the trading card game; that being direct competition between two units (or cards). This mechanic applies across multiple game genres and is demonstrated with respect to combat between two units as found in StarCraft II. The process applies equally to other games once interactions between pairs of units can be reduced to an outcome matrix. Immediate benefits achieved include the ability to detect bias in the engine implementation, identify over- or under-powered units, and to rank units in ways that could then be used to determine resource cost or prerequisites to access through a technology tree. This approach can be used to analyze existing game structures, or to iteratively validate and refine the design of new games as they are being built.

We are in the process of extending these ideas to other game design problems. Game design paradigms are envisaged that start with the canonical ranking and reverse engineer unit properties and simulation engine behaviour from these. Combinations of cards and alliances of units, typical of more complex trading card game mechanics, can be represented by power sets of $\mathscr{H}$, where the individual elements are groups of units and the analysis repeated with these. Since the complexity increases significantly in this case, the systematic and automated analysis methodology presented provides a support tool to enhance the designer's intuition.

CONCLUSIONS

The results of the case study provide insight into the strategic benefits of choosing particular units in StarCraft II. This alone is not sufficient to play the complete game unaided but provides a foundation for assessing the impact of unit design on the compound problems involving coalitions of units deployed over varied terrain and subject to resource constraints.

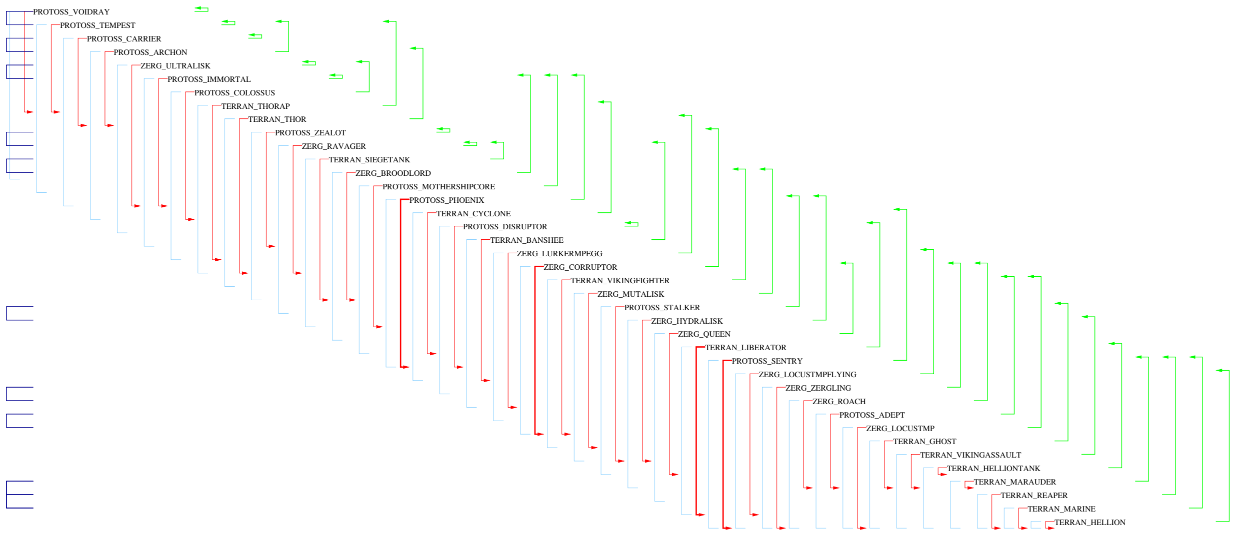

Figure 4. Dominance relationships for balanced unit set in StarCraft II. The light blue lines show the number of units with the band size for the game, with the red lines indicating the actual level of meaningful choice for units lower in the rankings. Green lines indicate how much higher in the rankings any unit can meaningfully compete. Dark blue lines indicate which units can be exchanged without affecting the outcome, and which are thus effectively equivalent.

The approach is validated as a way of reasoning about the analysis of other competitions that involve the choice of two competing units common to many other games with the ranking and measures presented able to yield further insight into a complex and already well studied context.

A number of dominated and dominating units are identified even in the established and balanced units of StarCraft II. It is acknowledged that the scenario tested is only one of the ways in which the units could be deployed in the full game. A core set of units offering meaningful choice have been identified and this offers the most direct insight to the game designer from this particular experiment. The band size of these units has been measured as 12 which is much less than the total number of 39 units in the core. The game thus has to be played as a series of stages, where sets of units available to each player at any stage need to overlap within this band size. Player must also be able to advance their units according to the ranking identified. The order of units in this ranking does somewhat correspond to that expected from characteristics of the various factions: with Protoss having more powerful (and expensive) units. A surprising reversal is evident where some cheaper Zerg units are ranked highly compared to more expensive Terran units which could explain why the Terran faction is underrepresented in professional games.

The lack of skew symmetry in StarCraft II is surprising and often favours the first player, all other factors being equal. This suggests that the engine implementation may not be completely consistent with the intended game rules. The analysis identifies the units which are responsible. Commonalities between these units may offer insight into which elements of the game engine are responsible for this bias.

The analysis process requires no prior insight into the nature of any particular game, allowing it to be applied to any scenario that can be represented using outcomes of binary competitions. The ranking of units discovered corresponds closely to that intended by the game designer as embodied in the technology tree suggesting that these results closely correspond with the intuitive understanding of the role of each unit. The analysis can be seen as characterizing units as offering meaningful choice, but also addressing the problem of comparing strategic decisions involving selection from a set (trading card mechanic). Such outcomes can be achieved from simulations of existing systems, but equally game design can be facilitated by defining the desired strategic opportunities in the form of the matrix representing optimal ranking and the game mechanics reversed engineered from this.

Games can now be represented and defined in terms of their strategic structure, with an examples of a visualization providing insight into game characteristics shown in Figure 4. In complex scenarios such as StarCraft II players may not be directly aware of the extent to which their strategic choices are meaningful but community theories do develop over time. Game designers are now able to produce evidence during the design and development phase of a game that the fundamental unit characteristics are balanced to the extent desired and that game progression is consistent with capabilities of each unit.

REFERENCES

1. G. Andrade, G. Ramalho, H. Santana, and V. Corruble. 2005. Extending Reinforcement Learning to Provide Dynamic Game Balancing. In *Nineteenth International Joint Conference on Artificial Intelligence*. Edinburgh, Scotland, 7–12.

2. Dennis Ang and Alex Mitchell. 2017. Comparing Effects of Dynamic Difficulty Adjustment Systems on Video Game Experience. In *Proceedings of the Annual Symposium on Computer-Human Interaction in Play (CHI PLAY '17)*. ACM, New York, NY, USA, 317–327.

3. Alexander Baldwin, Daniel Johnson, and Peta A. Wyeth. 2014. The Effect of Multiplayer Dynamic Difficulty Adjustment on the Player Experience of Video Games. In *CHI '14 Extended Abstracts on Human Factors in Computing Systems (CHI EA '14)*. ACM, New York, NY, USA, 1489–1494.
4. Shaun Bangay and Owen Makin. 2013. Modelling Attribute Dependencies in Single Unit Game Combat Settings. In *Proceedings of the 2013 IEEE International Games Innovation Conference*. 20–26.
5. Nicola Baumann, Christoph Lürig, and Stefan Engeser. 2016. Flow and enjoyment beyond skill-demand balance: The role of game pacing curves and personality. *Motivation and Emotion* 40, 4 (01 Aug 2016), 507–519.
6. Simona Botti and Ann L. McGill. 2006. When Choosing Is Not Deciding: The Effect of Perceived Responsibility on Satisfaction. *Journal of Consumer Research* 33, 2 (2006), 211–219.
7. D. Cartwright and F. Harary. 1956. Structural balance: a generalization of Heider's theory. *Psychological Review* 63, 5 (1956), 277–293.
8. H. Chen, Y. Mori, and I. Matsuba. 2012. Solving the Balance Problem of On-Line Role-Playing Games Using Evolutionary Algorithms. *Journal of Software Engineering and Applications* 5, 8 (2012), 574–582.
9. James A. Davis. 1967. Clustering and Structural Balance in Graphs. *Human Relations* 20, 2 (1967), 181–187.
10. Sebastian Deterding. 2016. Contextual Autonomy Support in Video Game Play: A Grounded Theory. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, New York, NY, USA, 3931–3943.
11. Joris Dormans. 2011. Simulating Mechanics to Study Emergence in Games. In *AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*.
12. T. Fayard. 2007. Using a Planner to Balance Real Time Strategy Video Game. In *Workshop on Planning in Games, ICAPS 2007*.
13. Kevin Gold. 2010. Why Games Must Be More Than Fair: Random Walks, Long Leads, and Tools to Encourage Close Games. (2010).
14. C. M. Grinstead and J. L. Snell. 1997. *Introduction to Probability*. American Mathematical Society.
15. Christian Guckelsberger, Christoph Salge, Jeremy Gow, and Paul Cairns. 2017. Predicting Player Experience Without the Player.: An Exploratory Study. In *Proceedings of the Annual Symposium on Computer-Human Interaction in Play (CHI PLAY '17)*. ACM, New York, NY, USA, 305–315.
16. Ben Hanson. 2003. *Massively Multiplayer Game Development*. Charles River Media, Inc., Chapter Game Balance for Massively Multiplayer Games.
17. Frank Harary. 1953. On the notion of balance of a signed graph. *Michigan Math. J.* 2, 2 (1953), 143–146.
18. John Harris, Mark Hancock, and Stacey D. Scott. 2016. Leveraging Asymmetries in Multiplayer Games: Investigating Design Elements of Interdependent Play. In *Proceedings of the 2016 Annual Symposium on Computer-Human Interaction in Play (CHI PLAY '16)*. ACM, New York, NY, USA, 350–361.
19. Fritz Heider. 1946. Attitudes and Cognitive Organization. *The Journal of Psychology* 21, 1 (1946), 107–112.
20. V. Hom and J. Marks. 2007. Automatic Design of Balanced Board Games. In *The Third AAI Conference on Artificial Intelligence and Interactive Digital Entertainment*.
21. Britton Horn, Amy K. Hoover, Jackie Barnes, Yetunde Folajimi, Gillian Smith, and Casper Harteveld. 2016. Opening the Black Box of Play: Strategy Analysis of an Educational Game. In *Proceedings of the 2016 Annual Symposium on Computer-Human Interaction in Play (CHI PLAY '16)*. ACM, New York, NY, USA, 142–153.
22. A. Jaffe, A. Miller, E. Anderson, Y-E Liu, A. Karlin, and Z. Popovic. 2012. Evaluating Competitive Game Balance with Restricted Play. In *The Eigth AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*. 26–31.
23. Yubo Kou, Xinning Gui, and Yong Ming Kow. 2016. Ranking Practices and Distinction in League of Legends. In *Proceedings of the 2016 Annual Symposium on Computer-Human Interaction in Play (CHI PLAY '16)*. ACM, New York, NY, USA, 4–9.
24. Krzysztof Kulakowski, Przemyslaw Gawronski, and Piotr Gronek. 2005. The Heider Balance: A continuous approach. *International Journal of Modern Physics C (IJMPC)* 16, 05 (2005), 707–716.
25. Jérôme Kunegis. 2014. Applications of Structural Balance in Signed Social Networks. *CoRR* abs/1402.6865 (2014).
26. R. Leigh, J. Schonfeld, and S. J. Louis. 2008. Using coevolution to understand and validate game balance in continuous games. In *Proceedings of the 10th annual conference on Genetic and evolutionary computation*. 1563–1570.
27. Jure Leskovec, Daniel Huttenlocher, and Jon Kleinberg. 2010. Signed Networks in Social Media. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '10)*. ACM, New York, NY, USA, 1361–1370.
28. Owen Makin and Shaun Bangay. 2013. Establishing competitive domination cycles for peer-to-peer game combat. In *Games Innovation Conference (IGIC), 2013 IEEE International*. 152–156.

29. Owen Makin and Shaun Bangay. 2017. Orthogonal Analysis of StarCraft II for Game Balance. In *Proceedings of the Australasian Computer Science Week Multiconference (ACSW '17)*. ACM, New York, NY, USA, Article 30, 4 pages.

30. Morgan McGuire and Odest Chadwicke Jenkins. 2008. *Creating Games: Mechanics, Content, and Technology*. Taylor & Francis.

31. F. Mueller, F. Vetere, M. Gibbs, D. Edge, S. Agamanolis, J. Sheridan, and J. Heer. 2012. Balancing Exertion Experiences. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*. Acm, New York, NY, USA, 1853–1862. DOI:http://dx.doi.org/10.1145/2207676.2208322

32. M. J. Nelson and M. Mateas. 2009. A requirements analysis for videogame design support tools. In *Proceedings of the Fourth International Conference on the Foundations of Digital Games*. 137–144.

33. Jeannie Novak. 2011. *Game Development Essentials: An Introduction*. Cengage Learning.

34. John M. Olsen. 2003. *Massively Multiplayer Game Development*. Charles River Media, Inc., Chapter Game Balance and AI Using Payoff Matrices.

35. S. Ontanon, G. Synnaeve, A. Uriarte, F. Richoux, D. Churchill, and M. Preuss. 2013. A Survey of Real-Time Strategy Game AI Research and Competition in StarCraft. *Computational Intelligence and AI in Games, IEEE Transactions on* 5, 4 (2013), 293–311. DOI: http://dx.doi.org/10.1109/Tciaig.2013.2286295

36. Andrew Rollings and Ernest Adams. 2003. *Andrew Rollings and Ernest Adams on Game Design*. New Riders.

37. A. Rollings and D. Morris. 2004. *Game architecture and design*. New Riders.

38. Katie Salen and Eric Zimmerman. 2004. *Rules of Play: Game Design Fundamentals*. MIT Press.

39. Jesse Schell. 2008. *The Art of Game Design: A Book of Lenses*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

40. Paul Schuytema. 2007. *Game Design: A Practical Approach*. Charles River Media.

41. Kristin Siu and Mark O. Riedl. 2016. Reward Systems in Human Computation Games. In *Proceedings of the 2016 Annual Symposium on Computer-Human Interaction in Play (CHI PLAY '16)*. ACM, New York, NY, USA, 266–275.

42. Harvey Smith. 2003. Orthogonal Unit Differentiation. In *Game Developers Conference*.

43. Oriol Vinyals, Timo Ewalds, Sergey Bartunov, Petko Georgiev, Alexander Sasha Vezhnevets, Michelle Yeo, Alireza Makhzani, Heinrich Küttler, John Agapiou, Julian Schrittwieser, John Quan, Stephen Gaffney, Stig Petersen, Karen Simonyan, Tom Schaul, Hado van Hasselt, David Silver, Timothy P. Lillicrap, Kevin Calderone, Paul Keet, Anthony Brunasso, David Lawrence, Anders Ekermo, Jacob Repp, and Rodney Tsing. 2017. StarCraft II: A New Challenge for Reinforcement Learning. *CoRR* abs/1708.04782 (2017). http://arxiv.org/abs/1708.04782

44. K. Werbach and D. Hunter. 2012. *For the Win: How Game Thinking Can Revolutionize Your Business*. Wharton Digital Press. https://books.google.com.au/books?id=abg0SnK3XdMC

45. Wen Xia and B. Anand. 2016. Game balancing with ecosystem mechanism. In *2016 International Conference on Data Mining and Advanced Computing (SAPIENCE)*. 317–324.

46. S. Xiong and H. Iida. 2014. Attractiveness of real time strategy games. In *The 2014 2nd International Conference on Systems and Informatics (ICSAI 2014)*. 271–276.