Generalized Quantum Tic-Tac-Toe

Ananya Kumar[1], Ang Yan Sheng[1], and Chai Ming Huang*[1]

[1]NUS High School, 20 Clementi Avenue 1, Singapore 129953

*Mentor/Teacher-Advisor

Abstract

Quantum tic-tac-toe (QT3) elegantly extends the popular game of tic-tac-toe, inspired loosely by quantum physics principles. Yet, despite the interesting and challenging gameplay, not much research has been done on it. Hence in this paper we explore the game in terms of extension, analysis and solution. We first conjecture and prove a graph theory theorem that enables a generalization of the game (GQT3). We then show that our generalized game can always be successfully completed in a finite number of moves. Then, we begin game analysis. Firstly, we investigate the game tree size; we find that QT3 has more than 18 trillion possible games, substantially higher than tic-tac-toe‘s 300 thousand. Next, we explore GQT3 games where players play their moves randomly; for a 3-by-3 board the expected score is a player 1 win by 0.452 points. Thereafter, we examine the Nash Equilibrium of the game; the result if two perfect players play the game against each other. We find that in this scenario, the first player will win by 0.5 points. To make the game fairer, we suggest minor variations which make the Nash Equilibrium a draw. Note that standard methods to analyze most of these would take at least a year, but our programs take under an hour due to various optimizations. Finally, we extend our programs into an artificial intelligence that is a perfect solution to the game.

1 Introduction

Alan Goff extended the popular game of tic-tac-toe based on the quantum physics principle of superposition, resulting in a game that is substantially more interesting and challenging: Quantum Tic-Tac-Toe (QT3). The rules are explained well with concrete examples in [1, 2], and are summarized in Appendix A. Understanding the game rules is essential for understanding our research; hence, we first refer the reader to Appendix A.

2 Theorem for Generalization

The rules for QT3 define entanglement using the notion of a cycle, which is difficult to generalize. Instead, we define entanglement to occur when there are some x pairs of pieces which are entirely contained within x squares. In this section, we prove the equivalence of these two definitions.

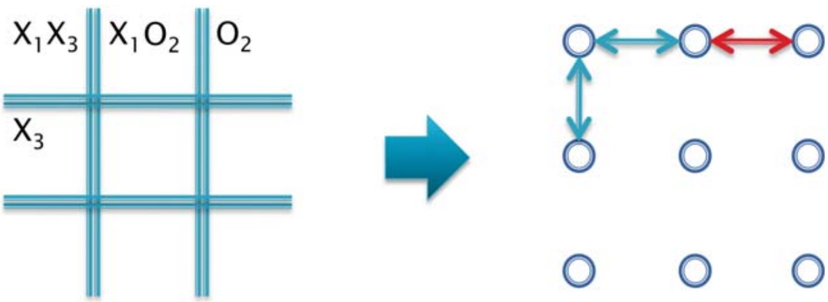

Figure 1: Example transformation of a QT3 game into a graph.

Consider a multigraph G with vertices corresponding to squares of the board. Draw an edge between 2 vertices for each pair of quantum moves placed on the 2 corresponding squares (Figure 1). By the game rules, the two endpoints of any edge in G are distinct, but a pair of vertices may be connected by more than one edge. By the rules of QT3 whenever a cycle appears on the board collapse will take place to remove it, thus without loss of generality we consider a graph that does not initially contain a cycle. With that, the equivalence of the two definitions can be stated as follows, with Goff's original definition as condition 1 and our definition as condition 3.

Theorem. *For any graph G which does not initially contain a cycle and in which edges are added one by one, the following conditions hold* for the first time *simultaneously:*

1. *G contains a cycle;*
2. *G contains exactly one cycle;*
3. *G contains a connected component with x edges and x vertices.*

Proof. We will use the fact that all trees (connected graphs with no cycles) have exactly one less edge than vertices [3].

1 $\Longrightarrow$ 2: If the last added edge AB produces two cycles $AC_1C_2 \dots C_iBA$, $AD_1D_2 \dots D_jBA$ then there must have been a cycle $AC_1C_2 \dots C_iBD_j \dots D_2D_1A$ before this edge was added, which is a contradiction. Hence G contains exactly one cycle.

2 $\Longrightarrow$ 3: Consider the maximal connected component $H \subseteq G$ that the cycle is in, which has v vertices and e edges. Then removing any edge of the cycle turns H into H', which has no cycles yet is still connected. Hence H' is a tree with v vertices and $e - 1$ edges, so $e = v$.

3 $\Longrightarrow$ 1: Let H be the stated connected component. If H contains no cycles then H is a tree, contradiction. Hence H (and therefore G) contains a cycle. □

This theorem is important and allows us to see that entanglement and collapse occur at the last possible moment, in the sense that a board with $x + 1$ pairs of pieces in x squares does not have a valid collapse by the Pigeonhole Principle. Furthermore, this definition of entanglement can be very easily generalized, as presented in the next section.

3 Generalized Rules of the Game

After formulating and proving the important theorem in the previous section, we now present the rules of our *Generalized QT3* (GQT3).

Two players first agree on 2 integers $m, n > 0$. The game is played on an $m \times m$ square board, where each square is either quantum or classical, with all squares initially quantum. Each turn, a player places n copies of the turn number (collectively called a *series*) on n quantum squares of his choice. (If there are less than n quantum squares, classical tic-tac-toe is played.) The maximal group of x squares that completely contains x series is considered entangled.

When a player causes entanglement, the other player collapses the board by fixing each of the x series part of the entanglement to one of the n squares that it occupies, such that each of the x squares is assigned to exactly one series. Then the series become classical and replace the quantum pieces in their respective squares, and all of the x squares become classical.

The game ends when all squares are classical and hence occupied by exactly one number. Points are then awarded to players with lines (rows, columns or diagonals) filled only with their numbers. The i^{th} line obtained (ordered by the maximal number in the line) is awarded $\frac{1}{i}$ points.

Note that GQT3 with $m = 3, n = 1$ is normal Tic-Tac-Toe, and $m = 3, n = 2$ is equivalent to QT3, as shown in the previous section.

4 Proof of Consistency

When entanglement occurs, it is not obvious whether a collapse is always possible. In this section, we prove the consistency of the game by showing that there must always exist *many* collapses.

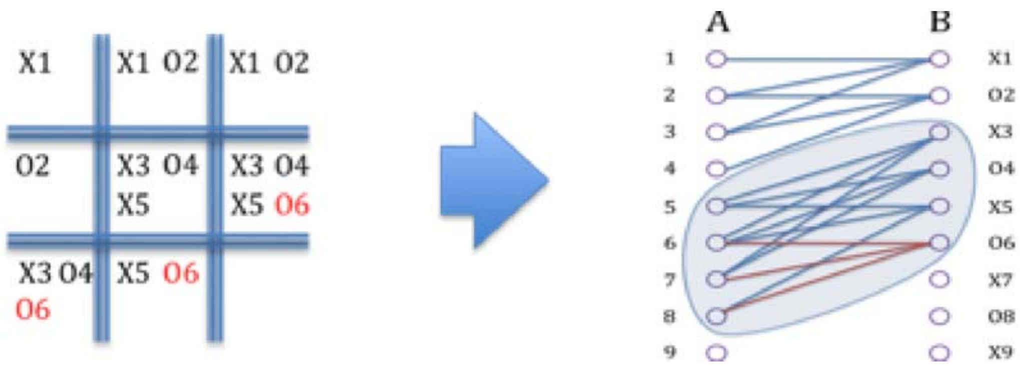

Figure 2: An example bipartite graph G, for $m = 3, n = 3$. Note that collapse corresponds to choosing a perfect matching on G'.

Represent the GQT3 board as a bipartite graph G with bipartition A and B, with A representing the squares of the board and B representing the series of pieces (Figure 2). When entanglement occurs, consider the maximal $E \in B$ such that $|n(E)| \leq |E|$. Then the graph G' induced on the vertices in $E \cup n(E)$ represents the entangled portion of the game board, and satisfies the following:

1. If for any nonempty $S \subseteq E$, we have $|n(S)| < |S|$, then entanglement would have occurred earlier in $S \cup n(S)$, contradiction. Thus $|n(S)| \geq |S|$. In particular, $|n(E)| = |E|$.

2. If this is the first entanglement of the game then $\deg(v) = n$ for all vertices $v \in E$, since each quantum series contains exactly n pieces.

   Else, if there exists $v \in E$ such that $\deg(v) = 1$ then consider the board at the previous entanglement, say on $E_0 \cup n(E_0)$. Note that $|n(E_0 + v)| = |n(E_0) \cup n(v)| = |E_0 + v|$, so the previous entanglement should have been on $(E_0 + v) \cup n(E_0 + v)$ instead of $E_0 \cup n(E_0)$ by maximality, contradiction. Hence $\deg(v) \geq 2$ for all $v \in E$.

Theorem. *Let $k_0 \in \mathbb{N}$. A bipartite graph G' with bipartition A, B is given such that:*

1. *$|A| = |B|$;*

2. *For all $S \subseteq B$, we have $|n(S)| \geq |S|$; and*

3. *For all vertices $v \in B$, $deg(v) \geq k_0$.*

Then G' has at least $k_0!$ perfect matchings.

Proof. We proceed by induction on k_0. For $k_0 = 1$ the theorem holds by Hall's Marriage Lemma.

Assume induction hypothesis for some $k_0 = k \geq 1$, and let graph G satisfy the premises of the induction hypothesis for $k_0 = k+1$. We then choose the smallest $S_0 \subseteq B$ such that $|n(S_0)| = |S_0|$.

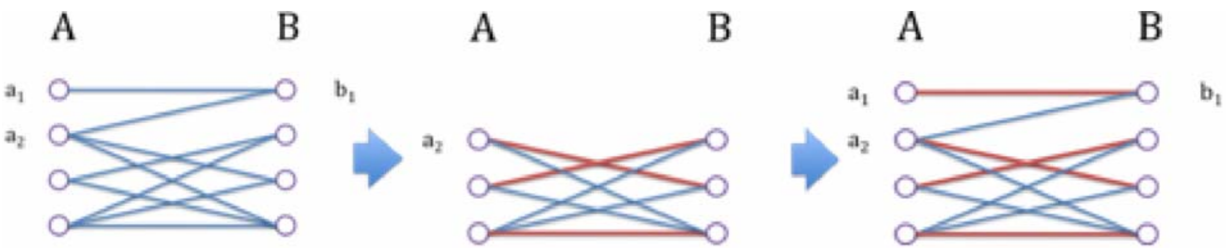

Figure 3: Left: Original graph. Center: After edge removal. Right: After adding back the edges.

This subset exists because $|n(B)| = |A| = |B|$. For any nonempty $S \subset S_0$, $|n(S)| > |S|$ by the minimality condition on S_0.

Select an arbitrary vertex b_1 from S_0 and $a_1, a_2, \ldots, a_{k+1} \in n(b_1) \subseteq n(S_0)$.

Now remove b_1 and a_1, along with all edges incident to them. Now for all $S \subset S_0$, $|n(S)| \geq |S|$ since only one vertex was removed from $n(S_0)$, and hence at most one vertex was removed from $n(S)$. Similarly, for all $b \in S_0$, $\deg(b) \geq k$. Hence the inductive hypothesis states that there are at least $k!$ perfect matchings on $S_0 \cup n(S_0)$.

Combined with edge (a_1, b_1), this means that there are at least $k!$ perfect matchings on $S \cup S_0$ with b_1 matched to a_1. Since we can repeat the argument for any edge from (a_1, b_1) to (a_{k+1}, b_1), there are at least $(k+1) \cdot k! = (k+1)!$ perfect matchings on $S \cup S_0$.

Let $X = B \setminus S_0$, and $T = A \setminus n(S_0)$. For any $S \subseteq X$, $n(S \cup S_0) \subseteq (n(S) \cap T) \cup n(S_0)$. Hence if $|n(S) \cap T| < |S|$ then

$$|n(S \cup S_0)| \leq |n(S) \cap T| + |n(S_0)| < |S| + |S_0| = |S \cup S_0|,$$

where the last equality holds because $S \subseteq X$ and S_0 are disjoint. This contradicts condition 2 of the induction hypothesis.

Therefore $|n(S) \cup T| \geq |S|$ for all $S \subseteq X$, so there is a perfect matching on $X \cup T$ by Hall's Marriage Lemma. Hence there are at least $(k+1)!$ perfect matchings on the original graph G, and induction is complete. □

Hence by applying the theorem on G', there are always ≥ 2 collapses possible. Further, at the first entanglement there are always $\geq n!$ collapses.

5 Entanglement Detection

The discussion in the previous section shows that a large number of collapses are possible when entanglement occurs on a GQT3 board. However, nothing has been mentioned about *how* to determine when an entanglement has occurred! In this section, we describe an efficient polynomial-time algorithm to determine for any GQT3 game whether there is an entanglement on the board.

$n(S)$ denotes the set of vertices adjacent to at least one vertex of S.

For any GQT3 game board, construct the bipartite graph G as in the previous section, with bipartition A (representing squares) and B (representing series). By a similar argument in the previous section, we deduce that $|n(S)| \geq |S|$ for all subsets $S \subseteq B$. This implies by Hall's Marriage Lemma that there exists a matching in G with size $|B|$, which is clearly maximal.

Choose any vertex $v \in B$, and suppose v is adjacent to vertices $a_1, \ldots, a_k$ in A. We add a new vertex v' to the graph, and add new edges $v'a_1, \ldots, v'a_k$, calling the resulting graph G_v. In effect, v' is a *duplicate* of v. Also, define $S_v = S + v'$ for any $S \subseteq B$.

We find the size of the maximal matching of G_v, by considering the following two cases.

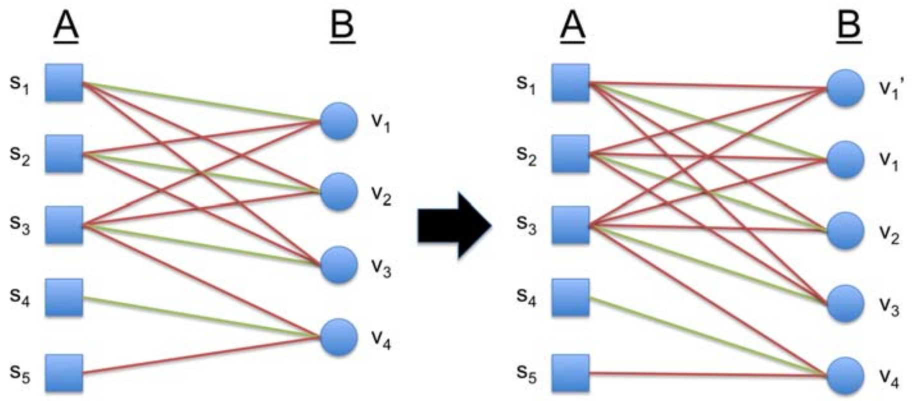

Figure 4: Example G (left) and G_v (right), with $v = v_1$. Maximal matchings are highlighted in green. Then $E = \{v_1, v_2, v_3\}$ satisfies the conditions in Case 1.

Case 1: *There exists a subset $E \subseteq B$, with $v \in E$, such that* $|n(E)| = |E|$, i.e. there is an entanglement on the board containing v (Fig. 1). Then by the above condition there must exist a perfect matching on $E \cup n(E)$, by Hall's Marriage Lemma.

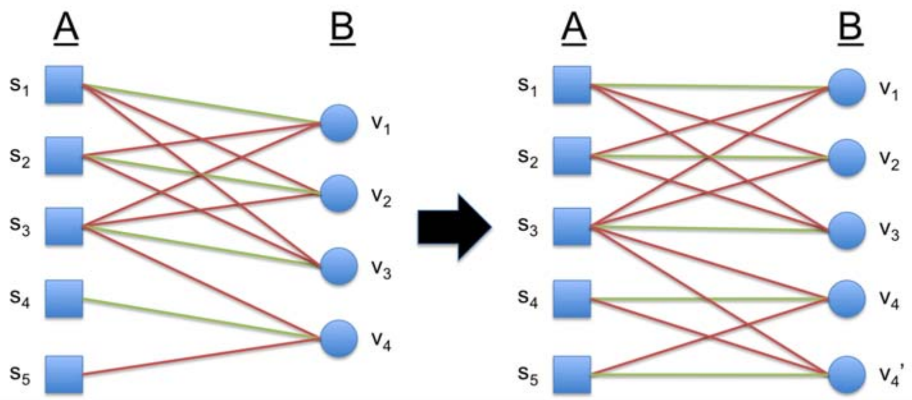

Figure 5: Example G (left) and G_v (right), with $v = v_4$. Maximal matchings are highlighted in green.

Consider the maximal matching on G_v. This matching has size at least $|B|$, since G is a subgraph of G_v. Moreover, note that $|n_{G_v}(E_v)| = |n_G(E)| = |E| = |E_v| - 1$. Hence there does not exist a matching in G_v of size $|B_v| = |B| + 1$. Thus the maximal matching on G_v contains exactly $|B|$ edges.

Case 2: *For all subsets $E \subseteq B$, with $v \in E$, we have $|n(E)| \geq |E| + 1$,* i.e. there is no entanglement on the board containing v (Fig. 2). Consider some $S \subseteq B_v$.

1. If at most one of v, v' is in S then note that we may swap v and v'; hence we may assume without loss of generality that $v' \notin S$, so $S \subseteq B$. Then $|n_{G_v}(S)| = |n_G(S)| \geq |S| + 1$.

2. If both v, v' are in S then $|n_{G_v}(S)| = |n_G(S - v')| \geq |S - v'| + 1 = |S|$.

Thus the maximal matching on G_v contains $|B_v| = |B| + 1$ edges by Hall's Marriage Lemma. □

The above discussion implies that *the size of the maximal matching in G_v is*

$$\begin{cases} |B| & \text{if } v \text{ is contained in an entanglement;} \\ |B| + 1 & \text{otherwise.} \end{cases}$$

This suggests a quick algorithm to determine the existence of entanglement in G: for all $v \in B$, compute the size of the maximal matching for G_v using well-known polynomial-time algorithms (e.g. the Hopcroft-Karp algorithm). The vertices v for which this size is $|B|$ constitute exactly the set in which entanglement occurs. In particular, if all the computed sizes are equal to $|B| + 1$ then entanglement has not yet occurred on the board.

6 Game Tree Size

The game tree size is the total number of possible games that can be played, or the number of leaves in the game tree. As we are not interested in determining a winner when counting the game tree size, the number of squares in the board does not need to be a perfect square and we define s to be the number of squares ($s \in \mathbb{N}$). It turns out that computing the game tree size through a brute force search in the game tree would take approximately a year and is hence not feasible.

The crux move in solving this conundrum is realizing that the number of move combinations that result in any particular endgame is the same. This is because we can simply rearrange the squares to transform one endgame and its move combinations into another. Thus we only need to count the number of ways to achieve a certain specified endgame and then multiply that value by $s!$. We choose to compute the number of moves to achieve an endgame where the classical piece i is on the i^{th} square of the board (Figure 4). To do this, on the i^{th} turn, we affix one piece of the i^{th} series on the i^{th} square and perform an exhaustive search for all possibilities for the other $n-1$ pieces. Whenever collapse occurs, we prune out possibilities that cannot end in the required end game state. For example, we ignore cases where the piece 3 (X_3) lands up on square 2, since we want it to land on square 3. Our Java implementation of this runs in about 15 seconds for $n = 2$,

$s = 9$ and yields an enormous value of about 18.5 trillion (18,539,269,580,160). As a comparison, this is roughly 73 million times the game tree size of classical tic-tac-toe, which is 255,168. This provides mathematical evidence for our intuition that GQT3 is a very challenging game that is difficult to play and analyze.

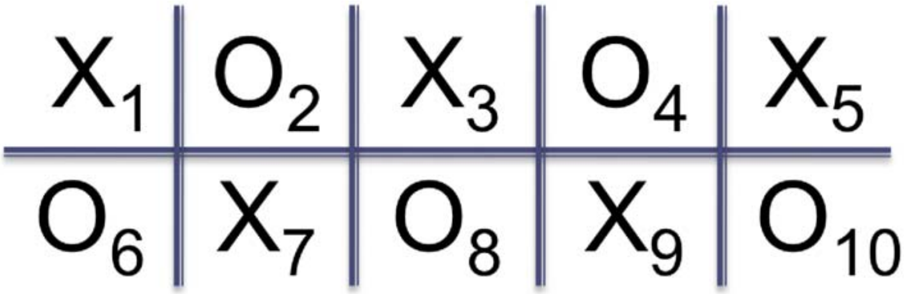

Figure 6: The end game state we use to compute game tree size

We can also estimate bounds for the game tree size. $\binom{s}{n}^s$ is an upper bound since at most s squares can be quantum at any turn. $\binom{s}{n}\binom{s-1}{n}\binom{s-2}{n}\cdots\binom{n}{n}$ is a lower bound since at least $s-t$ squares will be left quantum after t moves. For the case $n = 2$, we can do even better using our symmetry idea mentioned above. Using a similar method we obtain a lower bound of $s! \times (s-1)!$ and upper bound of $s! \times (s-1)^s$. Table 1 shows the upper bound, correct value and lower bound of the game tree size for $n = 2$, $1 \leq s \leq 9$.

s	1	2	3	4	5	6	7	8	9
Upper	1	2	48	1940	1.23×10^5	1.13×10^7	1.41×10^9	2.32×10^{11}	4.87×10^{13}
Actual	1	2	42	1370	7.33×10^4	5.86×10^6	6.53×10^8	9.71×10^{10}	$\mathbf{1.85 \times 10^{13}}$
Lower	1	2	12	144	2.88×10^3	8.64×10^4	3.63×10^6	2.03×10^8	1.46×10^{10}

Table 1: Values of the Game Tree Size for $n = 2$

7 Random Play

An interesting situation is a GQT3 game where both players play their moves randomly. In Section 5 we explained that there is a bijection in the move combinations resulting in any 2 distinct endgames. Thus in a random game the probability of any endgame occurring is equal. Therefore we can analyze this scenario by analyzing the distribution of the endgames. In a 2-by-2 board, player 1 will always win by 0.5 points since this is the result in every endgame. $m = 3$ (3-by-3 board) is more interesting. There are 362,880 endgames which can be distributed into the player 1 win margins as in table 2 below. These values were computed combinatorially and verified using a computer program. Using this we can calculate the probability of each win margin state occurring. *

	Player 1 Win Margin						
Type	-1	-0.5	0	0.5	1	1.5	2
Number of Endgames	34,560	69,984	46,080	33,696	115,200	50,688	12,672
Probability	9.5%	19.3%	12.7%	9.3%	31.7%	14.0%	3.5%
Total	28.8%		12.7%	58.5%			

Table 2: Analysis of Random Games for the 3-by-3 board

From this we can see that the most likely result (mode) is player 1 wins by 1 point. However, player 1 is expected to win the game by 0.452 (3sf) points.

8 Perfect Play

Even more interesting than random play is perfect play, where the 2 players choose the best move possible with the assumption that the other player will also play the best move possible. Most interesting in this analysis is the end result, the Nash Equilibrium [4]. †

We computationally determine the Nash Equilibrium, using the minimax algorithm as a skeleton. Without optimizations it would take about a year to traverse the huge game tree of Goff's QT3. Thus, we use alpha-beta pruning [5], memoization (using a hash table), and symmetry considerations (rotation and reflections) to bring down the run time to under an hour. Through this method, we find the Nash Equilibrium of Goff's QT3 is a Player 1 win by 0.5 points.

This is unfair, thus we attempted to tweak the game rules. If we subtract 1 from the subscripts of all player 2's pieces, then the Nash Equilibrium is a draw. Also, if player 2 chooses collapse then the game is a draw. On the other hand, if player 1 chooses collapse, then he wins by 2 points, the largest possible margin. If the person who causes collapse chooses it then the Nash Equilibrium is a Player 1 win as well.

Unfortunately, for generalized m and n, the game tree is too large for such searches, hence finding the Nash Equilibrium in these cases is room for further research.

9 Artificial Intelligence

We transform the Nash Equilibrium computation method into a perfect artificial intelligence that can play GQT3 from any move for n = 2. Practically, however, it can only solve cases $m \leq 3$ in a reasonable time frame. To enhance user interaction, we precompute and store the AI moves for the 1^{st} and 2^{nd} turn for $m = 3$. Thus with this, we have strongly solved GQT3 for $n \leq 2$ and $m \leq 3$.

*It can be proven that the probabilities of a loss, draw, and win are the same as ordinary tic-tac-toe.

†While the Nash Equilibrium outcome is not unique, the payoff of all Nash Equilibria are the same in this game

In our paper we are mainly interested with exact results or bounds, but an approximate solution for GQT3 can be developed using heuristics we have presented in a previous paper. With this, rather than scanning through the entire game tree, we can look ahead a few moves, rank the game states using a utility function based on our heuristics, and then finally use our perfect play algorithm. Analyzing the accuracy of this method is room for further research. For the small cases, we can do this by pitting this AI against our perfect playing AI.

10 Further Research

In this paper we have done significant research on Generalized Quantum Tic-Tac-Toe and have also conjectured and proven new theorems and algorithms in graph theory. Furthermore, we have done more research and devised more programs that we unfortunately cannot present in the scope of our paper. Yet, there remains much room for extension.

For the game tree size, ideally a mathematical method should be devised to determine the exact value. However, after a lot of investigation, we were unable to make much progress. Hence if this is not possible, perhaps better bounds could be deduced. Instead of explicit formulae, perhaps a quick algorithm for these can be developed. We have one such linear algorithm to improve the lower bound, but there is still a lot more progress possible.

For random games, we have conveniently shown that analysis does not depend on n. Yet we have not yet analyzed large boards. Perhaps a formula or a quick algorithm could be deduced to determine game state distribution amongst the various win margins.

For the Nash Equilibrium, the mathematical methods we attempted to use to determine the Nash Equilibrium (such as the Strategy Stealing argument) have shown no success. Finding a mathematical method for this would likely be difficult but rewarding.

Lastly, an effective AI can be developed to play more complicated versions of the game. This could make use of a utility function, or perhaps other observations of the game that we have missed out. Since this game can easily shown to be PSPACE-Complete (it is a superset of Tic-Tac-Toe), it may be advisable to focus on heuristics and approximate solutions instead of perfect play.

Ultimately, GQT3 is a challenging and interesting game with myriads of future exploration possibilities.

11 Bibliography

[1] Goff, A. (2004, January 18). Quantum Tic-Tac-Toe. Retrieved December 25, 2011, from Paradigm Puzzles: http://www.paradigmpuzzles.com/QT3Play.htm

[2] Goff, A. (2006). Quantum Tic-Tac-Toe: A teaching metaphor for superposition in quantum mechanics. American Journal of Physics, 962-973.

[3] Zeitz, P. (2007). The Art and Craft of Problem Solving. Hoboken, New Jersey: The Art and

Craft of Problem Solving.
[4] Nash, J. (1951). Non-Cooperative Games. The Annals of Mathematics, 286-295.
[5] Russell, S. J., Norvig, P. Artificial Intelligence: A Modern Approach (3rd ed.). Upper Saddle River, New Jersey: Pearson Education.

1 Appendix A: QT3 Game Rules

Quantum tic-tac-toe (QT3) essentially includes only one additional idea, that of quantum moves. As a metaphor to quantum mechanics, where a particle can be in two places at once until its position is measured, players can now play two quantum moves onto the board per turn. This rule is illustrated in Figure 5, where two X pieces have been played onto squares 1 and 2 of the board. (The subscript denotes the turn number.)

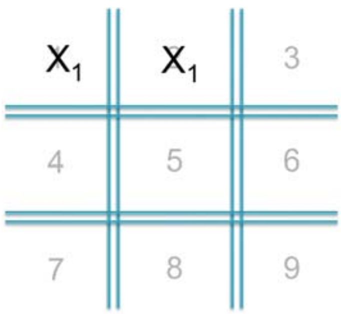

Figure 7: After 1^{st} move

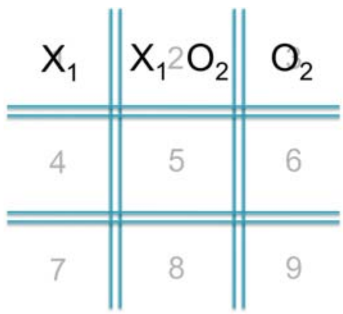

Figure 8: After 2^{nd} move

By the end of the game, one of these two quantum pieces will turn 'classical' and remain on the board, while the other is removed. Hence in this example, X1 will end up in either square 1 or square 2, but not both. How this 'collapse' occurs will be discussed later. Suppose next, Player 2 plays her pieces, labelled O2, into squares 2 and 3 (Figure 6). Now we obviously cannot have two pieces into the same square in normal (classical) tic-tac-toe, but here we are allowed to do so because all pieces are still quantum; we just have to make sure that after collapse, the two classical pieces X1 and O2 do not both land up in square 2.

We note that the pairs of moves X1 and O2 are no longer independent: if X1 collapses into square 2 (and its quantum partner in square 1 is removed), then O2 can no longer collapse into square 2, so it must collapse into square 3. This interdependence of quantum moves adds a new layer of complexity into the game.

Now we define the mechanism of collapse. Clearly we should not collapse after every turn, or else the game will get quite boring; we collapse at a point when there is an entanglement on the board. Goff defines this entanglement as a cycle between quantum pieces, on the condition that any (and only any) two pieces with the same subscript or on the same square are connected (have an edge between them). It turns out that this is the last time collapse can occur validly based on our definitions; we prove this in Section 2.

Hence, if we add a pair of quantum pieces to squares 1 and 3 (Figure 7), there will be entanglement, as there is a cycle (Square 1 X_1 to Square 1 X_3 to Square 3 X_3 to Square 3 O_2 to Square 2 O_2 to Square 2 X_1, and finally back to Square 1 X_1). The presence of cycles in QT3 can be checked efficiently using the depth-first search algorithm.

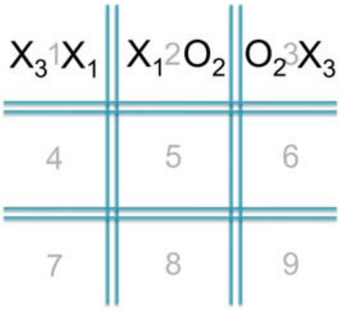

Figure 9: After 3^{rd} move

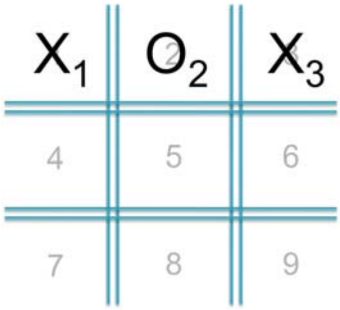

Figure 10: Possible Collapse 1

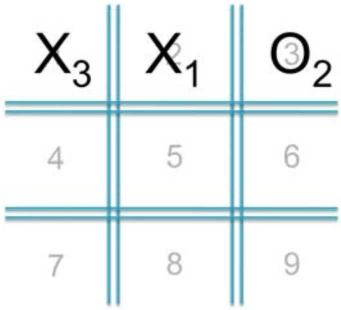

Figure 11: Possible Collapse 2

The person who did not play the latest move (in this case Player 2) will then perform collapse if the entanglement condition is satisfied. She will choose and collapse 1 of the quantum pieces to 1 of its 2 squares. That piece will hence become classical (meaning no other quantum pieces can be played there), and the other pieces in its square and with the same subscript will vanish. All squares containing only one piece will be collapsed to that square in a similar manner, and so on until no more changes occur. If any collapse is invalid (a pair of pieces vanish entirely) then collapse must be restarted. Note that there are always exactly 2 valid collapses that can be chosen by the player choosing collapse (Appendix B). After collapse to a square, *no* pieces can be played on that square. If at any point there is only 1 square remaining, the last classical piece X_9 automatically goes there.

The game ends when all squares are classical and hence occupied by one number exactly. Points are then awarded to players with rows/columns/diagonals (henceforth called lines) filled only with their numbers. The i^{th} line obtained (ranked based on how small the largest number in the line is) is awarded $\frac{1}{i}$ points.

2 Appendix B: Stricter QT3 Collapse Theorem

For the case of $n = 2$, we can show an even stronger result; there are exactly 2! collapses possible. If entanglement has occurred, then there is a connected component $H \subseteq G$ with exactly one cycle. During collapse, we choose one endpoint for each edge in H, such that each vertex in H is chosen exactly once. This is equivalent to choosing an orientation of H such that the indegree of each vertex is 1.

Theorem. *For every graph H with exactly one cycle, there are exactly two orientations of H such* *that* $\deg^-(v) = 1$ *for all vertices* $v \in H$.

Proof. Given an orientation satisfying the stated condition, consider the direction of a fixed edge AB contained in the cycle. If this edge is oriented $A \to B$ then remove AB to obtain H', a tree oriented such that $\deg^-(B) = 0$ and $\deg^-(v) = 1$ for every other v.

Root H' at B. Now for all vertices v_1 which are children of B, the edge Bv_1 must be oriented $B \to v_1$, as the indegree of B is 0. Also, for all vertices v_2 which are children of v_1, the edge v_1v_2 must be oriented $v_1 \to v_2$, since v_1 already has an edge pointing towards it.

In the same fashion we see that all edges in this orientation must point away from B. It is clear that this orientation satisfies the given condition, so there is an unique orientation with $A \to B$. Similarly there is an unique orientation with $B \to A$, and we are done. □

Thus *every QT3 entanglement has exactly 2 collapses.*