Conclusion & Future Work

In conclusion, the results of this project successfully validated the possibility of original goal. The original goal was to create a self-sufficient computer program, which self-generates music, through implementation of various concepts of algorithmic composition, computer science, music theory, and mathematics. In this process, the goal was to make a musical output that was random, but limited in randomness through utilization of parametric algorithms. In these aspects, the project was successful because it synthesized a product that utilized 9 different methods of scientific qualities to emulate in music. These 9 methods (Figure 2) illustrate success through their numbers and the extent to which they were successful. To add, the fields of algorithmic composition, music theory, and computer science were all combined in a systematic way to create a product that embodies each of their merits and interests.

Developing areas were also identified in ‘humanizing’ the final composition through use of musical structure and imperfection that is attributed to human execution. These developing areas have been identified and paired with proposals for improvement or future work. To add, the final product has been juxtaposed to both similar algorithmic compositional programs as well as human compositions to notice their differences and similarities both in quantitative and qualitative fields. Based on the data, an accurate statement is to say that as the factors that contribute to randomization of the next value go up and the limitations on that next value increase, the quality of the music increases as well. This means that as the algorithms further envelop the randomization and it becomes more systematic than random, the musical tendencies go up. This means that the timespan of this project should be essentially infinite as more and more combinations, permutations, and variations of algorithms are experimented with to create the ‘ultimate musical machine’. (See Figure 5)

The practical uses of this program are almost infinite. Among them entails mentioning the trend of computerization in the music industry these days. Much of the music produced and popularized today is becoming less and less instrumental and more computerized. Perhaps the creation of computer software that try their hand in composing music will not only advance this industry through introduction of various new algorithms to implement, but revolutionize it in a way that perhaps the new musical genius will not come from people using computers to make music, but rather people telling computers *how* to make music. After this, it is as easy as sitting back and ‘prooflistening’ the product. To add, it is quite possible that the use of music theory in order to patternize music will lead to advancements in music theory as well. It is quite possible that new patterns will arise as computers compute every musical possibility within a given set of parameters. Moreover, these algorithmic composition programs may be used recreationally. Because each new product is unique from the last, it yields the opportunity to listen infinitely, perhaps in replacement to hearing the same songs repeatedly on the radio. Perhaps, it could inspire human made music through introduction of a riff or pattern than can be utilized on a lrger scale without brain-racking thought processes and creative blocks. In all, with almost infinite practical uses, the future of algorithmic composition is bright.

*Algorithmic*. Christopher Ariza, n.d. Web. 18 Nov. 2013. <http://algorithmic.net>.

Antonoff, Jack, et al. *Some Nights*. N.p.: Alfred Music, 2012. Print.

Biedrzycki, Maciej. "cgMusic - Can computers create music?" *Casual Games Harmony*. N.p., n.d. Web. 9 Mar. 2014. <http://codeminion.com/blogs/maciek/2008/05/cgmusic-computers-create-music/>.

Edwards, Michael. "Algorithmic Composition: Computational Thinking in Music." *Communications of the ACM*. ACM, n.d. Web. 8 Mar. 2014. <http://cacm.acm.org/magazines/2011/7/109891-algorithmic-composition/fulltext>.

Essl, Karlheinz. "Algorithmic Composition." *University of Washington*. University of Washington, n.d. Web. 18 Nov. 2013. <http://wiki.dxarts.washington.edu/sandbox/groups/general/wiki/9e11b/attachments/0bead/K.%20Essl%20-%20Algorithmic%20composition.pdf?sessionID=a3f30c5d2db2af680e2cf41f3627b42c99b9bf5f>.

Harnum, Jonathan. "Basic Music Thoery." *Monarch Knights*. Monarch Knights High School, n.d. Web. 18 Nov. 2013. <http://www.monarchknights.com/teacherwebpages/halladay/documents/BasicMusicTheory1ed\_000.pdf>.

Kreidler, Johannes. "Programming Electronic Music in Pd." *Pure Data*. N.p., n.d. Web. 14 Feb. 2014. <http://www.pd-tutorial.com/english/index.html>.

Maurer, John A. "A Brief History of Algorithmic Composition." *Stanford*. Stanford University, n.d. Web. 8 Mar. 2014. <https://ccrma.stanford.edu/~blackrse/algorithm.html>.

Maurer, John A., IV. "A Brief History of Algorithmic Composition." *Center for Computer Research in Music and Acoustics*. Standford, n.d. Web. 18 Nov. 2013. <https://ccrma.stanford.edu/~blackrse/algorithm.html>.

Mungan, R. Yagiz. "Introduction to Algorithmic Composition and Pure Data." File last modified on July 2012. PDF file.

Nierhaus, Gerhard. *Algorithmic Composition: Paradigms of Automated Music Generation*. New York: Springer Wein New York, 2009. *Google Books*. Web. 18 Nov. 2013. <http://books.google.com/books?id=jaowAtnXsDQC&printsec=frontcover&dq=algorithmic+composition&hl=en&sa=X&ei=H9uKUsCzBq\_gsATtroKoDg&ved=0CC0Q6AEwAA#v=onepage&q=algorithmic%20composition&f=false>.

Puckette, Miller S. "Theory and Techniques of Electronic Music." *University of California, San Diego*. UC San Diego, n.d. Web. 14 Feb. 2014. <http://msp.ucsd.edu/techniques/latest/book-html/>.

Smith, Stuart. "Jazz Theory." *Computer Science Department of the University of Massachusetts*. Rev. 4th. University of Massachusetts, n.d. Web. 18 Nov. 2013. <http://www.cs.uml.edu/~stu/JazzTheory.pdf>.

Van Beethoven, Ludwig. *Rondo in C Major Op. 51, No. 1*. N.p.: n.p., n.d. Print.