Video Game Industry as a Complex Network

Tony Morelli
School of Computer Science and Engineering
University of Nevada, Reno
Reno, NV 89501
Email: morelli@cse.unr.edu

***Abstract*—Entry into the video game market can be a daunting task. The relationships between game titles and publishers may not be known. Some publishers are responsible for the majority of the titles, while others produce very few. This paper presents the current console video game industry (Wii, Xbox 360,PS3) as a complex network. In addition, the console video game industry from 20 years ago is also presented as a comparison. The evolution of the industry is also presented by researching a total of 14 video game consoles. The visual appearance of these networks is used to draw similarities and differences between the console video game industry over time. Background information will be presented to show how complex networks have been used to examine other similar types of data relationships.**

I. Introduction

Analyzing how an industry has changed over a period of time can help predict where that industry is going in the future. The video games industry is projected to bring in revenues over 60 Billion dollars in 2011 [**?**]. This value has been increasing since video games gained mass acceptance when arcades were popping up in the late 70's/early 80's. An industry this big has many people wanting to gain entry. However, without any knowledge of how the industry is shaped it can be a pretty daunting task. It has been shown that visual representations can give a better idea of the characteristics of a system than just raw numbers [**?**]. Although this is a good idea, there has been no work done to visually map out the video games industry. The focus of this paper will be on the console video games industry.

Console video games systems are gaming systems that players typically use at home. All video games are released for a specific console. For example, a game that runs on the Playstation 3 will not run on an Xbox 360. Games, however, can have separate releases that allow the same title to play on both consoles. Madden 2012 has a release for the Xbox 360 and the Playstation 3, but the discs are not interchangeable. In addition to a specific console, games are created by a developer and released by a publisher. A publisher may release games for several developers, and developers may release several titles. The relationship between titles, developers, publishers, and consoles give a good representation of the games industry.

The video games industry has been around for over 30 years. As technologies change, the industries behind them may change as well. It is unknown how the relationship between developers, publishers and consoles has changed over the course of this relatively young industry. To have a comparison, two generations of video game console industry network topologies will be compared: Classic Video Game Consoles and Current Video Game Consoles.

The first generation of video games will be called classic video games. The classic video games are from the 1970's and 1980's. The first console included in this category is the Atari 2600. It was released in 1977 and has been credited as popularizing cartridge based home gaming. The second console included in this group is the Nintendo Entertainment System (NES). At one point, the NES was the best selling console of all time and was released in 1985. The final console in this category is the Sega Master System. It was also released in 1985 as a direct competitor to the NES.

The current generation of video games consoles include products from Nintendo, Microsoft, and Sony. All of these consoles were released within the last 6 years. The first console included in this category is the Nintendo WII. This console revolutionized game play with its motion capturing controller. It is the best selling console of all time with over 87 million sold world wide and was released in 2006. The second console in this category is the Sony Playstation 3 (PS3). The PS3 was released in 2006 as the successor to Sony's highly successful Playstation 2 and includes a Blu-Ray player. The final console included in the current generation is Microsoft's Xbox 360. The Xbox 360 was released in 2005 as the successor to Microsoft's first consumer product, the Xbox.

In addition to the classic and current video games, we will also analyze all generations in between. The classic video games will be generation 1, followed by the Super Nintendo, Sega Genesis, and Turbo Grafx 16 in generation 2. Generation 3 will contain the Nintendo 64 and the Playstation. Generation 4 contains the Playstation 2, the Xbox, and the Gamecube. And finally, generation 5 will contain the current generation of video games described above.

Questions we hope to answer include: Who are the dominant players in today's industry? Does any small group of publishers dominate all consoles? Do different consoles have a different distribution of publishers to titles? Also the relationships of developers is important. Do most of the titles come from a small amount of developers, or is it evenly spread out? The results of this study will be beneficial to someone who is new to the industry, or looking to break into the industry. It will be revealed who the major players are (if any) at all three of the levels.

Another aspect that will be examined throughout this study is how the industry has changed. Are there more publishers now than before? Has the average number of titles per publisher changed over time? The answers to these questions may help predict where the industry is headed for the next 20 years.

The rest of the paper is organized as follows. Section II covers background information and previous work to justify the methods for displaying and analyzing the video game industry. Section III explains the methods of data collection and data analysis. Section IV reveals the results, and the paper is concluded in section VI.

II. Related Work

In order to compare the two networks representing classic console video games and current console video games, methods of comparing two networks must be determined. Existing work has shown that both a graphical and a numerical analysis are valid ways of comparing complex networks.

A. Graphical Analysis

It has been shown that it is easier to recognize differences when they are represented by position, size. or color as opposed to a strictly numerical display [?]. The following are examples of network comparisons that were done using visual inspections.

One research project [?] looked into the change in the organization of the global seed industry between 1996 and 2008. This study utilized animations of the network to show how the industry changed over the years. The complex network showed parent companies as major nodes represented by a size according to the number of smaller companies they owned. The smaller companies were the nodes, and over time as the graph changes, acquisitions and newcomers into the industry can be visualized.

The applicable piece of this graph is the animation. Watching this animation can quickly give a person an idea of what the main point is. Without any reading at all, or even any familiarity with the subject, a lot of information can be obtained from watching the short video.

Looking at soft drink selection at the local convenience store may give the appearance that there are many choices, however a 2008 study [?] shows that most of these choices trickle back to one of a few major parent companies. 89% of the choices are belong to one of three parent companies. Although this statistic seems to be fairly significant, when represented as a visual network where the node sizes are based on the amount of soft drink brands they own, it becomes very clean who the major players are. Unlike the seed industry, this graph contained no technical jargon and was easy for a person to glance at it and get a great understanding of what the graph was showing.

The graphical representation of the soft drink study is directly related to this study. The overall network topology of the video game industry is going to follow very closely to the topology of the soft drink industry.

Complex networks can be artificially generated. A study shows how graphs with nodes added at random and in random positions form. One technique used to draw the random graphs is to show the weight of the nodes by using color [?]. This allows the reader to quickly determine which nodes have more importance than others. The previous two graphical representations used a combination of color and size. This graph was harder to read because of the fact that it utilized color alone. With using color alone, a person reviewing the graph will have to become familiar with which color is more important, as opposed to size where it is inferred that larger is more important.

Visual representations of complex networks do not need to be for research purposes alone. Many organizations use a graphical representation of a network such that new people unfamiliar with the organization will quickly recognize major components. One organization, SHG [?], uses a network diagram to explain the relationships between the parent organization and all of the local chapters of the group.

B. Numerical Analysis

Although a visual comparison between complex networks is very useful, it is also useful to look at some numeric metrics. Using these metrics in conjunction with the visual representation may give a good indication of the type of change that has occurred in the video game industry over the past 25 years. The following are examples of studies, and what metrics were used to analyze networks.

One study [?] looks to compare simulated internet topologies. Network topology generators typically used GT-ITM and Tiers to generate a topology. This generated topology is used to run simulations to better enhance the internet or to find potential problems. In order to be useful, a simulated network must closely resemble the network it is modeling. It was shown that the structure of the internet is a power law. However, the most used generated networks for simulation did not follow power law. This might show that GT-ITM and Tiers may be insufficient when analyzing a simulated internet based on topologies generated by these methods. Researchers are creating modified topology generators that will create a simulated network that is power law. However, this research investigates whether or not power law is the most important metric. The authors introduce several different topology metrics and investigate how important each of them are. A good network topology generator for the internet is one that closely resembles the actual internet. The authors used three basic metrics to determine the accuracy of a generated network topology.

The first metric is *expansion*. They define this as "the average fraction of nodes in the graph that fall within a ball of radius r, centered at a node in the topology". In other words, this is the measure of the number of nodes the start node can connect to within a certain number of hops. The larger this number is, the more expansive the network is. A large expansion indicates the start node can reach a large number of nodes in a low number of hops.

The second metric used is *resilience*. This measures how much a graph is protected from an attack. For example, if you cut a single link in a tree the graph is no longer connected.

This is as opposed to a random graph, where it might require many links to be broken in order for the graph to become disconnected. To measure this, they calculate the average cut-set size within an n-node ball around any node in the topology.

The third metric used is *distortion*. Just as the measurement of resilience, the measurement of distortion is based on the average value from a subset of nodes within an n node ball around any node in the topology. Within this ball, a spanning tree is created on the graph. Then the average distance between any two vertices in the spanning tree that are connected in the graph is calculated. It measures the difference of the number of hops required to go from one side of an edge in the graph to the other when using the constraints of the spanning tree.

When looking to compare two generations of the video game industry, expansion seems to be a reasonable metric. Resilience and distortion are more related to the topology of the network, not its contents. Based on that, those metrics will not be used when comparing the networks of the video game industry over time.

Another use of comparing networks was used in the comparison of automated language translation techniques [**?**]. In this study, sentences were translated between languages by automatic tools, as well as by hand. The resulting automatic translations were compared to the human translations in order to determine which of the automatic translations were the best fit. Sentences were translated between Portuguese and English, and Portuguese and Spanish. The two automatic translations were performed by the software packages Free Translation and Intertran.

The translated texts were modeled as graphs. The resulting directed graph was a graph mapping how frequent word pairs appeared next to each other. Nodes were the words in the translation, and were connected when they appeared consecutively in the translated sentence. In order to account for frequently occurring pairs, the links were weighted based on how often those particular pairs occurred.

Once the three graphs were generated (human translated graph, Free Translation, and Intertran) different metrics were used to compare them. The idea is that based on these metrics, the graph that most closely resembles the human translated graph represents the most accurate translation software. Metrics used where in-degree, out-degree, clustering coefficient, and shortest paths. In-degree represents the frequency a particular node was the second word in the pair. Out-degree represents the frequency a particular node was the first word in the pair. The clustering coefficient represents how much the nodes in the graph tend to cluster together. Their results show that the clustering coefficient values were preserved in quality computer translated texts and not preserved in poor computer translated texts. In-degree and out-degree values were once again preserved for a good computer translator and were not preserved for a bad computer translator.

Out-degree and clustering coefficient will be used when comparing two generations of video games. In-degree is not useful in this study as the it will be the same for all levels due to the strict hierarchical nature of the network. Shortest paths are also irrelevant due to the nature of the topology.

Another analysis of complex networks involved looking at whether or not food web networks are considered small world or scale free networks [**?**]. Food webs depict networks of relationships in ecosystems. They analyzed a variety of food webs each with 26-172 nodes from both aquatic and terrestrial ecosystems.

They used several characteristics or metrics to analyze the complex network. These include characteristic path length, average shortest path length between all pairs of species, clustering coefficient, the average fraction of pairs of species one link away from a species that are also linked to each other, and cumulative degree distribution. They also measure the complexity of a network by connectance. That is the fraction of all possible links that are realized in a network.

From the food-web study, clustering coefficient, and degree distribution seem to be the most relevant. As shown in the previous studies these will be important when comparing the video game industry past and present.

III. Methodology

In order to analyze the video game industry over the past 30 years, several items needed to be completed. First, the data needed to be collected. Secondly, the data needed to be analyzed in a graphical manner, and finally the data needed to be analyzed in a numerical manner. The procedures for achieving these goals is outlined in the following subsections. The overall goal of this analysis is to predict certain characteristics of the video game industry for the next generation of consoles. If these predictions prove to be accurate, they could help a newcomer to the video game industry figure out a business plan that will be in line with the next generation.

A. Data Collection

The most important piece of this project was to get the data. Although the data does exist, it is scattered across the internet, and not available as one nice complete package. The data used for this project lives on two main websites.

The first website used to pull data from is www.games-db.com. This website was founded in 2003 and is dedicated to archiving data relating to video games, including console, publisher, developer and titles - all of which are needed for this project. Although this site is complete for the data is has, it is not updated frequently enough to contain the latest releases. This website was used to collect all data, except for the data on the current generation of consoles.

In order to get data that is up to date and accurate, the data for the current generation of consoles was pulled from www.ign.com. IGN is a global leader in three business verticals: media, digital distribution, and game technology. IGN estimates that their site is visited by 1 in 4 men. Their data is always current, and they have information on the latest game releases, however their data for past consoles is not as complete as their information on the current generation.

None of these websites offer direct access to their data. As a result, a custom web scraper was written from scratch to grab this data. The utility, Video Game Grabber (VG^2), was created. This utility was written in C#. The HttpWebRequest API was used to get the root webpage for each console, and then custom routines were written to get the information for all of the titles for that particular console.

In order to identify any potential errors in scraping the data, if a particular game's publisher, or developer were not able to be determined, the fields for that game were filled with the string ERROR. Examples of errors were when custom items such as footnotes were added to the websites to further describe some aspect of the game. When these were encountered, the information was initially listed as ERROR, and a modification was performed on the data by hand.

The output of the VG^2 were comma separated files (csv) that contained the game title, the publisher and the developer. The data for each console was saved in separate files and the specific console was determined by the name of the file (i.e. ps3.csv). The csv file format was chosen as it is easy to import that into common programs for analysis.

B. Graphical Analysis

As discussed earlier, a graphical representation of a complex network has been shown to be beneficial [**?**]. For this study, all of the consoles' complex networks were graphed based on the relationships of console, publisher, developer and title. Two different tools were used to generate these visual representations of the networks.

Pajek

The first tool used was Pajek. Pajek is a program for analyzing large networks. It is freely available from the program's website [**?**]. Unfortunately, Pajek does not take data in a csv format. As a result, the VG^2 software was modified to translate the csv file into a .net file that Pajek could interpret. The VG^2 program traversed the entire csv, and assigned a unique identifier to each node in the graph. Separate lists were maintained for each of the publishers, developers, and titles, while a global identifier was used to uniquely distinguish them from each other when generating the .net file. Once all the unique nodes were identified, the Vertices section of the .net file was written out. After that, the relationships between all the nodes were identified, and the Edges section of the Pajeck .net file were written out. The weights were all set to a 1.

With the data in the correct format, it was loaded into Pajek by using the folder icon in the networks section. Next, a partition was created by utilizing *Net->Partitions->Degree->All*. This creates the degree partition. The data was then drawn using the *Draw->Partition* command. Unfortunately the default graph is circular which does not show the data in a fashion that represents how the console's underlying structure. To remedy this, the graph was redrawn using *Layout->Energy->Kamada-Kawai->Separate Components*. This produces a nice graph with nodes colored by their degree.

GUESS

Although Pajek colored the nodes based on their degree, it was a little difficult to determine the important nodes. If the size of the nodes was altered such that nodes with high degree are drawn larger, it would be easier to identify who the important players are. GUESS does this very nicely.

GUESS is the Graph Exploration System [**?**]. Unfortunately, GUESS uses a different file format than Pajek, so the data needed to be converted. Instead of converting from the .net format used by Pajek to the .gdf format used by GUESS, the VG^2 software was modified to convert a csv to .gdf. The file format is similar, however instead of using numbers that uniquely identify the node, the .gdf format uses strings. In order to not confuse GUESS, strings in the network were stripped of their special characters. Characters such as spaces and apostrophes, were replaced with underscores.

Once the file was created, it was loaded into GUESS. The nodes were then drawn with the sizes based on the out degree of each node. 'g.nodes.outdegree' followed by 'resizeLinear(outdegree,1,75)' were entered into the GUESS command prompt. This produced a somewhat odd looking graph. In order to get a nice looking graph that shows the topology in a way that is easy to compare, the GUI was used to change the arrangement of the nodes. *Layout->Physics* was used to produce the final GUESS images.

C. Numerical Analysis

The graphical analysis will provide a nice overview of how the industry has changed over the years, but it will not directly show any results based on the statistics of the network. In order to quantitatively look at the change, a number of metrics must be looked at. The specific metrics are listed throughout this section. The metrics will be averaged for 3 consoles for each generation, then plotted to see if there can be any type of line or curve fit to the data. If the results show that there can be some kind of curve fit to the data, it will be important because it could potentially not only show where the industry has been, but where it might go in the future. This information could be used by someone who is starting up a game development company and wants to anticipate what the structure will be like in the next generation of consoles.

One tool used for the numerical analysis of the change of the video game industry will be Pajek. Pajek provides the ability to get the numbers needed to do a proper analysis. Also, the VG^2 software has already been written to output the data in the Pajek format which allows for it to easily be imported. Other tools will be written from scratch when needed and which tools are used to generate which metric are listed below.

Degree Distribution

As shown in [**?**], degree distribution is a useful network metric. To analyze the video game network, out degree will be used. The network of the video games is a directed network with publishers linking to developers who link to titles. Because of this structure, out degree will be used. Pajek provides a nice interface to generate this value:

Net->Partitions->Degree->Output

Just as in the analysis of the clustering coefficient, the values for each console of a particular generation will be averaged, and this average value will be used to generate a line of best fit. This will be used to make a prediction about the degree distribution of the next generation of consoles.

Because there are two different pieces of information that are relevant in this network, we will analyze the out degree at 2 different levels of the hierarchical. First we will look at the out degree at the publisher level. This will reveal the number of developers that typically work for each publisher. Then we will analyze the out degree at the developer level. This will reveal the number of titles created by each developer.

Power Law

We will also analyze the placement of data. The degree distribution will show the average values at each level, but we will need to look a little bit deeper to see how these averages were achieved. The out degree of the developers and the out degree of the publishers will be plotted for all members of each group. These will be compared from the first generation to the current generation to see if either generation more closely follows power law.

IV. Results

A. Graphical Results

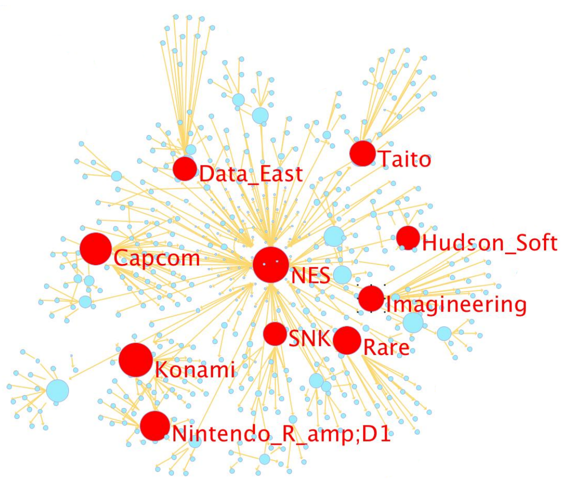

Fig. 1. NES Network

Although 14 consoles were analyzed for this research, 2 graphical representations are shown here. They are the network map for the Nintendo Entertainment System (Figure 1), and the Nintendo Wii (Figure 2). The nodes of the graphs are sized according to their out degree. Larger nodes are connected to more nodes. The largest 10 nodes in the network are colored red, as well as labeled. From these graphs it is interesting to see the overall increase of the number of nodes. The current generation of consoles have so many nodes, that it is difficult to really convey the underlying structure of the networks. The overall structure of the networks remains unchanged, the console industry is still a hierarchal structure.

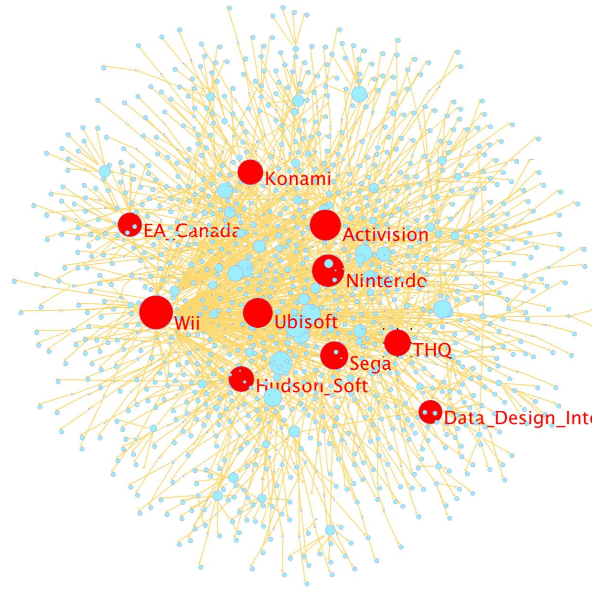

Fig. 2. Wii Network

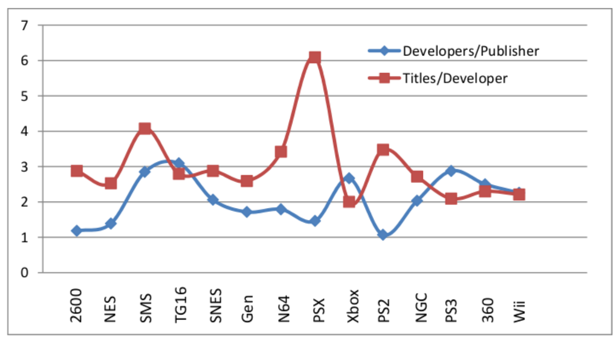

Fig. 3. Publisher, Developer, Title Distribution

B. Numerical Results

Figure 3 shows the degree distribution at both the Publisher and the Developer level of the hierarchical network for each console analyzed. The data is organized in sequence of console release with the Atari 2600 on the far left and the Playstation 3 on the far right. The spike in the middle section is due to a large number of Titles/Developer for the original Playstation console. The original Playstation checked in with an average of 6.01 titles per developer.

Figure 4 shows the degree distribution at both the Publisher and the Developer level of the hierarchical network averaged for each generation of consoles. This is a weighted average with each console for each generation given a weight equal to its share of the total number of nodes for that generation. For the third generation, only the original Playstation and the Nintendo 64 were analyzed. The Playstation accounted for 81.98% of the nodes in the network for that generation, and thus the spike remains when everything is averaged.

Figure 5 shows the out degree required to be in the top 10 of nodes for that particular console. In order to make

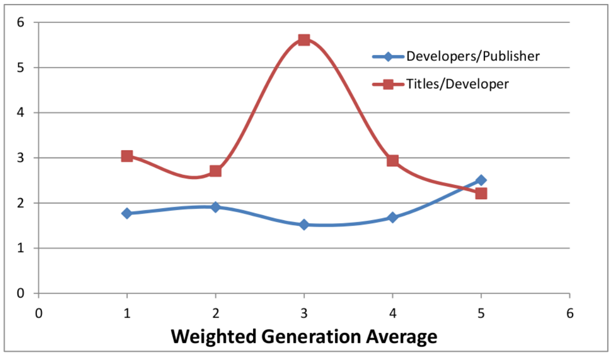

Fig. 4. Publisher, Developer, Title Distribution per Generation

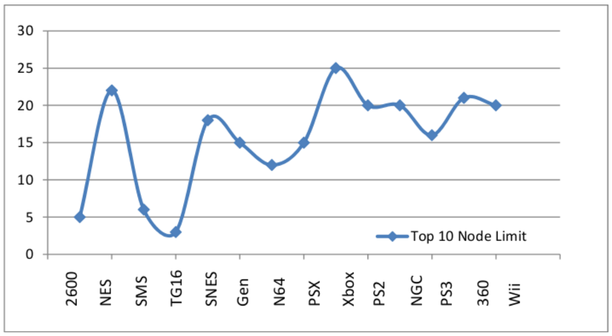

Fig. 5. Out degree required to be in the top 10 nodes for that console

the graphical representations more readable, the top 10 nodes for that console were colored red. This chart breaks that graphical representation down a little further showing the numbers required to be in the top 10 for each console. Recent consoles are stabilizing around the 20 connection mark.

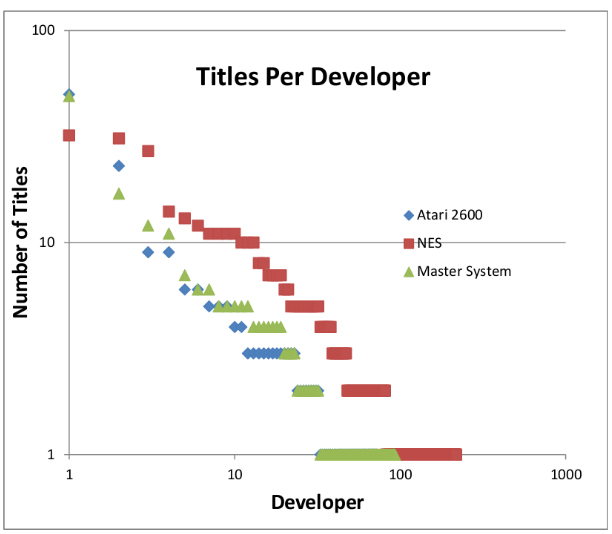

Fig. 6. Log log plot of Titles Per Developer (Classic)

Figures 6 and 7 show the distribution of developers for the classic and current generation in a log log plot. The current generation is very tight, while the classic generation is loose.

Figures 8 and 9 show the distribution of publishers for the classic and current generation in a log log plot. Just as in the developer analysis, the current generation is very tight, while the classic generation is loose.

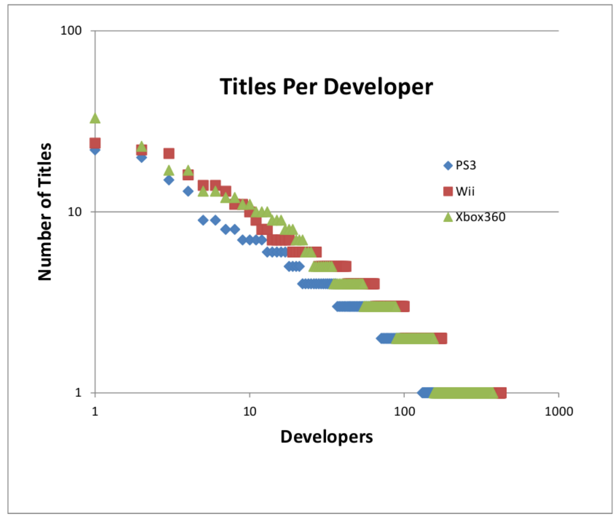

Fig. 7. Log log plot of Titles Per Developer (Current)

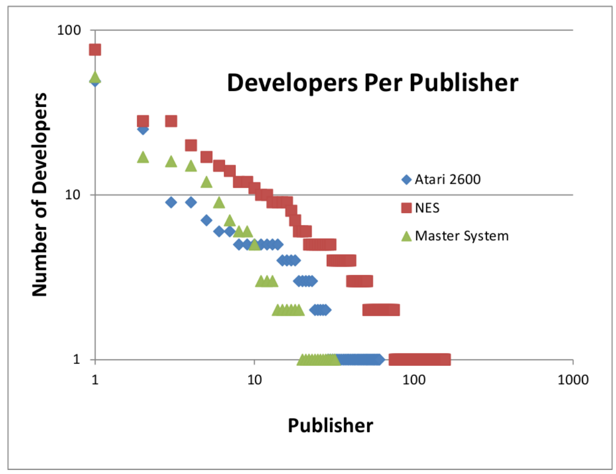

Fig. 8. Log log plot of Developers Per Publisher (Classic)

V. Discussion

The graphical representation of the data shows how much the industry has grown over the past 30 years. Looking at the shear number of nodes in the recent consoles shows how crowded the industry is. Simple graphs such as the NES easily show the hierarchical structure, however when looking at a recent console like the Nintendo Wii, the number of nodes appear as a large blob of data. If this trend continues, the next generation of consoles will probably be more crowded.

The industry as a whole seems to be settling on 2 developers per publisher and 2 titles per developer per console. Early on, these numbers seemed to fluctuate, but as the industry matures, the volatility has subsided and as a result these values appear to be where the future consoles will go. A company wishing to enter this industry must prepare itself to only be focussed

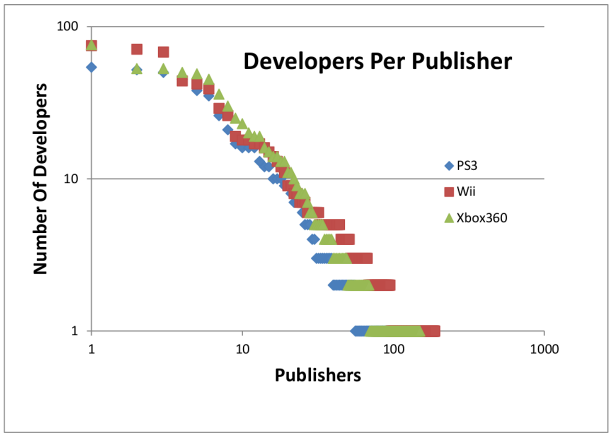

Fig. 9. Log log plot of Developers Per Publisher (Current)

on 2 projects over the lifetime of a console, either by writing two pieces of software if you are a developer, or by publishing for no more than two developers if you are a publisher.

The increase in the number of nodes required to be in the top 10 nodes for a console can be interpreted with the consolidation of the industry. For current consoles, a value of 20 connections is required when the average is just 2. Joining this group will take lots and lots of work. Although stabilizing, this value still seems to have some growth, and to join ranks of the top 10 in the next generation, companies will need to have more than 20 connections.

The log log plots show a power law distribution for the current generation of consoles when looking at the developer and publisher level of the hierarchical network structure. This is slightly different than the charts for the classic generation. If this type of distribution is to continue and the goal is to find a publisher, it may be a good idea to choose one of the few publishers who accept many developers. It can also be perceived that if an individual developer plans many titles, they all may not be completed unless the developer is one of the few that can produce that many titles.

VI. Conclusion

This paper presents a look at the video game console industry by using complex networks. It has been shown that the industry is getting more crowded as each generation of console is introduced. The industry is settling on an average of 2 developers per publisher and 2 titles per developer, however to be considered in the elite of any console, publishers must publish for 20 developers or developers must publish 20 titles. Both graphical analysis using GUESS and numerical analysis using Pajek were performed.

References

[1] J. Tidwell, *Designing Interfaces: Patterns for Effective Interaction Design*,. O'Reilly Media: Sebastopol, CA, USA, 2005.

[2] Philip H. Howard, *Visualizing Consolidation in the Global Seed Industry: 19962008*,. Sustainability 2009.

[3] Philip H. Howard, *The illusion of diversity: visualizing ownership in the soft drink industry*,. https://www.msu.edu/ howardp/softdrinks.html 2008.

[4] Steven H. Strogatz, *Exploring complex networks*,. Nature. March 2001.

[5] Pajek, *Pajek Software*,. http://vlado.fmf.uni-lj.si/pub/networks/pajek/.

[6] GUESS *Graph Exploration Software*,. http://graphexploration.cond.org/.

[7] Zimele, *Developing Community Self Reliance*,. http://www.zimelecommunity.org/programs/microbanks/ 2011.

[8] Hongsuda Tangmunarunkit and Ramesh Gordon and Sugih Jamin and Scott Shenker and Walter Willinger, *Network Topologies, Power Laws, and Hierarchy*,. ACM SIGCOMM January 2002.

[9] Diego R. Amancio and Lucas Antiqueira and Thiago A. S. Pardo and Luciano da F. Costa and Osvaldo N. Oliviera, Jr. and Maria G. V. Nunes, *Complex Networks Analysis Of Manual And Machine Translation*,. International Journal of Modern Physics Vol. 19, No. 4 (2008) 583-59.

[10] Jennifer A. Dunne and Richard J. Williams and Neo D. Martinez, *Food-web structure and network theory: The role of connectance and size*,. Procedings of the National Academy of Sciences of the United States of America.

[11] Paul Noglows, *Moving Online Will Help Video Games Capture More Ad Revenue*,. http://www.businessinsider.com/ 2011.